

UNIVERSIDADE FEDERAL DE JUIZ DE FORA  
INSTITUTO DE CIÊNCIAS EXATAS  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

# Uma Implementação Numérica Paralela para Simulação de Necrose de Células Tumerais por Hipertermia Local

Ruy Freitas Reis

JUIZ DE FORA  
AGOSTO, 2013

# Uma Implementação Numérica Paralela para Simulação de Necrose de Células Tumerais por Hipertermia Local

RUY FREITAS REIS

Universidade Federal de Juiz de Fora  
Instituto de Ciências Exatas  
Departamento de Ciência da Computação  
Bacharelado em Ciência da Computação

Orientador: Felipe dos Santos Loureiro

Coorientador: Marcelo Lobosco

JUIZ DE FORA

AGOSTO, 2013

UMA IMPLEMENTAÇÃO NUMÉRICA PARALELA PARA  
SIMULAÇÃO DE NECROSE DE CÉLULAS TUMORAIS POR  
HIPERTERMIA LOCAL

Ruy Freitas Reis

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

---

Felipe dos Santos Loureiro  
Doutor em Engenharia Civil (UFRJ)

---

Marcelo Lobosco  
Doutor em Engenharia de Sistemas e Computação (UFRJ)

---

Bernardo Martins Rocha  
Mestre em Modelagem Computacional (UFJF)

JUIZ DE FORA

16 DE AGOSTO, 2013

*Dedico este trabalho primeiramente a Deus,  
que me possibilitou a vida e conseqüências dela.  
A minha família que nunca deixou de me am-  
parar nesse processo evolutivo, e aos meus ami-  
gos que me acompanham durante essa cami-  
nhada.*

## Resumo

A hipertermia é amplamente utilizada no tratamento de câncer para destruir tumores. A ideia principal da hipertermia é aquecer uma região específica, como um tumor, de modo que acima de um limiar de temperatura as células tumorais sejam destruídas. Isto pode ser realizado por diversas técnicas de fornecimento de calor e a utilização de nanopartículas magnéticas, que geram calor quando um campo magnético alternado é aplicado, é atualmente uma técnica promissora. No presente trabalho, a equação de Pennes de bio-transferência de calor é adotada para modelar a ablação térmica de tumores, no contexto de nanopartículas magnéticas. O método das diferenças finitas explícito é empregado para resolver as equações. No entanto, uma grande quantidade de computação é necessária para este propósito. Portanto, este trabalho também apresenta uma primeira tentativa de melhorar o desempenho usando OpenMP, uma API de programação paralela. Os resultados experimentais foram bastante animadores: *speedups* de cerca de 35 foram obtidos em uma máquina de 64 *cores*.

**Palavras-chave:** Hipertermia, Biotranferência de Calor, Nanopartículas Magnéticas, OpenMP, Método das Diferenças Finitas.

## Abstract

Hyperthermia has been widely used in cancer treatment to destroy tumors. The main idea of the hyperthermia is to heat a specific region like a tumor so that above a threshold temperature the tumor cells are destroyed. This can be accomplished by many heat supply techniques and the use of magnetic nanoparticles that generate heat when an alternating magnetic field is applied has emerged as a promise technique. In the present paper, the Pennes bioheat transfer equation is adopted to model the thermal tumor ablation in the context of magnetic nanoparticles. Numerical simulations are carried out considering different injection sites for the nanoparticles in an attempt to achieve better hyperthermia conditions. Explicit finite difference method is employed to solve the equations. However, a large amount of computation is required for this purpose. Therefore, this work also presents an initial attempt to improve performance using OpenMP, a parallel programming API. Experimental results were quite encouraging: speedups around 35 were obtained on a 64-core machine.

**Keywords:** Bioheat, Hyperthermia, Magnetic Nanoparticles, OpenMP, Finite Difference Method

## Agradecimentos

Agradeço primeiramente a Deus, inteligência suprema, causa primária de todas as coisas, que me permitiu nascer e viver.

A minha família. Lane, por ser essa mãe maravilhosa que mesmo de longe sempre me acolheu e amou. José Robson, pai querido que nunca deixou de me apoiar nesta caminhada. Meu irmão Robson, o qual sempre me serviu de inspiração nesta caminhada e minha irmã Nágila, pelo apoio e carinho.

A todos os meus amigos queridos, os quais me acompanharam nesta caminhada. Em especial a: Aos irmãos Guilherme e Rafael e o amigo Leandro, irmãos que conheci, os quais nunca me negaram apoio, mesmo nas horas mais difíceis; ao “Bando” (Alessandra, Eduardo, Evelyn, Gustavo, João, Johnny e Joventino), amigos da faculdade, pelos melhores anos vividos nesta instituição, e que jamais me esquecerei.

Aos professores. Meu orientador, professor e amigo Felipe, que me confiou e possibilitou a criação deste projeto, além do inegável apoio durante esta caminhada. Meu coorientador, professor e amigo Marcelo Lobosco, pelo apoio profissional inegável, tanto como amigo como professor. E todos os demais professores que contribuíram para minha formação na primeira turma de engenharia computacional do Brasil.

*"Um homem precisa viajar. Por sua conta, não por meio de histórias, imagens, livros ou TV. Precisa viajar por si, com seus olhos e pés, para entender o que é seu. Para um dia plantar as suas árvores e dar-lhes valor. Conhecer o frio para desfrutar o calor. E o oposto. Sentir a distância e o desabrigo para estar bem sob o próprio teto. Um homem precisa viajar para lugares que não conhece para quebrar essa arrogância que nos faz ver o mundo como o imaginamos, e não simplesmente como é ou pode ser; que nos faz professores e doutores do que não vimos, quando deveríamos ser alunos, e simplesmente ir ver".*

*Amyr Klink (Mar sem fim)*



# Sumário

<b>Lista de Figuras</b>	<b>8</b>
<b>Lista de Tabelas</b>	<b>9</b>
<b>Lista de Abreviações</b>	<b>10</b>
<b>1 Introdução</b>	<b>11</b>
1.1 Apresentação do tema e contextualização do problema . . . . .	11
1.2 Justificativa . . . . .	12
1.3 Objetivos . . . . .	13
1.3.1 Objetivos Gerais . . . . .	13
1.3.2 Objetivos Específicos . . . . .	13
<b>2 Modelo Matemático</b>	<b>15</b>
2.1 Introdução . . . . .	15
2.2 Formulação Diferencial da Condução de Calor . . . . .	15
2.3 Biotransferência de Calor em Tecido Vivo . . . . .	19
2.3.1 Introdução . . . . .	19
2.3.2 Simplificações Assumidas . . . . .	19
2.3.3 Formulação da Equação de Pennes . . . . .	20
2.4 Hipertermia com Nanopartículas Magnéticas . . . . .	22
2.4.1 Introdução . . . . .	22
2.4.2 Formulação do Modelo . . . . .	22
<b>3 Método Numérico</b>	<b>25</b>
3.1 Introdução . . . . .	25
3.2 O Método das Diferenças Finitas em Meio Heterogêneo . . . . .	25
3.2.1 Predefinições do Modelo . . . . .	25
3.2.2 Condutividade Térmica em Meio Heterogêneo . . . . .	26
3.2.3 Formulação de Diferenças Finitas Explícita em Meio Heterogêneo para o Modelo . . . . .	28
3.2.4 Formulação das Condições de Contorno . . . . .	31
<b>4 Paralelização</b>	<b>33</b>
4.1 Contextualização Teórica . . . . .	33
4.2 Paralelização do Algoritmo . . . . .	34
<b>5 Resultados</b>	<b>38</b>
5.1 Introdução . . . . .	38
5.1.1 Definição do Domínio . . . . .	38
5.1.2 Parâmetros das Simulações . . . . .	39
5.1.3 Condições Inicial e de Contorno . . . . .	39
5.2 Resultados do Modelo . . . . .	40
5.3 Resultados da Paralelização . . . . .	43

<b>6</b>	<b>Conclusões</b>	<b>46</b>
6.1	Modelo Matemático e Método Numérico . . . . .	46
6.2	Paralelização . . . . .	46
6.3	Trabalhos Futuros . . . . .	47
	<b>Referências Bibliográficas</b>	<b>49</b>
<b>A</b>	<b>Fórmulas de Diferenças Finitas</b>	<b>51</b>
<b>B</b>	<b>Série de Taylor</b>	<b>52</b>

## Lista de Figuras

2.1	Elemento infinitesimal. Extraído de JIJI (2009) . . . . .	16
2.2	Fluxos nos elementos infinitesimais no plano 2D. Extraído de JIJI (2009) .	17
2.3	Capilares. Adaptado de (JIJI, 2009) . . . . .	21
2.4	Distribuição de temperatura com um ponto de injeção. Adaptado de SALLOUM <i>et al.</i> (2009) . . . . .	23
3.1	Pontos $i$ e $i + 1$ em um meio heterogêneo com o ponto fictício $i + \frac{1}{2}$ . . . . .	27
3.2	<i>Stencil</i> de segunda ordem no espaço e primeira ordem no tempo . . . . .	28
3.3	Meio heterogêneo com os pontos fictícios $i + \frac{1}{2}$ e $i - \frac{1}{2}$ . . . . .	29
3.4	Domínio discretizado com o ponto fantasma . . . . .	32
4.1	Sistema de memória compartilhada. Adaptado de PACHECO (2011) . . . .	33
4.2	Sistema de memória compartilhada <i>multicore</i> . Adaptado de PACHECO (2011) . . . . .	34
4.3	Processo <i>multithread</i> . Adaptado de PACHECO (2011) . . . . .	34
4.4	<i>Fork</i> e <i>join</i> entre as <i>threads</i> com a diretiva <i>omp parallel</i> . Adaptado de PACHECO (2011) . . . . .	35
4.5	Escalonamento estático com 4 <i>threads</i> . . . . .	37
5.1	Domínio da solução da equação com tumor. . . . .	38
5.2	Contorno do domínio. . . . .	40
5.3	Evolução da temperatura no tempo, dentro e fora do tumor em ambos os casos. . . . .	41
5.4	Distribuição da temperatura no domínio, quando utilizado somente um ponto de injeção. . . . .	41
5.5	Distribuição da temperatura no domínio, quando utilizado quatro pontos de injeção. . . . .	42
5.6	Tecido morto destacado em amarelo, quando utilizado somente um ponto de injeção. . . . .	42
5.7	Tecido morto destacado em amarelo, quando utilizado quatro pontos de injeção. . . . .	43
5.8	Compartilhamento de recursos do Chip AMD Opteron Série 6200 Interlagos. Adaptado de Numerical Algorithms Group (2011). . . . .	44
5.9	<i>Speedup</i> de execução do código nesta arquitetura. . . . .	45

## Lista de Tabelas

5.1	Parâmetros do modelo. . . . .	39
5.2	Parâmetros da aplicação do tratamento com hipertermia. . . . .	39
A.1	Fórmulas de Diferenças Finitas . . . . .	51

## Lista de Abreviações

API	<i>Application Programming Interface</i>
CPU	<i>Central Processing Unit</i>
CUDA	<i>Compute Unified Device Architecture</i>
DCC	Departamento de Ciência da Computação
FP	<i>Float Point</i>
FPU	<i>Float Point Unit</i>
GPU	<i>Graphic Process Unit</i>
Int	<i>Integer</i>
MDF	Método das Diferenças Finitas
MEC	Método dos Elementos de Contorno
MEF	Método dos Elementos Finitos
MVF	Método dos Volumes Finitos
MPI	<i>Message Passing Interface</i>
NUMA	<i>Non-Uniform Memory Access</i>
OpenMP	<i>Open Multi-Processing</i>
RAM	<i>Random Access Memory</i>
SAR	<i>Specific Absorption Rate</i>
UFJF	Universidade Federal de Juiz de Fora
UMA	<i>Uniform Memory Access</i>

# 1 Introdução

## 1.1 Apresentação do tema e contextualização do problema

Atualmente a sociedade enfrenta diversos problemas, alguns já parcialmente solucionados, mas que ainda tem muito a se explorar. Dentre os problemas enfrentados, existe a modelagem térmica. Neste caso pode-se destacar diversos problemas como aquecimento global, previsão do tempo, termodinâmica e no caso deste trabalho a biotransferência de calor.

A aplicação de métodos computacionais na modelagem de sistemas biológicos é alvo de interesse de várias áreas do conhecimento, dentre elas a física, matemática e engenharias. Este interesse é impulsionado, também, devido ao rápido avanço tecnológico da computação. No caso da modelagem térmica pode-se utilizar modelos matemáticos os quais serão utilizados para gerar distribuições de temperatura em um dado domínio. Esta solução pode ser utilizada, por exemplo, para fazer diagnósticos médicos (MIYAKAWA e BOLOMEY, 1996; LIU e XU, 2002). Outra aplicação de simulações com modelos de biotransferência de calor é no tratamento de tumor através de hipertermia (SALLOUM *et al.*, 2008, 2009).

