



Universidade Federal
do Rio de Janeiro
Escola Politécnica

COMPARAÇÃO E VALIDAÇÃO DE MÉTODOS NUMÉRICOS APLICADOS A TRANSFERÊNCIA DE CALOR

Rodrigo Seefelder de Assis Araujo

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.

Orientador: Gustavo Rabello dos Anjos

Rio de Janeiro
Dezembro de 2022

COMPARAÇÃO E VALIDAÇÃO DE MÉTODOS NUMÉRICOS APLICADOS A
TRANSFERÊNCIA DE CALOR

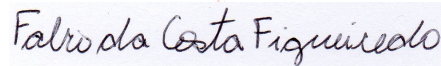
Rodrigo Seefelder de Assis Araujo

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.

Aprovado por:



Prof. Gustavo Rabello dos Anjos



Prof. Fabio da Costa Figueiredo



Prof. Nísio de Carvalho Lobo Brum

RIO DE JANEIRO, RJ - BRASIL
DEZEMBRO DE 2022

de Assis Araujo, Rodrigo Seefelder

Comparação e validação de métodos numéricos aplicados a Transferência de Calor/ Rodrigo Seefelder de Assis Araujo. – Rio de Janeiro: UFRJ/Escola Politécnica, 2022.

XI, 61 p.: il.; 29,7cm.

Orientador: Gustavo Rabello dos Anjos

Projeto de Graduação – UFRJ/ Escola Politécnica/ Curso de Engenharia Mecânica, 2022.

Referências Bibliográficas: p. 60 – 60.

1. Transferência de Calor. 2. Método de Elementos Finitos. 3. Método de Diferenças Finitas. I. Rabello dos Anjos, Gustavo. II. Universidade Federal do Rio de Janeiro, UFRJ, Curso de Engenharia Mecânica. III. Comparação e validação de métodos numéricos aplicados a Transferência de Calor.

*“A ciência, como um todo,
não é nada mais do que um
refinamento do pensar diário.”*

Albert Einstein

Agradecimentos

Agradeço aos meus pais, Mauro e Mônica, pela base que me deram. Pelo exercício diário de amor, compreensão e paciência que me permitiu ser quem sou hoje. Agradeço também ao meu irmão, Marcos, por todo o carinho e ensinamentos que me transmitiu e ainda transmite.

Agradeço aos amigos que tive o privilégio de fazer na faculdade e que tornaram mais divertidos meus estudos e almoços no CT.

Agradeço, também, ao meu orientador, Gustavo Rabello dos Anjos, pelas excelentes eletivas que tive a oportunidade de cursar e por sua disponibilidade, atenção e zelo para com o aprendizado de seus alunos.

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

COMPARAÇÃO E VALIDAÇÃO DE MÉTODOS NUMÉRICOS APLICADOS A TRANSFERÊNCIA DE CALOR

Rodrigo Seefelder de Assis Araujo

Dezembro/2022

Orientador: Gustavo Rabello dos Anjos

Programa: Engenharia Mecânica

O objetivo desse trabalho é analisar a aplicabilidade do Método de Diferenças Finitas e do Método de Elementos Finitos a problemas bidimensionais de condução de calor. Os resultados numéricos obtidos são utilizados como objeto de discussão e comparação, a fim de entender como os métodos se relacionam entre si e para com a solução analítica, quando disponível. Além disso, os conceitos acerca da aplicação dos métodos, bem como aqueles relativos à teoria de Transferência de Calor, são revisados antes da implementação dos métodos em si, a fim de que haja, no trabalho, um embasamento teórico condizente com as análises desenvolvidas.

Palavras-Chave: Condução de Calor, Método de Diferenças Finitas, Método de Elementos Finitos

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

COMPARISON AND VALIDATION OF NUMERICAL METHODS APPLIED
TO HEAT TRANSFER

Rodrigo Seefelder de Assis Araujo

December/2022

Advisor: Gustavo Rabello dos Anjos

Department: Mechanical Engineering

The objective of this work is to expose an analysis on the applicability of the Finite Difference Method and the Finite Element Method to bidimensional heat conduction problems. The obtained numerical results are used as an object of discussion and comparison, in order to elucidate how the methods correlate with each other and with the analytical solution (when the latter is available). Furthermore, the concepts related to the application of the methods, as well as those related to the theory of Heat Transfer are reviewed before the implementation of the methods, in order to construct a theoretical foundation compatible with the analyses carried out.

Key-Words: Heat Conduction, Finite Difference Method, Finite Element Method

Sumário

| | |
|--|-----------|
| Lista de Figuras | x |
| Lista de Tabelas | xi |
| 1 Introdução | 1 |
| 1.1 Motivação | 1 |
| 1.2 Organização do Trabalho | 2 |
| 2 Revisão de Conceitos Fundamentais | 3 |
| 2.1 Transferência de Calor | 4 |
| 2.2 Geração de Malha | 7 |
| 2.2.1 Geração de malha 1D | 8 |
| 2.2.2 Geração de malha 2D com espaçamento linear | 11 |
| 2.2.3 Geração de malha 2D com espaçamento variável | 14 |
| 2.3 Método de Diferenças Finitas | 16 |
| 2.3.1 Aproximação das derivadas | 16 |
| 2.3.2 Montagem e resolução de sistema linear | 19 |
| 2.3.3 Métodos Explícito, Implícito e Crank-Nicholson para proble- mas transientes | 19 |
| 2.4 Método de Elemento Finitos | 21 |
| 3 Análise de Convergência de Malha em Problema Permanente | 31 |
| 3.1 Solução Analítica | 31 |
| 3.2 Modelagem - MDF | 33 |
| 3.3 Convergência da malha - MDF | 34 |
| 3.4 Modelagem - MEF | 36 |

| | | |
|----------|--|-----------|
| 3.5 | Convergência da malha - MEF | 37 |
| 4 | Análise de Problema Transiente | 39 |
| 4.1 | Modelagem MDF | 39 |
| 4.2 | Modelagem MEF | 40 |
| 4.3 | MDF x MEF | 41 |
| 5 | Análise de Problema Transiente com Geração de Calor | 45 |
| 5.1 | Problema 1 | 45 |
| 5.1.1 | Modelagem MDF | 45 |
| 5.1.2 | Modelagem MEF | 46 |
| 5.1.3 | MDF x MEF | 47 |
| 5.2 | Problema 2 | 51 |
| 5.2.1 | Modelagem MDF | 52 |
| 5.2.2 | Modelagem MEF | 52 |
| 5.2.3 | MDF x MEF | 53 |
| 6 | Conclusão | 57 |
| | Referências Bibliográficas | 60 |
| A | Algoritmos Implementados | 61 |

Lista de Figuras

| | | |
|-----|--|----|
| 2.1 | Exemplo de malhas para placa vazada, adaptado de FISH/BELYTS-CHKO [1] | 7 |
| 2.2 | Comparação entre malhas 1D com diferentes espaçamentos | 10 |
| 2.3 | Malhas com $n_x=4$, $n_y=4$ | 12 |
| 2.4 | Malhas com $n_x=7$, $n_y=5$ | 13 |
| 2.5 | Comparação entre malhas com diferentes espaçamentos utilizando a função gaussiana | 15 |
| 2.6 | Elemento Triangular Linear [2] | 26 |
| 3.1 | Distribuição de temperatura na placa 2D em regime permanente . . . | 33 |
| 3.2 | Visualização das soluções numérica e analítica | 34 |
| 3.3 | Análise de convergência da malha - MDF | 35 |
| 3.4 | Tempo de Execução [s] por quantidade de pontos em cada eixo - MDF | 36 |
| 3.5 | Análise de convergência da malha - MEF | 37 |
| 3.6 | Tempo de Execução [s] por quantidade de pontos em cada eixo - MEF | 38 |
| 4.1 | Comparação entre perfis de temperatura utilizando MDF e MEF para $n_x = n_y = 21$ | 43 |
| 5.1 | Comparação entre perfis de temperatura utilizando MDF e MEF para $n_x = n_y = 25$ | 48 |
| 5.2 | Comparação entre perfis de temperatura utilizando MDF e MEF para $n_x = n_y = 41$ (MDF) e $n_x = n_y = 33$ (MEF) | 50 |
| 5.3 | Comparação entre perfis de temperatura utilizando MDF e MEF para $n_x = 48$, $n_y = 24$, $\Delta t = 1e-05$ (MEF) e $\Delta t = 1e-08$ (MDF) | 56 |

Lista de Tabelas

| | | |
|-----|--|----|
| 2.1 | Propriedades apresentadas no capítulo | 3 |
| 2.2 | Hipóteses consideradas na obtenção da equação 2.7 | 6 |
| 2.3 | Vetores posição das malhas unidimensionais geradas | 9 |
| 2.4 | IEN das malhas unidimensionais geradas | 11 |
| 2.5 | Comparação das IEN's para $n_x = 4$ e $n_y = 4$ | 12 |
| 2.6 | Comapração das IEN's para $n_x = 5$ e $n_y = 7$ | 13 |
| 2.7 | Influência do Parâmetro θ na Discretização Temporal | 25 |
| 4.1 | Temperatura média na placa para $n_x = n_y = 21$ | 42 |
| 4.2 | Parâmetros do Algoritmo | 44 |
| 5.1 | Temperatura média na placa para $n_x = n_y = 25$ | 47 |
| 5.2 | Temperatura média na placa para $n_x = n_y = 33$ (MEF) e $n_x = n_y =$ 41 (MDF) | 49 |
| 5.3 | Parâmetros do Algoritmo | 51 |
| 5.4 | Temperatura média na placa para $n_x = 36$, $n_y = 18$, $dt = 1e-06$ (MDF) e $n_x = 36$, $n_y = 18$ e $dt = 1e-04$ (MEF) | 53 |
| 5.5 | Temperatura média na placa para $n_x = 44$, $n_y = 22$, $dt = 1e-07$ (MDF) e $n_x = 44$, $n_y = 22$ e $dt = 1e-05$ (MEF) | 54 |
| 5.6 | Temperatura média na placa para $n_x = 48$, $n_y = 24$, $dt = 1e-08$ (MDF) e $n_x = 48$, $n_y = 24$ e $dt = 1e-05$ (MEF) | 54 |
| 5.7 | Parâmetros do Algoritmo | 55 |

Capítulo 1

Introdução

1.1 Motivação

A aplicação adequada de métodos numéricos é uma ferramenta de suma importância e proeminência na Engenharia contemporânea. Dentre esses métodos, o Método de Diferenças Finitas (MDF) e o Método de Elementos Finitos (MEF) têm sido amplamente utilizado na resolução de problemas de diversos ramos: Mecânica dos Fluidos, Transferência de Calor, Mecânica dos Sólidos, Vibrações, entre outros. Assim, convém estudar a aplicabilidade dos métodos, por meio da simulação de fenômenos naturais pareada com análise de convergência e validação em relação aos resultados registrados na literatura acadêmica.

Este relatório tem como objetivo a resolução de problemas multidimensionais de Transferência de Calor utilizando métodos numéricos. Além disso, constam no escopo do trabalho a comparação dos resultados obtidos pelos métodos entre si e, quando possível, com as soluções analíticas aceitas na literatura. Os problemas analisados ao longo dos capítulos apresentam graus crescentes de complexidade, sendo o mais avançado o de condução de calor bidimensional transiente com geração de calor.

Ao final deste trabalho, espera-se atingir uma compreensão mais aprofundada sobre a implementação de métodos numéricos, seu potencial, limitações e cuidados necessários ao aplicá-los.

1.2 Organização do Trabalho

O trabalho está organizado em 6 capítulos. O primeiro capítulo é meramente introdutório, expondo a motivação por trás do estudo e a organização adotada capítulo a capítulo.

O capítulo 2 (“Revisão de Conceitos Fundamentais”) revisita conceitos de transmissão de calor, geração de malha, Método de Diferenças Finitas (MDF) e Método de Elementos Finitos (MEF).

O capítulo 3 (“Análise de Convergência de Malha em Problema Permanente”) aborda um problema térmico de condução bidimensional permanente, abrangendo sua solução por MDF, MEF e comparação com a solução analítica. A comparação é utilizada como base para a análise de convergência de malha.

O capítulo 4 (“Análise de Problema Transiente”) aborda um problema térmico de condução bidimensional transiente, abrangendo sua solução por MDF, MEF e comparando as soluções obtidas pelos dois métodos. Além disso, a solução numérica permanente (para a qual o perfil de temperatura converge ao iterar os métodos por repetições o suficiente) é comparada com a solução permanente analítica, a fim de validar os resultados obtidos.

Já o capítulo 5 (“Análise de Problema Transiente com Geração de Calor”) aborda dois problemas térmicos de condução bidimensional transiente com geração de calor e suas soluções por MDF e MEF com diferentes graus de refino na discretização do problema.

O capítulo 6, por fim, fecha o trabalho com uma discussão final acerca do que pode ser observado ao longo dos demais capítulos.

Capítulo 2

Revisão de Conceitos

Fundamentais

Este capítulo revisita conceitos de Transferência de Calor, Geração de Malha, Método de Diferenças Finitas e Método de Elementos Finitos. A tabela 2.1 contém a descrição das propriedades presentes nas equações apresentadas no capítulo, bem como seus símbolos:

| Símbolo | Propriedade |
|----------|--------------------------------------|
| ρ | Massa Específica |
| e | Energia Interna |
| k | Condutividade Térmica |
| c_v | Calor Específico (volume constante) |
| c_p | Calor Específico (pressão constante) |
| c | Calor Específico |
| v | Velocidade |
| τ | Tensão |
| p | Pressão |
| T | Temperatura |
| q | Taxa de Geração de Calor |
| α | Difusividade Térmica |

Tabela 2.1: Propriedades apresentadas no capítulo

2.1 Transferência de Calor

Esta seção contém o desenvolvimento das equações diferenciais que regem o problema de condução de calor 2D a partir da equação de energia interna e aplicação de hipóteses simplificadoras do problema.

A equação da energia interna pode ser expressa pela seguinte equação, em notação de índices:

$$\frac{\partial \rho e}{\partial t} + v_j \frac{\partial \rho e}{\partial x_j} = p \frac{\partial v_j}{\partial x_j} + \tau_{ij} \frac{\partial v_i}{\partial x_j} + \frac{\partial^2 k T}{\partial x_j \partial x_j} + q \quad (2.1)$$

Na equação 2.1, cada termo têm diferentes significados físicos. O termo $\frac{\partial \rho e}{\partial t}$ representa a taxa de variação de energia cinética, enquanto $p \frac{\partial v_j}{\partial x_j}$ representa a variação de energia devido a trabalho. Já os termos $\tau_{ij} \frac{\partial v_i}{\partial x_j}$, $\frac{\partial^2 k T}{\partial x_j \partial x_j}$ e q representam, respectivamente, a dissipação viscosa, o fluxo líquido de calor e a taxa de geração de calor.

