



Ministério da  
**Ciência, Tecnologia  
e Inovação**



sid.inpe.br/mtc-m19/2014/01.26.01.19-TDI

**MÉTODO DE ALTA ORDEM PARA AJUSTE DE PASSO  
DE TEMPO LOCAL PARA A RESOLUÇÃO NUMÉRICA  
DE EQUAÇÕES DIFERENCIAIS EVOLUTIVAS COM  
USO DE ANÁLISE MULTIRRESOLUÇÃO ADAPTATIVA**

Müller Moreira Souza Lopes

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Margarete Oliveira Domingues, e Odim Mendes Júnior, aprovada em 24 de fevereiro de 2014.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/3FKS33H>>

INPE  
São José dos Campos  
2014

## **PUBLICADO POR:**

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

## **CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE (RE/DIR-204):**

### **Presidente:**

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

### **Membros:**

Dr. Antonio Fernando Bertachini de Almeida Prado - Coordenação Engenharia e Tecnologia Espacial (ETE)

Dr<sup>a</sup> Inez Staciarini Batista - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

Dr. Germano de Souza Kienbaum - Centro de Tecnologias Especiais (CTE)

Dr. Manoel Alonso Gan - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr<sup>a</sup> Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

### **BIBLIOTECA DIGITAL:**

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

### **REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:**

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

### **EDITORAÇÃO ELETRÔNICA:**

Maria Tereza Smith de Brito - Serviço de Informação e Documentação (SID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)



Ministério da  
**Ciência, Tecnologia  
e Inovação**



sid.inpe.br/mtc-m19/2014/01.26.01.19-TDI

**MÉTODO DE ALTA ORDEM PARA AJUSTE DE PASSO  
DE TEMPO LOCAL PARA A RESOLUÇÃO NUMÉRICA  
DE EQUAÇÕES DIFERENCIAIS EVOLUTIVAS COM  
USO DE ANÁLISE MULTIRRESOLUÇÃO ADAPTATIVA**

Müller Moreira Souza Lopes

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Margarete Oliveira Domingues, e Odim Mendes Júnior, aprovada em 24 de fevereiro de 2014.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/3FKS33H>>

INPE  
São José dos Campos  
2014

Dados Internacionais de Catalogação na Publicação (CIP)

---

Lopes, Müller Moreira Souza.

L881m Método de alta ordem para ajuste de passo de tempo local para a resolução numérica de equações diferenciais evolutivas com uso de análise multirresolução adaptativa / Müller Moreira Souza Lopes. – São José dos Campos : INPE, 2014.

xxiv + 101 p. ; (sid.inpe.br/mtc-m19/2014/01.26.01.19-TDI)

Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2014.

Orientadores : Drs. Margarete Oliveira Domingues, e Odin Mendes Júnior.

1. EDP. 2. Local time. 3. runge-kutta contínuos. 4. volumes finitos. 5. multirresolução adaptativa. I.Título.

CDU 519.63

---



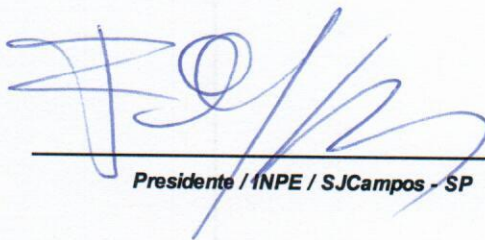
Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](https://creativecommons.org/licenses/by-nc/3.0/).

This work is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](https://creativecommons.org/licenses/by-nc/3.0/).

Aprovado (a) pela Banca Examinadora  
em cumprimento ao requisito exigido para  
obtenção do Título de **Mestre** em

**Computação Aplicada**

Dr. Fernando Manuel Ramos



---

Presidente / INPE / SJC Campos - SP

Dra. Margarete Oliveira Domingues



---

Orientador(a) / INPE / SJC Campos - SP

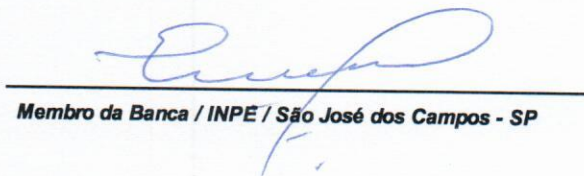
Dr. Odim Mendes Júnior



---

Orientador(a) / INPE / SJC Campos - SP

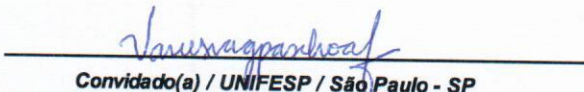
Dr. Elbert Einstein Nehrer Macau



---

Membro da Banca / INPE / São José dos Campos - SP

Dra. Vanessa Gonçalves Paschoa Ferraz



---

Convidado(a) / UNIFESP / São Paulo - SP

Este trabalho foi aprovado por:

maioria simples

unanimidade

Aluno (a): **Müller Moreira Souza Lopes**

São José dos Campos, 24 de Fevereiro de 2014



*“La venganza nunca es buena, mata el alma y la envenena”.*

DON RAMÓN  
em *“Don Ramón Zapatero”*, 1973





## AGRADECIMENTOS

Agradeço ao Instituto Nacional de Pesquisas Espaciais (INPE) e ao Curso de pós graduação em Computação Aplicada (CAP) a estrutura oferecida para a realização deste trabalho.

À Coordenação de Aperfeiçoamento de Pessoal de nível Superior (CAPES) o auxílio financeiro que viabilizou minhas pesquisas e estudos durante estes dois anos.

Aos orientadores Dra. Margarete Oliveira Domingues e Dr. Odim Mendes a oportunidade, o material, as discussões e a paciência que tornaram possível o desenvolvimento deste trabalho.

Ao Dr. Olivier Roussel, que desenvolveu e disponibilizou o código carmen, usado como base para a implementação da técnica proposta.

Aos pesquisadores Dr. C-D. Munz e Dr. K. Schneider as discussões científicas que motivaram este trabalho.

Ao Eng. Varlei Menconi o suporte técnico computacional a esta pesquisa.



## RESUMO

Resoluções numéricas de equações diferenciais parciais (EDPs) evolutivas que apresentam estruturas singulares localizadas em suas soluções têm ampla aplicação nas áreas das Ciências Espaciais, como em hidrodinâmica e magneto-hidrodinâmica. Os métodos tradicionais para resolver essas EDPs, tais como diferenças finitas ou volumes finitos, podem tornar a simulação de tais EDPs muito dispendiosa. Por essa razão, muitos estudos desenvolvem técnicas numéricas adaptativas capazes de tornar as resoluções numéricas dessas EDPs mais viáveis computacionalmente. Neste trabalho utiliza-se o método de volumes finitos auxiliado por técnicas de multirresolução adaptativa para o domínio espacial. O propósito é medir a regularidade local da solução a fim de gerar uma malha computacional adaptada espacialmente à solução, ou seja, produzir um maior refinamento nas regiões de comportamento menos suaves da solução. A evolução temporal é feita por meio de uma técnica em que o passo de tempo é ajustado localmente a malha espacial adaptativa. Dessa forma, cada célula é evoluída no tempo individualmente, com um passo de evolução temporal proporcional ao seu tamanho e de acordo com a condição de estabilidade respeitada pela malha mais refinada. Tal procedimento já foi aplicado com sucesso para métodos de baixa ordem. Entretanto, para ordens superiores existe o problema de sincronização da solução no tempo entre células adjacentes com níveis diferentes de resolução. Este trabalho resolve este problema ao propor uma abordagem inovadora para lidar com esse problema, a utilização de métodos de Runge-Kutta contínuos.



**HIGH ORDER METHOD FOR LOCAL TIME STEP ADJUSTMENT  
IN THE NUMERICAL RESOLUTION OF EVOLUTIVE  
DIFFERENTIAL EQUATIONS USING ADAPTIVE  
MULTIRESOLUTION ANALYSIS.**

**ABSTRACT**

Numerical solutions of evolutionary partial differential equations (PDEs) that have unique structures located in their solutions have wide applications in the areas of Space Science, as in hydrodynamics and magnetohydrodynamics. Traditional methods for solving those PDEs, such as finite differences or finite volumes, make the computations very expensive. For this reason, many studies have developed numerical techniques to make these PDEs resolution computationally feasible. Here, the efforts use the finite volume method aided by adaptive multiresolution techniques. The purpose is to measure the local regularity of the solution in order to generate a computational mesh spatially adapted to the solution, i.e, to produce a greater refinement in less smooth behavior regions of the solution. The time evolution is performed by a technique which the time step is locally adjusted to the adaptive mesh. In it, each cell is individually evolved in time with a step size proportional to its size, and in accordance to the stability condition observed by the finer mesh. Such a technique has already been applied to low orders methods. In case of higher orders, however, there is the problem of synchronizing the time between the solution of adjacent cells with different levels of resolution. This work solves this problem proposing a new approach to deal with it by applying the Natural Continuous Extensions for Runge-Kutta Methods.



## LISTA DE FIGURAS

	<u>Pág.</u>
3.1 Exemplo de estrutura multi-escala de valores pontuais com 3 níveis. . . .	15
3.2 Exemplo de estrutura multi-escala de médias celulares com 4 níveis. . . .	16
4.1 Hierarquia de malhas bidimensionais em diversos níveis de resolução. . .	25
4.2 Exemplo de malha adaptada. . . . .	26
4.3 Implementação da hierarquia de malhas como uma árvore. . . . .	29
6.1 Dessincronização da evolução temporal ao aplicar a técnica do LT. . . . .	46
6.2 Passos para a execução do LT. . . . .	48
6.3 Evolução temporal de duas células adjacentes utilizando o RK2. . . . .	49
6.4 Busca em profundidade: percorrimento das células de uma árvore. . . . .	52
6.5 Busca em largura: percorrimento das células de uma árvore. . . . .	53
7.1 Condição inicial do problema proposto da Equação de Burgers 1D. . . .	58
7.2 Condição inicial do problema proposto da Equação de Burgers 2D. . . .	58
8.1 Solução numérica: Equação de Burgers 1D – método VF. . . . .	64
8.2 Erros e malhas adaptadas para Eq. Burgers 1D – métodos adaptativos. .	65
8.3 Número de células por nível de refinamento – Equação de Burgers 1D. .	65
8.4 Solução numérica: Equação de Burgers 2D – método VF. . . . .	68
8.5 Erros e malhas adaptadas para Eq. Burgers 2D – métodos adaptativos. .	69
8.6 Número de células por nível de refinamento – Equação de Burgers 2D. .	69
8.7 Solução numérica: Problema de chammas 1D – método VF/RK3. . . . .	72
8.8 Malhas adaptadas e distribuição de células – Problema de chammas 1D. . .	73
8.9 Erros para o problema de chammas 1D – métodos adaptativos. . . . .	74
8.10 Número de células por nível de refinamento – Problema de chammas 3D. .	76
8.11 Solução do problema de chammas 3D em três instantes distintos. . . . .	77
8.12 Solução numérica: Eq. de Euler – método VF/RK3. . . . .	80
8.13 Malhas adaptadas e distribuição de células por nível – Eq. de Euler 2D. .	81
8.14 Erros para Equações de Euler 2D – métodos adaptativos: $\rho$ , $p$ e $T$ . . . . .	82
8.15 Erros para Equações de Euler 2D – métodos adaptativos: $E$ , $v_x$ e $v_y$ . . .	83





## LISTA DE TABELAS

	<u>Pág.</u>
5.1	Ordem de convergência máxima para um método RK de $s$ estágios. . . . . 34
5.2	Tabela de Butcher para um método RK explícito. . . . . 35
5.3	Tabela de Butcher: Método de Runge-Kutta de ordem 1. . . . . 35
5.4	Tabela de Butcher: Método de Runge-Kutta de ordem 2. . . . . 35
5.5	Tabela de Butcher: Método de Runge-Kutta de ordem 3. . . . . 35
5.6	Ordem de convergência máxima para um método RKC de $s$ estágios. . . 40
7.1	Configuração da condição inicial utilizada para a Equação de Euler. . . . 62
8.1	Erros, tempos de processamento e memória – Eq. de Burgers 1D. . . . . 66
8.2	Erros, tempos de processamento e memória (RK2) – Eq. de Burgers 2D . 68
8.3	Erros, tempos de processamento e memória (RK3) – Eq. de Burgers 2D . 70
8.4	Erros, tempos de processamento e memória – Problema de chamadas 1D . 72
8.5	Erros, tempos de processamento e memória – Problema de chamadas 3D . . 76
8.6	Erros, tempos de processamento e memória – Eq. de Euler 2D. . . . . 79
B.1	Computadores utilizados . . . . . 103