A temperatura da superfície corporal é controlada pela circulação sanguínea abaixo da pele, metabolismo, e a troca de calor entre a pele e o ambiente. Mudanças nestes parâmetros podem induzir mudança no fluxo de calor. Um tumor superficial super vascularizado pode causar aumento na temperatura na pele como, por exemplo, a temperatura sobre um tumor de mama é maior que em sua periferia. O alto metabolismo causado por inflamações também podem induzir a elevação da temperatura da pele. Por outro lado trombozes, ou escleroses vasculares na circulação periférica, diminuem o fluxo sanguíneo resultando em uma hipotermia na pele. Portanto, mudanças na temperatura ou no fluxo de calor da pele podem indicar circulação periférica irregular, podendo ser

utilizado para diagnósticos médicos (LIU e XU, 2002; CAO *et al.*, 2010).

Técnicas de medição da temperatura na superfície corporal requerem soluções generalizadas de equações de biotransferência de calor sob inúmeras condições de contorno e constantes térmicas, para simulação de casos reais. Comparado com outros métodos não-invasivos de termometria como ressonância magnética, micro-ondas, e ultrassom, métodos térmicos aparentemente são mais econômicos e seguros. Pesquisas a respeito de previsões de temperatura em tecidos vivos estão em constante desenvolvimento desde a proposição da equação de Pennes (PENNES, 1948). Métodos numéricos de resolução de equações diferenciais são utilizados para a resolução destas equações, dentre eles o método das diferenças finitas (MDF).

## 1.2 Justificativa

Baseando-se nas informações supracitadas, atualmente dentre os métodos utilizados em uma variedade de procedimentos para o tratamento contra o câncer, ou mesmo tratamento auxiliar a drogas e radiação, pode-se destacar a hipertermia. Hipertermia com nanopartículas magnéticas foi desenvolvido nos anos 50 e essa técnica consegue atingir até mesmo tumores situados em localizações profundas e/ou irregulares (GILCHRIST *et al.*, 1957).

Normalmente, a geração de calor por procedimentos com hipertermia é dado por uma taxa de absorção (SAR), adicionada a equação de biotransferência de calor usada. A distribuição do SAR e seus respectivos parâmetros podem ser encontrados em SALLOUM *et al.* (2008), por exemplo.

O objetivo deste estudo é analisar diferentes modos de aplicar as nanopartículas magnéticas usando a temperatura relativa do tecido e levando em consideração a temperatura estacionária do tumor.

Além disso, devido a grande intensidade de operações aritméticas em ponto flutuante exigida pelo método numérico, principalmente para modelos tridimensionais, adotou-se um esquema de paralelização usando a API (*application programming interface* - interface de programação de aplicativos) OpenMP, inicialmente, para reduzir o tempo da simulação *in silico*. A API OpenMP foi utilizada para o modelo bidimensional do código.

## 1.3 Objetivos

### 1.3.1 Objetivos Gerais

O objetivo principal deste trabalho é a geração de curvas de distribuição de temperatura através de simulações numéricas de tratamentos por hipertermia. Devido a biocompatibilidade do corpo humano com o óxido ferromagnético  $Fe_3O_4$  e magnetita  $\gamma-Fe_2O_3$  essas são as nanopartículas mais estudadas para este tipo de tratamento (MOROZ, 2002).

O estudo de transferência de calor no estado transiente é muito importante, possibilitando a análise e validação de possíveis tratamentos, principalmente aqueles que utilizam hipertermia para eliminar o câncer por necrose celular (SALLOUM *et al.*, 2009).

### 1.3.2 Objetivos Específicos

Este trabalho irá descrever o método numérico que será utilizado para resolver a equação de biotransferência de calor. Desta forma este trabalho apresenta um estudo do método das diferenças finitas em meios heterogêneos em paralelo. Este será o objetivo específico deste trabalho, bem como a utilização deste no âmbito da biotransferência de calor, neste caso simular o tratamento por hipertermia.

Além do aspecto numérico e do modelo que será utilizado neste trabalho, deve-se preocupar também com o lado computacional. A linguagem escolhida para o desenvolvimento desta pesquisa é o C++. A justificativa desta escolha é que ela possui a robustez da linguagem C, porém com a vantagem de poder utilizar a orientação a objeto. Além do mais, esta linguagem possui integração com API de programação paralela, *e. g.* OpenMP API, MPI e CUDA. Por ser também uma linguagem de médio nível permite otimizar detalhes aproveitando da arquitetura da máquina.

Após a implementação do modelo de biotransferência de calor, utilizando a técnica e método especificado, é feita uma comparação de duas maneiras de aplicar as nanopartículas e, então, analisar a influência que o modo de aplicação pode causar.

Outro fator importante é a técnica utilizada na paralelização. Primeiramente, será apresentado a arquitetura onde foi executada as simulações, bem como, a ideia utilizada na paralelização e a interferência que a arquitetura da máquina provocou no desempenho



do código.

## 2 Modelo Matemático

### 2.1 Introdução

No âmbito da biologia, a determinação da distribuição da temperatura corporal é alvo de estudos para o auxílio em inúmeros tratamentos médicos, principalmente em seres humanos. Dentre eles vale destacar a detecção de tumor baseado na temperatura corporal (LIU e XU, 2002), criocirurgia (THAOKAR e RABIN, 2012), queimaduras de tecidos de pele (ABRAHAM *et al.*, 2011), e no caso deste texto a hipertermia. O maior desafio desta etapa da modelagem computacional é a escolha/construção do modelo que melhor se encaixa no alvo de estudo, que neste caso é a biotransferência de calor.

No decorrer do tempo alguns modelos matemáticos foram elaborados para descrever a biotransferência de calor, e dentre eles pode-se citar o modelo elaborado por H. H. Pennes, o qual ficou conhecido por equação de Pennes (PENNES, 1948). Neste texto será feito um breve detalhamento da formulação, limitações e aplicabilidade deste modelo.

A ideia principal do tratamento por hipertermia é o aquecimento de uma determinada região, neste caso um tumor, até um limiar de temperatura na qual a célula é destruída por necrose.

### 2.2 Formulação Diferencial da Condução de Calor

Conforme dito anteriormente, o estudo da distribuição de temperatura em uma dada região tem grande importância não somente na área da saúde, mas também em inúmeras áreas de conhecimento, principalmente na engenharia, *e. g.* determinação de previsão do tempo (VU *et al.*, 2013), resfriamento de caldeiras e aquecimento global (BUSCH *et al.*, 2012).

Deve-se, inicialmente, determinar a distribuição de temperatura em uma região qualquer. Para isso, deve-se aplicar o princípio da conservação de energia em um certo volume de controle para se obter as equações diferenciais que regem o fenômeno físico

de interesse. Desta forma, destacando-se um elemento infinitesimal  $dV = dx dy dz$  em coordenadas cartesianas (JIJI, 2009):

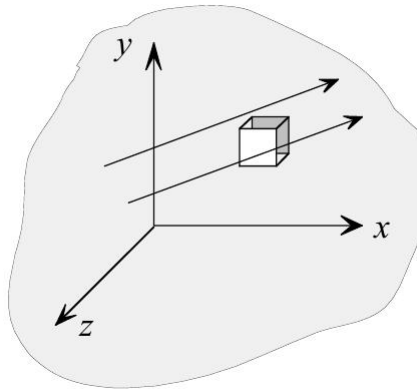


Figura 2.1: Elemento infinitesimal. Extraído de JIJI (2009)

Aplicando o princípio da conservação de energia (primeira lei da termodinâmica) no elemento infinitesimal da Figura 2.1, obtém-se a seguinte equação:

$$\dot{E}_{in} + \dot{E}_g - \dot{E}_{out} = \dot{E} \quad (2.1)$$

onde *Taxa de Energia Adicionada*, *Taxa de Energia Gerada*, *Taxa de Energia Removida* e *Taxa de Energia Alterada no Elemento* são  $\dot{E}_{in}$ ,  $\dot{E}_g$ ,  $\dot{E}_{out}$  e  $\dot{E}$ , respectivamente. Para simplificar a formulação deve-se assumir:

1. Velocidade uniforme;
2. Pressão constante;
3. Densidade constante;
4. Variação de energia potencial desprezível

A energia é trocada com o elemento por condução e transferência de massa. Os dois modos de transferência de energia são mostrados na Figura 2.2. Nesta figura não foi considerada a coordenada z, a qual pode ser analogamente incorporada na formulação. A energia entra no elemento por condução com um fluxo  $q_x$ ,  $q_y$  e  $q_z$ , nas direções x, y e z, respectivamente. Baseado-se que o fluxo representa a energia por unidade de área e por unidade de tempo, o fluxo deve ser multiplicado pela área normal a ele. A energia

também entra no elemento através de fluxo de massa. A taxa do fluxo de massa que entra no elemento na direção  $x$  é  $\rho U dy dz$ , onde  $\rho$  é a densidade e  $U$  a velocidade do componente na direção  $x$ . A taxa de energia carregada pela massa é  $\rho U \hat{h} dy dz$ , onde  $\hat{h}$  é a entalpia por unidade de massa. Os componente correspondentes nas direções  $y$  e  $z$  são  $\rho V \hat{h} dx dz$  e  $\rho W \hat{h} dx dy$ , onde  $V$  e  $W$  são as componentes da velocidade nas direções  $y$  e  $z$  respectivamente. Com isso pode-se expressar  $\dot{E}_{in}$  com a equação:

$$\dot{E}_{in} = q_x dy dz + q_y dx dz + q_z dx dy + \rho U \hat{h} dy dz + \rho V \hat{h} dx dz + \rho W \hat{h} dx dy \quad (2.2)$$

onde  $\vec{q}$  é o fluxo por unidade de volume.

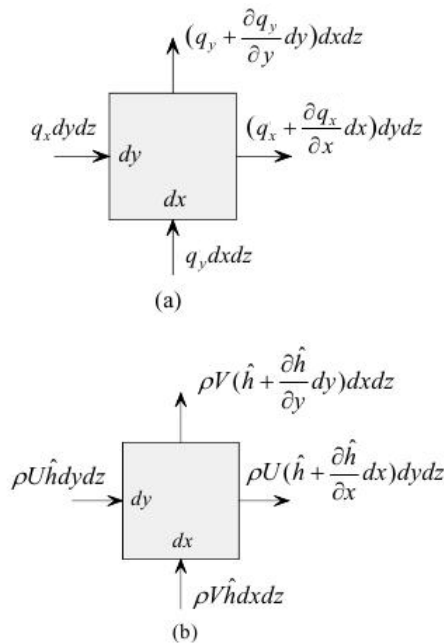


Figura 2.2: Fluxos nos elementos infinitesimais no plano 2D. Extraído de JIJI (2009)

A energia gerada  $\dot{E}_g$  é dada pela equação:

$$\dot{E}_g = q dx dy dz \quad (2.3)$$

A energia que deixa o elemento é construída usando expansão por série de Taylor (consultar apêndice B). A formulação é dada pela equação (note que  $U$ ,  $V$  e  $W$  são constantes):

$$\begin{aligned} \dot{E}_{out} = & (q_x + \frac{\partial q_x}{\partial x} dx) dydz + (q_y + \frac{\partial q_y}{\partial y} dy) dx dz + (q_z + \frac{\partial q_z}{\partial z} dz) dx dy + \\ & \rho U (\hat{h} + \frac{\partial \hat{h}}{\partial x} dx) dydz + \rho V (\hat{h} + \frac{\partial \hat{h}}{\partial y} dy) dx dz + \rho W (\hat{h} + \frac{\partial \hat{h}}{\partial z} dz) dx dy \end{aligned} \quad (2.4)$$

A energia trocada dentro do elemento  $\dot{E}$  resulta em:

$$\dot{E} = \rho \frac{\partial \hat{u}}{\partial t} dx dy dz \quad (2.5)$$

onde  $\hat{u}$  é a energia interna por unidade de massa e  $t$  é o tempo. Substituindo 2.2 - 2.5 na equação 2.1, obtém-se a equação:

$$-\frac{\partial q_x}{\partial x} - \frac{\partial q_y}{\partial y} - \frac{\partial q_z}{\partial z} - \rho U \frac{\partial \hat{h}}{\partial x} - \rho V \frac{\partial \hat{h}}{\partial y} - \rho W \frac{\partial \hat{h}}{\partial z} + q = \rho \frac{\partial \hat{u}}{\partial t} \quad (2.6)$$

A entalpia  $\hat{h}$  é definida pela equação:

$$\hat{h} = \hat{u} + \frac{P}{\rho} \quad (2.7)$$

onde  $P$  é a pressão, assumida como constante. Substituindo  $\hat{u} = \hat{h} - \frac{P}{\rho}$  em 2.6 e rearranjando os termos, obtém-se a equação:

$$-\frac{\partial q_x}{\partial x} - \frac{\partial q_y}{\partial y} - \frac{\partial q_z}{\partial z} + q = \rho \left( \frac{\partial \hat{h}}{\partial t} + U \frac{\partial \hat{h}}{\partial x} + V \frac{\partial \hat{h}}{\partial y} + W \frac{\partial \hat{h}}{\partial z} \right) \quad (2.8)$$

O próximo passo é expressar o fluxo e a entalpia em termos da temperatura  $T$ . A lei de Fourier de condução, dada pela Equação 2.9, relaciona o fluxo de calor com o gradiente da temperatura:

$$q_x = -k \frac{\partial T}{\partial x} \quad q_y = -k \frac{\partial T}{\partial y} \quad q_z = -k \frac{\partial T}{\partial z} \quad (2.9)$$

onde  $k$  é a condutividade térmica.