A utilização de hipóteses simplificadoras, baseadas nas características dos materiais e fenômenos analisados, permite transformar equações complexas em equações mais facilmente aplicáveis. Assumindo a hipótese de que as propriedades k e ρ são constantes, a equação 2.1 toma a seguinte forma:

$$\rho \left[\frac{\partial e}{\partial t} + v_j \frac{\partial e}{\partial x_j} \right] = p \frac{\partial v_j}{\partial x_j} + \tau_{ij} \frac{\partial v_i}{\partial x_j} + \frac{k \partial^2 T}{\partial x_j \partial x_j} + q \quad (2.2)$$

Para a análise em sólidos, a velocidade v é nula. Nesse caso, tem-se:

$$\rho \frac{\partial e}{\partial t} = \frac{k \partial^2 T}{\partial x_j \partial x_j} + q \quad (2.3)$$

Além disso, para materiais incompressíveis, as seguintes equações são verdadeiras (MORAN/SHAPIRO [3]):

$$e = c_v T \quad (2.4)$$

$$c_v = c_p \quad (2.5)$$

Chamando o calor específico c_v de c , para simplificar a notação, obtém-se:

$$e = cT \quad (2.6)$$

Substituindo 2.6 em 2.3 e considerando o calor específico como constante, obtém-se a seguinte equação:

$$\rho c \frac{\partial T}{\partial t} = \frac{k \partial^2 T}{\partial x_j \partial x_j} + q \quad (2.7)$$

Chamando $\alpha = k/\rho c$ e dividindo ambos os lados da equação 2.7 por ρc , a seguinte equação, em notação vetorial, é obtida:

$$\frac{\partial T}{\partial t} = \alpha \nabla^2 T + \frac{q}{\rho c} \quad (2.8)$$

Ou seja, ao tomar a equação 2.1 e aplicar as hipóteses descritas na tabela 2.2, foi possível obter a equação 2.7.

| Hipóteses Consideradas | |
|------------------------|---------------------------|
| 1 | $k = \text{constante}$ |
| 2 | $\rho = \text{constante}$ |
| 3 | Material Sólido: $v = 0$ |
| 4 | Material Incompressível |
| 5 | $c = \text{constante}$ |

Tabela 2.2: Hipóteses consideradas na obtenção da equação 2.7

Para problemas mais particulares, outras simplificações podem ser realizadas. No caso da análise transiente e sem geração de calor, tem-se $Q = 0$, e a equação que rege o problema é a seguinte:

$$\frac{\partial T}{\partial t} = \alpha \nabla^2 T \quad (2.9)$$

Já para problema permanente com geração de calor, $\frac{\partial T}{\partial t} = 0$, e o fenômeno passa a ser representado pela seguinte equação:

$$\nabla^2 T + \frac{q}{\rho c} = 0 \quad (2.10)$$

Por fim, o problema permanente sem geração de calor, onde $\frac{\partial T}{\partial t} = 0$ e $q = 0$, pode ser representado pela equação de Laplace:

$$\nabla^2 T = 0 \quad (2.11)$$

A solução dessas equações diferenciais, em um domínio determinado e com condições de contorno definidas para tal domínio resulta na obtenção do perfil de temperatura neste.

As condições de contorno podem ser uma prescrição para o valor da função de interesse no contorno do domínio (Condição de Contorno de Dirichlet), o valor da derivada dessa função no contorno (Condição de Contorno de Neumann), ou uma combinação das duas supracitadas (Condição de Contorno Mista).

Ao longo dos capítulos a seguir, as soluções analíticas e numéricas apresentadas para os problemas propostos levam em consideração as hipóteses expostas na tabela 2.2 e condições de contorno de Dirichlet.

2.2 Geração de Malha

A aplicação dos métodos de Elementos Finitos e Diferenças Finitas requer uma modelagem discreta do domínio espacial sob análise. Essa modelagem consiste em transformar um domínio contínuo em discreto, através da sua divisão em *elementos*. Os vértices desses elementos são chamados de *nós*. A esse conjunto de elementos e nós que constitui a modelagem discreta do domínio, atribui-se o nome de *malha*.

A figura 2.1 apresenta dois exemplos de malhas representativas de uma placa vazada. O exemplo da esquerda apresenta menor grau de refino, isto é, menos nós e elementos, ao passo que a figura da direita representa uma malha mais refinada. Esta seção almeja elucidar conceitos fundamentais à compreensão do conceito de malha, através da construção de malhas 1D e 2D.

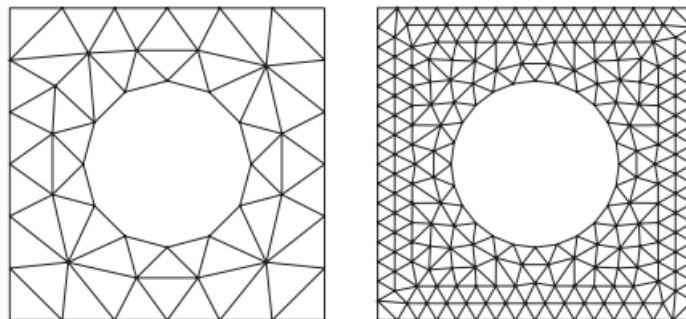


Figura 2.1: Exemplo de malhas para placa vazada, adaptado de FISH/BELYTS-CHKO [1]

Ao criar a malha, é fundamental que se armazene algumas informações a seu respeito:

- a. Coordenadas geométricas dos pontos na forma de vetores
- b. Matriz interconectividade, que é uma matriz que expressa a relação entre os pontos na forma de elementos. Convencionou-se chamar essa matriz de *IEN*

O acesso a esses vetores e matriz viabiliza a utilização de ferramentas como o Método de Diferenças Finitas ou o Método de Elementos Finitos para a resolução aproximada das equações diferenciais que governam problemas de física ou engenharia nos domínios discretizados.

A geração das distribuições de pontos, matrizes e gráficos apresentados daqui em diante neste capítulo foi inteiramente implementada utilizando a linguagem de programação “Python”. A biblioteca “matplotlib” foi utilizada para gerar as imagens representativas das malhas. Além disso, a biblioteca possibilita, também, gerar automaticamente a IEN para malhas bidimensionais de elementos triangulares baseado no fornecimento dos vetores contendo as coordenadas X e Y dos nós da malha. Já a biblioteca “numpy” foi utilizada para armazenar e operar com vetores. As malhas utilizadas na aplicação dos métodos numéricos nos capítulos 3, 4 e 5 também foram construídas inteiramente utilizando a linguagem Python e as bibliotecas supracitadas.

2.2.1 Geração de malha 1D

Em alguns problemas simples de engenharia, uma abordagem unidimensional basta para que se atinja uma resolução adequada. Nesta seção, estão expostos quatro modelos de malha 1D contemplados pelo gerador de malha desenvolvido neste estudo. São estes: malha 1D linear, malha 1D quadrática, malha 1D cúbica, malha 1D exponencial. Para padronizar a malha gerada, adotou-se uma malha com 5 nós e comprimento $L=1$, com uma extremidade em $x=0$ e outra em $x=1$.

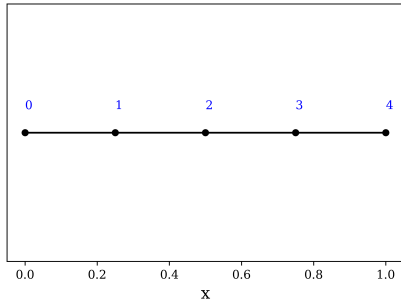
Para gerar essas malhas, criou-se um vetor base (\vec{X}_0) com 5 pontos linearmente

espaçados, cujos valores variam de 0 (no primeiro elemento) a 1 (no último elemento). Esse vetor é o próprio vetor posição da malha 1D linear. Operando com este vetor posição, é possível obter os vetores posição (\vec{X}) das demais malhas expostas na seção, como mostrado na tabela abaixo.

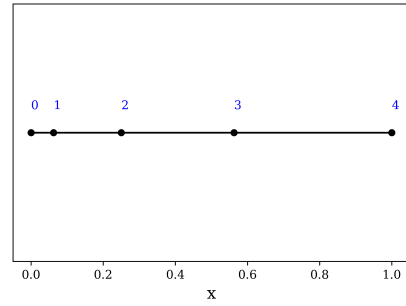
| | Vetor Posição (\vec{X}) |
|----------------------|-------------------------------|
| Malha 1D linear | $\vec{X}_1 = \vec{X}_0$ |
| Malha 1D quadrática | $\vec{X}_2 = \vec{X}_0^2$ |
| Malha 1D cúbica | $\vec{X}_3 = \vec{X}_0^3$ |
| Malha 1D exponencial | $\vec{X}_e = \exp(\vec{X}_0)$ |

Tabela 2.3: Vetores posição das malhas unidimensionais geradas

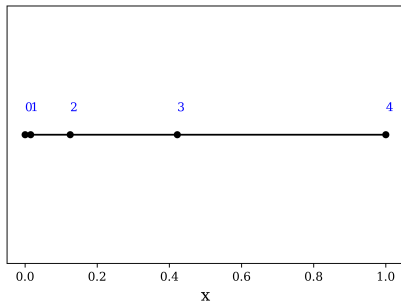
As 3 primeiras malhas têm naturalmente os valores 0 e 1 como mínimo e máximo, respectivamente do vetor posição. Já a quarta malha varia de 1 a e . Para manter a proporção no espaçamento e impor mínimo = 0 e máximo = 1, foi feita a operação $\vec{X}_{ea} = (\vec{X}_e - 1)/(e - 1)$, onde \vec{X}_{ea} é o vetor posição ajustado às condições impostas. Abaixo, a representação gráfica das malhas geradas, onde a numeração dos pontos consta em azul.



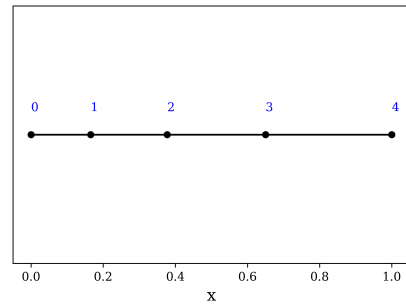
(a) Malha 1D Linear



(b) Malha 1D Quadrática



(c) Malha 1D Cúbica



(d) Malha 1D Exponencial

Figura 2.2: Comparação entre malhas 1D com diferentes espaçamentos

As malhas representadas acima tem os seguintes vetores posição:

- (a) $\vec{X}_1 = [0.0, 0.25, 0.50, 0.75, 1.0]$
- (b) $\vec{X}_2 = [0.0, 0.0625, 0.25, 0.5625, 1.0]$
- (c) $\vec{X}_3 = [0.0, 0.015625, 0.125, 0.421875, 1.0]$
- (d) $\vec{X}_{ea} = [0.0, 0.1653, 0.3775, 0.6501, 1.0]$

A malha 1D Linear possui espaçamento uniforme, ou seja, a distância dos pontos internos aos vizinhos é sempre a mesma. As demais malhas geradas possuem espaçamento variável. Além do vetor posição, o conceito de malha exige também que exista uma matriz interconectividade (IEN). Para as 4 malhas geradas, com 5 pontos cada, a IEN é a mesma:

| v1 | v2 |
|----|----|
| 0 | 1 |
| 1 | 2 |
| 2 | 3 |
| 3 | 4 |

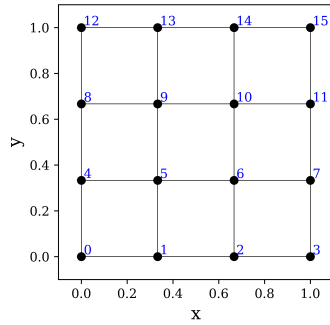
Tabela 2.4: IEN das malhas unidimensionais geradas

Na IEN, cada linha representa um elemento, e os valores constituintes dessa linha contém a numeração dos vértices que formam o elemento. Isto é, as linhas da tabela 2.4 representam os 4 elementos da malha 1D gerada, cujos vértices são representados pelos números $[0,1]$, $[1,2]$, $[2,3]$, $[3,4]$. Assim, o número de linhas é o número de elementos da malha. Já o número de colunas é a quantidade de vértices do elementos (nesse caso, há 2 vértices por elemento). Na próxima seção está a parte relativa à geração de malha bidimensional.

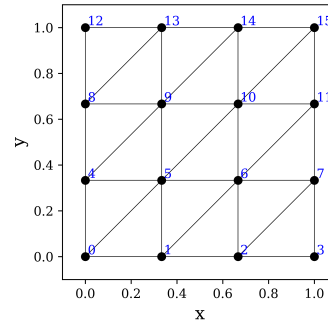
2.2.2 Geração de malha 2D com espaçamento linear

O objetivo primário desta seção é desenvolver uma malha bidimensional com espaçamento linear. Isto é, os diversos pontos que compõem a malha possuem a mesma distância entre si. A fim de padronizar os diversos resultados obtidos, o comprimento L_x e L_y das laterais da malha foi fixado como 1, com as coordenadas variando de 0 a 1 nos eixos x e y . O número de nós, entretanto, varia conforme necessidade do usuário.

Primeiramente, construiu-se uma malha com 4 elementos no eixo x e 4 elementos no eixo y , totalizando 16 pontos. Abaixo, mostra-se a diferença entre a malha de elementos quadrangulares e triangulares para esta quantidade de pontos.



(a) Elementos quadrangulares



(b) Elementos triangulares

Figura 2.3: Malhas com $n_x=4$, $n_y=4$

Além da diferença gráfica observável na figura 1, a opção por elementos triangulares ou quadrangulares afeta a IEN da malha.

| v1 | v2 | v3 | v4 |
|----|----|----|----|
| 0 | 1 | 5 | 4 |
| 1 | 2 | 6 | 5 |
| 2 | 3 | 7 | 6 |
| 4 | 5 | 9 | 8 |
| 5 | 6 | 10 | 9 |
| 6 | 7 | 11 | 10 |
| 8 | 9 | 13 | 12 |
| 9 | 10 | 14 | 13 |
| 10 | 11 | 15 | 14 |

(a) IEN de elementos quadrangulares

| v1 | v2 | v3 |
|-----|-----|-----|
| 0 | 5 | 4 |
| 0 | 1 | 5 |
| 1 | 6 | 5 |
| 1 | 2 | 6 |
| ... | ... | ... |
| 9 | 14 | 13 |
| 9 | 10 | 14 |
| 10 | 15 | 14 |
| 10 | 11 | 15 |

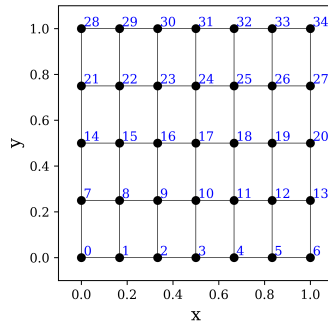
(b) IEN de elementos triangulares

Tabela 2.5: Comparação das IEN's para $n_x = 4$ e $n_y = 4$

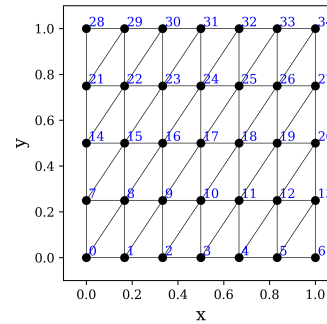
Assim como discutido na seção anterior, na IEN, cada linha representa um elemento, e os valores constituintes dessa linha contém a numeração dos vértices que formam o elemento. Isto é, as linhas da tabela 2.5 (a) representam os elementos quadrangulares da malha 2D gerada, cujos vértices são representados pelos números $[0,1,5,4]$, $[1,2,6,5]$, ..., $[10,11,15,14]$. Já as linhas da tabela 2.5 (b) representam os elementos triangulares da malha 2D gerada, cujos vértices são representados pelos números $[0,5,4]$, $[0,1,5]$, ..., $[10,11,15]$. Ou seja, o número de

colunas da IEN fornece o número de vértices da geometria do elemento utilizado, enquanto o número de linhas da IEN corresponde ao número de elementos da malha.

Fornecendo ao gerador $n_x=7$ e $n_y=5$, as seguintes malhas são obtidas:



(a) Elementos quadrangulares



(b) Elementos triangulares

Figura 2.4: Malhas com $n_x=7$, $n_y=5$

Mais uma vez, percebe-se a diferença entre as IEN's:

| v1 | v2 | v3 | v4 |
|-----|-----|-----|-----|
| 0 | 1 | 8 | 7 |
| 1 | 2 | 9 | 8 |
| 2 | 3 | 10 | 9 |
| 3 | 4 | 11 | 10 |
| ... | ... | ... | ... |
| 23 | 24 | 31 | 30 |
| 24 | 25 | 32 | 31 |
| 25 | 26 | 33 | 32 |
| 26 | 27 | 34 | 33 |

(a) IEN de elementos quadrangulares

| v1 | v2 | v3 |
|-----|-----|-----|
| 0 | 8 | 7 |
| 0 | 1 | 8 |
| 1 | 9 | 8 |
| 1 | 2 | 9 |
| ... | ... | ... |
| 25 | 33 | 32 |
| 25 | 26 | 33 |
| 26 | 34 | 33 |
| 26 | 27 | 34 |

(b) IEN de elementos triangulares

Tabela 2.6: Comparação das IEN's para $n_x = 5$ e $n_y = 7$

Percebe-se que, para o gerador desenvolvido até agora, a malha gerada é bem organizada e previsível, seguindo um padrão lógico facilmente identificável. O número de nós da malha corresponde a $n_x \times n_y$. O número de elementos da malha

(que coincide com o número de linhas da IEN) é de $(nx - 1) \times (ny - 1)$ para elementos quadrangulares e $2 \times (nx - 1) \times (ny - 1)$ para elementos triangulares. Portanto, utilizar esta malha como ponto de partida é uma boa estratégia para se obter familiaridade com o conceito de malha.