## LISTA DE ALGORITMOS

1	Obtenção de coeficientes wavelet. . . . .	18
2	Reconstrução dos valores no nível mais refinado. . . . .	18
3	Construção da malha adaptativa. . . . .	30
4	Evolução temporal com métodos compactos. . . . .	44
5	Execução do LT - Ciclo da evolução temporal. . . . .	50
6	Execução do LT - Subrotina 1. . . . .	51
7	Busca em profundidade sobre a árvore de armazenamento das células. . . . .	52
8	Busca em largura sobre a árvore de armazenamento das células. . . . .	53
9	Execução do método proposto. . . . .	55



## LISTA DE ABREVIATURAS E SIGLAS

1D	–	Unidimensional
2D	–	Bidimensional
3D	–	Tridimensional
CC	–	Condição de Contorno
CFL	–	Courant-Friedrich-Lewy, condição de
CI	–	Condição Inicial
EDO	–	Equações Diferenciais Ordinárias
EDP	–	Equações Diferenciais Parciais
HLL	–	Harten, Lax e van Leer, fluxo numérico de
HLLD	–	Harten, Lax e van Leer Descontinuidades
LT	–	Tempo local
MR	–	Análise Multirresolução
PVI	–	Problema de Valor Inicial
RK	–	Runge-Kutta, método de
RK1	–	RK de ordem 1
RK2	–	RK de ordem 2
RK3	–	RK de ordem 3
RKC	–	Runge-Kutta Contínuo, método de
VF	–	Volumes Finitos, método dos
VF/RK2	–	Método de volumes finitos utilizando o RK3 para integração no tempo
VF/RK3	–	Método de volumes finitos utilizando o RK3 para integração no tempo



## LISTA DE SÍMBOLOS

$a$	– Constante associada ao método de Runge–Kutta
$b$	– Constante associada ao método de Runge–Kutta
$c$	– Constante associada ao método de Runge–Kutta
$d_\Omega$	– Coeficiente wavelet referente à célula $\Omega$
$f$	– Função a ser integrada utilizando o método de Runge–Kutta
$h$	– Contador associado à localização de uma célula na malha
$i$	– Contador associado à localização de uma célula na malha
$j$	– Contador associado à localização de uma célula na malha
$k$	– Valor associado a execução dos métodos de Runge–Kutta
$\ell$	– Contador associado ao número de níveis de resolução do problema
$\ell_{\min}$	– Nível menos refinado da representação na estrutura multi-escala
$m$	– Contador associado ao domínio temporal
$n$	– Contador associado ao número de dimensões do problema
$p$	– Ordem de convergência do método numérico
	Pressão
$q$	– Contador auxiliar
$r$	– Constante associada a quantidade de decomposições realizadas nas operações de multirresolução
$s$	– Número de estágios do método numérico
$t$	– Variável temporal
$t_i$	– Instante inicial da solução do método numérico
$t_f$	– Instante final da solução do método numérico
$t^m$	– Instante $m$ da discretização temporal
$\vec{v}$	– Vetor velocidade
$v_x$	– Velocidade no eixo $x$
$w$	– Variável auxiliar no método RKC de três estágios
$x_n$	– Variáveis espaciais
$z$	– Número de células vizinhas, para cada direção, usadas para a interpolação da operação de predição
$A$	– Variável auxiliar usada nas operações de predição
$B$	– Variável auxiliar usada nas operações de predição
$C$	– Variável auxiliar usada nas operações de predição
$D$	– Variável auxiliar usada nas operações de predição
$E$	– Variável auxiliar usada nas operações de predição
$F_n$	– Fluxo físico na direção do eixo $x_n$
$G$	– Variável auxiliar usada nas operações de predição
$H$	– Variável auxiliar usada nas operações de predição
$L$	– Número de níveis da análise multirresolução
$N$	– Número de dimensões espaciais do problema estudado
$Q^m$	– Valor médio da solução dentro da célula $C$ no instante $t^m$

$\hat{Q}^m$	– Valor obtido para $Q^m$ utilizando um operador de predição
$\tilde{Q}^m$	– aproximação do valor $Q^m$ utilizando um método numérico
$Q^*$	– Variável auxiliar
$Q^{**}$	– Variável auxiliar
$T$	– Temperatura
$U$	– Função a ser integrada
$X$	– Variável auxiliar usada na modelagem do problema de chamadas
$Y$	– Concentração de gás não queimado Variável auxiliar usada na modelagem do problema de chamadas
$Z$	– Variável auxiliar usada na modelagem do problema de chamadas
$\mathbb{D}^\ell$	– Conjunto de coeficientes wavelet referentes ao nível $\ell$
$L^2(\mathbb{R})$	– Conjunto de funções de quadrado integrável
$\mathbb{N}$	– Conjunto dos números naturais
$\mathbb{R}$	– Conjunto dos números reais
$\mathbb{Z}$	– Conjunto dos números inteiros
$\Delta t^m$	– Passo de integração temporal no instante $t^m$
$\Delta t_\ell^m$	– Passo de integração temporal no instante $t^m$ em relação às células do nível $\ell$
$\Delta x_n$	– Comprimento da célula $C$ na direção $x_n$
$\mathcal{C}$	– Célula genérica do domínio discretizado utilizando VF
$\mathcal{F}_n$	– Fluxo numérico na direção do eixo $x_n$
$\beta$	– Polinômios utilizados para o método RKC
$\epsilon$	– Valor de <i>threshold</i> , associado ao refinamento da malha
$\gamma$	– Constante de interpolação usada nas operações de predição
$\tau$	– Taxa de conversão entre gás não queimado e gás queimado
$\omega$	– Taxa de reação
$\Omega_k^\ell$	– Célula de nível de refinamento $\ell$ e posição $k$ .
$\Omega^\ell$	– Célula do nível $\ell$ da estrutura multi-escala
$\lambda$	– Autovalor referente ao modelo estudado
$\rho$	– Densidade
$\theta$	– Variável dos polinômios $\beta$ .
$\sigma$	– Número de Courant
$\sigma_{MAX}$	– Valor máximo de $\sigma$ para que o método numérico seja estável
$Ze$	– Número de Zeldovich



## SUMÁRIO

	<u>Pág.</u>
<b>1 INTRODUÇÃO</b> . . . . .	<b>1</b>
<b>2 MÉTODOS DE DISCRETIZAÇÃO ESPACIAL</b> . . . . .	<b>5</b>
2.1 Método dos volumes finitos . . . . .	5
2.2 Forma primitiva e conservativa . . . . .	9
2.3 Cálculo dos fluxos numéricos . . . . .	10
<b>3 ANÁLISE MULTIRRESOLUÇÃO</b> . . . . .	<b>13</b>
3.1 Operadores de análise . . . . .	15
3.1.1 MR no contexto de valores pontuais . . . . .	15
3.1.2 MR no contexto de médias celulares . . . . .	16
3.2 Operadores de síntese – Contexto da médias celulares . . . . .	17
3.2.1 Operadores de predição – caso unidimensional . . . . .	21
3.2.2 Operadores de predição – caso bidimensional . . . . .	21
3.2.3 Operadores de predição – caso tridimensional . . . . .	22
<b>4 CONSTRUÇÃO DA MALHA ADAPTATIVA</b> . . . . .	<b>25</b>
4.1 Implementação da malha adaptativa . . . . .	27
4.2 Multirresolução adaptativa . . . . .	29
<b>5 MÉTODOS DE EVOLUÇÃO TEMPORAL</b> . . . . .	<b>33</b>
5.1 Métodos de Runge–Kutta tradicionais explícitos . . . . .	34
5.2 Formulação compacta . . . . .	36
5.2.1 Formulação compacta: método RK de ordem 2 . . . . .	36
5.2.2 Formulação compacta: método RK de ordem 3 . . . . .	37
5.3 Métodos de Runge–Kutta Contínuos . . . . .	39
5.3.1 RKC: Formulação compacta . . . . .	43
<b>6 TÉCNICA DE TEMPO LOCAL</b> . . . . .	<b>45</b>
6.1 Abordagem tradicional . . . . .	49
6.2 Abordagem proposta . . . . .	49
6.3 Varredura da malha computacional . . . . .	52
6.3.1 Busca em profundidade . . . . .	52
6.3.2 Busca em largura . . . . .	53

<b>7 APLICAÇÕES</b>	<b>57</b>
7.1 Equação de Burgers	57
7.2 Problema de chamadas	58
7.2.1 Caso unidimensional	58
7.2.2 Caso tridimensional	59
7.3 Equações de Euler	61
<b>8 RESULTADOS</b>	<b>63</b>
8.1 Equação de Burgers	63
8.1.1 Caso unidimensional	63
8.1.2 Caso bidimensional	67
8.2 Problema de chamadas	71
8.2.1 Caso unidimensional	71
8.2.2 Caso tridimensional	75
8.3 Equações de Euler bidimensionais	78
<b>9 CONCLUSÕES</b>	<b>85</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>87</b>
<b>APÊNDICE A – COEFICIENTES PARA MÉTODOS RK</b>	<b>89</b>
A.1 Série de Taylor	89
A.2 Ordem de convergência	89
A.3 Obtenção dos coeficientes de um método RK	90
A.4 Métodos RK de primeira ordem	91
A.5 Métodos RK de segunda ordem	92
A.6 Métodos RK de terceira ordem	95
<b>APÊNDICE B – COMPUTADORES UTILIZADOS</b>	<b>103</b>

## 1 INTRODUÇÃO

Atualmente, as pesquisas em Ciências Espaciais e da Atmosfera têm disposto de uma quantidade muito maior de sistemas de aquisição de dados e de uma maior qualidade na resolução dos dados. Para exemplo de um desafio consideravelmente recente, as pesquisas espaciais têm contado com um maior número de satélites ou sondas espaciais que monitoram, com melhores resoluções temporais e já apresentam uma simultaneidade de coleta em pontos espalhados desde a proximidade do Sol a regiões das atmosferas planetárias (KIVELSON; RUSSELL, 1996; BOTHMER; DAGLIS, 2007). No entanto, muitas vezes essas situações de monitoramento não são suficientes para a compreensão adequada dos fenômenos. As simulações numérico-computacionais firmam-se como instrumentos indispensáveis para completar estudos sobre os processos de cenários conhecidos e, não raras vezes, desvendar aspectos inéditos relacionados às complexidades inerentes a essas vastas fenomenologias, sejam dos plasmas espaciais ou dos fluidos planetários (JARDIN, 2010; BÜCHNER et al., 2003; CHOUDHURI, 2004). O INPE investe árduos esforços no uso, participação e desenvolvimento de códigos de simulação para atender as suas pesquisas na área de modelagens, inclusive visando áreas de vanguarda como a do Clima Espacial (<http://www2.inpe.br/climaespacial/>), que motivam este tipo de trabalho.

A base dos modelos de hidrodinâmica e magnetohidrodinâmica espacial são as Equações Diferenciais Parciais (EDPs) evolutivas, cujas resoluções numéricas podem ser realizadas utilizando inúmeros métodos. O método dos Volumes Finitos (VF) é um método amplamente utilizado devido a sua facilidade de implementação e sua boa precisão no caso de soluções com descontinuidades (TORO, 1999). Ao aplicar esse método em casos de EDPs evolutivas que possuem estruturas localizadas em suas soluções, o método de VF pode necessitar de um refinamento na malha que representa a solução apenas em algumas regiões do domínio. Entretanto, nesse método, toda a malha é igualmente refinada. Isso torna a evolução temporal computacionalmente dispendiosa. Uma alternativa eficaz para resolver esse tipo de desafio é o uso de métodos com malha computacional adaptável (DOMINGUES et al., 2011; DEITERDING, 2011). Nesse caso, a malha se torna mais refinada onde existem variações mais bruscas na solução e menos refinada em regiões suaves. Métodos adaptativos necessitam de uma ferramenta capaz de avaliar a suavidade da solução. Uma ferramenta utilizada para este fim é a análise multirresolução (MR), proposta por Mallat (1989), no contexto adaptativo descrito por Harten (1993), como é explicado nos próximos parágrafos.

A análise multirresolução faz decomposições multinível de um sinal utilizando uma transformada wavelet. Com essa transformada, o sinal passa a ser representado pelos coeficientes wavelet. A amplitude desses coeficientes tem a propriedade de indicar a regularidade local (MALLAT, 1998). Dessa forma, Harten (1993) propôs um método combinando o método de VF com MR de forma a se obter um método adaptativo espacial para resolução de EDPs evolutivas com estruturas localizadas. Nesse método, a evolução temporal é feita em malhas adaptadas espacialmente para a solução.

A inexistência de coeficientes significativos nos níveis mais altos indica regularidade na região referente ao coeficiente, o que permite que a região seja representada em um nível menor de refinamento sem que ocorram perdas. Neste contexto, Harten (1994) propõe uma abordagem em que, para cada iteração, é criada uma nova malha adaptada espacialmente à solução de acordo com a regularidade local dessa. Nessa malha, o grau de refinamento local depende de onde se encontram os coeficientes wavelet de maior amplitude, denominados significativos. Com isso, tem-se uma malha localmente refinada de forma automática e ajustada à solução. Esta técnica é estendida para o caso bidimensional em Müller (2003), e para o caso tridimensional em Roussel et al. (2003).