A mudança da entalpia em pressão constante é dada pela equação:

$$d\hat{h} = c_p dT \quad (2.10)$$

onde  $c_p$  é o calor específico sob pressão constante. Substituindo a Equação 2.9 e 2.10 na Equação 2.8 resulta em:

$$\frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left( k \frac{\partial T}{\partial z} \right) + q = \rho c_p \left( \frac{\partial T}{\partial t} + U \frac{\partial T}{\partial x} + V \frac{\partial T}{\partial y} + W \frac{\partial T}{\partial z} \right) \quad (2.11)$$

Desconsiderando o termo convectivo na a Equação 2.11, obtém-se a seguinte equação:

$$\nabla \cdot k \nabla T + q = \frac{\partial T}{\partial t} \rho c_p \quad (2.12)$$

onde  $\nabla T$  é o gradiente de temperatura e  $\nabla \cdot ()$  é o operador divergente.

## 2.3 Biotransferência de Calor em Tecido Vivo

### 2.3.1 Introdução

A complexa natureza da transferência de calor em tecido vivo impede a modelagem matemática exata. Algumas suposições e simplificações devem ser adotadas para tratar deste problema, porém capturando as características essenciais. Segue uma resumida formulação de uma equação que determina a distribuição de temperatura em tecido vivo. Essa modelagem se baseou na perfusão sanguínea, propriedades térmicas e simplificação da geometria vascular.

### 2.3.2 Simplificações Assumidas

A equação de Pennes é baseada principalmente em quatro simplificações. São elas:

1. **Equilíbrio Local:** A principal troca de calor entre sangue e o tecido acontece entre os vasos capilares e arteríolas de fornecimento de sangue aos capilares e as vênulas de drenagem. Então, toda a transferência de calor entre as pre-arteríolas e pós-vênulas

e os tecidos são desconsideradas.

2. **Perfusão Sanguínea:** O fluxo sanguíneo nos pequenos capilares é assumido como sendo isotrópico. Conseqüentemente isso desconsidera o efeito da direção do fluxo sanguíneo.
3. **Arquitetura Vascular:** As grandes veias sanguíneas na vizinhança dos vasos capilares não desempenham nenhum papel de troca de energia entre tecido e sangue capilar. Conseqüentemente este modelo não considera a geometria vascular local.
4. **Temperatura do Sangue:** Assumi-se que o sangue o qual alcança as arteríolas suprindo os vasos capilares na temperatura interna do corpo  $T_a$ . Isso instantaneamente troca calor e equilibra com a temperatura  $T$  do tecido local. Baseado nestas suposições, Pennes modela o efeito sanguíneo como sendo uma fonte ou sorvedouro de calor isotrópica o qual é proporcional a taxa de fluxo sanguíneo e na diferença de temperatura entre a temperatura central do corpo  $T_a$  e o tecido local  $T$ . Neste modelo, o sangue originalmente submetido a uma temperatura  $T_a$  não perde ou ganha energia conforme circula nos vasos sanguíneos do corpo.

### 2.3.3 Formulação da Equação de Pennes

O elemento de tecido mostrado na Figura 2.3, é grande o suficiente para ser saturado com arteríolas, vênulas e capilares, mas pequeno se comparado com a dimensão da região considerada.

A taxa de energia adicionada ao elemento é devido a condução e convecção (transferência de massa). Aqui o componente convectivo é eliminado e substituído pela energia adicionada devido a perfusão sanguínea. O modo mais simples de considerar este efeito é trata-lo como energia gerada  $\dot{E}_g$ . Chamam-se  $q_b$  e  $q_m$  a taxa de energia adicionada ao sangue por unidade de volume do tecido e a taxa de energia metabólica produzida por unidade de volume do tecido, respectivamente. Então a Equação 2.3 se torna:

$$q_b = \rho_b c_b \dot{\omega}_b (T_a - T(\vec{x}, t)) \quad (2.13)$$

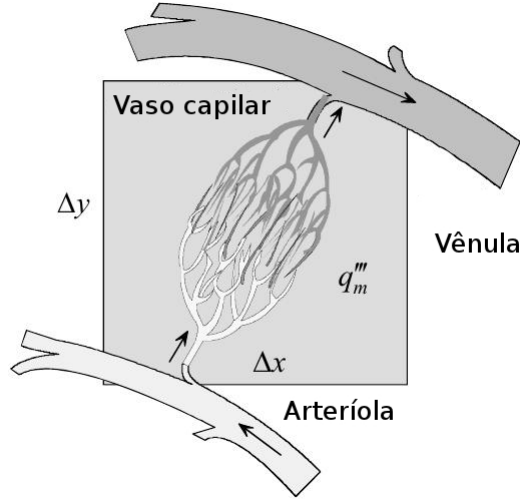


Figura 2.3: Capilares. Adaptado de (JILI, 2009)

onde  $c_b$ ,  $\dot{\omega}_b$  e  $\rho_b$  são o calor específico do sangue, taxa de fluxo sanguíneo volumétrico por unidade de volume de tecido e densidade do sangue, respectivamente.

Substituindo a Equação 2.13 em 2.11, obtém-se a equação:

$$q = q_m + \rho_b c_b \dot{\omega}_b (T_a - T(\vec{x}, t)) \quad (2.14)$$

Retornando à equação de condução de calor 2.12 e usando-se a Equação 2.14, obtém-se a seguinte equação, conhecida por equação de Pennes:

$$\nabla \cdot k \nabla T(\vec{x}, t) + \rho_b c_b \dot{\omega}_b (T_a - T(\vec{x}, t)) + q_m = \rho c \frac{\partial T(\vec{x}, t)}{\partial t} \quad (2.15)$$

onde  $c$ ,  $k$  e  $\rho$  são o calor específico do tecido, condutividade térmica do tecido e densidade do tecido, respectivamente.

Tomando  $t \rightarrow \infty$ , pode-se obter a equação no estado estacionário, dada por:

$$\nabla \cdot k \nabla T(\vec{x}) + \rho_b c_b \dot{\omega}_b (T_a - T(\vec{x})) + q_m = 0 \quad (2.16)$$



## 2.4 Hipertermia com Nanopartículas Magnéticas

### 2.4.1 Introdução

Hipertermia é atualmente muito utilizada como método de terapia não invasiva em uma variedade de procedimentos em tratamentos contra o câncer, ajudando assim outras técnicas como drogas e radioterapia. Hipertermia com nanopartículas magnéticas foi um método inicialmente desenvolvido nos anos 50 (GILCHRIST *et al.*, 1957), e ainda está no estágio inicial do desenvolvimento para tratamento em câncer. Esta técnica é baseada em nanopartículas magnéticas que aquecem tecidos tumorais irregulares e/ou profundos quando expostas a um campo magnético de baixa frequência. O calor produzido por estas partículas, quando expostas a um campo magnético alternado, é gerado, principalmente, por relaxação de Néel e/ou movimento Browniano das mesmas (ROSENSWEIG, 2002). Devido a biocompatibilidade ao tecido humano, óxido ferromagnético  $Fe_3O_4$  e magnetita  $\gamma - Fe_2O_3$  são as nanopartículas mais estudadas (MOROZ, 2002).

Esta abordagem é muito utilizada no tratamento de tumores do fígado (MATSUKI *et al.*, 1994) e mama (HILGER *et al.*, 2005). Há estudos clínicos realizados por Johannsen e Jordan (JOHANNSEN *et al.*, 2005) que abordam testes de hipertermia com fluido magnético em câncer de próstata. Baseados em análises histológicas dos tecidos cancerígenos que mostram uma necrose parcial das células após este tipo de tratamento. Portanto o sucesso deste tratamento é elevar a temperatura acima de  $43^\circ C$  no tumor inteiro com o mínimo de dano às células da vizinhança saudável, provocando assim a necrose celular. Por outro lado o estudo feito em Kawai *et al.* (2005) mostra que tratamentos repetidos com hipertermia matam células com câncer não somente por aquecimento, mas também por resposta imune induzida.

### 2.4.2 Formulação do Modelo

A distribuição de temperatura é determinada pela equação de condução de calor no tecido (Equação 2.15) e uma distribuição do calor gerado  $q_r(\vec{x}, t)$ . Geralmente a distribuição de temperatura gerada em um processo de hipertermia é modelada por uma taxa de absorção específica (do inglês *specific absorption rate* ou SAR)(SALLOUM *et al.*, 2008). Para o

tipo de partícula e força de campo magnético utilizado neste tipo de procedimento, a distribuição espacial das partículas dispersas no tecido é dado pela distribuição de SAR.

Um recente estudo experimental, executado em membros posteriores de ratos (SALLOUM *et al.*, 2008), mostrou que a taxa de absorção em volta do ponto de injeção pode ser aproximado por uma curva de distribuição gaussiana:

$$SAR = Ae^{\frac{-r^2}{r_0^2}} \quad (2.17)$$

onde  $r$  é a distância espacial do ponto de injeção,  $A$  a amplitude máxima da geração de calor e  $r_0$  o limite que a geração de calor pode afetar o tecido.

A distribuição da taxa de absorção foi determinada baseando-se nas medidas de elevação de temperatura nos membros posteriores do rato. O conhecimento da expressão 2.17 permite determinar a distribuição de temperatura na vizinhança do ponto, baseado em um valor inicial do fluido ferromagnético injetado. Conforme será mostrado neste texto, em um caso de somente um ponto de injeção, os danos térmicos terão forma de uma esfera em 3D, conforme pode ser observado na Figura 2.4, ou em forma de uma circunferência no plano 2D. O grande desafio deste tipo de procedimento é qual a melhor forma de aplicar os pontos de injeção de forma a ter o máximo de dano no tumor e o mínimo no tecido saudável.

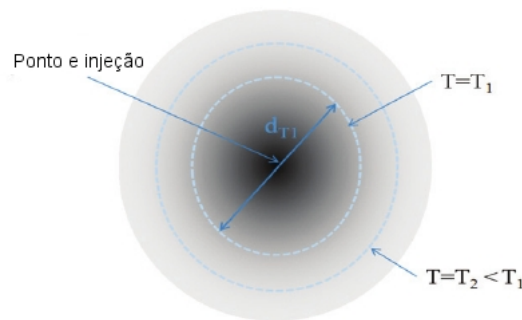


Figura 2.4: Distribuição de temperatura com um ponto de injeção. Adaptado de SALLOUM *et al.* (2009)

Munido destas informações, pode-se evoluir o modelo matemático que prevê a distribuição de temperatura em um tecido vivo com a seguinte equação:

$$\nabla \cdot k \nabla T(\vec{x}, t) + \rho_b c_b \dot{\omega}_b (T_a - T(\vec{x}, t)) + q_m + \sum_{i=1}^M A_i e^{\frac{-r_i^2}{r_{0,i}^2}} = \rho c \frac{\partial T(\vec{x}, t)}{\partial t} \quad (2.18)$$

onde  $M$  é o número de pontos de injeção.

## 3 Método Numérico

### 3.1 Introdução

A solução de equações diferenciais parciais ainda é um grande desafio para a ciência e engenharias. Fundamentando-se no fato que a maioria das equações diferenciais parciais não possui solução analítica, a proposição de métodos numéricos capazes de solucionar estas equações é tema de grandes linhas de pesquisa, principalmente para profissionais da área de modelagem matemática e computacional. Este trabalho utiliza o método das diferenças finitas, o qual já é consagrado pela robustez em solucionar vários tipos de equações diferenciais. Esta técnica numérica é utilizada para solucionar a equação proposta por Pennes (PENNES, 1948), com algumas alterações para realizar simulações *in silico* de tratamentos por hipertermia com nanopartículas magnéticas em tumores.