Nos exemplos desta seção, foi possível gerar as malhas e obter suas informações manualmente através de loops no *Python*. Para malhas mais desordenadas, o problema de conectar os nós em elementos torna-se mais complexo. No entanto, sabendo os vetores posição (em x e y) dos nós, pode-se utilizar a função `matplotlib.tri.Triangulation` para gerar uma IEN de elementos triangulares. Esta função utiliza o algoritmo de Delaunay para realizar essa tarefa, algoritmo este que visa obter elementos triangulares o mais equiláteros possível, o que é interessante para a qualidade da malha (este assunto será abordado posteriormente). Daqui para a frente, as IEN's de malhas de elementos triangulares serão geradas utilizando esta função.

2.2.3 Geração de malha 2D com espaçamento variável

Na resolução de problemas utilizando métodos numéricos, a busca pela eficiência é de suma importância. Nesse contexto, a malha utilizada tem um grande papel. Em determinados problemas, deseja-se observar, com bastante precisão, fenômenos concentrados em determinadas regiões do domínio, ao passo que outras regiões não demandam uma precisão tão alta. Para isso, é interessante que um gerador de malha tenha flexibilidade de trabalhar com espaçamento variável entre os nós do domínio.

A fim de obter esta maleabilidade no espaçamento entre nós, utilizou-se a função gaussiana para especificar pontos de maior refinamento da malha (utilizando um espaçamento inversamente proporcional a tal função, de modo que a malha é mais refinada na vizinhança de onde a função gaussiana é máxima). Para especificar os pontos de maior refinamento em x e y , definiu-se valores px e py , que variam de 0 a 1. Esse valor representa o valor do parâmetro μ da função gaussiana, e é o ponto onde a função assume valor máximo. Assim, como o espaçamento foi definido

inversamente à função, p_x e p_y indicam os pontos nos eixos x e y onde a malha é mais refinada. O outro parâmetro utilizado na função gaussiana foi o desvio padrão $\sigma = 0.4$. As seguintes malhas foram obtidas com esse método.

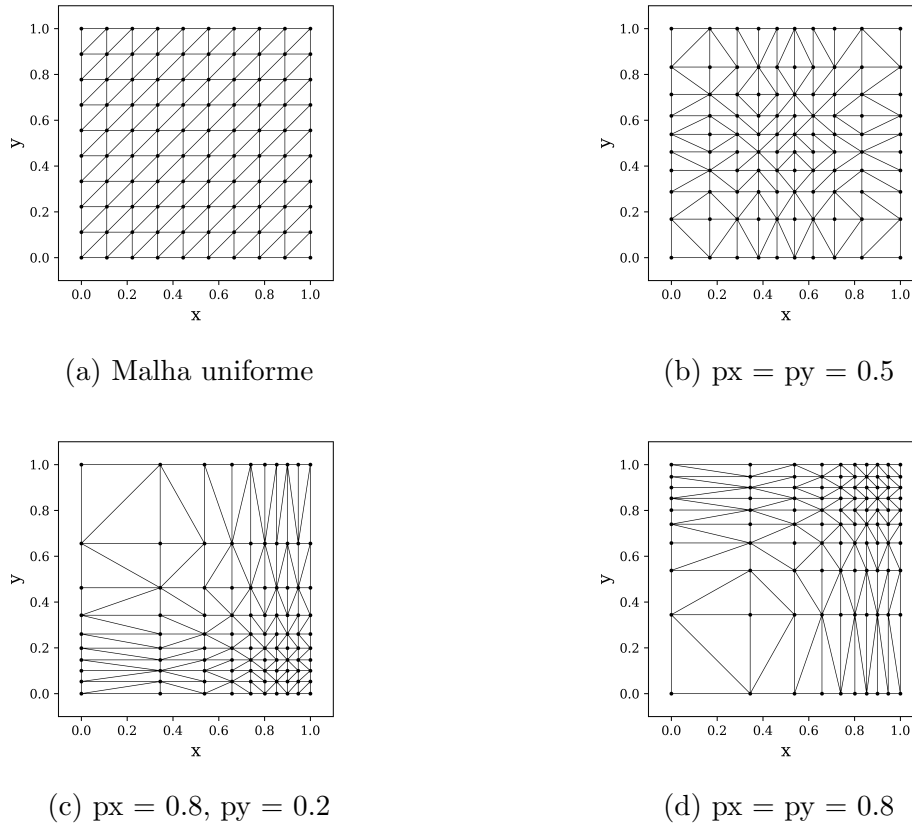


Figura 2.5: Comparação entre malhas com diferentes espaçamentos utilizando a função gaussiana

Esse método desenvolvido objetiva, meramente, fornecer um exemplo de geração de malha 2D com espaçamento variável. Seria possível, assim como na subseção 2.2.1, utilizar, por exemplo, vetores quadráticos, cúbicos ou exponenciais nos eixos para obter esse espaçamento variável. Além disso, diversos softwares já existentes, como o *Gmsh*, podem ser utilizados para a construção de malhas com parâmetros customizáveis. Apesar de, no presente trabalho, ter-se optado pela geração de malhas por meio da linguagem *Python*, a utilização de softwares específicos de geração de malha é uma alternativa ótima para a execução dessa etapa.

É importante ressaltar que, embora a IEN e os vetores posição não estejam aqui explicitados, é fundamental que essas informações sejam armazenadas ao gerar a

malha. Além disso, a variação do espaçamento entre os nós não modifica a IEN, apenas os vetores posição.

Além disso, convém acrescentar que, nas simulações realizadas nos capítulos a seguir, foram utilizadas apenas malhas uniformes. Portanto, a presente seção expõe a possibilidade de geração de malhas não uniformes como um elemento didático, embora a utilização desse tipo de malha não chegue a ocorrer ao longo das soluções numéricas implementadas no trabalho.

2.3 Método de Diferenças Finitas

Uma maneira de resolver problemas governados por equações diferenciais é a aplicação do Método de Diferenças Finitas. Para aplicá-lo é necessário discretizar o domínio espacial - isto é, representá-lo na forma de uma malha, conforme discutido na última seção - e aproximar as derivadas por equações algébricas. A aplicação do método transforma o problema contínuo em um problema algébrico. Além disso, “devido à sua facilidade de aplicação, o método de diferenças finitas é bem apropriado para um tratamento introdutório das técnicas numéricas” (INCROPERA [4]).

2.3.1 Aproximação das derivadas

A base teórica para o desenvolvimento da aproximação de derivadas por diferenças finitas é a sua expansão em série de Taylor (ÖZİŞİK [5]). Essa aproximação é chamada de atrasada quando baseia-se na expansão de uma função $f(x)$ em torno de um ponto deslocado na direção negativa do eixo x , e adiantada quando baseia-se na expansão em torno de um ponto deslocado na direção positiva. Já a aproximação centrada é obtida a partir da combinação das equações provenientes das expansões atrasada e adiantada. A equação 2.12 representa a expansão em série de Taylor em um ponto atrasado, ao passo que a 2.13 representa a expansão em um ponto adiantado.

$$f(x_0 - \Delta x) = f(x_0) - \frac{df}{dx}\Big|_{x=x_0} \Delta x + \frac{d^2 f}{dx^2}\Big|_{x=x_0} \frac{(\Delta x)^2}{2!} - \frac{d^3 f}{dx^3}\Big|_{x=x_0} \frac{(\Delta x)^3}{3!} + \dots \quad (2.12)$$

$$f(x_0 + \Delta x) = f(x_0) + \frac{df}{dx}\Big|_{x=x_0} \Delta x + \frac{d^2 f}{dx^2}\Big|_{x=x_0} \frac{(\Delta x)^2}{2!} + \frac{d^3 f}{dx^3}\Big|_{x=x_0} \frac{(\Delta x)^3}{3!} + \dots \quad (2.13)$$

A aproximação atrasada requer o valor da função no próprio ponto e em pontos anteriores.

$$\frac{\partial f(x, y)}{\partial x}\Big|_{i,j} = \frac{f_{i,j} - f_{i-1,j}}{\Delta x} + 0(\Delta x) \quad (2.14)$$

$$\frac{\partial f(x, y)}{\partial y}\Big|_{i,j} = \frac{f_{i,j} - f_{i,j-1}}{\Delta y} + 0(\Delta y) \quad (2.15)$$

A adiantada requer no próprio ponto e em pontos posteriores.

$$\frac{\partial f(x, y)}{\partial x}\Big|_{i,j} = \frac{f_{i+1,j} - f_{i,j}}{\Delta x} + 0(\Delta x) \quad (2.16)$$

$$\frac{\partial f(x, y)}{\partial y}\Big|_{i,j} = \frac{f_{i,j+1} - f_{i,j}}{\Delta y} + 0(\Delta y) \quad (2.17)$$

A central requer tanto em pontos anteriores quanto posteriores.

$$\frac{\partial f(x, y)}{\partial x}\Big|_{i,j} = \frac{f_{i+1,j} - f_{i-1,j}}{2\Delta x} + 0[(\Delta x)^2] \quad (2.18)$$

$$\frac{\partial f(x, y)}{\partial y} \Big|_{i,j} = \frac{f_{i,j+1} - f_{i,j-1}}{2\Delta y} + 0[(\Delta y^2)] \quad (2.19)$$

Os termos $0(\dots)$ representam o erro de truncamento associado à aproximação por diferenças finitas. Das equações acima, a aproximação central é considerada de segunda ordem, já que seu erro é da ordem de Δx^2 ou Δy^2 , e é mais acurada que as aproximações atrasada e adiantada, de primeira ordem.

Convencionou-se, neste trabalho, utilizar o índice i para a posição em x na malha e o índice j para a posição em y . Além disso, Δx e Δy representam, respectivamente, a distância entre dois pontos consecutivos da malha em x e y .

É possível, também, obter aproximações para derivadas de ordem superior. A equação 2.20 representa a aproximação central da segunda derivada em x da função $f(x,y)$.

$$\frac{\partial^2 f(x, y)}{\partial x^2} \Big|_{i,j} = \frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{\Delta x^2} + 0[(\Delta x^2)] \quad (2.20)$$

As derivadas temporais também podem ser discretizadas, e o índice n foi o escolhido para convencionar a iteração no tempo. O termo Δt quantifica o incremento temporal a cada iteração do método.

$$\frac{df(x, y, t)}{dt} \Big|_{i,j}^n = \frac{f_{i,j}^{n+1} - f_{i,j}^n}{\Delta t} + 0(\Delta t) \quad (2.21)$$

2.3.2 Montagem e resolução de sistema linear

De posse da forma discreta do problema e das equações aproximadas, é possível transformá-lo em um sistema linear da forma $Ax = b$. Por exemplo, para a equação 2.11, a seguinte equação discretizada pode ser obtida via aproximação centrada:

$$\frac{T_{i-1,j} - 2T_{i,j} + T_{i+1,j}}{\Delta x^2} + \frac{T_{i,j-1} - 2T_{i,j} + T_{i,j+1}}{\Delta y^2} = 0 \quad (2.22)$$

Essa equação é denominada *Equação do Nó Geral* e deve ser aplicada aos pontos internos da malha. A montagem da matriz A e do vetor b é realizada através da aplicação da equação do nó geral aos pontos internos da malha, aliada à imposição das condições de contorno estipuladas aos pontos no contorno. Invertendo a matriz A , obtém-se o vetor x de interesse: $AA^{-1}x = x = A^{-1}b$.

2.3.3 Métodos Explícito, Implícito e Crank-Nicholson para problemas transientes

Na resolução de problemas transientes, é necessário efetuar a discretização temporal do problema. Ou seja, dividir o domínio temporal em diversos passos no tempo (demarcados pelo índice n), com um espaçamento de Δt entre esses passos. Além disso, ao discretizar o problema no tempo, diferentes caminhos podem ser tomados. Analisemos a seguinte equação como exemplo:

$$\frac{\partial T}{\partial t} = \nabla^2 T(x, y) \quad (2.23)$$

Sua discretização no espaço (aproximação centrada) e no tempo (aproximação adiantada) resulta na seguinte equação:

$$\frac{T_{i,j}^{n+1} - T_{i,j}^n}{\Delta t} = \frac{T_{i-1,j}^{n+1} - 2T_{i,j}^{n+1} + T_{i+1,j}^{n+1}}{\Delta x^2} + \frac{T_{i,j-1}^{n+1} - 2T_{i,j}^{n+1} + T_{i,j+1}^{n+1}}{\Delta y^2} \quad (2.24)$$

Para o lado direito da equação, é possível adotar tanto o passo n de tempo quanto o passo $n + 1$. A escolha do passo n caracteriza o Método Explícito. Com esse método, é possível isolar $T_{i,j}^{n+1}$ do lado direito da equação, e obter seu resultado a cada iteração sem necessidade de montagem de sistema linear e inversão de matriz:

$$T_{i,j}^{n+1} = T_{i,j}^n + \Delta t \left(\frac{T_{i-1,j}^n - 2T_{i,j}^n + T_{i+1,j}^n}{\Delta x^2} + \frac{T_{i,j-1}^n - 2T_{i,j}^n + T_{i,j+1}^n}{\Delta y^2} \right) \quad (2.25)$$

Já a escolha do passo $n + 1$ caracteriza o Método Implícito. Com esse método, é necessário montar e resolver o sistema linear a cada iteração no tempo:

$$T_{i,j}^{n+1} - \Delta t \left(\frac{T_{i-1,j}^{n+1} - 2T_{i,j}^{n+1} + T_{i+1,j}^{n+1}}{\Delta x^2} + \frac{T_{i,j-1}^{n+1} - 2T_{i,j}^{n+1} + T_{i,j+1}^{n+1}}{\Delta y^2} \right) = T_{i,j}^n \quad (2.26)$$

O Método Explícito é, via de regra, menos custoso computacionalmente, uma vez que não requer a resolução de um sistema linear a cada iteração. No entanto, é necessária adicional cautela na solução do Δt da discretização temporal, uma vez que valores longos demais resultam em um algoritmo instável, incapaz de retornar resultados acurados. É possível também implementar uma combinação dos métodos implícito e explícito, adotando “ n ” para alguns dos termos do lado direito da equação 2.24 e “ $n+1$ ” para outros. Esse método misto é conhecido como *Crank-Nicholson*.

Além disso, a escolha entre Método Explícito, Implícito ou *Crank-Nicholson* tem implicações também quanto à estabilidade do algoritmo. Uma solução numérica é considerada estável se os erros envolvidos nos cálculos não são amplificados ilimitadamente conforme as iterações se sucedem. Os métodos de discretização temporal Implícito e Crank-Nicholson são incondicionalmente estáveis, enquanto o Explícito precisa atender a um critério de estabilidade a depender da equação do nó geral do problema. Para a equação 2.25, o critério de estabilidade a ser atendido é [5]

$$\frac{\alpha\Delta t}{\Delta x^2} + \frac{\alpha\Delta t}{\Delta y^2} \leq \frac{1}{2} \quad (2.27)$$

que pode ser reescrito como

$$\Delta t \leq \frac{1}{2\alpha \left[\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right]} \quad (2.28)$$

Ao longo deste trabalho, será realizada a verificação do critério de estabilidade para as soluções por Método Explícito.