A evolução temporal também pode ser construída de acordo com a malha adaptada (DOMINGUES et al., 2008). A técnica de tempo local (*local time*, LT) permite realizar a evolução temporal localmente com um passo proporcional ao tamanho da célula. Porém, para realizar a evolução temporal em uma célula, é necessário o valor do fluxo numérico em todos os instantes da evolução temporal. Isso faz com que, no caso de duas células adjacentes e tamanhos diferentes, a evolução no tempo da célula menor dependa do valor da célula maior em um instante de tempo intermediário, o que a princípio impediria o uso do LT. Em Domingues et al. (2008), utilizou-se um método de Runge-Kutta (RK) explícito de ordem 2 para realizar o MR/LT. Nesse caso, o tempo intermediário  $t^m + \frac{\Delta t^m}{2}$  usado para o cálculo do fluxo numérico pode ser obtido a partir do próprio método de Runge-Kutta 2.

O caso de métodos de Runge-Kutta de ordem superior ( $\geq 3$ ) é um desafio, pois os passos intermediários da evolução temporal das células maiores não sincronizam com a evolução temporal das células menores (DOMINGUES et al., 2008). A abordagem proposta neste trabalho para lidar com o problema da sincronização é o uso de métodos de Runge-Kutta contínuos, desenvolvidos por Zennaro (1986), de modo que a solução contínua forneça, de forma barata, os valores intermediários para a realização da evolução temporal nas células menores.

Para a realização deste trabalho, um conteúdo embasador é apresentado, envolvendo os aspectos considerados para a plena realização, como também situações de aplicações e, como consequência das implementações, os resultados, para a obtenção das conclusões sobre o desafio estudado. No Capítulo 2, são apresentados os métodos de discretização espacial para criar a malha não-uniforme que será usada conforme um método proposto. No Capítulo 3, a análise multirresolução é explicada. No Capítulo 4, procede-se com a explicação da construção da malha adaptativa. No Capítulo 5, são apresentadas métodos de evolução temporal de interesse. No Capítulo 6, a técnica de tempo local, conforme se pretende implementar, é apresentada. No Capítulo 7, situações de aplicações são consideradas e tratadas. No Capítulo 8, os resultados de todo o estudo realizado são apresentados e discutidos. No Capítulo 9, por fim, as conclusões são estabelecidas.

A justificativa e a importância deste trabalho podem ser aqui ressaltados. A procura de uma maneira para realizar a sincronização dos passos temporais é o desafio significativo para a utilização de métodos RK de ordem superior no contexto do MR/LT. A falta de uma abordagem para este problema limitou, até agora, o uso do MR/LT a métodos de ordem 1 e 2. Este trabalho propõe uma abordagem para estender a técnica do MR/LT para métodos de ordem superior, de modo a obter melhores soluções para os problemas estudados. O que foi feito com sucesso.



## 2 MÉTODOS DE DISCRETIZAÇÃO ESPACIAL

Neste capítulo, é tratado o método dos volumes finitos, em que se expõe a técnica de discretização espacial utilizada no método proposto neste trabalho. Isto serve de base para técnica de análise multirresolução adaptativa, explicada com mais detalhes no Capítulo 3.

### 2.1 Método dos volumes finitos

O método dos Volumes Finitos (VF) é um método adequado para resolver problemas de leis de conservação, como de transmissão de energia, massa, calor, etc (LEVEQUE, 2004). Em um espaço  $N$ -dimensional, com  $N \in \mathbb{N}$ , a discretização do método dos VF é feita dividindo o domínio em intervalos de  $N$  dimensões espaciais, estes intervalos são denominados células de grade ou volumes finitos. As equações de leis de conservação têm a forma:

$$\frac{\partial U}{\partial t} + \frac{\partial F_1(U)}{\partial x_1} + \dots + \frac{\partial F_N(U)}{\partial x_N} = S(U) \quad (2.1)$$

Em que  $U(x_1, \dots, x_N, t)$  representa a grandeza conservativa estudada, as funções  $F_n(U)$  são denominadas fluxos físicos de  $U$  e as variáveis  $x_n$  são variáveis espaciais, e  $S(U)$  é um termo fonte. O método VF trabalha com a ideia de um “escoamento” entre células de grade adjacentes. Esse “escoamento” ocorre em função dos fluxos físicos.

A discretização de um Problema de Valor Inicial (PVI) utilizando o método VF é realizada particionando o domínio de interesse em células  $N$ -dimensionais  $\mathcal{C}$  de dimensões  $\Delta x_n$  na direção do respectivo eixo  $x_n$ , em que o conjunto de todas as células  $\mathcal{C}$  é denominado malha computacional. Para facilitar a notação, considera-se que todas as células na malha computacional possuem as mesmas dimensões  $\Delta x_n$ , desta forma é possível indexar as células do domínio utilizando uma sequência  $(e_1, \dots, e_N)$  em que cada valor  $e_n \in \mathbb{N}$  associa a localização da célula na direção  $x_n$  a partir de uma origem. Para cada célula, atribui-se um valor médio dado a partir da condição inicial do problema. Esta média é estimada de forma que represente a condição inicial no domínio de interesse, utilizando do teorema do valor médio:

$$Q_{e_1, \dots, e_N}^m = \frac{1}{\prod_{n=1}^N \Delta x_n} \int_{\mathcal{C}} U(x_1, \dots, x_N, t^m) \prod_{n=1}^N dx_n \quad (2.2)$$

A resolução de um problema de valor inicial utilizando um método numérico consiste em representar a condição inicial no domínio discretizado e evoluir sucessivamente esta solução no tempo até atingir o instante desejado. O processo de evolução temporal é detalhado no Capítulo 5. Além dessa parte temporal, utiliza-se uma discretização espacial sobre as funções  $F_n$ , o tipo de discretização utilizado para estas funções caracteriza o método numérico utilizado. Para o método VF, esta discretização é realizada conforme dado no Teorema 1.

**Teorema 1.** *A discretização espacial de uma função a ser integrada em uma malha quadrangular, utilizando o método dos volumes finitos, é realizada conforme a expressão:*

$$\frac{dQ^m}{dt} = \sum_{n=1}^N \frac{(\mathcal{F}_n^- - \mathcal{F}_n^+)}{\Delta x_n} + S(Q), \quad (2.3)$$

em que  $\mathcal{F}_n^-$  é o fluxo numérico entre as células  $\mathcal{C}_{e_1, \dots, e_{n-1}, \dots, e_N}$  e  $\mathcal{C}_{e_1, \dots, e_N}$ , enquanto  $\mathcal{F}_n^+$  é o fluxo numérico entre as células  $\mathcal{C}_{e_1, \dots, e_N}$  e  $\mathcal{C}_{e_1, \dots, e_{n+1}, \dots, e_N}$ .

*Demonstração.* A equação para a evolução temporal do método VF pode ser obtida modificando a Equação 2.1 para:

$$\frac{\partial U}{\partial t} = - \sum_{n=1}^N \frac{\partial F_n(U)}{\partial x_n} + S(U) \quad (2.4)$$

Integrando-a em relação às variáveis espaciais e no intervalo de uma célula arbitrária, obtém-se:

$$\int_{\mathcal{C}} \frac{\partial U}{\partial t} \prod_{n=1}^N dx_n = - \sum_{n=1}^N \int_{\mathcal{C}} \frac{\partial F_n(U)}{\partial x_n} \prod_{q=1}^N dx_q + \int_{\mathcal{C}} S(U) \prod_{q=1}^N dx_q \quad (2.5)$$

Aplicando a regra de Leibniz na integral à esquerda obtém-se:

$$\frac{d}{dt} \int_{\mathcal{C}} U \prod_{n=1}^N dx_n = - \sum_{n=1}^N \int_{\mathcal{C}} \frac{\partial F_n(U)}{\partial x_n} \prod_{q=1}^N dx_q + \int_{\mathcal{C}} S(U) \prod_{q=1}^N dx_q \quad (2.6)$$

Modificando a Equação 2.2, tem-se que:

$$\int_{\mathcal{C}} U(x_1, \dots, x_N, t^m) \prod_{n=1}^N dx_n = Q^m \prod_{q=1}^N \Delta x_q \quad (2.7)$$



Substituindo o valor da integral na Equação 2.6, obtém-se:

$$\frac{d}{dt} \left( Q^m \prod_{q=1}^N \Delta x_q \right) = - \sum_{n=1}^N \int_{\mathcal{C}} \frac{\partial F_n(U)}{\partial x_n} \prod_{q=1}^N dx_q + \int_{\mathcal{C}} S(U) \prod_{q=1}^N dx_q \quad (2.8)$$

Aplicando a propriedade de linearidade do operador diferencial:

$$\prod_{q=1}^N \Delta x_q \frac{dQ^m}{dt} = - \sum_{n=1}^N \int_{\mathcal{C}} \frac{\partial F_n(U)}{\partial x_n} \prod_{q=1}^N dx_q + \int_{\mathcal{C}} S(U) \prod_{q=1}^N dx_q \quad (2.9)$$

Dividindo ambos os termos por  $\prod_{q=1}^N \Delta x_q$ :

$$\frac{dQ^m}{dt} = - \frac{1}{\prod_{q=1}^N \Delta x_q} \sum_{n=1}^N \int_{\mathcal{C}} \frac{\partial F_n(U)}{\partial x_n} \prod_{q=1}^N dx_q + \frac{1}{\prod_{q=1}^N \Delta x_q} \int_{\mathcal{C}} S(U) \prod_{q=1}^N dx_q \quad (2.10)$$

Seja  $S(Q)$  o valor médio da integral sobre  $S(U)$ , obtido de acordo com o Teorema do Valor Médio:

$$\frac{dQ^m}{dt} = - \frac{1}{\prod_{q=1}^N \Delta x_q} \sum_{n=1}^N \int_{\mathcal{C}} \frac{\partial F_n(U)}{\partial x_n} \prod_{q=1}^N dx_q + S(Q) \quad (2.11)$$

Aplicando a propriedade de soma de integrais em relação as mesmas variáveis, tem-se que:

$$\frac{dQ^m}{dt} = - \frac{1}{\prod_{q=1}^N \Delta x_q} \int_{\mathcal{C}} \sum_{n=1}^N \frac{\partial F_n(U)}{\partial x_n} \prod_{q=1}^N dx_q + S(Q) \quad (2.12)$$

Utilizando a notação de divergência para o termo  $\sum_{n=1}^N \frac{\partial F_n(U)}{\partial x_n}$  e definindo o diferencial de volume  $dV := \prod_{q=1}^N dx_q$  da célula  $\mathcal{C}$ :

$$\frac{dQ^m}{dt} = - \frac{1}{\prod_{q=1}^N \Delta x_q} \int_{\mathcal{C}} \nabla \cdot F dV + S(Q) \quad (2.13)$$

Aplicando o Teorema de Gauss, obtém-se:

$$\frac{dQ^m}{dt} = - \frac{1}{\prod_{q=1}^N \Delta x_q} \iint_{\partial \mathcal{C}} F \cdot \vec{v} dS + S(Q) \quad (2.14)$$

em que  $dS$  é o diferencial e  $\vec{v}$  é o vetor normal em relação à superfície de  $\mathcal{C}$ . Dividindo a integral sobre a superfície da célula em integrais sobre as faces, obtém-se:

$$\frac{dQ^m}{dt} = -\frac{1}{\prod_{q=1}^N \Delta x_q} \sum_{k=1}^{NFaces} \oint_{Face_k} F \cdot \vec{v} dS + S(Q), \quad (2.15)$$

em que  $NFaces$  é o número de faces de uma célula da malha. Considerando a geometria da malha quadrangular criada, para cada direção  $x_n$ , existem duas faces paralelas cujo o vetor normal aponta para a direção  $x_n$ . Reescrevendo o somatório, obtém-se:

$$\frac{dQ^m}{dt} = -\frac{1}{\prod_{q=1}^N \Delta x_q} \sum_{n=1}^N \left( \oint_{Face_n^+} F \cdot \vec{v} dS + \oint_{Face_n^-} F \cdot \vec{v} dS \right) + S(Q), \quad (2.16)$$

em que  $Face_n^+$  é a face mais à direita com vetor normal na direção  $x_n$ , enquanto  $Face_n^-$  é a face mais à esquerda na mesma direção. É importante observar que apesar dos vetores normais possuírem a mesma direção, seus sentidos são opostos. Resolvendo o produto interno para cada integral, tem-se:

$$\frac{dQ^m}{dt} = -\frac{1}{\prod_{q=1}^N \Delta x_q} \sum_{n=1}^N \left( \oint_{Face_n^+} F_n dS - \oint_{Face_n^-} F_n dS \right) + S(Q) \quad (2.17)$$

Segundo o Teorema do Valor Médio, a igualdade:

$$\oint_{Face_n^{+/-}} F_n dS = \mathcal{F}_n^{+/-} \prod_{\substack{q=1 \\ q \neq n}}^N \Delta x_q \quad (2.18)$$

é válida para algum ponto  $\mathcal{F}_n^{+/-}$  na face correspondente da célula. Estes valores são denominados fluxos numéricos. Substituindo estes valores na Equação 2.17, tem-se que:

$$\frac{dQ^m}{dt} = -\frac{1}{\prod_{q=1}^N \Delta x_q} \sum_{n=1}^N \left( \mathcal{F}_n^+ \prod_{\substack{q=1 \\ q \neq n}}^N \Delta x_q - \mathcal{F}_n^- \prod_{\substack{q=1 \\ q \neq n}}^N \Delta x_q \right) + S(Q) \quad (2.19)$$

Reordenando os termos, obtém-se:

$$\frac{dQ^m}{dt} = \sum_{n=1}^N \frac{(\mathcal{F}_n^- - \mathcal{F}_n^+) \prod_{\substack{q=1 \\ q \neq n}}^N \Delta x_q}{\prod_{q=1}^N \Delta x_q} + S(Q) \quad (2.20)$$

Eliminando os termos  $\Delta x_q$  que se repetem no numerador e no denominador, obtém-

se a formulação para malhas quadriculares do método VF, conforme apresentada na Equação 2.3.