### 3.2 O Método das Diferenças Finitas em Meio Heterogêneo

#### 3.2.1 Predefinições do Modelo

Baseando-se na Equação 2.18 desenvolvida no capítulo anterior, pode-se fazer algumas suposições sobre as condições de contorno e inicial.

$$\begin{cases} \rho c \frac{\partial T_1(\vec{x}, t)}{\partial t} = \nabla \cdot k \nabla T_1(\vec{x}, t) + \dot{\omega}_b \rho_b c_b (T_a - T_1(\vec{x}, t)) + q_m(\vec{x}) + q_r(\vec{x}, t) & \text{em } \Omega, t > 0 \\ \alpha T_1(\vec{x}, t) + \beta \nabla T_1(\vec{x}, t) \cdot \vec{n} = \hat{f}(\vec{x}, t) & \text{em } \Gamma, t > 0 \\ T_1(\vec{x}, 0) = T_2(\vec{x}) & \text{em } \Omega \end{cases} \quad (3.1)$$

onde  $\Omega$  é o domínio,  $\Gamma$  é o contorno do domínio,  $\alpha$  e  $\beta$  são funções que definem a forma com que será a condição de contorno, conforme será melhor detalhado subseção 3.2.4,  $\vec{n}$  é um vetor normal ao contorno  $\Gamma$ ,  $q_r(\vec{x}, t)$  é o calor imposto pela hipertermia com nanopartículas

magnéticas que será dado pela Equação 2.17 e a temperatura inicial  $T_2(\vec{x})$  em todo tecido pode ser calculada pela solução da Equação 2.16 com  $q_r = 0$ . Basicamente a utilização da temperatura  $T_2(\vec{x})$  como temperatura inicial, ao invés da temperatura padrão do corpo humano de  $37^\circ \text{C}$ , é justificada devido a elevação de temperatura que ocorre em um tecido com a presença de câncer, conforme é mostrado por LIU e XU (2002).

Uma vez que a temperatura  $T_2(\vec{x})$  é conhecida, pode-se seguir para resolução do modelo, porém, para facilitar a análise da solução, usar-se-á a temperatura relativa dada por  $T(\vec{x}, t) = T_1(\vec{x}, t) - T_2(\vec{x})$ , e desta forma, obtém-se a seguinte equação:

$$\begin{cases} \rho c \frac{\partial T(\vec{x}, t)}{\partial t} = \nabla \cdot k \nabla T(\vec{x}, t) - \dot{\omega}_b \rho_b c_b T(\vec{x}, t) + q_r(\vec{x}, t) & \text{em } \Omega, t > 0 \\ \alpha T(\vec{x}, t) + \beta \nabla T(\vec{x}, t) \cdot \vec{n} = f(\vec{x}, t) & \text{em } \Gamma, t > 0 \\ T(\vec{x}, 0) = 0 & \text{em } \Omega \end{cases} \quad (3.2)$$

### 3.2.2 Condutividade Térmica em Meio Heterogêneo

Considere um domínio discretizado em pontos  $x_i$  dados por:

$$x_i = x_0 + ih_x \quad (3.3)$$

onde  $h_x = \frac{x_b - x_a}{n_x}$ ,  $x_a$  é o ponto inicial,  $x_b$  o ponto final,  $n_x$  o número de divisões do domínio. Isto para um domínio unidimensional, mas que analogamente pode ser expandido para dimensões superiores.

Na Equação 3.2 considere que a constante  $k$  agora seja uma função descontínua  $k : \mathbb{R}^n \rightarrow \mathbb{R}^+$ . Deste modo deve-se fazer algumas considerações para encontrar um valor de  $k$  efetivo, de forma a garantir a continuidade do fluxo  $\vec{q}$

Considere o fluxo  $q_x$  dado por:

$$q_x = -k \frac{\partial T}{\partial x} \quad (3.4)$$

Conforme é mostrado na Figura 3.1 e considerando  $q_i$  o fluxo no ponto  $x_i$ , pode-se considerar um ponto fictício  $i + \frac{1}{2}$  e os fluxos  $q_{i+\frac{1}{2}}^-$  e  $q_{i+\frac{1}{2}}^+$  serão aproximados pela diferença

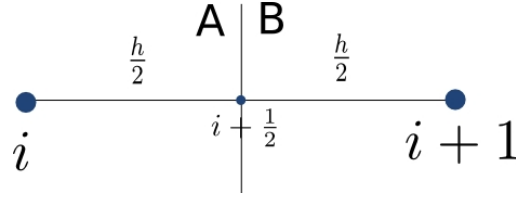


Figura 3.1: Pontos  $i$  e  $i + 1$  em um meio heterogêneo com o ponto fictício  $i + \frac{1}{2}$ .

progressiva conforme segue:

$$q_{i+\frac{1}{2}}^- \approx \frac{-2k_A}{h}(T_{i+\frac{1}{2}} - T_i) \quad (3.5)$$

onde  $k_A$  é a condutividade térmica no meio  $A$ .

$$q_{i+\frac{1}{2}}^+ \approx \frac{-2k_B}{h}(T_{i+1} - T_{i+\frac{1}{2}}) \quad (3.6)$$

onde  $k_B$  é a condutividade térmica no meio  $B$ .

Além disso, pode-se considerar o fluxo  $q_{i+\frac{1}{2}}$  aproximado pela diferença centrada no ponto  $i + \frac{1}{2}$  dado por:

$$q_{i+\frac{1}{2}} \approx \frac{-k}{h}(T_{i+1} - T_i) \quad (3.7)$$

E considerando a continuidade do fluxo no domínio, pode-se considerar que:

$$q_{i+\frac{1}{2}}^- = q_{i+\frac{1}{2}}^+ = q_{i+\frac{1}{2}} \quad (3.8)$$

ou seja

$$\frac{-2k_A}{h}(T_{i+\frac{1}{2}} - T_i) = \frac{-2k_B}{h}(T_{i+1} - T_{i+\frac{1}{2}}) = \frac{-k}{h}(T_{i+1} - T_i) \quad (3.9)$$

Isolando o termo  $T_{i+\frac{1}{2}}$  da parte 1 da equação com a parte 2, e na parte 2 com a parte 3, tem-se:

$$k(k_A + k_B)(T_{i+\frac{1}{2}} - T_i) = 2k_A k_B (T_{i+\frac{1}{2}} - T_i) \quad (3.10)$$

Desta forma, tem-se que a condutividade térmica efetiva  $k$  dada pela média harmônica de  $k_A$  e  $k_B$ :

$$k = \frac{2k_A k_B}{k_A + k_B} \quad (3.11)$$

### 3.2.3 Formulação de Diferenças Finitas Explícita em Meio Heterogêneo para o Modelo

Agora munido das relações de diferenças dadas pela tabela A.1, pode-se manipular o modelo dado pela Equação 3.2 e discretizá-lo conforme é mostrado no *stencil* dado pela Figura 3.2, onde  $T_{i,j}^n$  é a temperatura no ponto  $T(x_i, y_j, t_n)$ .

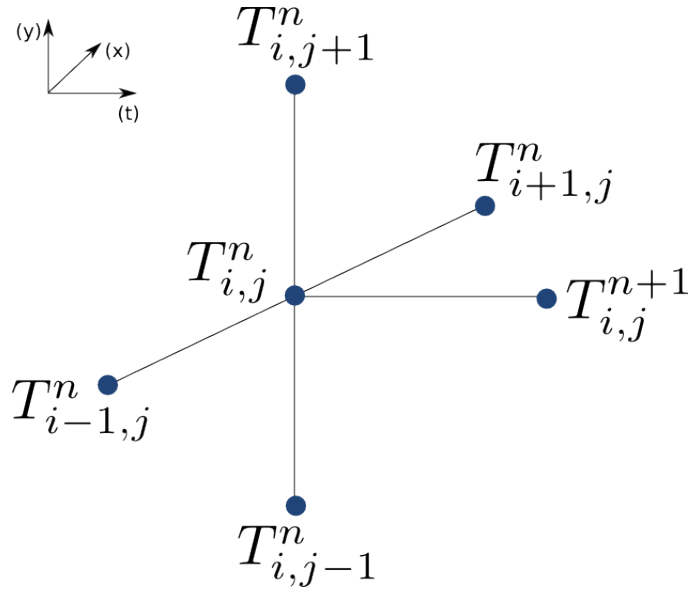


Figura 3.2: *Stencil* de segunda ordem no espaço e primeira ordem no tempo

Desta forma, deseja-se fazer uma relação de diferenças no qual o tempo  $n + 1$  dependa dos valores dos pontos no tempo  $n$ .

Dado o operador laplaciano no domínio bidimensional aplicado à temperatura  $T$ :

$$\nabla^2 T = \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \quad (3.12)$$

onde pode-se aproximar a derivada de segunda ordem em relação a  $x$  pela diferença centrada no ponto  $x_i$ :

$$\frac{\partial^2 T}{\partial x^2} \approx \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{h_x^2} \quad (3.13)$$

e analogamente a derivada em relação a  $y$  no ponto  $y_j$

$$\frac{\partial^2 T}{\partial y^2} \approx \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{h_y^2} \quad (3.14)$$

Porém, conforme demonstrado na seção anterior, devido a não continuidade do termo de condutividade térmica, deve-se considerar um  $k$  efetivo, o que obriga a reestruturar a aproximação numérica de forma a levar em consideração que  $k$  agora é uma função.

Pode-se redividir o domínio incluindo dois pontos fictícios  $i + \frac{1}{2}$  e  $i - \frac{1}{2}$  conforme pode ser observado na Figura 3.3:

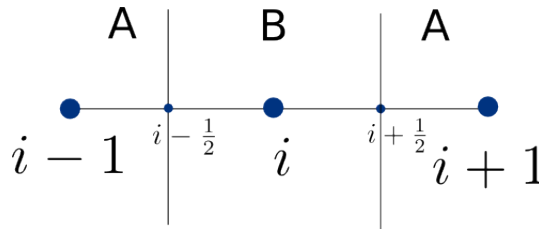


Figura 3.3: Meio heterogêneo com os pontos fictícios  $i + \frac{1}{2}$  e  $i - \frac{1}{2}$

Desta maneira pode-se calcular a diferença centrada no ponto  $i$  pelos pontos fictícios  $i \pm \frac{1}{2}$  como:

$$\frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) \approx \frac{k_{i+\frac{1}{2}} \frac{\partial T_{i+\frac{1}{2}}}{\partial x} - k_{i-\frac{1}{2}} \frac{\partial T_{i-\frac{1}{2}}}{\partial x}}{h_x} \quad (3.15)$$

Agora deve-se aproximar novamente as derivadas nos pontos  $i \pm \frac{1}{2}$  por diferença centrada em relação aos pontos  $i \pm 1$ :

$$\frac{\partial T_{i-\frac{1}{2}}}{\partial x} \approx \frac{T_i - T_{i-1}}{h} \quad (3.16)$$

$$\frac{\partial T_{i+\frac{1}{2}}}{\partial x} \approx \frac{T_{i+1} - T_i}{h} \quad (3.17)$$

Substituindo as diferenças centradas dos pontos fictícios na Equação 3.15 obtém-se:

$$\frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) \approx \frac{k_{i+\frac{1}{2}}(T_i - T_{i-1}) - k_{i-\frac{1}{2}}(T_{i+1} - T_i)}{h_x^2} \quad (3.18)$$

onde os  $k_{i+\frac{1}{2}}$  e  $k_{i-\frac{1}{2}}$  serão dados pela média harmônica (Equação 3.11), ou seja, pelo  $k$



efetivo no ponto do domínio em questão, garantindo a continuidade do fluxo.

Analogamente na derivada em relação a  $y$  tem-se:

$$\frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) \approx \frac{k_{j+\frac{1}{2}}(T_j - T_{j-1}) - k_{j-\frac{1}{2}}(T_{j+1} - T_j)}{h_y^2} \quad (3.19)$$

Portanto tem-se:

$$\nabla \cdot k \nabla T \approx \frac{k_{i+\frac{1}{2},j}(T_{i,j} - T_{i-1,j}) - k_{i-\frac{1}{2},j}(T_{i+1,j} - T_{i,j}) + k_{i,j+\frac{1}{2}}(T_{i,j} - T_{i,j-1}) - k_{i,j-\frac{1}{2}}(T_{i,j+1} - T_{i,j})}{h^2} \quad (3.20)$$

considerando  $h_x = h_y = h$ .

Vale destacar que se considerar  $k_{i+\frac{1}{2},j} = k_{i-\frac{1}{2},j} = k$ , obtém-se a mesma relação da equação 3.13 multiplicada por  $k$ , ou seja quanto o meio é homogêneo a relação 3.18 recai na relação 3.13.