2.4 Método de Elemento Finitos

Muitos fenômenos físicos em engenharia e ciência podem ser descritos através de equações diferenciais parciais. O Método de Elementos Finitos é uma forma de obter soluções aproximadas para as equações que regem tais fenômenos. O método é aplicável a problemas de engenharia de naturezas variadas, como análise de tensão, transferência de calor, mecânica dos fluidos e eletromagnetismo [1].

Sua aplicação, quando bem feita, obtém resultados que aproximam a realidade com bastante acurácia e precisão. O método contempla uma discretização do problema real contínuo. A aplicação do método em si engloba as seguintes etapas, conforme descrito em ANJOS [6]:

Passo A. Transformação do problema da forma forte para a forma variacional (também conhecida como forma fraca)

Passo B. Integração por partes que separa os termos de contorno dos termos internos, resultando na forma fraca final

Passo C. Substituição de funções contínuas por funções discretas

Passo D. Montagem das matrizes do MEF

Passo E. Imposição das condições de contorno

Passo F. Solução do sistema linear $Ax = b$

Como exemplo, tomemos a equação 2.8 a fim de aplicar as etapas do MEF, descritas acima, e atingir maior compreensão acerca do método e de conceitos fundamentais a sua aplicação.

A forma variacional (ou fraca) da equação 2.8 é

$$\int_{\Omega} w \left(\frac{\partial T}{\partial t} - \alpha \nabla^2 T - \frac{q}{\rho c} \right) d\Omega = 0 \quad (2.29)$$

onde o termo representado por w é chamado *função peso*. A equação pode ser reescrita, aplicando-se a distributiva, como

$$\int_{\Omega} w \frac{\partial T}{\partial t} d\Omega - \oint_{\Omega} w \alpha \nabla^2 T d\Omega - \int_{\Omega} w \frac{q}{\rho c} d\Omega = 0 \quad (2.30)$$

Ao aplicar a Identidade de Green para separar os termos de contorno dos termos internos (Passo B), obtém-se

$$\int_{\Omega} w \frac{\partial T}{\partial t} d\Omega - \oint_{\Gamma} w \alpha \nabla T d\Gamma + \int_{\Omega} \alpha \nabla T \nabla w d\Omega - \int_{\Omega} w \frac{q}{\rho c} d\Omega = 0 \quad (2.31)$$

onde Ω é o domínio do problema e Γ o seu contorno.

Os termos T , w e q devem então, dando início ao Passo C, ser representados por

$$T = N_i(x, y) T_i(t) \quad (2.32)$$

$$w = N_j(x, y) w_j \quad (2.33)$$

$$q = N_i(x, y)q_i \quad (2.34)$$

Os termos N_i e N_j são conhecidos como *funções de forma* ou *funções interpoladoras*. Essas funções variam de acordo com a forma do elemento escolhido na discretização do domínio. Mais adiante no capítulo, na execução do "Passo D", será exposta, com mais detalhes, a formulação algébrica dessas funções, baseada na seleção do formato dos elementos da malha. Além disso, nesse relatório, será aplicado o MEF *Galerkin*, o que significa que as funções de forma assumem o mesmo valor, isto é:

$$N_i = N_j \quad (2.35)$$

Tendo-se as equações 2.32, 2.33 e 2.34, é possível reescrever os termos da equação 2.31 como

$$\int_{\Omega} w \frac{\partial T}{\partial t} d\Omega = \sum_i \sum_j w_j \frac{\partial T_i}{\partial t} \int_{\Omega} N_i N_j d\Omega \quad (2.36)$$

onde

$$\int_{\Omega} N_i N_j d\Omega = M_{ij} \quad (2.37)$$

$$\int_{\Omega} \alpha \nabla T \nabla w d\Omega = \sum_i \sum_j w_j \alpha T_i \int_{\Omega} \nabla N_i \nabla N_j d\Omega \quad (2.38)$$

onde

$$\int_{\Omega} \nabla N_i \nabla N_j d\Omega = K_{ij} \quad (2.39)$$

e

$$\int_{\Omega} w \frac{q}{\rho c} d\Omega = \sum_i \sum_j w_j \frac{q_i}{\rho c} \int_{\Omega} N_i N_j d\Omega \quad (2.40)$$

Além disso, para condições de contorno prescritas (Dirichlet), tem-se, para o contorno, que $w = 0$, ou seja

$$\oint_{\Gamma} w \alpha \nabla T d\Gamma = 0 \quad (2.41)$$

As matrizes M_{ij} e K_{ij} são conhecidas, respectivamente, como *Matriz de Massa* e *Matriz de Rigidez*.

Combinando as equações acima, obtém-se, por fim, a forma final da discretização espacial da equação 2.8 pelo MEF Galerkin

$$M_{ij} \frac{\partial T_i}{\partial t} + \alpha K_{ij} T_i - M_{ij} \frac{q_i}{\rho c} \quad (2.42)$$

Nota-se que o problema trata-se de um problema transiente, ou seja, a temperatura varia também como o tempo. Assim, é necessário, também, adotar um método que torne possível avaliar esse comportamento transiente do tempo. No presente trabalho, a discretização temporal adotada em todas as soluções numéricas

desenvolvidas, foi feita utilizando o MDF, e o procedimento detalhado para essa discretização temporal através do método pode ser consultada na subseção 2.3.3.

Ao realizar a discretização espacial por MEF Galerkin, desenvolvida no capítulo, aliada à discretização temporal pelo MDF, obtém-se a seguinte equação

$$\left[\frac{M_{ij}}{\Delta t} + \theta \alpha K_{ij} \right] T_i^{n+1} = [M_{ij} - (1 - \theta) \alpha K_{ij}] T_i^n + M_{ij} \frac{q_i^n}{\rho_i C_i} \quad (2.43)$$

O parâmetro θ possibilita uma formulação geral da discretização temporal, condensando os métodos explícito, implícito e crank-nicholson em apenas uma equação. Adotando-se $\theta = 0$, $\theta = 1/2$ e $\theta = 1$ obtém-se, respectivamente, os métodos Explícito, Crank-Nicholson e Implícito.

| Valores de θ | |
|---------------------|------------------------|
| $\theta = 0$ | Método Explícito |
| $\theta = 1/2$ | Método Crank-Nicholson |
| $\theta = 1$ | Método Implícito |

Tabela 2.7: Influência do Parâmetro θ na Discretização Temporal

Tendo a formulação geral expressa em 2.43, deve-se prosseguir para o Passo D (Montagem das Matrizes do MEF). Essa montagem depende do formato dos elementos selecionados na discretização espacial. No atual exemplo - assim como nos problemas resolvidos nesse trabalho através do MEF - será utilizada uma malha de elementos triangulares e funções de formas lineares.

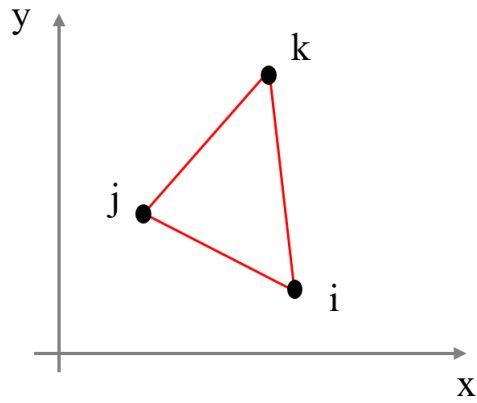


Figura 2.6: Elemento Triangular Linear [2]

Para esse modelo de elemento a função de peso é um vetor da forma

$$N(x, y) = [N_i \ N_j \ N_k] \quad (2.44)$$

onde

$$N_i = \frac{1}{2A}(a_i + b_i x + c_i y) \quad (2.45)$$

$$N_j = \frac{1}{2A}(a_j + b_j x + c_j y) \quad (2.46)$$

$$N_k = \frac{1}{2A}(a_k + b_k x + c_k y) \quad (2.47)$$

Já A é a área do elemento triangular, que pode ser calculada como

$$A = \frac{1}{2} \det \begin{bmatrix} 1 & x_i & y_i \\ 1 & x_j & y_j \\ 1 & x_k & y_k \end{bmatrix} \quad (2.48)$$

Os parâmetros a , b e c dependem das coordenadas dos vértices dos elementos, e adotam os seguintes valores:

$$a_i = x_j y_k - x_k y_j \quad (2.49)$$

$$a_j = x_k y_i - x_i y_k \quad (2.50)$$

$$a_k = x_i y_j - x_j y_i \quad (2.51)$$

$$b_i = y_j - y_k \quad (2.52)$$

$$b_j = y_k - y_i \quad (2.53)$$

$$b_k = y_i - y_j \quad (2.54)$$

$$c_i = x_k - x_j \quad (2.55)$$

$$c_j = x_i - x_k \quad (2.56)$$

$$c_k = x_j - x_i \quad (2.57)$$

A montagem das matrizes do MEF consiste de montagens de *matrizes locais*, isto é, elemento a elemento, que são unidas (na etapa conhecida, também, como *assembly*) para formar as *matrizes globais*. As matrizes elementares de rigidez são expressas por:

$$k^{elem} = \frac{1}{4A} \begin{bmatrix} b_i^2 + c_i^2 & b_i b_j + c_i c_j & b_i b_k + c_i c_k \\ b_j b_i + c_j c_i & b_j^2 + c_j^2 & b_j b_k + c_j c_k \\ b_k b_i + c_k c_i & b_k b_j + c_k c_j & b_k^2 + c_k^2 \end{bmatrix} \quad (2.58)$$

Já as matrizes elementares de massa apresentam a seguinte expressão:

$$m^{elem} = \frac{A}{12} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \quad (2.59)$$

Assim, tendo a IEN da malha (vide seção 2.2) e as matrizes elementares, é possível montar as matrizes globais do MEF através do seguinte algoritmo.

Listing 2.1: *Assembly* das matrizes do MEF

```

for elem in range(0, NE)
    for ilocal in range(0,3)
        iglobal = IEN[elem, ilocal]
        for jlocal in range(0,3)
            jglobal = IEN[elem, jlocal]
            K[iglobal, jglobal] += kelem[ilocal, jlocal]
            M[iglobal, jglobal] += melem[ilocal, jlocal]

```

O algoritmo acima usa a linguagem de *Python* como referência, e implementa a montagem das matrizes para malhas de elementos triangulares com NE elementos.

Tendo a equação 2.43 já com as matrizes montadas, basta realizar os Passos E e F (imposição das condições de contorno e solução do sistema linear $Ax = b$). Como esse relatório se limita a resolver problemas com condições de contorno de Dirichlet, a imposição das condições de contorno é simples. Tomando a equação 2.43, tem-se:

$$A = \left[\frac{M_{ij}}{\Delta t} + \theta \alpha K_{ij} \right] \quad (2.60)$$

$$x = T_i^{n+1} \quad (2.61)$$

$$b = [M_{ij} - (1 - \theta)\alpha K_{ij}] T_i^n + M_{ij} \frac{q_i^n}{\rho_i c_i} \quad (2.62)$$

Vale comentar que, para uma malha com a quantidade de pontos NP, a matriz A tem dimensão NP x NP, enquanto o vetor b tem dimensão NP. Assim, para a realização do Passo E, basta que, para cada ponto do contorno da malha, atribua-se o valor prescrito daquele ponto no vetor b , aliado à atribuição do valor 1 para a posição da matriz correspondente e 0 para as demais posições daquela linha da matriz. Ou seja, sendo cc uma lista com as posições das condições de contorno na malha e bcc uma lista com o valor da temperatura prescrito para essas posições, é possível implementar o seguinte algoritmo (utilizando a linguagem de *Python* como referência) para manipular o sistema linear, impondo as condições de contorno.

Listing 2.2: Imposição das Condições de Contorno

```
for i in cc:
    A[i ; ] = 0.0
    A[i , i] = 1.0
    b[i] = bcc[i]
```

Além disso, para problemas transientes, é necessário impor uma temperatura inicial para o domínio analisado, antes de iniciar o loop de implementação do método. Por fim, para o passo F, basta resolver o sistema linear. Para problemas transientes, o sistema linear deve ser solucionado para cada passo temporal. Além disso, conforme discutido anteriormente, é necessário selecionar o valor de θ , que determina se a implementação da discretização temporal será feita pelo método explícito, implícito ou crank-nicholson.

Portanto, ao seguir os 6 passos descritos na seção, é possível tomar as equações diferenciais regentes de determinado fenômeno, discretizá-las no espaço e no tempo, obtendo, assim, um sistema linear a ser resolvido para cada passo de tempo. Assim, problemas complexos e, muitas vezes, sem soluções analíticas documentadas,

podem ser solucionados aproximadamente por sistemas algébricos, através da implementação de métodos numéricos.

Capítulo 3

Análise de Convergência de Malha em Problema Permanente

Este capítulo é dedicado à solução analítica e numérica (por Método de Diferenças Finitas e Método de Elementos Finitos) de um problema térmico permanente. A resolução do problema serviu de base para uma análise de convergência de malha, baseada no erro absoluto obtido da comparação entre as soluções numérica e analítica. O problema selecionado para a análise é o problema térmico de condução permanente com condições de contorno prescritas (condição de contorno de Dirichlet). A equação diferencial que rege o problema, e as condições de contorno impostas estão descritas abaixo.

$$\nabla^2 T(x, y) = 0 \tag{3.1}$$

$$T(x, 0) = 0 \text{ e } T(x, 1) = 0, \quad \text{para } 0 \leq x \leq 1, \tag{3.2}$$

$$T(0, y) = 0 \text{ e } T(1, y) = -4y^2 + 4y, \quad \text{para } 0 \leq y \leq 1. \tag{3.3}$$

A equação diferencial que representa o problema foi desenvolvida na seção 2.1 e é conhecida como Equação de Laplace.

3.1 Solução Analítica

De acordo com BOYCE/DIPRIMA [7], o problema tem a seguinte solução analítica, onde a é a largura do domínio retangular em x , b a altura deste em y

e $f(y)$ a condição de contorno espacial não nula:

$$T(x, y) = \sum_{n=1}^{\infty} c_n \operatorname{senh} \frac{n\pi x}{b} \operatorname{sen} \frac{n\pi y}{b} \quad (3.4)$$

$$c_n \operatorname{senh} \frac{n\pi a}{b} = \frac{2}{b} \int_0^b f(y) \operatorname{sen} \frac{n\pi y}{b} dy \quad (3.5)$$

Aplicando $a = b = 1$, $f(y) = -4y^2 + 4y$, resolvendo a integral e substituindo a equação 3.5 em 3.4, obtém-se:

$$T(x, y) = \sum_{n=1}^{\infty} \frac{2}{\operatorname{senh}(n\pi)} \frac{8 - 4\pi n \operatorname{sen}(\pi n) - 8\cos(\pi n)}{\pi^3 n^3} \operatorname{senh}(n\pi x) \operatorname{sen}(n\pi y) \quad (3.6)$$

Para uma visualização inicial do resultado, um algoritmo computacional foi implementado para calcular a temperatura em 441 pontos igualmente espaçados no domínio. O somatório da equação 3.6 foi realizado 100 vezes. A figura 3.1 ilustra a distribuição de temperatura calculada por tal operação:

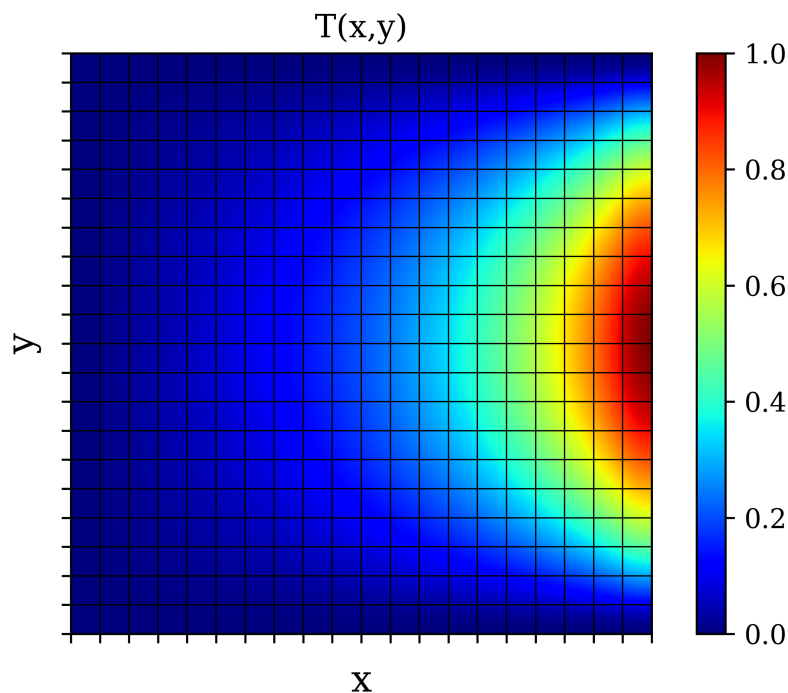


Figura 3.1: Distribuição de temperatura na placa 2D em regime permanente

3.2 Modelagem - MDF

As equações e condições de contorno da seção anterior podem ser discretizadas através do Método de Diferenças Finitas, a fim de que seja possível obter uma solução numérica para o problema baseada na obtenção do vetor x de um sistema linear da forma $Ax = b$.