□

A Equação 2.3, que descreve a evolução temporal do método VF, depende do cálculo dos fluxos numéricos  $\mathcal{F}_n^+$  e  $\mathcal{F}_n^-$ , que dependem dos fluxos físicos. O cálculo do fluxo numérico é o responsável pela maior parte da complexidade computacional do método proposto.

A Equação 2.3 é uma equação do tipo  $\frac{dQ}{dt} = f(t, Q)$ , que pode ser resolvida numericamente utilizando métodos de Runge-Kutta, tratados no Capítulo 5. Os valores  $\Delta x_n$  usados em cada célula serão determinados com a técnica da multirresolução adaptativa, em que essa determinará o tamanho de cada célula a cada passo temporal com base nos coeficientes wavelet da solução no tempo anterior. Os coeficientes wavelet são calculados de maneira eficiente utilizando-se a análise multirresolução, apresentada no Capítulo 3.

## 2.2 Forma primitiva e conservativa

A representação das equações do problema estudado na forma da Equação 2.1, denominada forma conservativa, permite calcular os valores dos fluxos de forma simples. Porém, tal representação não permite uma análise da estabilidade numérica do problema. Para isso é necessário converter as equações do problema estudado para a sua forma primitiva, representada como:

$$\frac{\partial W}{\partial t} + P_1(W) \frac{\partial W}{\partial x_1} + \dots + P_N(W) \frac{\partial W}{\partial x_N} = 0 \quad (2.21)$$

em que  $W$  é denominado vetor de variáveis primitivas e os valores  $P_n$  são matrizes.

Utilizando as matrizes  $P_n$ , calcula-se os autovalores do sistema. Tais valores serão usados para calcular o passo temporal  $\Delta t^m$  adequado ao sistema. Em alguns casos, os autovalores são utilizados para resolver o problema de Riemann associado ao fluxo de cada par de células.

O passo temporal  $\Delta t^m$  é calculado a partir da condição de CFL do problema (MOURA; KUBRUSLY, 2012):

$$\sigma = \frac{\lambda_1 \Delta t^m}{\Delta x_1} + \dots + \frac{\lambda_N \Delta t^m}{\Delta x_N} < \sigma_{MAX} , \quad (2.22)$$

em que  $\sigma$  é denominado número de CFL do problema. Reescrevendo a Equação 2.22, obtém-se:

$$\Delta t^m = \sigma \left( \frac{\lambda_1}{\Delta x_1} + \dots + \frac{\lambda_N}{\Delta x_N} \right)^{-1}, \quad (2.23)$$

em que os valores  $\lambda_n$  são os maiores autovalores, em módulo, das respectivas matrizes  $P_n$ . Caso o valor de CFL  $\sigma$  supere o valor  $\sigma_{MAX}$ , o método numérico utilizado para resolver as equações é considerado instável, com isso não se obterá uma solução satisfatória. O valor  $\sigma$  é definido pelo usuário; enquanto o valor  $\sigma_{MAX}$  possui um valor distinto para cada problema a ser resolvido.

De forma geral, a representação nas formas primitiva e conservativa são utilizadas para:

- Forma conservativa:
  - Cálculo dos fluxos;
  - Evolução temporal;
- Forma primitiva:
  - Obtenção dos autovalores;
  - Cálculo do passo temporal;

### 2.3 Cálculo dos fluxos numéricos

Como definidos na demonstração do Teorema 1, os valores numéricos  $\mathcal{F}_n^+$  e  $\mathcal{F}_n^-$  são os valores médios das funções  $F_n$  nas faces  $Face_n^+$  e  $Face_n^-$ , respectivamente. Estes valores nas faces podem ser aproximados por diversas técnicas (TORO, 1999). De forma geral, estas técnicas consistem em aproximar a solução  $Q^m$  nas faces das células, essa aproximação é denotada por  $Q_{\mathcal{C}}^m$ . O próximo passo é calcular os valores  $F_n$  nas faces de cada célula a partir dos valores  $Q_{\mathcal{C}}^m$ ; porém, verifica-se que para cada par de células adjacentes  $\mathcal{C}_1$  e  $\mathcal{C}_2$ , devido suas faces estarem coincidindo, o cálculo dos valores  $F_n$  possui uma condição inicial do tipo:

$$Q(x_n, t = 0) = \begin{cases} Q_{\mathcal{C}_1}^m, & \text{se } x_n \leq 0 \\ Q_{\mathcal{C}_2}^m, & \text{se } x_n > 0 \end{cases} \quad (2.24)$$

Um problema caracterizado por este tipo de condição inicial é denominado problema de Riemann. Neste contexto, o problema de Riemann pode ser interpretado como:

dados duas células adjacentes, como se comporta a transferência de informação entre estas células adjacentes. A solução deste problema é o fluxo numérico entre as células  $\mathcal{C}_1$  e  $\mathcal{C}_2$ , denotado por  $\mathcal{F}_n^+$  para a célula  $\mathcal{C}_1$  e  $\mathcal{F}_n^-$  para a célula  $\mathcal{C}_2$ .

O método para encontrar a solução do problema de Riemann é descrito em Toro (1999). Para aplicar tal método, é necessária uma decomposição em termos de auto-vetores dos valores  $Q_{\mathcal{C}_1}^m$  e  $Q_{\mathcal{C}_2}^m$ .

Resolvendo o problema de Riemann obtém-se uma solução denotada por  $\mathcal{Q}_n$ . Com isso, o fluxo numérico  $\mathcal{F}_n$  entre as células  $\mathcal{C}_1$  e  $\mathcal{C}_2$  é calculado como (TORO, 1999):

$$\mathcal{F}_n = F_n(\mathcal{Q}_n). \quad (2.25)$$

As computações para se obter esses valores podem ser muito dispendiosas. Afim de reduzir este custo, o cálculo da solução do problema de Riemann, dado na Equação 2.25 pode ser calculada utilizando métodos numéricos denominados *Riemann solvers*. Alguns *Riemann solvers* amplamente discutidos são: Harten-Lax-van Leer (HLL), HLLC, Gudonov (TORO, 1999) e HLL Descontinuidades (HLLD) (MIYOSHI; KUSANO, 2005).



### 3 ANÁLISE MULTIRRESOLUÇÃO

O próximo passo da construção do método proposto é o uso de uma malha que se adapte espacialmente à solução. Para construir esta malha, é necessária uma ferramenta matemática que seja capaz de avaliar a suavidade da solução em cada intervalo do domínio. Dessa forma, é possível construir uma malha que é mais refinada nas regiões do domínio que apresentam estruturas localizadas e mais grosseira nas regiões mais suaves. O uso desse tipo de malha computacional visa minimizar o custo computacional na simulação de equações que apresentam estruturas localizadas em sua solução, pois a parte mais dispendiosa do método VF é o cálculo dos fluxos numéricos. Com isso, o uso da malha adaptada reduz o número de fluxos numéricos a serem calculados, pois esta malha apresenta menos células do que a malha regular.

Considerando a solução atual do Problema de Valor Inicial (PVI) como um sinal, uma análise deste sinal no domínio das frequências pode indicar a existência de estruturas localizadas, que aparecem como componentes de alta frequência. Porém, esta análise, utilizando a transformada de Fourier não indica a localização da estrutura localizada no domínio discretizado. Portanto, é necessária uma ferramenta que seja capaz de analisar tanto as frequências quanto a localização dessas no domínio discretizado.

Uma ferramenta matemática criada para esse propósito é a análise wavelet. A análise wavelet surgiu das aplicações na área de geofísica com o intuito de auxiliar no processamento de sinais não estacionários. Posteriormente esta ferramenta foi formalizada por matemáticos.

Tais técnicas wavelet foram aprimoradas por Mallat, que apresenta seus resultados em [Mallat \(1998\)](#), surgindo então a análise multirresolução. Tal abordagem permitiu uma análise de um sinal não estacionário utilizando funções mais simples e algoritmos mais eficientes.

A análise multirresolução divide um sinal em diversos níveis de resolução  $\ell$ . Cada nível de resolução está associado ao número de pontos  $r^\ell$ , igualmente espaçados, usados na representação do sinal. Em que  $r$  é a razão do número de pontos entre dois níveis de refinamento adjacentes. Em geral utiliza-se o valor  $r = 2$  devido a menor complexidade dos algoritmos de decomposição. Com isso, um nível de refinamento superior possui  $2^N$  pontos para cada ponto de um nível inferior, sendo  $N$  o número de dimensões espaciais do problema.

Os valores do maior nível de resolução do sinal, denotado por  $L$ , são os obtidos pela amostragem do mesmo, enquanto os valores dos demais níveis de resolução são calculados utilizando operações de análise a partir do nível mais refinado.

**Definição 1.** *Análise multirresolução constitui-se de uma sequência  $\{V^\ell, \phi\}_{\ell \in \mathbb{Z}}$  de subespaços lineares fechados  $V^\ell$  de  $\mathbb{L}^2(\mathbb{R})$  e uma função associada  $\phi$ , denominada função escala, que satisfaz as seguintes condições:*

$$a) Q(x) \in V^\ell \leftrightarrow Q(x - 2^\ell \kappa) \in V^\ell, \forall (\ell, \kappa) \in \mathbb{Z}^2 ;$$

$$b) V^{\ell+1} \supset V^\ell, \forall \ell \in \mathbb{Z} ;$$

$$c) Q(x) \in V^{\ell+1} \leftrightarrow Q\left(\frac{x}{2}\right) \in V^\ell, \forall \ell \in \mathbb{Z} ;$$

$$d) \lim_{\ell \rightarrow \infty} V^\ell = \bigcap_{\ell=-\infty}^{\infty} V^\ell = \{0\} ;$$

$$e) \lim_{\ell \rightarrow -\infty} V^\ell = \overline{\left(\bigcup_{\ell=-\infty}^{\infty} V^\ell\right)} = \mathbb{L}^2(\mathbb{R});$$

$$f) \text{ Existe uma função } \phi \text{ tal que } \{\phi(x - n)\}_{n \in \mathbb{Z}} \text{ é uma base de Riesz de } V^0.$$

A análise multirresolução de uma função  $Q$  é a representação da decomposição desta função a partir de seu nível mais refinado  $L$  em uma estrutura multi-escala, conforme a equação:

$$Q^L \rightarrow Q_{MR}^L = Q^{L-1} \cup \mathbb{D}^L, \quad (3.1)$$

em que  $Q^{L-1}$  é a representação de  $Q$  em um nível menos refinado, enquanto  $\mathbb{D}^{L-1}$  é a diferença de informação entre os níveis consecutivos  $L$  e  $L - 1$ . O conjunto  $\mathbb{D}^\ell$  é denominado o conjunto de coeficientes wavelet do nível  $\ell$ .

Neste contexto, constrói-se uma estrutura de malhas aninhadas de forma que, a cada espaço  $V^\ell$  representa o domínio discretizado do problema em um nível de refinamento com  $2^{N^\ell}$  elementos, em que  $N$  é o número de dimensões espaciais do problema. Essa estrutura de malhas aninhadas deve respeitar as propriedades da análise multirresolução descritas.

A representação da função  $Q$  nos níveis inferiores da estrutura multi-escala é realizada utilizando as operações de análise adequadas. Posteriormente, é possível reconstruir a malha em seu nível mais refinado utilizando as operações de síntese adequadas e o conjunto de coeficientes wavelet  $\mathbb{D}^\ell$  referente a cada nível  $\ell$  a ser reconstruído.