Para aproximar a derivada no tempo no ponto  $n + 1$  será usada a derivada progressiva (A.1):

$$\frac{\partial T}{\partial t} \approx \frac{T_{i,j}^{n+1} - T_{i,j}^n}{h_t} \quad (3.21)$$

Agora munido de todas estas aproximações dadas pelas equações 3.21 e 3.20, substituindo no modelo dado pela Equação 3.2 e isolando o termo  $T_{i,j}^{n+1}$ , se obtém o seguinte operador de diferenças finitas explícito:

$$\begin{aligned} T_{i,j}^{n+1} = T_{i,j}^n - \frac{h_t}{(h^2)(\rho_b c_b)} & (k_{i+\frac{1}{2},j}(T_{i+1,j}^n - T_{i,j}^n) - k_{i-\frac{1}{2},j}(T_{i,j}^n - T_{i-1,j}^n) \\ & + k_{i,j+\frac{1}{2}}(T_{i,j+1}^n - T_{i,j}^n) - k_{i,j-\frac{1}{2}}(T_{i,j}^n - T_{i,j-1}^n)) \\ & - h_t \omega T_{i,j}^n + \frac{h_t}{\rho_b c_b} q_r \end{aligned} \quad (3.22)$$

A Equação 3.22 será rearranjada na seguinte forma compacta:

$$T_{i,j}^{n+1} = \frac{h_t}{\rho_{i,j}c_{i,j}} \left[ -\frac{(q_{i+\frac{1}{2},j}^n - q_{i-\frac{1}{2},j}^n)}{h_x} - \frac{(q_{i,j+\frac{1}{2}}^n - q_{i,j-\frac{1}{2}}^n)}{h_y} - \omega_{bi,j}\rho_{bi,j}c_{bi,j}T_{i,j}^n + q_{ri,j}^n \right] + T_{i,j}^n \quad (3.23)$$

onde

$$q_{i+\frac{1}{2},j}^n = -k_{i+\frac{1}{2},j} \frac{\partial T_{i,j}^n}{\partial x} \approx -k_{i+\frac{1}{2},j} \frac{T_{i+1,j}^n - T_{i,j}^n}{h_x} \quad (3.24)$$

e os outros fluxos podem ser definidos de maneira análoga.

### 3.2.4 Formulação das Condições de Contorno

Em matemática um problema de valor de contorno consiste em uma equação ou sistema de equações diferenciais munido de um conjunto de restrições sobre o contorno do domínio. A solução de um problema de valor inicial deve satisfazer todas as condições de contorno e inicial. Entre os principais tipos de condições, pode-se destacar as condições de Neumann e Dirichlet.

#### Condições de Contorno de Dirichlet

Condição de contorno de Dirichlet, quando aplicada a uma equação diferencial parcial, especifica os valores que a solução deve tomar sobre o contorno do domínio  $\Gamma$ , ou seja, assumindo  $\alpha = 1$  e  $\beta = 0$  na Equação 3.2, *i. e.*  $T(\vec{x}, t) = f(\vec{x}, t)$  para  $\vec{x} \in \Gamma$ .

Basicamente para aplicar este tipo de condição de contorno no MDF explícito, como o que foi utilizado neste trabalho, deve-se atribuir o valor da condição nos pontos  $x_i, y_j \in \Gamma$  a cada iteração do algoritmo, como pode ser observado a seguir:

$$T_{i,j}^n = f(x_i, y_j, t_n) \text{ para } x_i, y_j \in \Gamma \quad (3.25)$$

onde  $\Gamma$  é o contorno do domínio onde se aplica a condição de Dirichlet.

#### Condições de Contorno de Neumann

A condição de contorno de Neumann, quando aplicada à uma equação diferencial parcial, especifica o valor da derivada no ponto do contorno do domínio, ou seja, assumindo  $\alpha = 0$

e  $\beta = k$  na Equação 3.2, *i. e.*  $k\nabla T(\vec{x}, t) \cdot \vec{n} = f(\vec{x}, t)$  para  $\vec{x} \in \Gamma$ .

Uma das formas de se aplicar uma condição de Neumann consiste na criação de um ponto fictício chamado ponto fantasma, garantindo, assim, uma aproximação de segunda ordem. Este ponto ficará além do contorno conforme pode se visto no Figura 3.4 para  $\vec{n} = (1, 0)^T$ . O ponto fantasma será aproximado pela diferença centrada no ponto  $i$ , que pertence ao contorno do domínio, da seguinte forma:

$$k \frac{\partial T}{\partial x} \approx k \frac{T_{i+1} - T_{i-1}}{2h_x} = f(x_i, y_j, t_n) \quad (3.26)$$

Isolando o ponto fantasma  $T_{i+1}$  obtém-se:

$$T_{i+1} = \frac{2}{k} f(x_i, y_j, t_n) h_x + T_{i-1} \quad (3.27)$$

Então, aplica-se a função  $f(\vec{x}, t)$  no ponto  $x_i$ ,  $y_j$  e  $t_n$  dado pela condição de Neumann definida para o problema, e utiliza-se o operador de diferenças finitas obtido na seção anterior, dado pela Equação 3.23, levando em consideração a existência deste ponto fantasma no contorno, para todos os pontos  $x_i, y_j \in \Gamma$ .

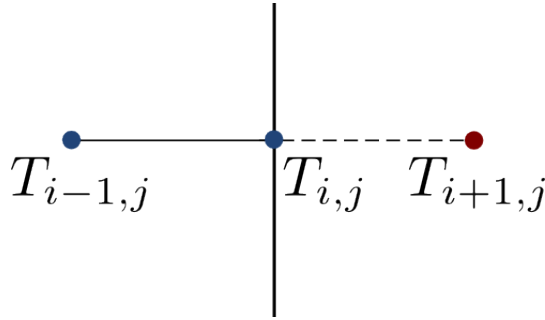


Figura 3.4: Domínio discretizado com o ponto fantasma

## 4 Paralelização

### 4.1 Contextualização Teórica

A paralelização que foi realizada neste trabalho se baseia em um sistema de memória compartilhada de acesso uniforme, ou seja, uma arquitetura de computadores UMA (*Uniform Memory Access*), conforme a Figura 4.1. Porém, atualmente, as máquinas que possuem duas ou mais unidades centrais de processamento (chamadas de *cores* ou núcleos) independentes em um mesmo chip, denominadas por *multicore*, estão cada vez mais comuns. Isto é, um sistema de memória compartilhada *multicore*, conforme ilustrado na Figura 4.2.

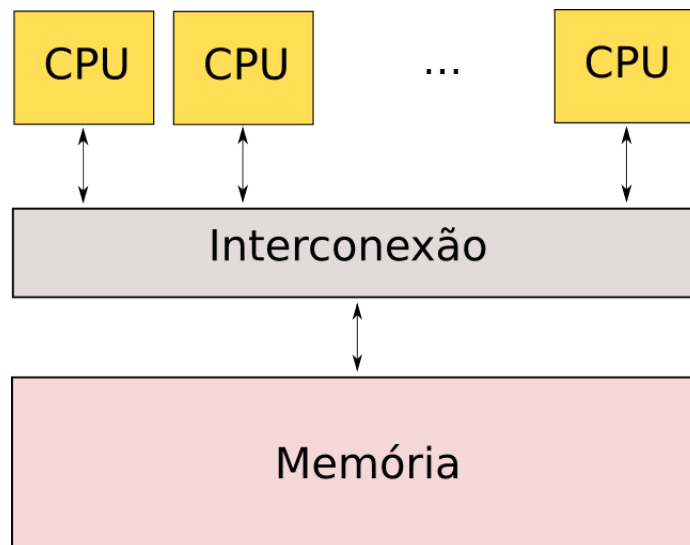


Figura 4.1: Sistema de memória compartilhada. Adaptado de PACHECO (2011)

Este trabalho se focou em um sistema de memória compartilhada *multicore* de acesso uniforme. Para criar as tarefas a serem executadas simultaneamente pelos *cores* utilizou-se uma paralelização baseada em *threads*. Uma *thread* é a menor sequência programada de instruções que pode ser gerenciada independentemente pelo escalonador de tarefas do sistema operacional. Elas são chamadas de processos leves, ou seja, usam menos recursos da máquina. A implementação de processos e *threads* variam de um sistema operacional para outro, mas na maioria dos casos elas estão contidas nos processos. Um

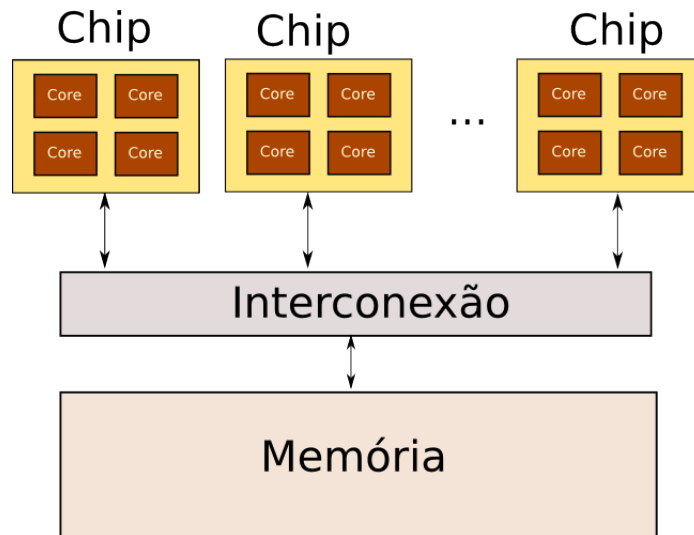


Figura 4.2: Sistema de memória compartilhada *multicore*. Adaptado de PACHECO (2011)

processo *multithread* pode ser observado na Figura 4.3.

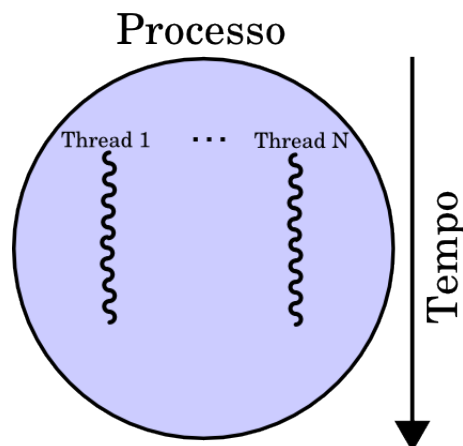


Figura 4.3: Processo *multithread*. Adaptado de PACHECO (2011)

## 4.2 Paralelização do Algoritmo

O algoritmo para a solução de uma equação diferencial parcial, através do MDF explícito, bidimensional transiente, como o formulado na seção anterior, pode ser implementado pelo Algoritmo 1. As variáveis *novaTemperatura* e *antigaTemperatura* são matrizes bidimensionais  $n_x \times n_y$ , onde  $n_x$  e  $n_y$  são o número de divisões do domínio no eixo  $x$  e  $y$ , respectivamente. A função *computaMDF* aplica a Equação 3.23, levando em consideração as condições de contorno (Dirichlet e/ou Neumann), a partir dos dados da matriz *antigaTemperatura* nos pontos  $i$  e  $j$  do domínio, e assim obter temperatura no

passo de tempo seguinte.

---

**Algoritmo 1:** Implementação de diferenças finitas sequencial

---

```

início
  para  $t \leftarrow t_i$  até  $t_f$  faça
    para  $i \leftarrow x_i$  até  $x_f$  faça
      para  $j \leftarrow y_i$  até  $y_f$  faça
        novaTemperatura[i,j] = computaMDF(antigaTemperatura,
        i,j);
      fim para
    fim para
     copia de novaTemperatura para antigaTemperatura;
  fim para
fim

```

---

A partir disto, pode-se notar que existe dependência de dados somente no passo de tempo. Portanto pode-se executar em paralelo, sem prejuízo, os passos espaciais. Para isto, este trabalho usou a API de programação paralela OpenMP. Esta API permite trabalhar com *threads* utilizando diretivas de compilação de pré-processamento chamados *pragmas*.

A diretiva mais básica para programação paralela com OpenMP é `# pragma omp parallel`. Este comando tem a função de criar as *threads* que irão executar um bloco de instruções. Este *pragma* possui parâmetros (chamados em OpenMP de clausulas), entre os principais podem-se citar os definem o número de *threads* que serão criadas e a privacidade das variáveis. Além do mais, ao término do bloco de instruções abaixo da diretiva, as *threads* são destruídas. Esse comando faz a *thread* principal do processo criar cópias de seus dados privados em um endereço de memória separado (*fork*) e depois a junção destas *threads* (*join*), conforme pode ser observado na Figura 4.4.