Nesta seção, a discretização será realizada utilizando aproximação central para as derivadas (mais informações sobre aproximações de derivadas na subseção 2.3.1). Assim, para um nó interno à malha, obtém-se a seguinte equação do nó geral:

$$\frac{T_{i-1,j} - 2T_{i,j} + T_{i+1,j}}{\Delta x^2} + \frac{T_{i,j-1} - 2T_{i,j} + T_{i,j+1}}{\Delta y^2} = 0 \quad (3.7)$$

No sistema linear $Ax = b$ a ser resolvido, os pontos da fronteira, por terem temperatura prescrita (Dirichlet), entram como 1 na matriz A e com seu valor prescrito no vetor b . Já para os pontos internos à malha, os coeficientes inseridos

na construção da matriz A e do vetor b devem obedecer à equação 3.7.

Para uma malha com 441 pontos igualmente espaçados ($n_x = n_y = 21$), o seguinte resultado foi obtido.

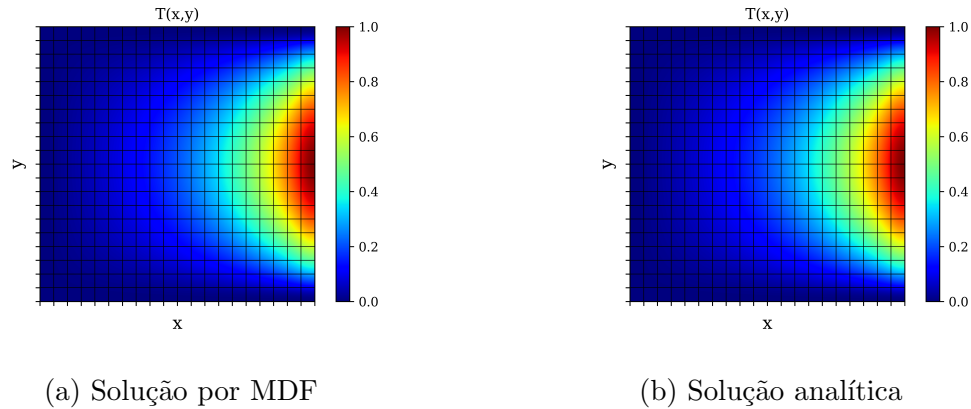


Figura 3.2: Visualização das soluções numérica e analítica

A comparação entre o perfil de temperatura obtido por solução numérica (441 pontos) e analítica disponível na figura 3.2 sugere aplicação bem sucedida do método, assim como seleção adequada de malha. No entanto, uma análise mais criteriosa acerca da malha está exposta na seção a seguir.

3.3 Convergência da malha - MDF

Convém estudar como o erro da solução numérica com relação à analítica varia a depender da quantidade de elementos da malha selecionada. Na figura 3.3, é possível observar esse comportamento do erro a depender do nível de refino da malha. Os dados foram obtidos registrando o erro absoluto máximo para malhas quadradas desde $n_x = n_y = 3$ até $n_x = n_y = 40$.

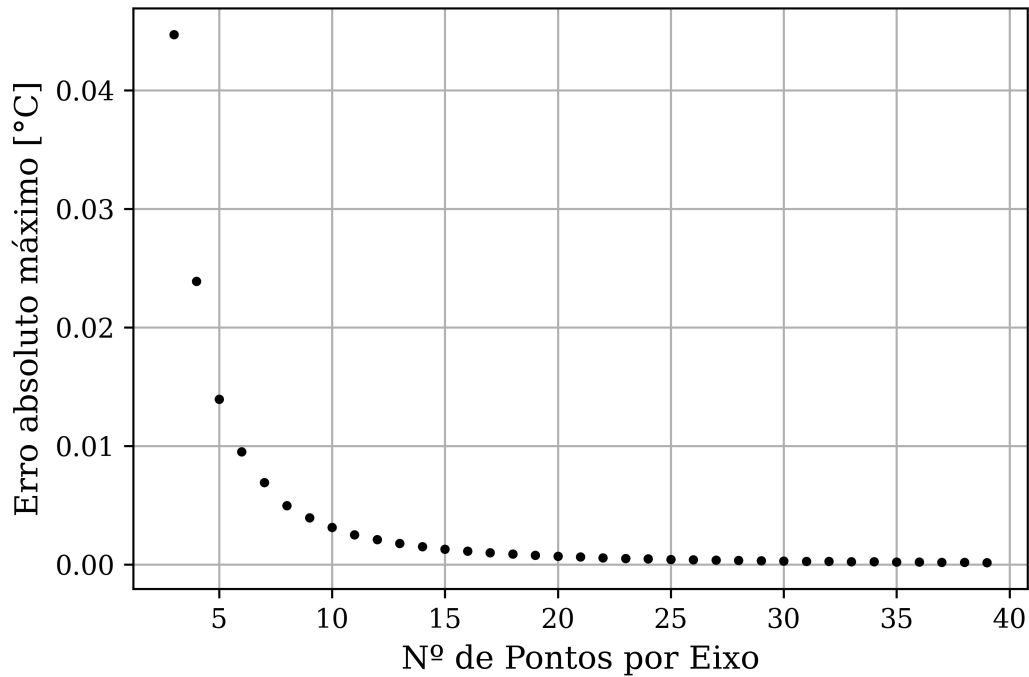


Figura 3.3: Análise de convergência da malha - MDF

É possível perceber um comportamento assintótico do erro, com este tendendo a zero para uma quantidade elevada de elementos de malha. Ao localizar o ponto referente à malha com $n_x = n_y = 21$, totalizando 441 elementos, conclui-se que esta seleção de malha fornece, para o problema posto neste capítulo, uma aproximação adequada ao caso real. O aumento de quantidade de elementos, apesar de resultar, como sugere o gráfico, em redução do erro absoluto, resulta também em um maior custo computacional para solucionar o problema. A figura 3.4 expressa a tendência crescente do tempo de execução do código para a resolução numérica para malhas mais refinadas. Assim, a análise de convergência é uma ferramenta que permite dimensionar uma malha que seja adequada a determinado problema sem que seja necessário custo de processamento em demasia.

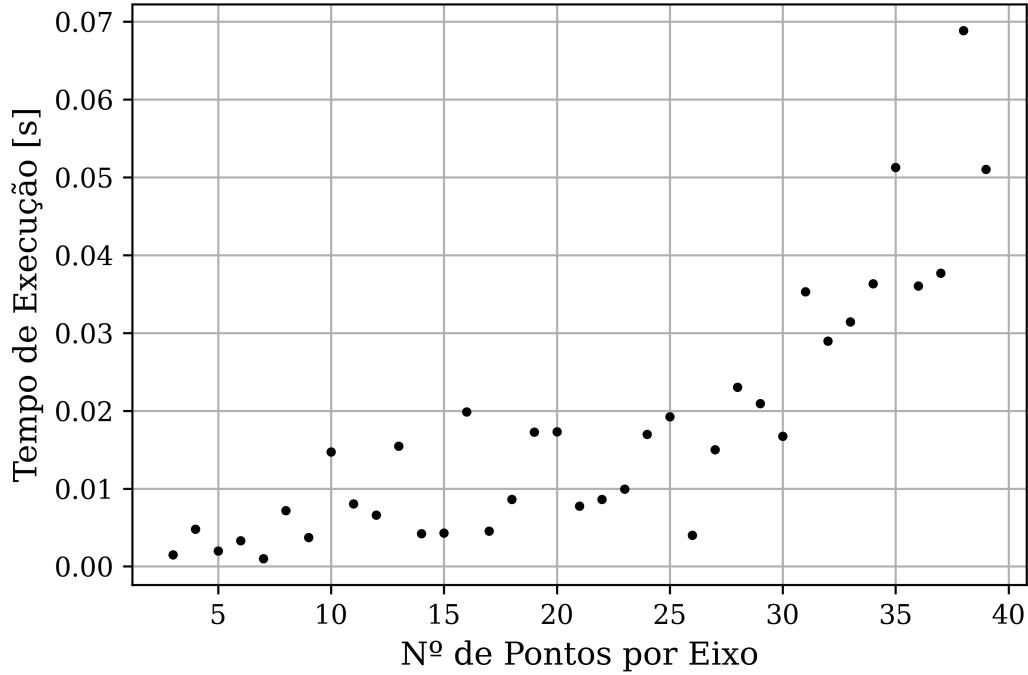


Figura 3.4: Tempo de Execução [s] por quantidade de pontos em cada eixo - MDF

3.4 Modelagem - MEF

Nesta seção, a resolução do problema será feita via MEF-Galerkin. Além disso, optou-se por utilizar função interpoladora linear, elementos triangulares e espaçamento linear entre os pontos da malha. Na seção 2.4, demonstrou-se que, para essas escolhas de método, elementos e função de forma, a seguinte equação é obtida:

$$\left[\frac{M_{ij}}{\Delta t} + \theta \alpha K_{ij} \right] T_i^{n+1} = [M_{ij} - (1 - \theta) \alpha K_{ij}] T_i^n + M_{ij} \frac{q_i^n}{\rho_i c_i} \quad (3.8)$$

Como o problema é permanente e sem geração de calor, é possível simplificar a última equação, obtendo:

$$K_{ij} T_i = 0 \quad (3.9)$$

Tendo a equação 3.9, basta impor as condições de contorno - conforme detalhado na seção 2.4 - e resolver o sistema linear.

3.5 Convergência da malha - MEF

Conforme feito anteriormente para a resolução pelo MDF, esta seção expõe a análise de convergência da malha comparando o erro absoluto entre a solução analítica e a solução numérica obtida pelo MEF.. Na figura 3.5, é possível observar esse comportamento do erro a depender do nível de refino da malha. Os dados foram obtidos registrando o erro absoluto máximo para malhas quadradas desde $n_x = n_y = 3$ até $n_x = n_y = 40$.

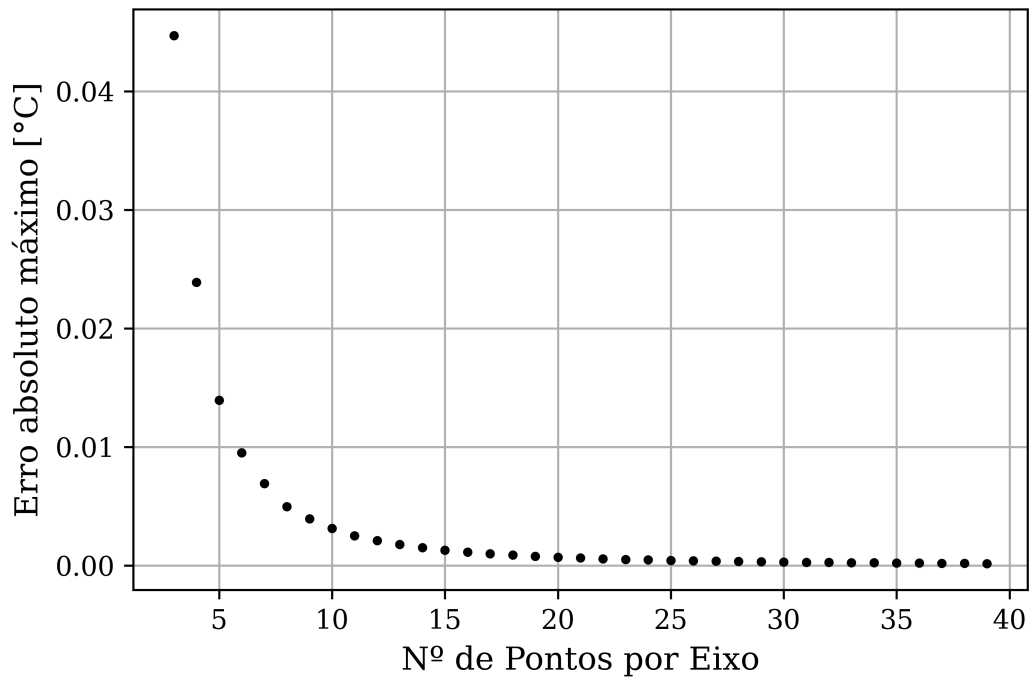


Figura 3.5: Análise de convergência da malha - MEF

Novamente, percebe-se o comportamento assintótico do erro, com este tendendo a zero para uma quantidade elevada de elementos de malha. Além disso, na figura 3.6, percebe-se, novamente, a tendência crescente do tempo de execução do código para a resolução numérica para malhas mais refinadas.

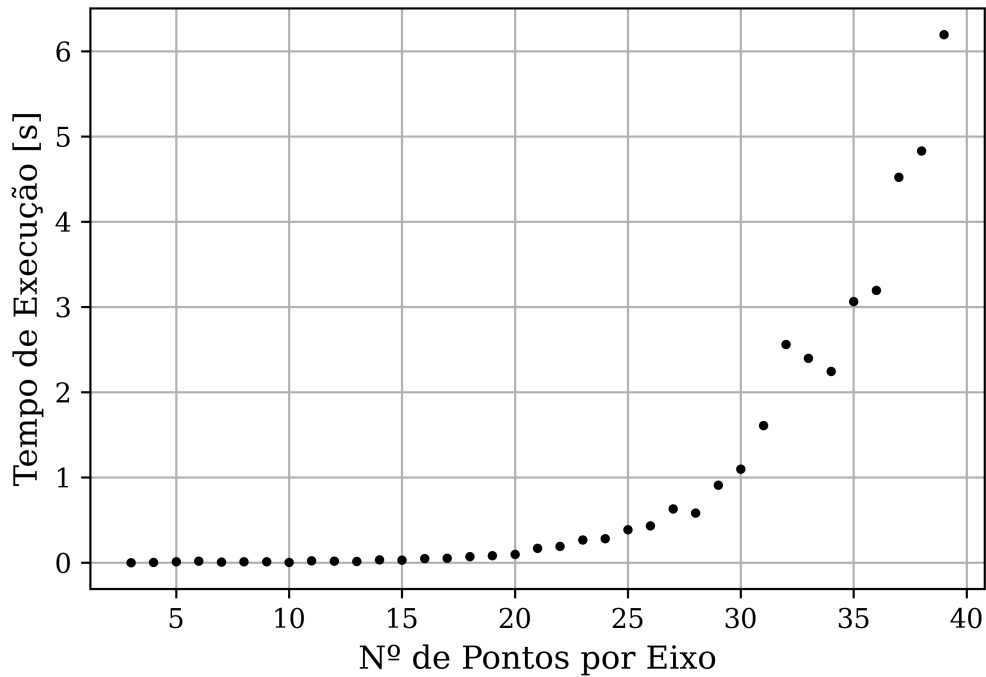


Figura 3.6: Tempo de Execução [s] por quantidade de pontos em cada eixo - MEF

Assim, fica claro que, para esse problema, ambos os métodos são capazes de aproximar a solução analítica de forma bem acurada. Para problemas com solução analítica documentada, a análise de convergência permite que o dimensionamento da malha seja realizado de forma a garantir a precisão da aproximação do método sem que o custo computacional seja excessivamente alto.