### 3.1 Operadores de análise

Os operadores de análise, responsáveis pela representação da malha em um nível de resolução inferior, são definidos de acordo com o tipo de estrutura multi-escala utilizado. Ao construir uma estrutura multi-escala, existem duas abordagens, que possuem operações de análise e síntese distintas. A primeira abordagem lida com valores pontuais e a segunda, utilizada neste trabalho, lida com médias celulares.

#### 3.1.1 MR no contexto de valores pontuais

O primeiro contexto é no sentido de uma malha de valores pontuais, ou seja, cada elemento de uma malha de nível  $\ell$  é o valor de um ponto do domínio discretizado.

Desta forma, para um caso unidimensional, considerando uma malha uniforme de pontos  $x_i^\ell$  de dimensão  $\Delta x^\ell$ , define-se uma hierarquia de malhas aninhadas de forma que  $x_i^\ell = x_{2i}^{\ell+1}$  com  $\Delta x^{\ell-1} = 2^l \Delta x^\ell$ ,  $\ell \in \mathbb{Z}$ .

A Figura 3.1 ilustra uma estrutura multi-escala gerada a partir de uma malha de 8 pontos, em que cada linha representa um nível de refinamento, a primeira representa o nível mais refinado e a última representa o nível mais grosseiro, no qual a função é representada por um número mínimo de pontos. As colunas da estrutura multi-escala indicam a região do espaço em que os pontos estão localizados. A Equação

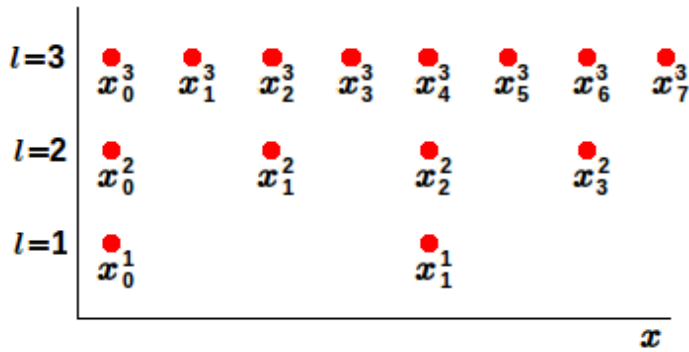


Figura 3.1 - Exemplo de estrutura multi-escala de valores pontuais com 3 níveis.

ção 3.2 representa a operação de análise para uma estrutura multi-escala em que os elementos são valores pontuais:

$$Q_{\ell-1, i}^m = Q_{\ell, 2i}^m \quad (3.2)$$

Tal abordagem, que utiliza valores pontuais, é utilizada em malhas cujos elementos são valores pontuais. Este tipo de representação é adequado para utilizar técnicas como a de diferenças finitas.

### 3.1.2 MR no contexto de médias celulares

A outra abordagem de estrutura multi-escala, é interpretá-la como um conjunto de células com valores médios referentes ao intervalo que cada célula representa.

Desta forma, para um caso unidimensional, considerando uma malha uniforme de células  $\Omega_i^\ell = [x_i^\ell, x_{i+1}^\ell]$  de dimensão  $\Delta x^\ell$ , define-se uma hierarquia de malhas aninhadas de forma que  $\Omega_i^\ell = \Omega_{2i}^{\ell+1} \cup \Omega_{2i+1}^{\ell+1}$  com  $\Delta x^{\ell-l} = 2^l \Delta x^\ell$ ,  $l \in \mathbb{Z}$ .

A Figura 3.2 ilustra uma estrutura multi-escala gerada a partir de uma malha de 8 células, em que cada linha representa um nível de refinamento. A primeira representa o nível mais refinado e a última representa o nível mais grosseiro, em que a função é representada por um número mínimo de células. As colunas da estrutura multi-escala indicam a região do espaço em que as células estão representadas.

$\Omega_0^3$	$\Omega_1^3$	$\Omega_2^3$	$\Omega_3^3$	$\Omega_4^3$	$\Omega_5^3$	$\Omega_6^3$	$\Omega_7^3$
$\Omega_0^2$		$\Omega_1^2$		$\Omega_2^2$		$\Omega_3^2$	
$\Omega_0^1$				$\Omega_1^1$			
$\Omega_0^0$							

Figura 3.2 - Exemplo de estrutura multi-escala de médias celulares com 4 níveis.

As Equações 3.3, a seguir, representam as operações de análise usadas no contexto de médias celulares. Estas operações são interpretadas como a média dos valores das células do nível superior. As Equações 3.3a, 3.3b e 3.3c são utilizadas para os casos unidimensional, bidimensional e tridimensional, respectivamente.

$$Q_{\ell-1, i}^m = \frac{1}{2} \left( Q_{\ell, 2i}^m + Q_{\ell, 2i+1}^m \right) \quad (3.3a)$$

$$Q_{\ell-1, i, j}^m = \frac{1}{4} \left( Q_{\ell, 2i, 2j}^m + Q_{\ell, 2i+1, 2j}^m + Q_{\ell, 2i, 2j+1}^m + Q_{\ell, 2i+1, 2j+1}^m \right) \quad (3.3b)$$

$$\begin{aligned}
Q_{\ell-1, i, j, h}^m = \frac{1}{8} & \left( Q_{\ell, 2i, 2j, 2h}^m + Q_{\ell, 2i+1, 2j, 2h}^m + Q_{\ell, 2i, 2j+1, 2h}^m \right. \\
& + Q_{\ell, 2i+1, 2j+1, 2h}^m + Q_{\ell, 2i, 2j, 2h+1}^m + Q_{\ell, 2i+1, 2j, 2h+1}^m \\
& \left. + Q_{\ell, 2i, 2j+1, 2h+1}^m + Q_{\ell, 2i+1, 2j+1, 2h+1}^m \right) \quad (3.3c)
\end{aligned}$$

Como este trabalho lida com método dos volumes finitos, a abordagem de uma estrutura multi-escala por médias celulares é a mais adequada, pois os elementos de cada célula do domínio discretizado são justamente médias celulares.

Definidas as operações de análise, resta definir as operações de síntese, que são responsáveis pela obtenção dos coeficientes wavelet, usados para medir a regularidade local da malha.

### 3.2 Operadores de síntese – Contexto da médias celulares

As operações de síntese, utilizadas para reconstruir a malha em um nível mais refinado, dependem, além dos valores da malha em um nível menos refinado, dos coeficientes wavelet referentes a cada célula do nível a ser reconstruído.

Os coeficientes wavelet são calculados a partir de operações de predição. Seja  $d_{\Omega}^{\ell}$  o coeficiente wavelet de uma célula  $\Omega$  pertencente ao nível  $\ell$ , este valor pode ser calculado como:

$$d_{\Omega}^{\ell} = Q_{\Omega^{\ell}}^m - \hat{Q}_{\Omega^{\ell}}^m, \quad (3.4)$$

em que  $\hat{Q}_{\Omega^{\ell}}^m$  é o resultado da operação de predição sobre o valor da célula. O coeficiente wavelet é interpretado como o erro cometido pela predição em relação ao valor exato da célula, por isso este valor é usado para medir a regularidade local da solução, pois um coeficiente alto significa um alto erro de predição, isso indica uma não suavidade local da solução.

Desta forma, os valores dos níveis mais refinados podem ser reconstruídos, sem perdas, como:

$$Q_{\Omega^{\ell}}^m = d_{\Omega}^{\ell} + \hat{Q}_{\Omega^{\ell}}^m \quad (3.5)$$

O Algoritmo 1, a seguir, representa o processo de obtenção dos coeficientes wavelet, enquanto o Algoritmo 2 representa a reconstrução da malha a partir dos coeficientes wavelet.

Ambos os processos discutidos dependem dos valores  $\hat{Q}^m$  calculados a partir das ope-

---

**Algoritmo 1** Obtenção de coeficientes wavelet.

---

**Require:** Seja  $Q^\ell$  o valor da média celular de uma célula  $\Omega$  no nível de refinamento  $\ell$ .

**Require:** Seja  $L$  o nível mais refinado da malha.

**Require:** Entrada: Todas as médias celulares no nível mais refinado  $L$ .

**for**  $\ell = L - 1 \rightarrow 1$  **do**

**for** toda célula  $\Omega$  pertencente ao nível  $\ell$  **do**

    Calcular  $\hat{Q}_{\Omega_\ell}^m$  usando a Sub-equação 3.3 adequada ao número de dimensões;

    Calcular os coeficientes wavelet  $d^{\ell+1}$  referentes as células do nível mais refinado da célula  $\Omega$  utilizando a Equação 3.4;

**end for**

**end for**

---

---

**Algoritmo 2** Reconstrução dos valores no nível mais refinado.

---

**Require:** Seja  $Q^\ell$  o valor da média celular de uma célula  $\Omega$  no nível de refinamento  $\ell$ .

**Require:** Seja  $L$  o nível mais refinado da malha.

**Require:** Entrada: Todas as médias celulares no nível mais refinado  $L$ .

**for**  $\ell = 1 \rightarrow L$  **do**

**for** toda célula  $\Omega$  pertencente ao nível  $\ell$  **do**

    Reconstruir os valores referentes as células do nível mais refinado da célula  $\Omega$  utilizando a Equação 3.5 e os coeficientes wavelet obtidos com o Algoritmo 1;

**end for**

**end for**

---

rações de predição. Em Roussel (2003) são apresentados os operadores de predição utilizados neste trabalho.

No contexto da análise multirresolução com uso de médias celulares, os operadores de predição consistem em operações de interpolação, utilizando as médias celulares de um nível inferior, afim de estimar o valor das médias celulares do nível superior.

Essas interpolações são realizadas, independente do número de dimensões, utilizando constantes  $\gamma \in \mathbb{R}$  que dependem do número de células adjacentes  $z \in \mathbb{N}$  usadas em cada sentido na interpolação, ou seja, caso sejam utilizadas, além da célula central, uma célula anterior e uma posterior para o cálculo da interpolação, utiliza-se  $z = 1$ .

Neste trabalho, as operações de predição são realizadas utilizando  $z = 1$ , pois pretende-se minimizar a complexidade computacional do processo de obtenção dos coeficientes wavelet. Em Roussel (2003), é apresentado o valor  $\gamma_1 = \frac{1}{8}$ , associado ao valor  $z = 1$ , conforme o Teorema 2.

**Teorema 2.** *Interpolação de médias celulares em níveis mais refinados: Seja uma célula  $\Omega_i^\ell$ , que representa um intervalo  $[\alpha_1; \alpha_2]$  do domínio, cuja média celular  $Q_{\ell, i}^m$  é decomposta em células  $\Omega_{2i}^{\ell+1}$  e  $\Omega_{2i+1}^{\ell+1}$  de valores  $Q_{\ell+1, 2i}^m$  e  $Q_{\ell+1, 2i+1}^m$ , respectivamente. Estes valores são aproximados pelos valores  $\hat{Q}$  definidos como:*

$$\hat{Q}_{\ell+1, 2i}^m = Q_{\ell, i}^m - \frac{1}{8}(Q_{\ell, i+1}^m - Q_{\ell, i-1}^m) \quad (3.6a)$$

$$\hat{Q}_{\ell+1, 2i+1}^m = Q_{\ell, i}^m + \frac{1}{8}(Q_{\ell, i+1}^m - Q_{\ell, i-1}^m) \quad (3.6b)$$

*Demonstração.* Utilizando o Teorema do Valor Médio, os valores  $Q_{\ell, i}^m$ ,  $Q_{\ell+1, 2i}^m$  e  $Q_{\ell+1, 2i+1}^m$  podem ser escritos como:

$$Q_{\ell, i}^m = \frac{1}{\Delta x} \int_{\alpha_1}^{\alpha_2} U dx \quad (3.7a)$$

$$Q_{\ell+1, 2i}^m = \frac{1}{\frac{\Delta x}{2}} \int_{\alpha_1}^{\alpha_1 + \frac{\Delta x}{2}} U dx \quad (3.7b)$$

$$Q_{\ell+1, 2i+1}^m = \frac{1}{\frac{\Delta x}{2}} \int_{\alpha_1 + \frac{\Delta x}{2}}^{\alpha_2} U dx \quad (3.7c)$$

Somando as Equações 3.7, obtém-se:

$$Q_{\ell, i}^m + Q_{\ell+1, 2i}^m + Q_{\ell+1, 2i+1}^m = \frac{1}{\Delta x} \int_{\alpha_1}^{\alpha_2} U dx + \frac{1}{\frac{\Delta x}{2}} \int_{\alpha_1}^{\alpha_1 + \frac{\Delta x}{2}} U dx + \frac{1}{\frac{\Delta x}{2}} \int_{\alpha_1 + \frac{\Delta x}{2}}^{\alpha_2} U dx \quad (3.8)$$