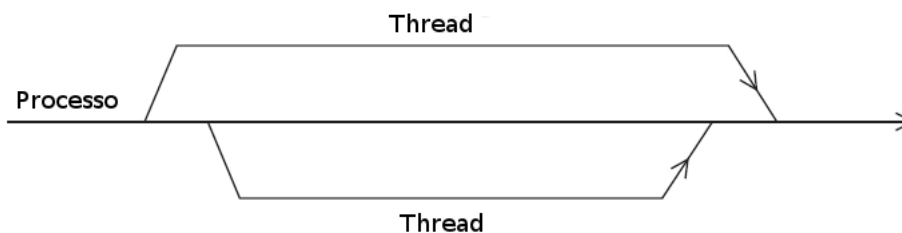


Figura 4.4: *Fork* e *join* entre as *threads* com a diretiva `omp parallel`. Adaptado de PACHECO (2011)

Outra diretiva muito importante é a `# pragma omp for`, que distribui as iterações de um laço `for` entre as `threads` que foram criadas. Além de definir a distribuição das iterações, este comando permite ao programador a escolha da forma que a distribuição se dará. As opções permitidas são estático, dinâmico ou guiado, automático e *runtime*, onde:

- **Estático:** As iterações podem ser atribuídas para as `threads` antes do laço ser executado.
- **Dinâmico** ou guiado: As iterações são atribuídas às `threads` enquanto o laço é executado, então depois que termina de executar suas tarefas, ela pode requerer mais ao sistema *run-time*.
- **Automático:** O compilador e/ou o sistema *run-time* determina o escalonamento.
- **Run-time:** O escalonamento é determinado em tempo de execução por variáveis de ambiente.

Um esquema inicial de paralelização feito para o algoritmo de solução do MDF citado anteriormente é ilustrado pelo pseudo-código do Algoritmo 2.

---

**Algoritmo 2:** Implementação de Diferenças Finitas em Paralelo com OpenMP

---

```

início
  # pragma para criar as threads
  para  $t \leftarrow t_i$  até  $t_f$  faça
    #pragma para dividir o trabalho entre as threads
    para  $i \leftarrow x_i$  até  $x_f$  faça
      para  $t \leftarrow y_i$  até  $y_f$  faça
        novaTemperatura[i,j] = computaMDF(antigaTemperatura,
        i,j);
      fim para
    fim para
     copia de novaTemperatura para antigaTemperatura;
  fim para
fim

```

---

Neste pseudo-código acima vale a pena ressaltar que as `threads` são criadas antes de entrar no passo de tempo, pois assim evita-se o *overhead* de criação/destruição delas a cada passo de tempo.

Outra técnica utilizada para ganhar desempenho neste algoritmo foi através de uma estratégia para evitar a cópia de dados da matriz *novaTemperatura* para *antigaTemperatura* (uma operação de ordem  $O(n^2)$ ) a cada passo de tempo. Para isto, considere a criação de somente uma matriz tridimensional  $2 \times n_x \times n_y$ , ao invés de duas  $n_x \times n_y$ . Com uma simples troca de valores das variáveis *nova* e *antiga* se pode executar o algoritmo sem a necessidade de cópia de dados entre as matrizes, como ilustra o Algoritmo 3.

---

**Algoritmo 3:** Implementação de Diferenças Finitas em Paralelo com OpenMP

---

```

início
  nova = 1;
  antiga = 0;
  # pragma para criar as threads
  para  $t \leftarrow t_i$  até  $t_f$  faça
    #pragma para dividir o trabalho entre as threads
    para  $i \leftarrow x_i$  até  $x_f$  faça
      para  $t \leftarrow y_i$  até  $y_f$  faça
        Temperatura[nova,i,j] = computaMDF(Temperatura[antiga],
        i,j);
      fim para
    fim para
    nova = (nova == 1?0:1);
    antiga = (antiga == 1?0:1);
  fim para
fim

```

---

A divisão de tarefas entre as *threads* foi feita de forma estática. Basicamente com este tipo de escalonamento, cada uma das  $n$  *threads* será responsável por  $n_y/n$  linhas da matriz da solução do problema, onde  $n_y$  é o número de linhas desta matriz. A Figura 4.5 ilustra a divisão de tarefas entre, por exemplo, 4 *threads*.



Figura 4.5: Escalonamento estático com 4 *threads*



## 5 Resultados

### 5.1 Introdução

Após a implementação de todo o modelo matemático e numérico, além do esquema de paralelização aqui mostrado, pode-se executar as simulações *in silico*. Antes de realizar as simulações, ainda falta definir alguns parâmetros do modelo, os quais foram retirados da literatura. Além disto, também deve-se definir o domínio onde o modelo se aplicará. Quanto à paralelização, é importante definir a arquitetura em que se executará as simulações em paralelo, de forma a poder compreender melhor os resultados que serão apresentados neste capítulo.

#### 5.1.1 Definição do Domínio

Para a resolução do modelo 3.2 foi proposto a utilização de um meio heterogêneo, ou seja, deseja-se modelar um tecido vivo, porém, devido a existência do tumor, algumas propriedades do material variam neste domínio. A Figura 5.1 ilustra o domínio da função, diferenciando o tecido saudável do tumoral. O câncer está localizado no quadrado central.

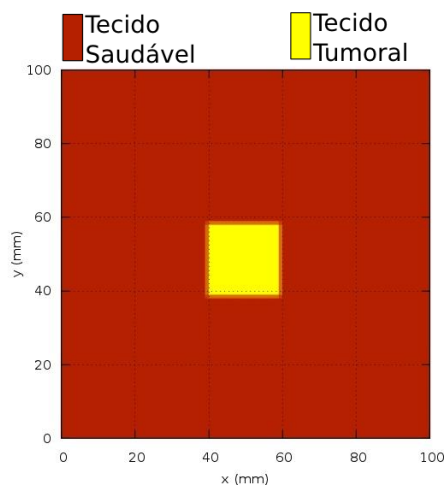


Figura 5.1: Domínio da solução da equação com tumor.

### 5.1.2 Parâmetros das Simulações

Como pode ser observado, a Equação 3.2 possui parâmetros que dizem respeito às propriedades do tecido que a simulação se executará. Alguns destes valores podem ser diferentes dependendo do meio que se está. Todos os parâmetros e descrições, além de seus respectivos valores utilizados nas simulações, podem ser consultados na Tabela 5.1 (CAO *et al.*, 2010).

Tabela 5.1: Parâmetros do modelo.

Símbolo	Descrição	Unidade	Tecido Saudável	Tecido Tumoral
$c$	Calor específico do Tecido	$(J/Kg^{\circ}C)$	4200.0	4200.0
$c_b$	Calor específico do sangue	$(J/Kg^{\circ}C)$	4200.0	4200.0
$k$	Condutividade Térmica do Tecido	$(W/m^{\circ}C)$	0.5	0.55
$\rho$	Densidade do Tecido	$(Kg/m^3)$	1000.0	1000.0
$\rho_b$	Densidade do sangue	$(Kg/m^3)$	1000.0	1000.0
$\omega_b$	Perfusão sanguínea	$(ml/s/ml)$	0.00051	0.00125

Além destes parâmetros do modelo, existem outros que dizem respeito à hipertermia, ou seja, a distribuição de temperatura baseada no SAR, os quais compõem a função  $q_r(\vec{x}, t)$  dada pela Equação 2.17. Estes valores variar-se-ão de acordo com a forma da aplicação dos pontos de injeção. Este trabalho simula dois modos de se injetar as nanopartículas, com quatro pontos de aplicação e com somente um. Em cada uma destas simulações irão se utilizar diferentes parâmetros para o SAR, conforme a Tabela 5.2.

Tabela 5.2: Parâmetros da aplicação do tratamento com hipertermia.

Símbolo	Descrição	Unidade	Um Ponto	Quatro Pontos
$A$	Amplitude máxima da geração de calor	$(W/m^3)$	$1.3 \times 10^6$	$0.325 \times 10^6$
$r_0$	Raio de distancia que as partículas espalham	$(m)$	$3.1 \times 10^{-3}$	$3.1 \times 10^{-3}$

### 5.1.3 Condições Inicial e de Contorno

Em todas as simulações executadas neste trabalho foram utilizadas condições de Neumann e de Dirichlet. Nos lados  $I, II$  e  $III \in \Gamma$  foram aplicadas condição de contorno de Neumann  $\nabla T(\vec{x}, t) \cdot \vec{n} = 0$ , e no lado  $IV$  condição de Dirichlet  $T(\vec{x}, t) = 0$ , conforme pode ser visto na Figura 5.2.

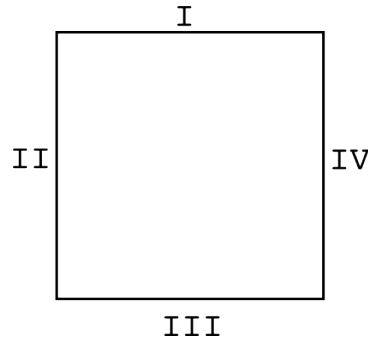


Figura 5.2: Contorno do domínio.

Além disto, as simulações realizadas neste trabalho utilizaram condição inicial  $T(\vec{x}, 0) = 0$ , de forma que somando o resultado da solução da Equação 2.16 no resultado final da simulação *i. e.*  $T_1(\vec{x}, t) = T(\vec{x}, 0) + T_2(\vec{x}, 0)$ , obtém-se a temperatura corporal em cada ponto da discretização do domínio.

## 5.2 Resultados do Modelo

Após a execução deste modelo de tratamento por hipertermia, utilizando injeção de nanopartículas magnéticas com o método das diferenças finitas explícito, pode-se observar os resultados das simulações. Elas foram basicamente divididas em dois tipos. O primeiro, utilizando somente um ponto de injeção, e o segundo, com quatro. Os parâmetros da fonte de calor aplicada estão na Tabela 5.2. Já os parâmetros da equação utilizada para modelar a biotransferência de calor estão apresentados na Tabela 5.1, e o domínio da simulação é o supracitado.

Vale a pena destacar, que as células sofrem necrose quando a temperatura chega a aproximadamente  $43^{\circ}C$  (SALLOUM *et al.*, 2008).

A Figura 5.3 apresenta a evolução da temperatura em ambas as simulações. O gráfico, basicamente, apresenta a temperatura em 2 pontos distintos, um localizado no tecido saudável em  $(x = 0.030m, y = 0.050m)$  e o outro dentro do tumor em  $(x = 0.045m, y = 0.050m)$ , em cada uma das simulações.

O gráfico apresentado na Figura 5.4 apresenta a distribuição de temperatura em  $t = 50$  min, no caso onde fez-se um ponto de injeção.

O gráfico apresentado na Figura 5.5 apresenta a distribuição de temperatura em

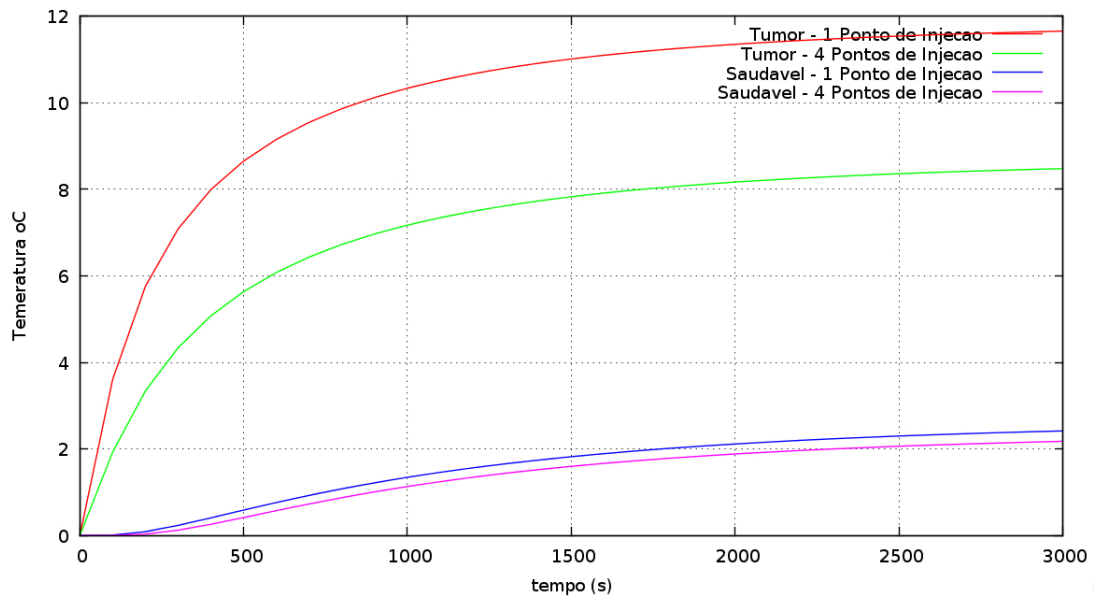


Figura 5.3: Evolução da temperatura no tempo, dentro e fora do tumor em ambos os casos.