Capítulo 4

Análise de Problema Transiente

Este capítulo é dedicado à solução numérica do problema de temperatura transiente na placa bidimensional com condições de contorno de Dirichlet, que pode ser representado pela equação diferencial e condições de contorno abaixo:

$$\alpha \nabla^2 T(x, y) = \frac{\partial T}{\partial t} \quad (4.1)$$

$$T(x, 0) = 0 \text{ e } T(x, 1) = 0, \quad \text{para } 0 \leq x \leq 1, \quad (4.2)$$

$$T(0, y) = 0 \text{ e } T(1, y) = -4y^2 + 4y, \quad \text{para } 0 \leq y \leq 1 \quad (4.3)$$

$$T(x, y)|_{t=0} = 0 \quad (4.4)$$

Nota-se que a equação que governa o fenômeno analisado foi desenvolvida na seção 2.1.

Os métodos de Diferenças Finitas e Elementos Finitos foram implementados para a obtenção comportamento transiente da distribuição de temperatura no material. Além disso, a solução analítica em regime permanente foi utilizada para analisar a convergência de temperatura encontrada na placa através dos dois métodos.

4.1 Modelagem MDF

Para obter o perfil de temperatura permanente da placa pelo método de diferenças finitas foi realizada uma discretização espacial do domínio, com aproximação das derivadas por diferenças finitas. Para o regime transiente, além da discretização

espacial, é necessário realizar, também, uma discretização temporal do problema. Ou seja, as iterações da resolução vão retornar resultado para passos definidos do tempo.

Ao discretizar a equação 4.1 usando aproximação central para as derivadas espaciais e aproximação adiantada para a derivada temporal $\frac{\partial T}{\partial t}$, obtém-se:

$$\frac{T_{i,j}^{n+1} - T_{i,j}^n}{\Delta t} = \alpha \frac{T_{i-1,j}^{n+1} - 2T_{i,j}^{n+1} + T_{i+1,j}^{n+1}}{\Delta x^2} + \alpha \frac{T_{i,j-1}^{n+1} - 2T_{i,j}^n + T_{i,j+1}^{n+1}}{\Delta y^2} \quad (4.5)$$

Adotando o método explícito (mais informações sobre os métodos explícito e implícito na subseção 2.3.3), a equação se torna:

$$T_{i,j}^{n+1} = T_{i,j}^n + \alpha \Delta t \left(\frac{T_{i-1,j}^n - 2T_{i,j}^n + T_{i+1,j}^n}{\Delta x^2} + \frac{T_{i,j-1}^n - 2T_{i,j}^n + T_{i,j+1}^n}{\Delta y^2} \right) \quad (4.6)$$

A partir da equação 4.6, é possível resolver um sistema linear a cada iteração temporal, obtendo a solução numérica para o perfil de temperatura nos passos de tempo desejados.

4.2 Modelagem MEF

Nesta seção, a resolução do problema será feita via MEF-Galerkin. Além disso, optou-se por utilizar função interpoladora linear, elementos triangulares e espaçamento linear entre os pontos da malha. Na seção 2.4, demonstrou-se que, para essas escolhas de método, elementos e função de forma, a seguinte equação é obtida:

$$\left[\frac{M_{ij}}{\Delta t} + \theta \alpha K_{ij} \right] T_i^{n+1} = [M_{ij} - (1 - \theta) \alpha K_{ij}] T_i^n + M_{ij} \frac{q_i^n}{\rho_i c_i} \quad (4.7)$$

Como o problema é transiente, é necessário, também, realizar uma discretização temporal. Realizando esta etapa via Método de Diferenças Finitas (MDF) implícito (isto é, adotando $theta = 1$ na equação 4.7), obtém-se:

$$\left(\frac{M_{ij}}{\Delta t} + \alpha_i K_{ij}\right)T_i^{n+1} = \frac{M_{ij}}{\Delta t}T_i^n + M_{ij} \frac{q_i k_i}{\rho_i c_i} \quad (4.8)$$

Como não há geração de calor no problema analisado, a equação pode ser simplificada para:

$$\left(\frac{M_{ij}}{\Delta t} + \alpha_i K_{ij}\right)T_i^{n+1} = \frac{M_{ij}}{\Delta t}T_i^n \quad (4.9)$$

Tendo a equação 4.9, basta impor as condições iniciais e de contorno - conforme detalhado na seção 2.4 - e resolver o sistema linear, para cada passo temporal, a fim de se obter as soluções numéricas para o problema.

4.3 MDF x MEF

Os resultados de perfil de temperatura transiente obtidos por meio dos dois métodos podem ser observados na Figura 4.1. Utilizou-se malha com 21 pontos igualmente espaçados em cada eixo, com elementos quadrados e triangulares para MDF e MEF, respectivamente. A análise foi feita considerando $\alpha = 1$ m/s². Além disso, utilizou-se passo de tempo Δt de 1e-4 para o MEF e 1e-7 para o MDF. Percebe-se que os perfis são praticamente idênticos, o que pode ser comprovado analisando a temperatura média nos nós para cada método (Tabela 4.1).

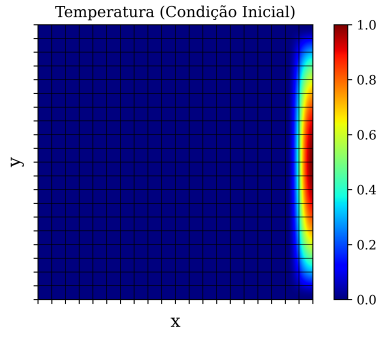
| Tempo [s] | Tmed (MDF) [°C] | Tmed (MEF) [°C] |
|-----------|-----------------|-----------------|
| 0,02 | 0,111 | 0,106 |
| 0,04 | 0,141 | 0,136 |
| 0,08 | 0,170 | 0,166 |
| 0,12 | 0,182 | 0,179 |
| 0,15 | 0,186 | 0,184 |
| 0,18 | 0,188 | 0,187 |
| 0,25 | 0,190 | 0,189 |
| 0,50 | 0,190 | 0,190 |

Tabela 4.1: Temperatura média na placa para $n_x = n_y = 21$

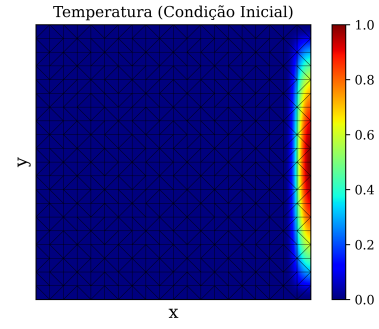
Observa-se que ambos os métodos retornam resultados semelhantes. Além disso, analisar a temperatura média transiente na placa, é possível perceber que ambos os métodos convergem para uma temperatura média de $0,190^\circ\text{C}$ no regime permanente.

Na seção 3.1, a equação 3.6 foi utilizada para calcular a solução analítica da temperatura em regime permanente nos 441 pontos e plotar um mapa de cor representativo desta. Ao calcular a média da temperatura nesses 441 pontos, obtém-se, também, um valor de $0,190^\circ\text{C}$. Essa comparação é um indicativo de que ambos os métodos numéricos foram implementados corretamente. Para garantir maior segurança quanto à acurácia da simulação, uma validação em relação à solução analítica transiente seria a estratégia mais adequada. Porém esta validação de soluções durante o regime transiente não está contemplada no escopo do presente trabalho.

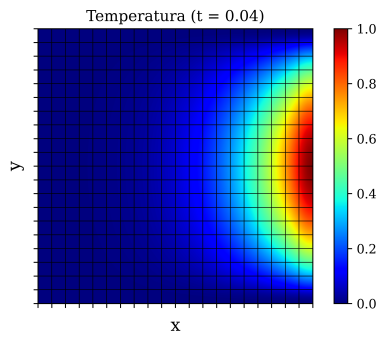
Além disso, vale comentar que, nesse capítulo, considerou-se $\alpha = 1 \text{ m/s}^2$ para facilitar obtenção de resultados e comparação dos métodos. No entanto, ao simular um problema com um material definido, seria necessário utilizar os valores específicos para o material selecionado na simulação, uma vez que o termo α é uma constante do material que depende de κ (condutividade térmica), ρ (massa específica) e c_v (calor específico).



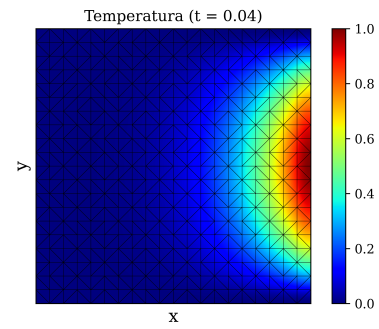
(a) MDF em $t = 0,00$ s



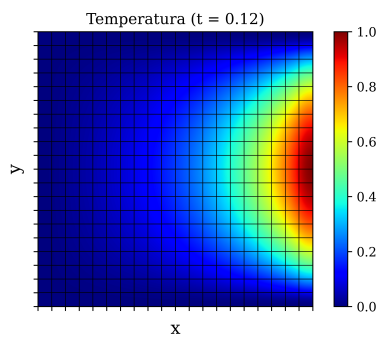
(b) MEF em $t = 0,00$ s



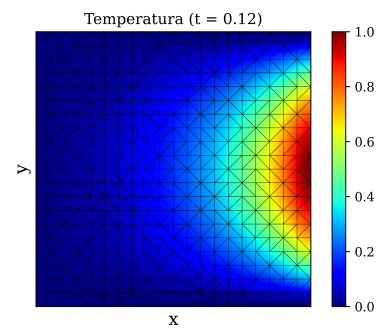
(c) MDF em $t = 0,04$ s



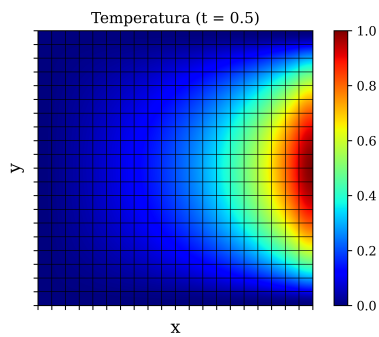
(d) MEF em $t = 0,04$ s



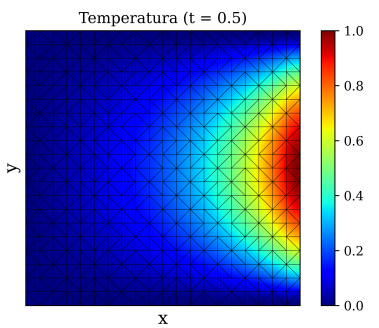
(e) MDF em $t = 0,12$ s



(f) MEF em $t = 0,12$ s



(g) MDF em $t = 0,50$ s



(h) MEF em $t = 0,50$ s

Figura 4.1: Comparação entre perfis de temperatura utilizando MDF e MEF para $n_x = n_y = 21$

Por fim, a tabela 4.2 reúne algumas informações sobre os algoritmos implementados nas resoluções do capítulo. Percebe-se que os tempos de iteração são extremamente superiores aos apresentados no capítulo 3, o que é esperado uma vez que, para problemas transientes, o perfil de temperatura deve ser obtido em cada passo de tempo estipulado, isto é, a cada iteração temporal, é necessário solucionar um sistema linear. Os sistemas foram resolvidos dentro do algoritmo de *Python* usando a função *numpy.linalg.solve*. Sistemas lineares podem ser resolvidos por diferentes métodos, e a seleção destes afeta o tempo de execução do algoritmo. Assim, os tempos de execução expostos neste trabalho poderiam ser diferentes a depender da forma adotada para sua resolução.

| | Δt | Qtd. Iterações | Tempo de Execução [s] | nx | ny |
|---------------|------------|----------------|-----------------------|----|----|
| MDF Explícito | 1e-8 | 6e7 | 134 | 21 | 21 |
| MEF Implícito | 1e-5 | 6e4 | 199,5 | 21 | 21 |

Tabela 4.2: Parâmetros do Algoritmo

Além disso, o critério demonstrado na equação 2.27 é aplicável à equação do nó geral desenvolvida para a resolução por MDF explícito, e precisa ser atendido para garantir que a implementação do método foi estável. Calculando $\frac{\alpha \Delta t}{\Delta x^2} + \frac{\alpha \Delta t}{\Delta y^2}$ obtém-se 8e-6, que é menor que $\frac{1}{2}$. Ou seja, fica comprovada a estabilidade do algoritmo.

Capítulo 5

Análise de Problema Transiente com Geração de Calor

5.1 Problema 1

Nessa seção, será realizada uma análise de um problema similar ao do capítulo anterior, com a adição de geração de calor na placa. O problema contínuo é regido pelas seguintes equações.

$$\alpha \nabla^2 T(x, y) + \frac{Q}{\rho c_v} = \frac{\partial T}{\partial t} \quad (5.1)$$

$$T(x, 0) = 0 \text{ e } T(x, 1) = 0, \quad \text{para } 0 \leq x \leq 1, \quad (5.2)$$

$$T(0, y) = 0 \text{ e } T(1, y) = -4y^2 + 4y, \quad \text{para } 0 \leq y \leq 1, \quad (5.3)$$

$$Q = 2W/m^3, \quad \text{para } 0 \leq x \leq 0,5 \quad (5.4)$$

$$T(x, y)|_{t=0} = 0 \quad (5.5)$$

A solução será obtida por MDF e MEF, e os resultados serão comparados.

5.1.1 Modelagem MDF

Ao discretizar a equação 5.1 usando aproximação central para as derivadas espaciais e aproximação adiantada para a derivada temporal $\frac{\partial T}{\partial t}$, obtém-se:

$$\frac{T_{i,j}^{n+1} - T_{i,j}^n}{\Delta t} = \alpha \frac{T_{i-1,j}^{n+1} - 2T_{i,j}^{n+1} + T_{i+1,j}^{n+1}}{\Delta x^2} + \alpha \frac{T_{i,j-1}^{n+1} - 2T_{i,j}^{n+1} + T_{i,j+1}^{n+1}}{\Delta y^2} + \frac{Q_{i,j}^n}{\rho c_v} \quad (5.6)$$

Adotando o método explícito, a equação se torna:

$$T_{i,j}^{n+1} = T_{i,j}^n + \alpha \Delta t \left(\frac{T_{i-1,j}^n - 2T_{i,j}^n + T_{i+1,j}^n}{\Delta x^2} + \frac{T_{i,j-1}^n - 2T_{i,j}^n + T_{i,j+1}^n}{\Delta y^2} \right) + \Delta t \frac{Q_{i,j}^n}{\rho c_v} \quad (5.7)$$

Assim, é possível obter a temperatura nos pontos a cada iteração temporal.