Reordenando os termos:

$$Q_{\ell+1, 2i}^m = -Q_{\ell+1, 2i+1}^m - Q_{\ell, i}^m + \frac{3}{\Delta x} \int_{\alpha_1}^{\alpha_2} U dx \quad (3.9)$$

Substituindo  $\frac{1}{\Delta x} \int_{\alpha_1}^{\alpha_2} U dx$ , conforme a Equação 3.7a:

$$Q_{\ell+1, 2i}^m = -Q_{\ell+1, 2i+1}^m + 2Q_{\ell, i}^m \quad (3.10)$$

Substituindo o termo  $Q_{\ell+1, 2i+1}^m$ :

$$Q_{\ell+1, 2i}^m = -\frac{1}{\frac{\Delta x}{2}} \int_{\alpha_1 + \frac{\Delta x}{2}}^{\alpha_2} U dx + 2Q_{\ell, i}^m \quad (3.11)$$

A integral pode ser aproximada como:

$$\int_{\alpha_1 + \frac{\Delta x}{2}}^{\alpha_2} U dx \approx \left[ U \left( \alpha_1 + \frac{\Delta x}{2} \right) + U(\alpha_2) \right] \frac{1}{2} \frac{\Delta x}{2} \quad (3.12)$$

Substituindo na Equação 3.11:

$$Q_{\ell+1, 2i}^m \approx -\frac{1}{2} \left[ U \left( \alpha_1 + \frac{\Delta x}{2} \right) + U(\alpha_2) \right] + 2Q_{\ell, i}^m \quad (3.13)$$

O valor  $U(\alpha_2)$  pode ser interpolado linearmente como:

$$U(\alpha_2) \approx U \left( \alpha_1 + \frac{\Delta x}{2} \right) + \left( \frac{U(\alpha_2 + \frac{\Delta x}{2}) - U(\alpha_1 - \frac{\Delta x}{2})}{2\Delta x} \right) \frac{\Delta x}{2} \quad (3.14)$$

Substituindo na Equação 3.13:

$$Q_{\ell+1, 2i}^m \approx -\frac{1}{2} \left[ 2U \left( \alpha_1 + \frac{\Delta x}{2} \right) + \frac{U(\alpha_2 + \frac{\Delta x}{2}) - U(\alpha_1 - \frac{\Delta x}{2})}{4} \right] + 2Q_{\ell, i}^m \quad (3.15)$$

Considerando o valor central das células como uma aproximação de sua média, obtém-se:

$$U \left( \alpha_1 + \frac{\Delta x}{2} \right) \approx Q_{\ell, i}^m \quad (3.16a)$$

$$U \left( \alpha_1 - \frac{\Delta x}{2} \right) \approx Q_{\ell, i-1}^m \quad (3.16b)$$

$$U \left( \alpha_2 + \frac{\Delta x}{2} \right) \approx Q_{\ell, i+1}^m \quad (3.16c)$$

Substituindo na Equação 3.15:

$$Q_{\ell+1, 2i}^m \approx -\frac{1}{2} \left[ 2Q_{\ell, i}^m + \frac{Q_{\ell, i+1}^m - Q_{\ell, i-1}^m}{4} \right] + 2Q_{\ell, i}^m \quad (3.17)$$

Reordenando os termos, obtém-se:

$$Q_{\ell+1, 2i}^m \approx Q_{\ell, i}^m - \frac{1}{8} (Q_{\ell, i+1}^m - Q_{\ell, i-1}^m) \quad (3.18)$$

Desta forma, define-se o operador de predição  $\hat{Q}_{\ell+1, 2i}^m$ , que aproxima o valor de  $Q_{\ell+1, 2i}^m$ . Substituindo esta aproximação na Equação 3.10, obtém-se a aproximação utilizada para a predição de  $Q_{\ell+1, 2i+1}^m$ . Desta forma, é possível verificar o valor  $\gamma_1 = \frac{1}{8}$ .  $\square$

### 3.2.1 Operadores de predição – caso unidimensional

Para o caso unidimensional (HARTEN, 1993) propõe um operador de predição baseado em uma interpolação linear das médias celulares utilizando os valores das células adjacentes.

Seja  $Q_{\ell, i}^m$  a média celular da célula de posição  $i$  do nível  $\ell$  da estrutura multi-escala. A predição dos valores médios  $\hat{Q}_{\ell+1, 2i}^m$  e  $\hat{Q}_{\ell+1, 2i+1}^m$ , referentes ao nível superior da célula  $Q_{\ell, i}^m$  na estrutura multi-escala de uma análise multirresolução com  $r = 2$ , é calculada como:

$$\hat{Q}_{\ell+1, 2i}^m = Q_{\ell, i}^m - A \quad (3.19a)$$

$$\hat{Q}_{\ell+1, 2i+1}^m = Q_{\ell, i}^m + A \quad (3.19b)$$

em que:

$$A = \sum_{n=1}^z \gamma_n (Q_{\ell, i+n}^m - Q_{\ell, i-n}^m) \quad (3.20)$$

Utilizando os valores  $z = 1$  e  $\gamma_1 = \frac{1}{8}$  propostos, reescreve-se o valor  $A$  da Equação 3.20 como:

$$A = \frac{1}{8}(Q_{\ell, i+1}^m - Q_{\ell, i-1}^m) \quad (3.21)$$

### 3.2.2 Operadores de predição – caso bidimensional

Para o caso bidimensional, (BIHARI; HARTEN, 1997) obtiveram os seguintes valores ao aplicar uma abordagem com produtos tensoriais.

Seja  $Q_{\ell, i, j}^m$  a média celular da célula de posição  $i$  na direção  $x_1$  e posição  $j$  na direção  $x_2$  do nível  $\ell$  da estrutura multi-escala, a predição dos valores médios referentes às células do nível superior da célula  $Q_{\ell, i, j}^m$  na estrutura multi-escala de uma análise multirresolução, com  $r = 2$ , é calculada como:

$$\hat{Q}_{\ell+1, 2i, 2p}^m = Q_{\ell, i, j}^m - A - B + D \quad (3.22a)$$

$$\hat{Q}_{\ell+1, 2i+1, 2p}^m = Q_{\ell, i, j}^m + A - B - D \quad (3.22b)$$

$$\hat{Q}_{\ell+1, 2i, 2p+1}^m = Q_{\ell, i, j}^m - A + B - D \quad (3.22c)$$

$$\hat{Q}_{\ell+1, 2i+1, 2p+1}^m = Q_{\ell, i, j}^m + A + B + D \quad (3.22d)$$

em que:

$$A = \sum_{n=1}^z \gamma_n (Q_{\ell, i+n, j}^m - Q_{\ell, i-n, j}^m) \quad (3.23a)$$

$$B = \sum_{q=1}^z \gamma_q (Q_{\ell, i, j+q}^m - Q_{\ell, i, j-q}^m) \quad (3.23b)$$

$$D = \sum_{n=1}^z \gamma_n \sum_{q=1}^z \gamma_q \left( \begin{aligned} &Q_{\ell, i+n, j+q}^m - Q_{\ell, i+n, j-q}^m \\ &- Q_{\ell, i-n, j+q}^m + Q_{\ell, i-n, j-q}^m \end{aligned} \right) \quad (3.23c)$$

Utilizando os valores  $z = 1$  e  $\gamma_1 = \frac{1}{8}$  propostos, reescrevem-se os valores  $A$ ,  $B$  e  $D$  das Equações 3.23 como:

$$A = \frac{1}{8} (Q_{\ell, i+1, j}^m - Q_{\ell, i-1, j}^m) \quad (3.24a)$$

$$B = \frac{1}{8} (Q_{\ell, i, j+1}^m - Q_{\ell, i, j-1}^m) \quad (3.24b)$$

$$D = \frac{1}{64} (Q_{\ell, i+1, j+1}^m - Q_{\ell, i+1, j-1}^m - Q_{\ell, i-1, j+1}^m + Q_{\ell, i-1, j-1}^m) \quad (3.24c)$$

### 3.2.3 Operadores de predição – caso tridimensional

Para o caso tridimensional, (ROUSSEL, 2003) aplicou a mesma metodologia usada no caso bidimensional.

Seja  $Q_{\ell, i, j, h}^m$  a média celular da célula de posição  $i$  na direção  $x_1$ , posição  $j$  na direção  $x_2$  e posição  $h$  na direção  $x_3$  do nível  $\ell$  da estrutura multi-escala, a predição dos valores médios referentes às células do nível superior da célula  $Q_{\ell, i, j, h}^m$  na estrutura



multi-escala de uma análise multirresolução, com  $r = 2$ , é calculada como<sup>1</sup>:

$$\hat{Q}_{\ell+1, 2i, 2p, 2h}^m = Q_{\ell, i, j, h}^m - A - B - C + D + E + G - H \quad (3.25a)$$

$$\hat{Q}_{\ell+1, 2i+1, 2p, 2h}^m = Q_{\ell, i, j, h}^m + A - B - C - D + E - G + H \quad (3.25b)$$

$$\hat{Q}_{\ell+1, 2i, 2p+1, 2h}^m = Q_{\ell, i, j, h}^m - A + B - C - D - E + G + H \quad (3.25c)$$

$$\hat{Q}_{\ell+1, 2i, 2p, 2h+1}^m = Q_{\ell, i, j, h}^m - A - B + C + D - E - G + H \quad (3.25d)$$

$$\hat{Q}_{\ell+1, 2i+1, 2p+1, 2h}^m = Q_{\ell, i, j, h}^m + A + B - C + D - E - G - H \quad (3.25e)$$

$$\hat{Q}_{\ell+1, 2i+1, 2p, 2h+1}^m = Q_{\ell, i, j, h}^m + A - B + C - D - E + G - H \quad (3.25f)$$

$$\hat{Q}_{\ell+1, 2i, 2p+1, 2h+1}^m = Q_{\ell, i, j, h}^m - A + B + C - D + E - G - H \quad (3.25g)$$

$$\hat{Q}_{\ell+1, 2i+1, 2p+1, 2h+1}^m = Q_{\ell, i, j, h}^m + A + B + C - D - E - G + H \quad (3.25h)$$

em que:

$$A = \sum_{n=1}^z \gamma_n (Q_{\ell, i+n, j, h}^m - Q_{\ell, i-n, j, h}^m) \quad (3.26a)$$

$$B = \sum_{q=1}^z \gamma_q (Q_{\ell, i, j+q, h}^m - Q_{\ell, i, j-q, h}^m) \quad (3.26b)$$

$$C = \sum_{w=1}^z \gamma_w (Q_{\ell, i, j, h+w}^m - Q_{\ell, i, j, h-w}^m) \quad (3.26c)$$

$$D = \sum_{n=1}^z \gamma_n \sum_{q=1}^z \gamma_q \left( Q_{\ell, i+n, j+q, h}^m - Q_{\ell, i+n, j-q, h}^m - Q_{\ell, i-n, j+q, h}^m + Q_{\ell, i-n, j-q, h}^m \right) \quad (3.26d)$$

$$E = \sum_{q=1}^z \gamma_q \sum_{w=1}^z \gamma_w \left( Q_{\ell, i, j+q, h+w}^m - Q_{\ell, i, j+q, h-w}^m - Q_{\ell, i, j-q, h+w}^m + Q_{\ell, i, j-q, h-w}^m \right) \quad (3.26e)$$

$$G = \sum_{n=1}^z \gamma_n \sum_{w=1}^z \gamma_w \left( Q_{\ell, i+n, j, h+w}^m - Q_{\ell, i+n, j, h-w}^m - Q_{\ell, i-n, j, h+w}^m + Q_{\ell, i-n, j, h-w}^m \right) \quad (3.26f)$$

<sup>1</sup>Tais valores foram obtidos ao corrigir as formulações apresentadas em (ROUSSEL, 2003)

$$\begin{aligned}
H = \sum_{n=1}^z \gamma_n \sum_{q=1}^z \gamma_q \sum_{w=1}^z \gamma_w & \left( Q_{\ell, i+n, j+q, h+w}^m - Q_{\ell, i+n, j+q, h-w}^m \right. \\
& - Q_{\ell, i+n, j-q, h+w}^m - Q_{\ell, i-n, j+q, h+w}^m \\
& + Q_{\ell, i+n, j-q, h-w}^m + Q_{\ell, i-n, j+q, h-w}^m \\
& \left. + Q_{\ell, i-n, j-q, h+w}^m - Q_{\ell, i-n, j-q, h-w}^m \right) \quad (3.26g)
\end{aligned}$$