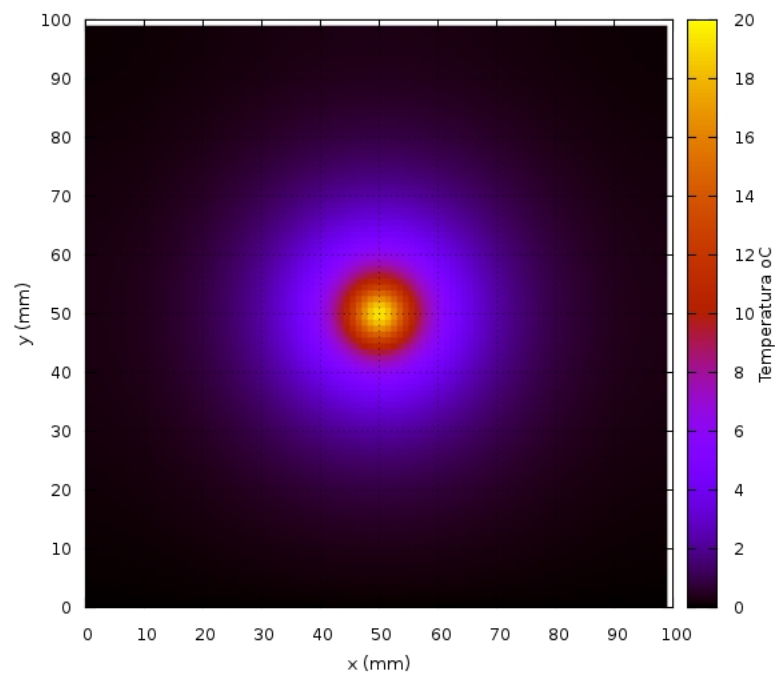


Figura 5.4: Distribuição da temperatura no domínio, quando utilizado somente um ponto de injeção.

$t = 50$  min, no caso onde fez-se quatro pontos de injeção.

Os gráficos das Figuras 5.6 e 5.7 são basicamente os mesmos das Figuras 5.4 e 5.5, porém com a escala de cores com máximo em  $6^{\circ}C$ , ou seja, onde o tecido está morto. Portanto no centro da simulação na cor amarela está o tecido morto.

Vale a pena destacar a efetividade da necrose do tumor em ambos os casos si-

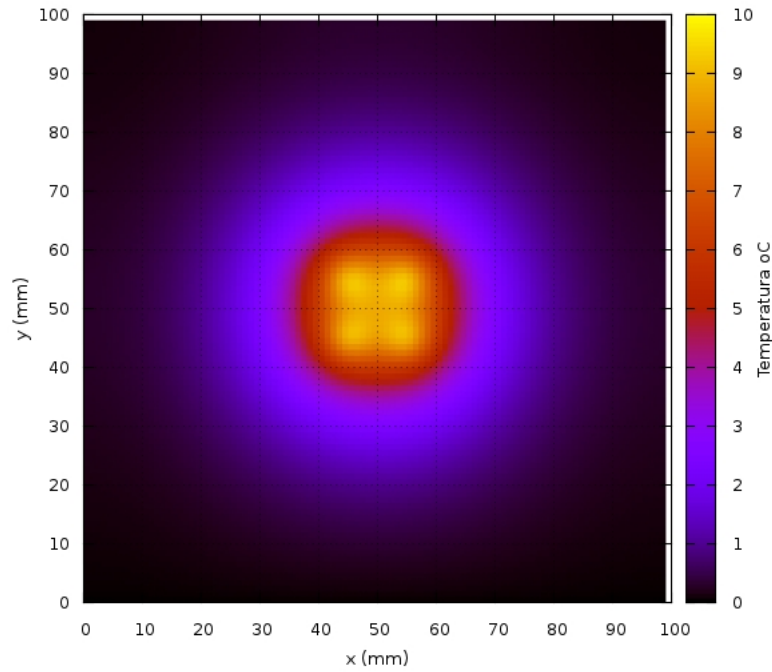


Figura 5.5: Distribuição da temperatura no domínio, quando utilizado quatro pontos de injeção.

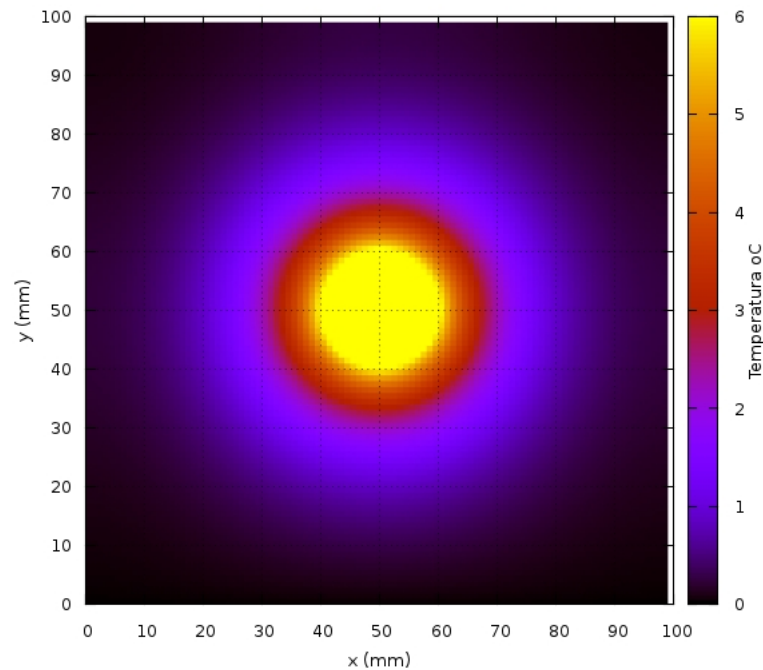


Figura 5.6: Tecido morto destacado em amarelo, quando utilizado somente um ponto de injeção.

mulados. Na simulação com um ponto de injeção, obteve-se 78,75% de necrose de tumor e 0,125% de tecido saudável. Já na simulação com quatro pontos de injeção, pôde-se conseguir 91,75% de necrose de tumor e 0,260% de tecido saudável. O ponto importante a se ressaltar nestes resultados, é que com 4 pontos de injeção, pôde-se causar necrose

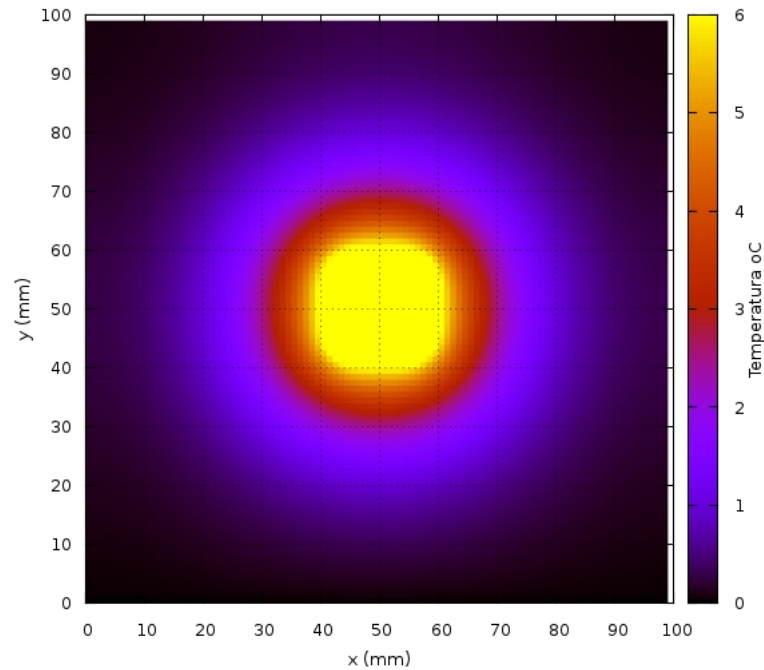


Figura 5.7: Tecido morto destacado em amarelo, quando utilizado quatro pontos de injeção.

celular em mais células tumorais que com somente um ponto de injeção, para o tumor modelado.

### 5.3 Resultados da Paralelização

A execução da simulação sequencial utilizando um *grid* onde o  $h = 10^{-4}$ m, ou seja, 1.000.000 pontos, chega a durar cerca de 10 horas. Para melhorar a velocidade de execução das simulações, optou-se por fazer um esquema de paralelização com memória compartilhada, conforme o descrito no Capítulo 4. Para executar a paralelização realizada neste trabalho, usou a ideia do pseudo-código mostrado no Algoritmo 3.

Além disso, vale destacar a arquitetura onde as simulações em paralelo se sucederam. Todas elas foram executadas em um computador SMP Linux (3.9.2-200), que possui 4 CPU AMD Opteron 6272 e 128 GB de RAM. Cada CPU possui 16 núcleos, em um total de 64. Porém, nesta arquitetura da AMD, os *cores* são pareados em estruturas denominadas por módulos, o qual compartilham significativa proporção de recursos, *e.g.* cada 2 *cores* compartilham 1 unidade de ponto flutuante (FPU) e isto interfere na performance das simulações executadas significativamente. Conforme a maneira com que

as *threads* são escalonadas, pode causar disputa pela FPU, mesmo quando o número de *threads* é inferior à quantidade de FPU disponíveis. Um esquema de como fica os recursos compartilhados neste processador, pode ser observado na Figura 5.8

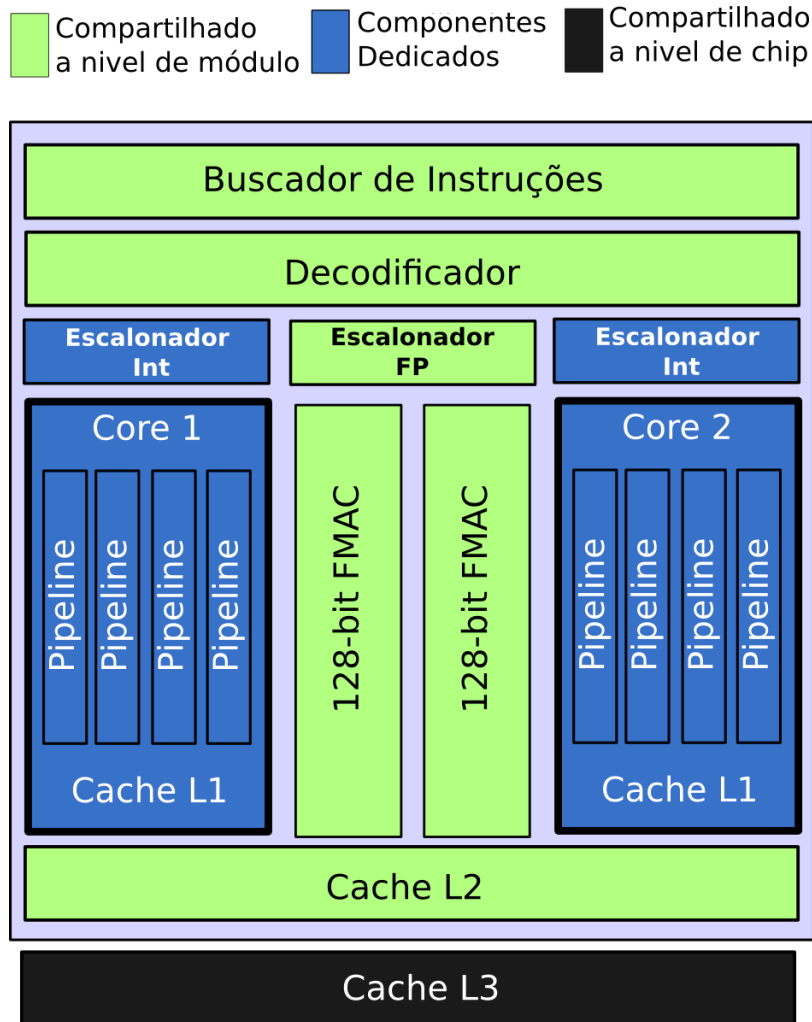


Figura 5.8: Compartilhamento de recursos do Chip AMD Opteron Série 6200 Interlagos. Adaptado de Numerical Algorithms Group (2011).

Uma medida utilizada para medir a aceleração do tempo de execução do código paralelo comparado ao sequencial é o *speedup*. Esta medida pode ser feita com a seguinte equação:

$$speedup = \frac{T_s}{T_p} \quad (5.1)$$

onde  $T_s$  é o tempo de execução do código na versão sequencial e  $T_p$  o tempo de execução do código na versão paralela.

A Figura 5.9 apresenta o *speedup* alcançado com a paralelização proposta, e exe-

cutada nesta arquitetura supra especificada.