5.1.2 Modelagem MEF

Nesta seção, a resolução do problema será feita via MEF-Galerkin. Além disso, optou-se por utilizar função interpoladora linear, elementos triangulares e espaçamento linear entre os pontos da malha. Na seção 2.4, demonstrou-se que, para essas escolhas de método, elementos e função de forma, a seguinte equação é obtida:

$$\left[\frac{M_{ij}}{\Delta t} + \theta \alpha K_{ij} \right] T_i^{n+1} = [M_{ij} - (1 - \theta) \alpha K_{ij}] T_i^n + M_{ij} \frac{q_i^n}{\rho_i c_i} \quad (5.8)$$

Como o problema é transiente, é necessário, também, realizar uma discretização temporal. Realizando esta etapa via Método de Diferenças Finitas (MDF) implícito (isto é, adotando $\theta = 1$ na equação 5.8), obtém-se:

$$\left(\frac{M_{ij}}{\Delta t} + \alpha_i K_{ij} \right) T_i^{n+1} = \frac{M_{ij}}{\Delta t} T_i^n + M_{ij} \frac{q_i k_i}{\rho_i c_i} \quad (5.9)$$

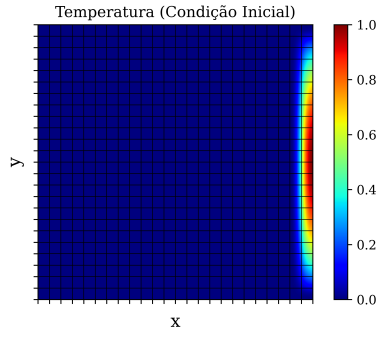
Tendo a equação 5.2.2, basta impor as condições iniciais e de contorno - conforme detalhado na seção 2.4 - e resolver o sistema linear, para cada passo temporal, a fim de se obter as soluções numéricas para o problema.

5.1.3 MDF x MEF

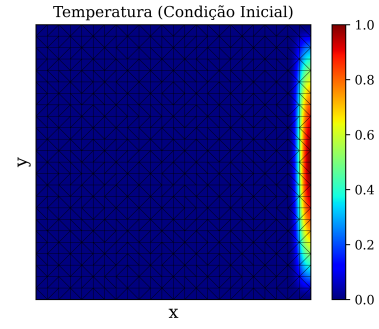
Os resultados de perfil de temperatura transiente obtidos por meio dos dois métodos podem ser observados na Figura 5.1. Utilizou-se malha com 25 pontos igualmente espaçados em cada eixo, com elementos quadrados e triangulares para MDF e MEF, respectivamente. A análise foi feita considerando $\kappa = 1 \text{ W/m.K}$, $c_v = 1 \text{ J/kg.K}$, $\rho = 1 \text{ kg/m}^3$ e $\alpha = \frac{\kappa}{\rho c_v} = 1 \text{ m/s}^2$. Além disso, utilizou-se passo de tempo Δt de $1e-4$ para o MEF e $1e-7$ para o MDF. Percebe-se que os perfis são similares entre si, o que pode ser analisado com mais detalhes ao comparar a temperatura média nos nós para cada método (Tabela 5.1). A discrepância entre a temperatura média na placa pelos dois métodos foi calculada fazendo-se $|\frac{T_{med(MDF)} - T_{med(MEF)}}{T_{med(MDF)}}|$

| Tempo [s] | Tmed (MDF) [°C] | Tmed (MEF) [°C] | Discrepância % |
|-----------|-----------------|-----------------|----------------|
| 0,02 | 0,132 | 0,116 | 11,8 |
| 0,04 | 0,174 | 0,152 | 12,6 |
| 0,08 | 0,216 | 0,190 | 11,9 |
| 0,12 | 0,232 | 0,207 | 10,9 |
| 0,15 | 0,238 | 0,213 | 10,3 |
| 0,18 | 0,240 | 0,220 | 9,9 |
| 0,25 | 0,243 | 0,221 | 9,4 |
| 0,50 | 0,243 | 0,221 | 9,2 |
| 1,00 | 0,243 | 0,221 | 9,2 |

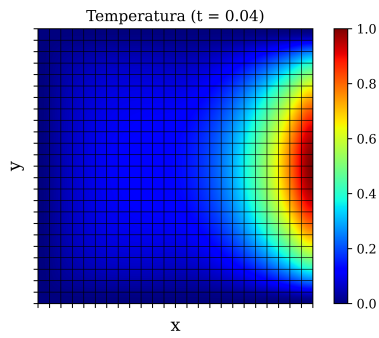
Tabela 5.1: Temperatura média na placa para $n_x = n_y = 25$



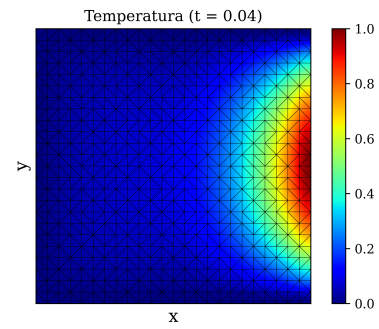
(a) MDF em $t = 0,00$ s



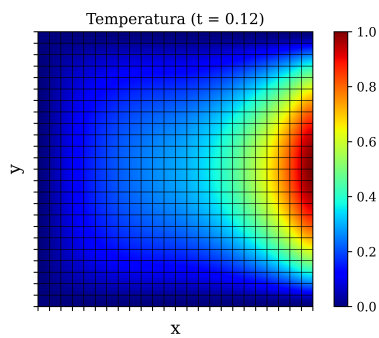
(b) MEF em $t = 0,00$ s



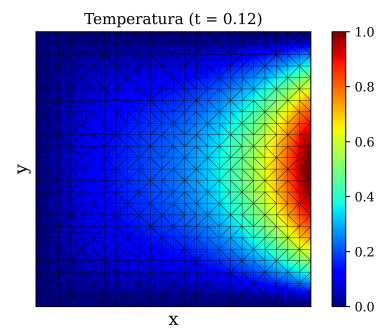
(c) MDF em $t = 0,04$ s



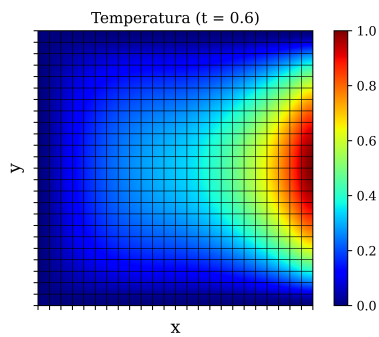
(d) MEF em $t = 0,04$ s



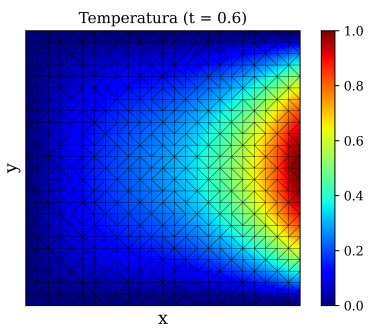
(e) MDF em $t = 0,12$ s



(f) MEF em $t = 0,12$ s



(g) MDF em $t = 0,60$ s



(h) MEF em $t = 0,60$ s

Figura 5.1: Comparação entre perfis de temperatura utilizando MDF e MEF para $n_x = n_y = 25$

Ao analisar os perfis de temperatura obtidos pelos dois métodos, percebe-se que há certa discrepância entre os resultados, apesar de apresentarem tendência similar.

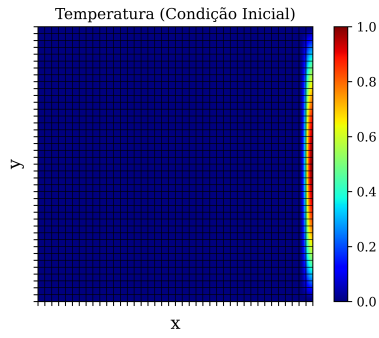
Ao refinar a malha e reduzir o passo temporal, é esperado que a acurácia da solução melhore. Os resultados utilizando MDF com malha 41x41 e $\Delta t = 1e-8$ e MEF com malha 33x33 e $\Delta t = 1e-5$ estão expressos na tabela 5.2. Já as imagens da evolução do perfil de temperatura no tempo estão expressas na figura 5.2.

| Tempo [s] | Tmed (MDF) [°C] | Tmed (MEF) [°C] | Discrepância % |
|-----------|-----------------|-----------------|----------------|
| 0,02 | 0,118 | 0,115 | 3,0 |
| 0,04 | 0,156 | 0,152 | 2,7 |
| 0,08 | 0,194 | 0,191 | 1,7 |
| 0,12 | 0,210 | 0,208 | 0,9 |
| 0,15 | 0,216 | 0,215 | 0,5 |
| 0,18 | 0,219 | 0,218 | 0,2 |
| 0,25 | 0,221 | 0,222 | 0,2 |
| 0,50 | 0,222 | 0,223 | 0,3 |
| 0,60 | 0,222 | 0,223 | 0,3 |

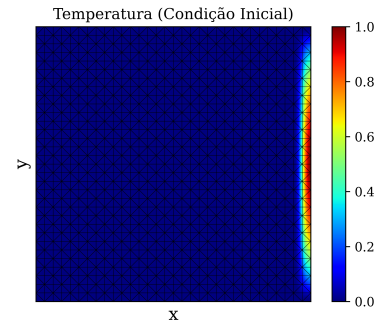
Tabela 5.2: Temperatura média na placa para $n_x = n_y = 33$ (MEF) e $n_x = n_y = 41$ (MDF)

Ao comparar os dados disponíveis na tabela 5.2 com aqueles da tabela 5.1, percebe-se que, utilizando uma malha mais refinada e passo temporal mais curto, a discrepância entre a temperatura média obtida na placa pelos dois métodos ficou mais reduzida. O resultado é um indicativo de que o método foi bem implementado, e que as escolhas de malha e passo temporal foram adequadas. No entanto, para garantir que os resultados obtidos são representativos da evolução teórica do perfil de temperatura na placa, o ideal seria comparar o resultado com a solução analítica do problema, como feito para o problema permanente na seção 3.3. Contudo, a solução analítica não é trivial, de forma que essa validação não foi realizada no presente trabalho.

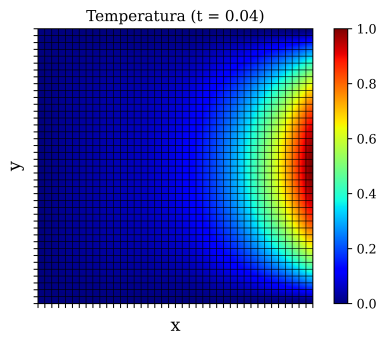
Por fim, a tabela 4.2 reúne algumas informações sobre os algoritmos implemen-



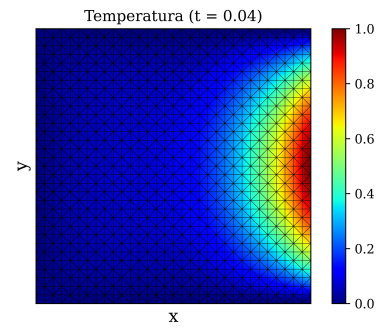
(a) MDF em $t = 0,00$ s



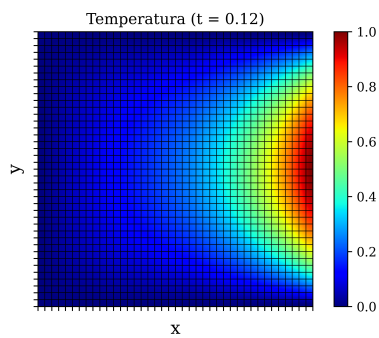
(b) MEF em $t = 0,00$ s



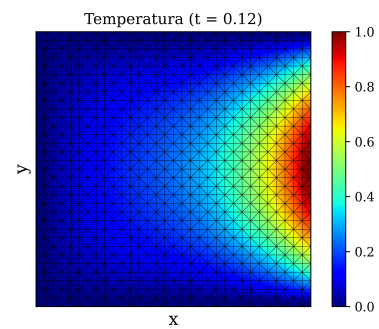
(c) MDF em $t = 0,04$ s



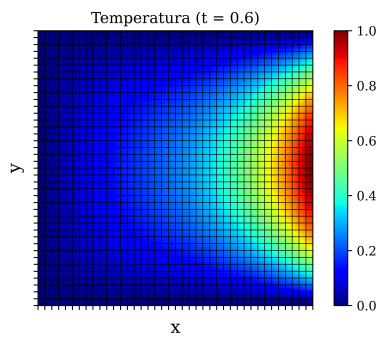
(d) MEF em $t = 0,04$ s



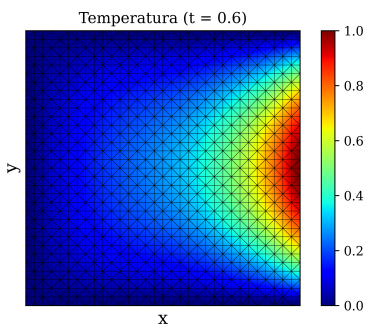
(e) MDF em $t = 0,12$ s



(f) MEF em $t = 0,12$ s



(g) MDF em $t = 0,60$ s



(h) MEF em $t = 0,60$ s

Figura 5.2: Comparação entre perfis de temperatura utilizando MDF e MEF para $n_x = n_y = 41$ (MDF) e $n_x = n_y = 33$ (MEF)

tados nas resoluções do capítulo. Novamente, os sistemas lineares foram resolvidos dentro do algoritmo de *Python*, para cada passo temporal, por meio da função *numpy.linalg.solve*. A solução por MEF Implícito com Δt é um bom exemplo de uma solução custosa computacionalmente. Para os parâmetros selecionados, seu tempo de execução foi de 1668 s. Assim, fica evidente a importância de se escolher um grau de refino que propicie acurácia adequada sem custo computacional excessivo, para evitar que o tempo de execução do algoritmo se torne um empecilho na obtenção dos resultados buscados.

| | Δt | Qtd. Iterações | Tempo de Execução [s] | nx | ny |
|---------------|------------|----------------|-----------------------|----|----|
| MEF Implícito | 1e-4 | 6e3 | 39 | 25 | 25 |
| MDF Explícito | 1e-8 | 6e7 | 175 | 25 | 25 |
| MEF Implícito | 1e-5 | 6e4 | 1668 | 33 | 33 |
| MDF Explícito | 1e-8 | 6e7 | 157 | 41 | 41 |

Tabela 5.3: Parâmetros do Algoritmo

Além disso, o critério demonstrado na equação 2.27 é aplicável à equação do nó geral desenvolvida para a resolução por MDF explícito, e precisa ser atendido para garantir que a implementação do método foi estável. Calculando $\frac{\alpha \Delta t}{\Delta x^2} + \frac{\alpha \Delta t}{\Delta y^2}$ para a primeira simulação via MDF Explícito do Problema 1 do presente capítulo, obtém-se $1,15e-5$, que é menor que $\frac{1}{2}$. Ou seja, fica comprovada a estabilidade do algoritmo. Já para a segunda simulação via MDF Explícito do Problema 1, $\frac{\alpha \Delta t}{\Delta x^2} + \frac{\alpha \Delta t}{\Delta y^2} = 3,2e-5$, comprovando, também, a estabilidade.

5.2 Problema 2

Esta seção objetiva dar continuidade ao estudo da aplicação de métodos numéricos via análise de condução transiente com geração de calor. Para isso, o seguinte problema foi proposto:

$$\alpha \nabla^2 T(x, y) + \frac{Q}{\rho c_v} = \frac{\partial T}{\partial t} \quad (5.10)$$

$$T(x, 0) = 0 \text{ e } T(x, 1) = 0, \quad \text{para } 0 \leq x \leq 2, \quad (5.11)$$

$$T(0, y) = 0 \text{ e } T(2, y) = 0, \quad \text{para } 0 \leq y \leq 1, \quad (5.12)$$

$$Q = 5W/m^3, \quad \text{para } 0,8 < x < 1,2 \text{ e } 0,4 < y < 0,6 \quad (5.13)$$

$$T(x, y)|_{t=0} = 0 \quad (5.14)$$

Além disso, será considerado $\kappa = 1 \text{ W/m.K}$, $c_v = 1 \text{ J/kg.K}$, $\rho = 1 \text{ kg/m}^3$ e $\alpha = \frac{\kappa}{\rho c_v} = 1 \text{ m/s}^2$ na análise. A solução será obtida por MDF e MEF, e os resultados serão comparados.

5.2.1 Modelagem MDF

Assim, como na subseção 5.1.1, foi adotado o Método de Diferenças Finitas Explícito, com aproximação adiantada para a derivada temporal e aproximação central para as derivadas espaciais. Ou seja, a equação do nó geral é dada por 5.7. Assim, implementando as condições de contorno e condição inicial estipuladas, é a equação 5.7 deve ser resolvida em cada passo temporal para que se obtenha a solução numérica do perfil de temperatura transiente na placa.