Utilizando os valores  $z = 1$  e  $\gamma_1 = \frac{1}{8}$  propostos, reescrevem-se os valores  $A, B, C, D, E, G$  e  $H$  das Equações 3.26 como:

$$A = \frac{1}{8}(Q_{\ell, i+1, j, h}^m - Q_{\ell, i-1, j, h}^m) \quad (3.27a)$$

$$B = \frac{1}{8}(Q_{\ell, i, j+1, h}^m - Q_{\ell, i, j-1, h}^m) \quad (3.27b)$$

$$C = \frac{1}{8}(Q_{\ell, i, j, h+1}^m - Q_{\ell, i, j, h-1}^m) \quad (3.27c)$$

$$\begin{aligned}
D = \frac{1}{64} & \left( Q_{\ell, i+1, j+1, h}^m - Q_{\ell, i+1, j-1, h}^m \right. \\
& \left. - Q_{\ell, i-1, j+1, h}^m + Q_{\ell, i-1, j-1, h}^m \right) \quad (3.27d)
\end{aligned}$$

$$\begin{aligned}
E = \frac{1}{64} & \left( Q_{\ell, i, j+1, h+1}^m - Q_{\ell, i, j+1, h-1}^m \right. \\
& \left. - Q_{\ell, i, j-1, h+1}^m + Q_{\ell, i, j-1, h-1}^m \right) \quad (3.27e)
\end{aligned}$$

$$\begin{aligned}
G = \frac{1}{64} & \left( Q_{\ell, i+1, j, h+1}^m - Q_{\ell, i+1, j, h-1}^m \right. \\
& \left. - Q_{\ell, i-1, j, h+1}^m + Q_{\ell, i-1, j, h-1}^m \right) \quad (3.27f)
\end{aligned}$$

$$\begin{aligned}
H = \frac{1}{512} & \left( Q_{\ell, i+1, j+1, h+1}^m - Q_{\ell, i+1, j+1, h-1}^m - Q_{\ell, i+1, j-1, h+1}^m \right. \\
& - Q_{\ell, i-1, j+1, h+1}^m + Q_{\ell, i+1, j-1, h-1}^m + Q_{\ell, i-1, j+1, h-1}^m \\
& \left. + Q_{\ell, i-1, j-1, h+1}^m - Q_{\ell, i-1, j-1, h-1}^m \right) \quad (3.27g)
\end{aligned}$$

## 4 CONSTRUÇÃO DA MALHA ADAPTATIVA

Tradicionalmente, a malha computacional utilizada na implementação de um método numérico tem a mesma forma e tamanho em todas as células. Isso faz com que a qualidade da solução numérica obtida dependa de seu refinamento, além das condições de consistência e estabilidade. Uma malha pouco refinada faz com que algumas informações da solução passem despercebidas. Por outro lado, uma malha excessivamente refinada acarreta em um custo computacional muito grande. O uso de uma malha adaptativa visa mesclar a qualidade de uma solução com grande refinamento com o baixo custo computacional de uma malha pouco refinada.

A malha adaptativa é uma representação da malha computacional em diversos níveis de resolução  $\ell = 0, \dots, L$ , com  $\ell \in \mathbb{N}$ . Cada nível de resolução representa a mesma malha regular com diferentes números de células, conforme a Figura 4.1.

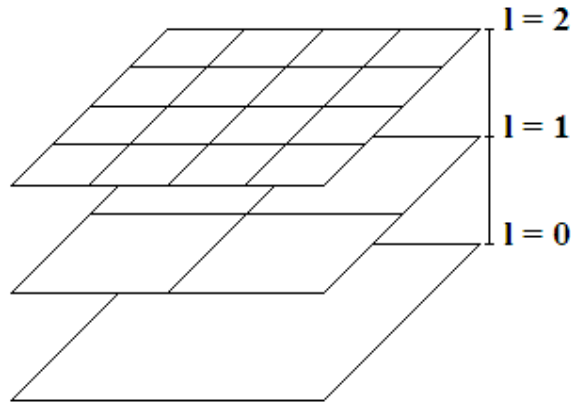


Figura 4.1 - Hierarquia de malhas bidimensionais em diversos níveis de resolução.

O número de células da malha do nível  $\ell$  é dado por  $r^{N\ell}$ , em que  $N$  é o número de dimensões espaciais do problema e  $r \in \mathbb{N}$ , com  $r \geq 2$ . ou seja uma malha em um nível superior é  $r^N$  vezes mais refinada do que uma malha em um nível inferior. O valor  $r$  depende da estratégia de multirresolução adotada. Neste trabalho, são utilizadas decomposições em dois elementos, portanto tem-se  $r = 2$ . Seja  $\Delta x_{n,\ell}$  o tamanho de uma célula do nível  $\ell$  e na direção  $x_n$ , como as malhas propostas neste trabalho são regulares em todos os níveis, tem-se que:

$$\Delta x_{1,\ell} = \Delta x_{2,\ell} = \dots = \Delta x_{N,\ell} \quad \forall \ell \in [0; L] \quad (4.1)$$

Então, define-se  $\Delta x_\ell$  como o valor das arestas das células do nível de resolução  $\ell$ .

Uma malha adaptada, Figura 4.2, é construída a partir das células dos diversos níveis de resolução da malha de forma que:

- As células ocupem todo o domínio, sem sobreposição;
- Seja tão refinada quanto o necessário para representar a solução sem perdas;
- Duas células adjacentes devem estar ou no mesmo nível ou com diferença de apenas um nível de refinamento. (*Graded-tree*) (COHEN, 2003)

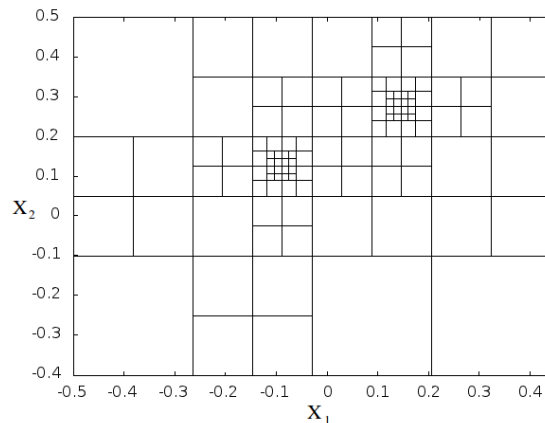


Figura 4.2 - Exemplo de malha adaptada.

Na prática, considerando o segundo item, uma região suave da solução pode ser representada por poucas células sem haver perdas, enquanto uma região menos suave precisa ser representada por mais células. Isto torna a escolha dos níveis de refinamento das células que representarão as diferentes regiões da solução dependente de uma ferramenta capaz de avaliar a sua suavidade em todas as regiões do seu domínio. A análise multirresolução, discutida na Capítulo 3, é uma ferramenta adequada para a avaliação da suavidade da solução (HARTEN, 1993).

A exigência do terceiro item da lista está relacionada à Cohen (2003), que obteve resultados comprovando a maior eficiência de malhas com esta característica, além de ser importante para a construção de algoritmos para a execução da técnica de tempo local, proposta no Capítulo 6.

Em problemas evolutivos com estruturas localizadas na solução, o uso de uma malha adaptada à solução permite minimizar o custo computacional de sua resolução numérica, pois representa a solução com uma quantidade mínima de células, diminuindo o número de fluxos intercelulares que devem ser cálculos para a evolução temporal.

O método proposto garante que a solução no instante  $t^m$  estará bem representada na malha adaptada, mas não garante que a malha estará adequada para receber a solução após a evolução temporal. Então, como precaução, após a malha adaptada ser criada para a solução no instante  $t^m$ , refinam-se todas as células pertencentes a níveis de resolução inferiores a  $L$ . Os seguintes passos descrevem o procedimento usado em cada iteração do método proposto. Para toda iteração do método numérico faça:

- a) Usar a análise multirresolução para decompor a solução em vários níveis de resolução e obter os coeficientes wavelet (Algoritmo 1).
- b) Conforme a técnica apresentada na Seção 4.2, construir a malha adaptada a partir dos coeficientes wavelet obtidos (Algoritmo 3).
- c) Refinar todas as células que estão em um nível de refinamento inferior a  $L$ .
- d) Realizar a evolução temporal.
- e) Reconstruir a malha no nível  $L$  (Algoritmo 2).

Nos Capítulos 5 e 6 são discutidas as técnicas de evolução temporal utilizadas neste trabalho.

#### 4.1 Implementação da malha adaptativa

A implementação computacional das malha adaptativa é realizada sobre a hierarquia de malhas discutida no início do capítulo. A hierarquia de malhas proposta pode ser implementada como uma estrutura de árvore.

**Definição 2.** *Árvore: árvore é uma estrutura de dados caracterizada por:*

- *Ou não tem elemento algum (árvore vazia);*

- Ou possui um elemento e  $r^N$  ponteiros para outras árvores que não pertencem a hierarquia. Estas novas árvores são denominadas filhos da árvore que contém os ponteiros para elas.

As seguintes definições são importantes para o desenvolvimento da malha adaptativa:

**Definição 3.** *Nó de uma árvore:* Toda árvore que possui filhos e é filho de outra árvore.

**Definição 4.** *Folha de uma árvore:* Toda árvore que não possui filhos e é filho de outra árvore.

**Definição 5.** *Raiz de uma árvore:* Toda árvore que não é filho de outra árvore.

**Definição 6.** *Profundidade de um nó:* É o número de nós que se percorre para ir da raiz até o nó.

**Definição 7.** *Nível de uma árvore:* O conjunto de nós com a mesma profundidade.

A estrutura de uma árvore pode ser implementada para representar os diferentes níveis da malha, de forma que os nós de cada nível da árvore representam as células da malha neste nível e a raiz da árvore é a malha menos refinada. O refinamento de uma célula é representado pelos filhos desta célula.

Como um exemplo, para uma malha bidimensional ( $N = 2$ ) e utilizando uma estratégia de multirresolução com decomposições de dois níveis ( $r = 2$ ), a árvore proposta para representar esta malha deve possuir  $r^N = 4$  filhos. A Figura 4.3 ilustra a relação entre esta árvore e a hierarquia de malhas bidimensionais. A Figura 4.3a representa as diversas malhas da hierarquia, cada célula é representada por uma estrutura de dados (círculos vermelhos). A Figura 4.3b ilustra uma árvore quartenária em que cada nó tem 4 tipos diferentes de filhos, cada nível de refinamento da malha computacional representa um nível na árvore. Neste exemplo, as células podem ser divididas em 4 grupos: verde, marrom, azul e laranja. Cada um destes grupos apresenta uma maneira distinta para procurar seus vizinhos para calcular os fluxos numéricos.

A Figura 4.3c ilustra a relação entre o refinamento da malha e os filhos do nó que representa a célula refinada, é importante observar que o padrão da posição das cores permanece o mesmo em todos os refinamentos realizados.

Devido a esta implementação, que é muito similar a estrutura formada na análise multirresolução, as células, além de conter a solução no espaço que ela representa, podem conter o coeficiente wavelet referente a aquela região, tornando o processo de multirresolução integrado a malha.

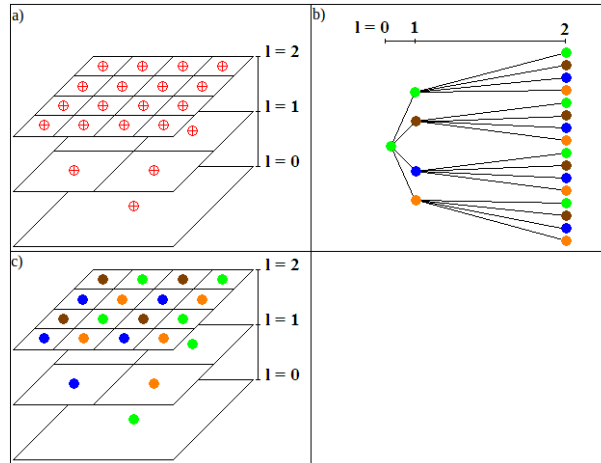


Figura 4.3 - Implementação da hierarquia de malhas como uma árvore.

## 4.2 Multirresolução adaptativa

Criada a estrutura computacional para representar a malha adaptativa nos diversos níveis de refinamento, resta definir as condições em que determinada região da malha será representada em um nível de resolução inferior.

A Multirresolução adaptativa é uma técnica capaz de determinar a suavidade de diferentes intervalos de uma função utilizando seus coeficientes wavelet, e a partir daí determinar o nível de refinamento adequado para aquele intervalo. Conforme apresentado na Seção 3.2, um coeficiente wavelet baixo indica suavidade na região referente a esse coeficiente; enquanto um coeficiente alto indica o contrário (HARTEN, 1993; MALLAT, 1998; DOMINGUES et al., 2011).