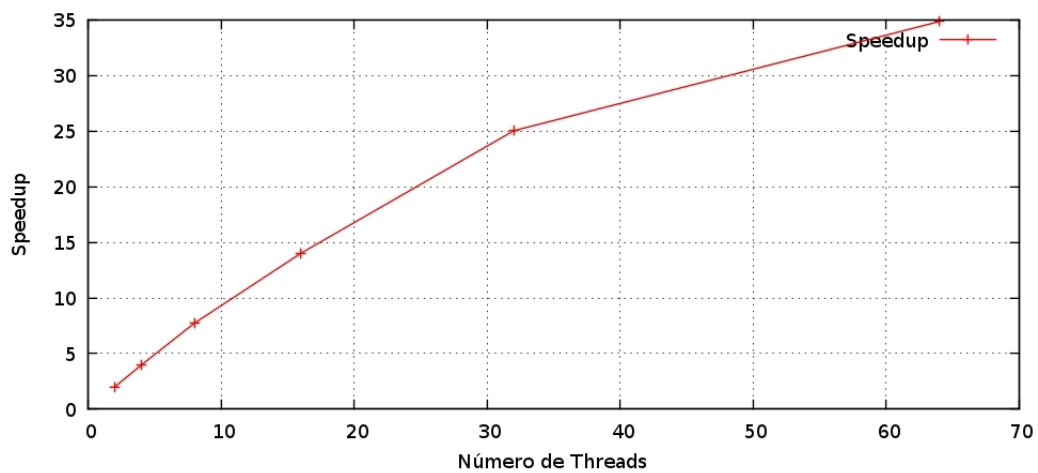


Figura 5.9: *Speedup* de execução do código nesta arquitetura.

O *speedup* apresentado, foi obtido usando a média de três execuções do código, o qual teve um desvio padrão abaixo de 0,4957. Vale destacar que o *speedup* conseguido foi aproximadamente 35, o que reduziu o tempo de execução de 10h para apenas cerca de 16 minutos.



## 6 Conclusões

### 6.1 Modelo Matemático e Método Numérico

O modelo matemático, inicialmente proposto por Pennes na década de 40, modela a distribuição de temperatura ao longo de um domínio. Neste trabalho, fez-se um estudo inicial baseado nesta equação em um domínio bidimensional. Para complementar este modelo incluiu-se a função  $q_r(\vec{x}, t)$ , conforme o estudo feito por Salloum em 2008 e 2009, utilizando a geração de calor aproximado pela taxa de absorção específica (SAR).

Munido das equações deste modelo, usou-se um método numérico para encontrar a sua solução, pois não existe solução analítica. Outro ponto importante, é o método numérico escolhido. Neste trabalho optou-se pelo método das diferenças finitas, um método que se baseia na discretização da equação, usando equações de diferenças. Optou-se pela diferença centrada, no caso da diferença de segunda ordem no espaço e, no tempo, diferença progressiva de primeira ordem, conforme a Tabela A.1. Vale destacar que mesmo no contorno, onde se usou pontos fantasmas para aproximar a condição de Neumann, preocupou-se em escolher também, a diferença centrada. Deste modo, o erro deste método numérico se mantém na ordem de  $O(h^2, h_t)$ , o que garante uma solução numérica com erro razoavelmente baixo.

Assim, se propôs dois tipos de simulações, com o intuito de minimizar a morte de células saudáveis e maximizar a morte das tumorais. Conforme observado nas simulações do tumor modelado neste trabalho, os resultados foram melhores para a simulação com quatro pontos de injeção das nanopartículas magnéticas. Além disso, com o maior número de pontos, pôde-se diminuir a potência das fontes de calor.

### 6.2 Paralelização

Devido a longa duração das simulações executadas em uma malha refinada, optou-se por fazer um esquema inicial de paralelização, baseado em memória compartilhada, e uti-

lizando a API de programação OpenMP. Por ser uma API de programação razoavelmente de simples uso, pôde-se obter resultados estimulantes com um custo benefício alto, ou seja, conseguiu-se bons resultados em tempo ágil. Além disto, este foi um esquema inicial, entretanto, bom o suficiente para haver o interesse na aplicação de técnicas mais avançadas de programação paralela.

Inicialmente o *speedup* conseguido foi menor que o esperado, pois se esperava que com uma máquina munida de uma arquitetura de processadores que totalizava 64 núcleos, o ganho de desempenho fosse maior. Porém isto foi explicado, ao realizar um estudo detalhado da arquitetura do processador, que contém uma unidade de pontos flutuantes para cada módulo, onde cada módulo é composto por dois núcleos de processamento. Como o método das diferenças finitas executa muitos cálculos onde se necessita a FPU, pôde-se inferir que houve uma grande disputa por esta unidade durante a execução das simulações.

## 6.3 Trabalhos Futuros

No que se diz respeito ao modelo matemático utilizado, foram adotadas algumas simplificações, *e. g.* o parâmetro  $\omega_b$  escolhido é constante, além disto a fonte de calor utilizada varia somente no espaço, não adotando a sua variação no tempo. Assim, existem parâmetros que podem ser alterados, possibilitando evoluir o modelo adotado, e com a utilização de menos simplificações, tornando-o mais próximo da realidade. Outro ponto importante é a escolha pelo domínio bidimensional, que pode ser evoluído para o tridimensional.

Quanto ao método numérico, existe a possibilidade da implementação do método das diferenças finitas com ordem de convergência superior, *e.g.*  $O(h^4)$ , podendo-se conseguir resultados mais precisos, além da possibilidade de comparação com o método atualmente implementando, realizando um comparativo da qualidade das soluções. Além disto, ainda existe a possibilidade da utilização de outros métodos numéricos, como o método dos elementos finitos, método dos volumes finitos ou elementos de contorno, que são métodos robustos para solução numérica de equações diferenciais parciais.

Com o intuito de se conseguir resultados melhores do ponto de vista da paralelização, tem-se como trabalho futuro a implementação em outros ambientes de programação

---

paralela, como o *Message Passing Interface* (MPI) e *Compute Unified Device Architecture* (CUDA). Com a utilização de um domínio tridimensional e o ambiente CUDA, se espera conseguir resultados com *speedup* maior.

## Referências Bibliográficas

- Abraham, J.; Hennessey, M. ; Minkowycz, W. A simple algebraic model to predict burn depth and injury. **International Communications in Heat and Mass Transfer**, v.38, n.9, p. 1169 – 1171, 2011.
- Busch, S.; Kirillin, G. ; Mehner, T. Plasticity in habitat use determines metabolic response of fish to global warming in stratified lakes. **Oecologia**, v.170, n.1, p. 275–287, 2012.
- Cao, L.; Qin, Q.-H. ; Zhao, N. An rbf-mfs model for analysing thermal behavior of skin tissues. **International Journal of Heat and Mass Transfer**, v.53, p. 2827–2839, 2010.
- Gilchrist, R. K.; Medal, R.; Shorey, W. D.; Hanselman, R. C.; Parrott, J. C. ; Taylor, C. B. Selective inductive heating of lymph nodes. **Annals of Surgery**, v.146, n.4, p. 596–606, 1957.
- Hilger, I.; Hergt, R. ; Kaiser, W. A. Towards breast cancer treatment by magnetic heating. **Journal of Magnetism and Magnetic Materials**, v.293, n.1, p. 314 – 319, 2005. Proceedings of the Fifth International Conference on Scientific and Clinical Applications of Magnetic Carriers.
- Holmes, M. H. **Introduction to Numerical Methods in Differential Equations**. 1. ed., Springer, 2007.
- Jiji, L. M. **Heat Conduction**. Springer-Verlag Berlin Heidelberg, 2009.
- Johannsen, M.; Thiesen, B.; Jordan, A.; Taymoorian, K.; Gneveckow, U.; Waldofner, N.; Scholz, R.; Koch, M.; Lein, M.; Jung, K. ; Loening, S. A. Magnetic fluid hyperthermia (mfh) reduces prostate cancer growth in the orthotopic dunning r3327 rat model. **The Prostate**, v.64, n.3, p. 283–292, 2005.
- Kawai, N.; Ito, A.; Nakahara, Y.; Futakuchi, M.; Shirai, T.; Honda, H.; Kobayashi, T. ; Kohri, K. Anticancer effect of hyperthermia on prostate cancer mediated by magnetite cationic liposomes and immune-response induction in transplanted syngeneic rats. **The Prostate**, v.64, n.4, p. 373–381, 2005.
- Liu, J.; Xu, L. Boundary information based diagnostics on the thermal states of biological bodies. **International Journal of Heat and Mass Transfer**, v.2473, p. 285–300, 2002.
- Matsuki, H.; Yanada, T.; Sato, T.; Murakami, K. ; Minakawa, S. Temperature-sensitive amorphous magnetic flakes for intratissue hyperthermia. **Materials Science and Engineering: A**, v.181, n.0, p. 1366 – 1368, 1994. Proceedings of the Eighth International Conference on Rapidly Quenched and Metastable Materials: Part 2.
- Miyakawa, M.; Bolomey, J. C. **Non-Invasive Thermometry of the Human Body**. CRC Press, 1996.

- Moroz, P.; Jones, S. K. ; Gray, B. N. Magnetically mediated hyperthermia: current status and future directions. **International Journal of Hyperthermia**, v.18, n.4, p. 267–284, 2002. PMID: 12079583.
- Group, N. A. **How to make best use of the AMD interlagos processor**. Technical report, 11 2011.
- Pacheco, P. **An Introduction to Parallel Programming**, volume 1. 1. ed., MK, 2011.
- Pennes, H. H. Analysis of tissue and arterial blood temperature in the restind human forearm. **Journal of Applied Phisiology**, v.1, p. 93–122, 1948.
- Rosensweig, R. Heating magnetic fluid with alternating magnetic field. **Journal of Magnetism and Magnetic Materials**, v.252, n.0, p. 370 – 374, 2002. Proceedings of the 9th International Conference on Magnetic Fluids.
- Salloum, M.; Ma, R. ; Zhu, L. An in-vivo experimental study of temperature elevations in animal tissue during magnetic nanoparticle hyperthermia. **International Journal of Hyperthermia**, v.24, n.7, p. 589–601, 2008.
- Salloum, M.; Ma, R. ; Zhu, L. Enhancement in treatment planning for magnetic nanoparticle hyperthermia: Optimization of the heat absorption pattern. **International Journal of Hyperthermia**, v.25, n.4, p. 309–321, 2009. PMID: 19670098.
- Thaokar, C.; Rabin, Y. Temperature field reconstruction for minimally invasive cryosurgery with application to wireless implantable temperature sensors and/or medical imaging. **Cryobiology**, v.65, n.3, p. 270 – 277, 2012.
- Vu, V. T.; Cats, G. ; Wolters, L. Graphics processing unit optimizations for the dynamics of the hirlam weather forecast model. **Concurrency and Computation: Practice and Experience**, v.25, n.10, p. 1376–1393, 2013.

## A Fórmulas de Diferenças Finitas

Considerando pontos  $x_1, x_2, x_3, \dots$  igualmente espaçados com um passo de tamanho  $h = x_{i+1} - x_i$ . O ponto  $\eta_i$  está localizado entre o ponto mais à esquerda e o ponto mais à direita usado na formula (HOLMES, 2007)

Tabela A.1: Fórmulas de Diferenças Finitas

Tipo	Fórmulas de Diferenças Finitas	Termo de Truncamento
Progressiva	$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{h} + \tau_i$	$\tau_i = -\frac{h}{2} f''(\eta_i)$
Regressiva	$f'(x_i) = \frac{f(x_i) - f(x_{i-1}))}{h} + \tau_i$	$\tau_i = -\frac{h}{2} f''(\eta_i)$
Centrada	$f'(x_i) = \frac{f(x_{i+1}) - f(x_{i-1}))}{2h} + \tau_i$	$\tau_i = -\frac{h^2}{6} f'''(\eta_i)$
Lateral à direita	$f'(x_i) = \frac{-f(x_{i+2}) + 4f(x_{i+1}) - 3f(x_i))}{2h} + \tau_i$	$\tau_i = -\frac{h^2}{3} f'''(\eta_i)$
Lateral à esquerda	$f'(x_i) = \frac{3f(x_i) - 4f(x_{i-1}) + 3f(x_{i-2}))}{2h} + \tau_i$	$\tau_i = -\frac{h^2}{3} f'''(\eta_i)$
Centrada	$f''(x_i) = \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1}))}{h^2} + \tau_i$	$\tau_i = -\frac{h^2}{12} f''''(\eta_i)$

## B Série de Taylor

A série de Taylor associada a uma função  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  infinitamente diferenciável (real ou complexa) definida em uma bola aberta centrada em  $\vec{x}_0$  e raio  $r$ , ou seja  $B(\vec{x}_0, r)$ , onde é a série de potências de  $f : \mathbb{R} \rightarrow \mathbb{R}$  é dada por:

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x - a)^n \quad (\text{B.1})$$

onde  $a$  é o ponto central da expansão em série de Taylor,  $f^{(n)}(a)$  é a  $n$ -ésima derivada de  $f$  no ponto  $a$  e  $n!$  é o fatorial de  $n$ .

No caso da série de Taylor para  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  tem-se:

$$f(\vec{x}) = \sum_{|k| \geq 0} \frac{1}{k!} \frac{\partial^{(k)} f}{\partial x_i^{(k)}}(\vec{x}_0) (x_i - x_i^0)^k \quad (\text{B.2})$$