5.2.2 Modelagem MEF

Assim, como na subseção 5.1.2, foi adotado o Método de Elementos Finitos Galerkin com discretização temporal implícita. Além disso, optou-se por utilizar função interpoladora linear, elementos triangulares e espaçamento linear entre os pontos da malha. A equação representa a solução numérica pelo método supracitado para o problema proposto.

Tendo a equação 5.2.2, basta impor as condições iniciais e de contorno - conforme detalhado na seção 2.4 - e resolver o sistema linear, para cada passo temporal, a fim de se obter as soluções numéricas para o problema.

5.2.3 MDF x MEF

Inicialmente, optou-se por utilizar uma malha com $n_x = 36$ e $n_y = 18$. Para a solução por MEF, utilizou-se $\Delta t = 1e-4$; já para a solução por MDF, $\Delta t = 1e-6$. Os resultados obtidos foram registrados na tabela 5.4. A discrepância entre a temperatura média na placa pelos dois métodos foi calculada fazendo-se $|\frac{T_{med}(MDF) - T_{med}(MEF)}{T_{med}(MDF)}|$.

| Tempo [s] | Tmed (MDF) [10^{-4} °C] | Tmed (MEF) [10^{-4} °C] | Discrepância % |
|-----------|----------------------------|----------------------------|----------------|
| 0,01 | 7,7 | 1,9 | 76,0 |
| 0,02 | 13,1 | 3,7 | 71,9 |
| 0,05 | 19,2 | 8,5 | 55,9 |
| 0,075 | 20,3 | 11,5 | 43,2 |
| 0,1 | 20,5 | 13,8 | 32,6 |
| 0,2 | 20,6 | 18,6 | 10,0 |
| 0,5 | 20,6 | 20,5 | 0,7 |
| 1,0 | 20,6 | 20,5 | 0,5 |

Tabela 5.4: Temperatura média na placa para $n_x = 36$, $n_y = 18$, $dt = 1e-06$ (MDF) e $n_x = 36$, $n_y = 18$ e $dt = 1e-04$ (MEF)

Percebe-se que há uma grande discrepância entre os valor da temperatura média obtido na placa pelos diferentes métodos no regime transiente. A discrepância diminui conforme a placa aproxima-se do regime permanente. Uma estratégia para tentar reduzir essas discrepâncias é refinar as discretizações utilizadas nos métodos. Portanto, como uma segunda tentativa, optou-se por utilizar uma malha com $n_x = 44$, $n_y = 22$, $\Delta t = 1e-05$ para o MEF e $\Delta t = 1e-07$ para o MDF. Os resultados obtidos foram registrados na tabela 5.5.

| Tempo [s] | Tmed (MDF) [10^{-4} °C] | Tmed (MEF) [10^{-4} °C] | Discrepância % |
|-----------|----------------------------|----------------------------|----------------|
| 0,01 | 2,2 | 1,7 | 23,05 |
| 0,02 | 4,2 | 3,3 | 22,28 |
| 0,05 | 9,4 | 7,6 | 18,58 |
| 0,075 | 12,3 | 10,4 | 15,53 |
| 0,1 | 14,3 | 12,5 | 12,76 |
| 0,2 | 17,7 | 16,8 | 5,12 |
| 0,5 | 18,5 | 18,5 | 0,03 |
| 1,0 | 18,5 | 18,5 | 0,16 |

Tabela 5.5: Temperatura média na placa para $n_x = 44$, $n_y = 22$, $dt = 1e-07$ (MDF) e $n_x = 44$, $n_y = 22$ e $dt = 1e-05$ (MEF)

Com o refino da malha e do passo temporal, obteve-se um comportamento transiente menos discrepante entre os dois métodos empregados. No entanto, o tempo de execução do algoritmo aumentou consideravelmente. A tabela 5.7 reúne as informações de tempo de execução para os algoritmos implementados nesta seção.

Como simulação final, optou-se por utilizar uma malha ainda mais refinada, com $n_x = 48$, $n_y = 24$, $\Delta t = 1e-05$ para o MEF e $\Delta t = 1e-07$ para o MDF. Os resultados obtidos foram registrados na tabela 5.6.

| Tempo [s] | Tmed (MDF) [10^{-4} °C] | Tmed (MEF) [10^{-4} °C] | Discrepância % |
|-----------|----------------------------|----------------------------|----------------|
| 0,01 | 1,73 | 1,75 | 0,95 |
| 0,02 | 3,44 | 3,48 | 1,03 |
| 0,05 | 7,97 | 8,04 | 0,98 |
| 0,075 | 10,9 | 11,0 | 0,91 |
| 0,1 | 13,1 | 13,2 | 0,84 |
| 0,2 | 17,6 | 17,7 | 0,66 |
| 0,5 | 19,4 | 19,5 | 0,53 |
| 1,0 | 19,4 | 19,5 | 0,52 |

Tabela 5.6: Temperatura média na placa para $n_x = 48$, $n_y = 24$, $dt = 1e-08$ (MDF) e $n_x = 48$, $n_y = 24$ e $dt = 1e-05$ (MEF)

Percebe-se que o aumento no grau de refino resultou em discrepâncias bem reduzidas na temperatura média na placa entre os diferentes métodos. Essa convergência entre os métodos é um indicativo de que a implementação da solução numérica foi bem sucedida. No entanto, para se ter certeza de que os métodos foram bem implementados, o ideal seria comparar os resultados obtidos com a solução analítica para o problema proposto. Essa comparação com soluções analíticas está listada como sugestão de trabalhos futuros no capítulo ??.

Por fim, com o intuito de ilustrar o problema desenvolvido na seção, a figura 5.3 apresenta os desenvolvimento do perfil de temperatura na placa pelos dois métodos conforme a terceira proposta de solução numérica da seção (registrada pela tabela 5.6).

| | Δt | Qtd. Iterações | Tempo de Execução [s] | nx | ny |
|---------------|------------|----------------|-----------------------|----|----|
| MEF Implícito | 1e-4 | 1e4 | 6 | 36 | 18 |
| MDF Explícito | 1e-6 | 1e6 | 74 | 36 | 18 |
| MEF Implícito | 1e-5 | 1e5 | 2174 | 44 | 22 |
| MDF Explícito | 1e-7 | 1e7 | 27 | 44 | 22 |
| MEF Implícito | 1e-5 | 1e5 | 3967 | 48 | 24 |
| MDF Explícito | 1e-8 | 1e8 | 290 | 48 | 24 |

Tabela 5.7: Parâmetros do Algoritmo

Além disso, o critério demonstrado na equação 2.27 é aplicável à equação do nó geral desenvolvida para a resolução por MDF explícito, e precisa ser atendido para garantir que a implementação do método foi estável. Calculando $\frac{\alpha\Delta t}{\Delta x^2} + \frac{\alpha\Delta t}{\Delta y^2}$ para a primeira simulação via MDF Explícito do Problema 2 do presente capítulo, obtém-se $\frac{\alpha\Delta t}{\Delta x^2} + \frac{\alpha\Delta t}{\Delta y^2} = 5,95e-4$, que é menor que $\frac{1}{2}$. Ou seja, fica comprovada a estabilidade do algoritmo. Já para a segunda simulação via MDF Explícito do Problema 2, $\frac{\alpha\Delta t}{\Delta x^2} + \frac{\alpha\Delta t}{\Delta y^2} = 9,03e-5$, comprovando, também, a estabilidade. Para a terceira simulação via MDF Explícito, $\frac{\alpha\Delta t}{\Delta x^2} + \frac{\alpha\Delta t}{\Delta y^2} = 1,08e-5$, comprovando a estabilidade, também, para esse último caso.

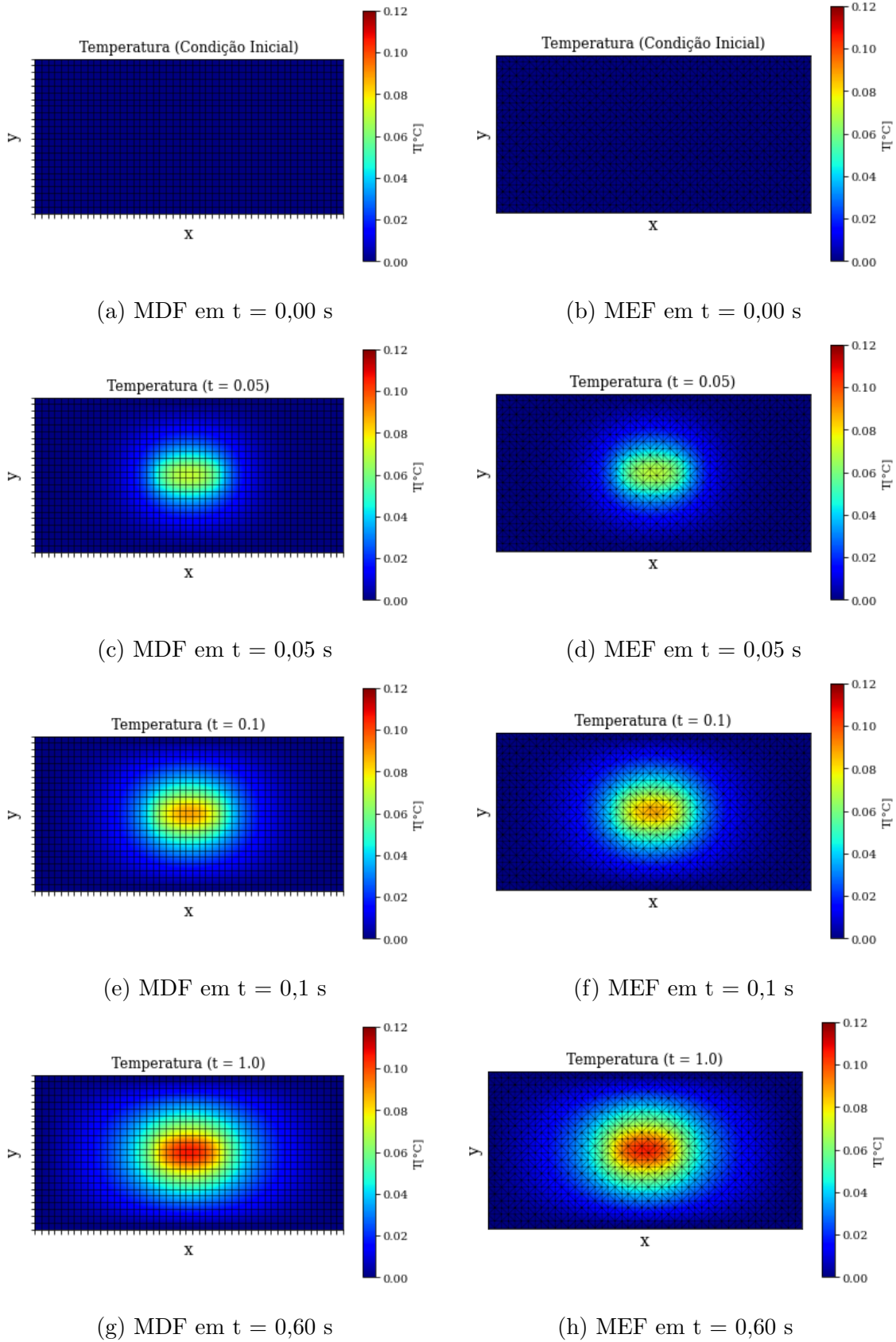


Figura 5.3: Comparação entre perfis de temperatura utilizando MDF e MEF para $n_x = 48$, $n_y = 24$, $\Delta t = 1e-05$ (MEF) e $\Delta t = 1e-08$ (MDF)

Capítulo 6

Conclusão

Ao longo desse trabalho, o Método de Diferenças Finitas e o Método de Elementos Finitos foram utilizados na resolução de diversos problemas térmicos bidimensionais. Os problemas abordados foram crescendo gradativamente em complexidade e a comparação entre os métodos entre si e a solução analítica serviu como um estudo sobre a aplicabilidade e as limitações dos métodos.

Ao longo dos capítulos, ficou evidenciado que a utilização de malhas mais refinadas (e passos temporais mais reduzidos, em problemas transientes) possibilita a obtenção de soluções mais acuradas. Além disso, é possível notar que, conforme o grau de complexidade do problema aumenta, mais necessário se torna tal refino na discretização do problema. No entanto, há um limite até onde esse refino é válido, uma vez que, a partir de certo ponto, aumenta-se o custo computacional sem acompanhamento do aumento de acurácia da solução obtida, conforme discutido nas seções 3.3 e 3.5.

O problema de maior complexidade, proposto no capítulo 5, foi o de condução transiente bidimensional com geração de calor. No capítulo, a solução analítica desse problema específico não foi abordada, impossibilitando uma análise ideal de convergência da malha com relação à solução teórica. No entanto, a aplicação de ambos os métodos numéricos com uma malha mais grosseira a princípio e mais refinada por fim serviu como indicativo de que ambos os métodos numéricos convergiram para um comportamento plausível de temperatura na placa.

Assim, por meio da utilização os dois métodos numéricos propostos, com variados graus de refino na discretização, a fim de analisar problemas térmicos de crescente grau de complexidade, foi possível ilustrar:

- Teoria e aplicação do MDF e MEF a problemas de condução de calor
- Influência da complexidade do problema sobre o grau de refinamento da malha
- Impacto dos parâmetros selecionados na discretização dos domínios sobre a acurácia e custo computacional das soluções numéricas
- Limitações na aplicação dos métodos

Por fim, vale lembrar que o presente trabalho se limitou à análise de problemas térmicos condutivos, bidimensionais, com condições de contorno de Dirichlet. O estudo pode ser continuado abordando a comparação entre os métodos numéricos para um elenco mais variado de problemas térmicos, por exemplo:

- Análise de problemas com condições de contorno de Neumann
- Análise de problemas com condições de contorno mistas
- Comparação dos métodos numéricos para problemas térmicos com convecção
- Aplicação e comparação dos métodos numéricos para problemas envolvendo três dimensões espaciais
- Aplicação e comparação dos métodos para problemas heterogêneos (propriedades físicas do material variando no domínio)
- Comparação dos resultados numéricos obtidos com resultados analíticos disponíveis na literatura
- Utilização de malhas não-uniformes na discretização espacial dos problemas, e análise dos impactos dessa utilização sobre a acurácia e o tempo de execução dos algoritmos
- Implementação de MEF Explícito e MDF Implícito, uma vez que este trabalho se limitou ao MEF Implícito e MDF Explícito em suas simulações

Essa análise de um elenco mais amplo de problemas térmicos possibilitaria reforçar as lições extraídas do presente trabalho, tornando o estudo comparativo ainda mais relevante. A adição de domínios variados ao estudo seria útil para evidenciar melhor quais as vantagens e limitações de um método numérico com relação a outro. Além disso, o acréscimo de simulações por MEF Explícito e MDF Implícito permitiria maior compreensão acerca da adequação de cada método no que diz respeito ao tempo de execução e acurácia obtidos.

Referências Bibliográficas

- [1] J. Fish and T. Belytschko, *A First Course in Finite Elements*.
- [2] G. dos Anjos Rabello, *Projeto Final - Mecânica dos Fluidos e Transmissão de Calor Computacional*.
- [3] M. J. Moran and H. N. Shapiro, *Fundamentals of Engineering Thermodynamics*.
- [4] F. P. Incropera, *Fundamentos da Transferência de Calor*.
- [5] M. N. Özışık, *Finite Difference Methods in Heat Transfer*.
- [6] G. dos Anjos Rabello, *Computação Científica para Engenheiros*. 2019.
- [7] W. E. Boyce and R. C. DiPrima, *Elementary differential equations and boundary value problems*.

Apêndice A

Algoritmos Implementados

Os códigos utilizados na implementação dos métodos, nos capítulos 3, 4 e 5 estão disponíveis no repositório do link a seguir: <https://github.com/rodrigoseefelder/ProjetoGraduacao>.