Para a construção de uma malha adaptada para o instante  $t^m$  e que seja capaz de representar o instante  $t^{m+1}$ , inicialmente deve-se representar a solução em  $t^m$  no nível mais refinado. Então verificar, para cada conjunto de células implementadas como irmãs, se todas estão presentes na malha atual. Caso estejam, essas células podem ser mescladas em uma célula menos refinada, representada em um nível inferior, caso sejam satisfeitas as condições:

- a) Os valores absolutos dos coeficientes wavelet  $d_j^k$  das células irmãs devem ser menores do que um valor  $\epsilon$  escolhido previamente. Este valor  $\epsilon$  é chamado de valor de limiar (*threshold*), ele indica se o coeficiente wavelet tem um valor significativo ou não. Caso o coeficiente wavelet não seja significativo, a região referente a este é suave e pode ser representada em um nível menos

refinado sem perdas;

- b) Nenhuma célula adjacente ao conjunto de células a serem mescladas deve estar em um nível de resolução superior ao das células do conjunto antes do refinamento. Esta condição faz com que a malha atenda as condições do *Graded-tree* citado anteriormente.

Este processo de verificação da malha e mescla de conjuntos de células irmãs é repetido até que nenhuma célula possa mais ser mesclada.

Montada a malha adaptada, aplica-se o refinamento de um nível em todas as células da malha com o intuito de adaptá-la também à solução no instante  $t^{m+1}$ . Este processo deve ser repetido para todos os níveis de resolução, conforme o Algoritmo 3.

---

**Algoritmo 3** Construção da malha adaptativa.

---

**Require:** Considere a malha adaptada com todas as células no nível mais refinado  $L$ .

**Require:** O valor de *threshold*  $\epsilon$  escolhido previamente.

Obter os conjunto de coeficientes wavelet  $\mathbb{D}^\ell$  referentes a todos os níveis  $\ell$ , obtidos através do Algoritmo 1;

**for**  $\ell = L - 1 : \ell \geq \ell_{\min} : \ell = \ell - 1$  **do**

**for** Para toda célula  $\mathcal{C}$  pertencente ao nível  $\ell$  **do**

**if** Todas as célula filhas de  $\mathcal{C}$  pertencem a malha adaptada. **then**

**if** O maior coeficiente wavelet pertencente a célula  $\mathcal{C}$  for menor que  $\epsilon$  **then**

**if** O nível de refinamento de todas as células adjacentes ao conjunto a ser mesclado for maior do que  $\ell$  **then**

                    Remover as células filhas de  $\mathcal{C}$  da malha adaptada;

                    Inserir célula  $\mathcal{C}$  na malha adaptada;

**end if**

**end if**

**end for**

**end for**

**end for**

Refinar todas as células que estão em um nível de refinamento menor do que  $L$ ;

---

Construída a malha adaptada, a evolução dos valores da solução depende do cálculos dos fluxos numéricos, discutidos na Seção 2.3. Porém, as técnicas discutidas são válidas para células do mesmo nível de refinamento. Para a malha proposta, em que as células adjacentes podem possuir diferença de até 1 nível de refinamento, precisa-



se do valor das células adjacentes no mesmo nível de refinamento da célula cujos fluxos serão calculados. Estes valores podem obtidos utilizando as Equações 3.3 e 3.5.

No próximo capítulo são apresentados os métodos de discretização temporal utilizados para realizar a evolução temporal do método proposto.



## 5 MÉTODOS DE EVOLUÇÃO TEMPORAL

Neste capítulo são apresentados métodos de Runge-Kutta explícitos tradicionais, e contínuo, e a versão compacta desses métodos. Estes métodos são utilizados para realizar a evolução temporal do método VF com multirresolução adaptativa e utilizando técnicas de tempo local. Inicialmente são apresentados os métodos de Runge-Kutta (RK) de ordem 1, 2 e 3, e, em seguida, métodos RK contínuos (RKC). Os métodos RKC são a nova estratégia temporal de tempo local proposta nesta dissertação para ordens de aproximação superiores a 2.

Métodos numéricos para resolução de equações diferenciais ordinárias (EDO) aproximam a solução no domínio discretizado. Seja uma função  $Q(t) \in C[t_i; t_f]$ , a discretização de  $Q$  é o conjunto de pontos denotados por  $t^m$  de forma que  $t^0 = t_i$ ,  $t^m = t^{m-1} + \Delta t^{m-1}$  e  $t^m \leq t_f$ . O valor  $\Delta t^m$  é denominado passo temporal. Um método numérico aproxima o valor de  $Q^m := Q(t^m)$  por um valor  $\bar{Q}^m$ .

Os métodos de Runge-Kutta (RK) são uma família de métodos numéricos usados na aproximação da solução de uma EDO na forma  $\frac{dQ}{dt} = f(t, Q)$ , em que  $f$  é uma função contínua no intervalo estudado. Essa EDO deve atender uma condição inicial  $Q(t_i) = \alpha$  previamente conhecida. Os métodos de RK se diferenciam por não precisarem do cálculo do valor de várias derivadas em diversos pontos; mas sim por precisar do valor do ponto anterior da solução (que já foi calculado) e de cálculos da função  $f$  (BURDEN; FAIRES, 2008).

No contexto das EDOs resultantes da discretização espacial de EDPs evolutivas, a formulação tradicional dos métodos de Runge-Kutta requer um custo de memória muito alto. Nesta formulação deve-se armazenar valores que serão usados para a evolução temporal, considerando que cada célula tem seus próprios valores que devem ser armazenados. Isso faz com que seja necessário mais memória a medida em que a malha se torna mais refinada ou adicionam-se mais dimensões ao problema.

Devido a interesses práticos, em simulações de EDPs evolutivas multidimensionais, faz-se necessária uma formulação alternativa desses métodos, que seja capaz de efetuar a evolução temporal utilizando uma menor área de memória. Em ordens superiores ( $p \geq 3$ ), o uso da formulação compacta reduz, em relação à formulação tradicional, a área de memória para armazenamentos auxiliares à medida em que são usadas malhas mais refinadas.

## 5.1 Métodos de Runge–Kutta tradicionais explícitos

A complexidade de um método RK está associada, além do número de pontos da discretização espacial e temporal, ao seu número de estágios. O número de estágios determina quantas vezes a função  $f$  deve ser calculada por iteração. De forma geral, a evolução temporal utilizando um método RK de  $s$  estágios é definida como:

$$\bar{Q}^{m+1} = \bar{Q}^m + \sum_{i=1}^s b_i k_i, \quad (5.1)$$

em que

$$k_i = \Delta t^m f \left( t^m + c_i \Delta t^m, \bar{Q}^m + \sum_{j=1}^{i-1} a_{i,j} k_j \right) \quad (5.2)$$

O método RK depende dos valores das constantes  $a_{i,j}$ ,  $b_i$  e  $c_i$ . Para calcular estes valores, é necessário definir previamente o número de estágios ( $s$ ) do método RK, que é associado ao número de constantes  $k_i$  que devem ser calculadas; e a ordem de convergência ( $p$ ) desejada para o método. A ordem de convergência que um método RK pode atingir é limitada pelo seu número de estágios (BURDEN; FAIRES, 2008), A Tabela 5.1 define a maior ordem de convergência que se pode obter usando métodos RK com diferentes números de estágios.

Tabela 5.1 - Ordem de convergência máxima para um método RK de  $s$  estágios.

Número de estágios ( $s$ )	1	2	3	4	$5 \leq s \leq 7$	$8 \leq s \leq 9$	$10 \leq s$
Melhor ordem ( $p$ ) possível	1	2	3	4	$s - 1$	$s - 2$	$s - 3$

A ordem de convergência do método RK é melhor explicada na Seção A.2, de forma geral, um método dito de ordem de convergência  $p$  ao ser aplicado em uma solução  $\bar{Q}^m$ , deve produzir uma solução no instante  $t^{m+1}$ , denotada por  $\bar{Q}^{m+1}$ , que satisfaça:

$$\|Q^{m+1} - \bar{Q}^{m+1}\| = O((\Delta t^m)^{p+1}) \quad (5.3)$$

em que  $Q^{m+1}$  é a solução exata do problema no instante  $t^{m+1}$ . A técnica usada para obter os coeficientes  $a_{i,j}$ ,  $b_i$  e  $c_i$  para o método RK desejado é apresentada na Seção A.3.

## Representação dos parâmetros de um método RK explícito

Os parâmetros  $a_{i,j}$ ,  $b_i$  e  $c_i$  de um método RK de  $s$  estágios podem ser organizados como mostrado na Tabela 5.2. Este tipo de tabela é conhecido como tabela de Butcher (BURDEN; FAIRES, 2008).

Tabela 5.2 - Tabela de Butcher para um método RK explícito.

0				
$c_2$	$a_{21}$			
$c_3$	$a_{31}$	$a_{32}$		
$\vdots$	$\vdots$	$\vdots$	$\ddots$	
$c_s$	$a_{s1}$	$a_{s2}$	$\dots$	$a_{ss}$
	$b_1$	$b_2$	$\dots$	$b_s$

Os coeficientes para métodos RK de ordens 1, 2 e 3 são dados, respectivamente, pelas Tabelas 5.3, 5.4 e 5.5. Estes coeficientes serão os utilizados na evolução temporal das equações deste trabalho.

Tabela 5.3 - Tabela de Butcher: Método de Runge-Kutta de ordem 1.

0		
	1	

Tabela 5.4 - Tabela de Butcher: Método de Runge-Kutta de ordem 2.

0		
1	1	
	$\frac{1}{2}$	$\frac{1}{2}$

Tabela 5.5 - Tabela de Butcher: Método de Runge-Kutta de ordem 3.

0			
1	1		
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	
	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{2}{3}$

## 5.2 Formulação compacta

A estratégia para reduzir o custo de memória de um método RK usada neste trabalho consiste em substituir os valores  $\bar{Q}^m + \sum_{j=1}^{i-1} a_{i,j} k_j$  por variáveis auxiliares e reordenar a equação de evolução do método RK, de forma que dependa apenas da última variável auxiliar computada.

A seguir será realizada a formulação compacta dos métodos RK apresentados na Seção anterior. Os métodos descritos aqui são justamente os utilizados no software *carmen* desenvolvido por (ROUSSEL, 2003).

### 5.2.1 Formulação compacta: método RK de ordem 2

**Teorema 3.** *Considerando a Equação 5.1 com os coeficientes do método RK de segunda ordem e dois estágios, utilizando os coeficientes dados na Tabela 5.4. A evolução temporal pode ser executada utilizando os dois passos:*

1.  $\bar{Q}^* = \bar{Q}^m + \Delta t^m f(t^m, \bar{Q}^m)$
2.  $\bar{Q}^{m+1} = \frac{1}{2}\bar{Q}^m + \frac{1}{2}\bar{Q}^* + \frac{1}{2}\Delta t^m f(t^m + \Delta t^m, \bar{Q}^*)$

*Demonstração.* Dada a equação de evolução temporal:

$$\bar{Q}^{m+1} = \bar{Q}^m + \frac{k_1}{2} + \frac{k_2}{2} \quad (5.4)$$

Substituindo  $k_1$  e  $k_2$  conforme a Equação 5.2:

$$\bar{Q}^{m+1} = \bar{Q}^m + \frac{\Delta t^m f(t^m, \bar{Q}^m)}{2} + \frac{\Delta t^m f(t^m + \Delta t^m, \bar{Q}^m + \Delta t^m f(t^m, \bar{Q}^m))}{2}. \quad (5.5)$$

Definindo a variável auxiliar  $\bar{Q}^*$ :

$$\bar{Q}^* := \bar{Q}^m + \Delta t^m f(t^m, \bar{Q}^m) \quad (5.6)$$

e isolando o termo  $\Delta t^m f(t^m, \bar{Q}^m)$ , obtém-se:

$$\Delta t^m f(t^m, \bar{Q}^m) = \bar{Q}^* - \bar{Q}^m. \quad (5.7)$$

Substituindo as Equações 5.6 e 5.7 na Equação 5.5:

$$\bar{Q}^{m+1} = \bar{Q}^m + \frac{\bar{Q}^* - \bar{Q}^m}{2} + \frac{\Delta t^m f(t^m + \Delta t^m, \bar{Q}^*)}{2} \quad (5.8)$$

Reordenando os termos, obtém-se:

$$\bar{Q}^{m+1} = \frac{\bar{Q}^m}{2} + \frac{\bar{Q}^*}{2} + \frac{\Delta t^m f(t^m + \Delta t^m, \bar{Q}^*)}{2} \quad (5.9)$$

Assim, o método RK de segunda ordem e dois estágios pode ser calculado em duas etapas, dadas pelas Equações 5.6 e 5.9.  $\square$

Apesar de não haver um ganho em memória ao utilizar a formulação proposta para um método de dois estágios, sua implementação em um software com diferentes esquemas de evolução temporal torna-se mais simples considerando que o método de três estágios terá uma implementação semelhante. Pois neste caso haverá um ganho significativo.

### 5.2.2 Formulação compacta: método RK de ordem 3

De maneira semelhante a realizada anteriormente, o método RK de ordem 3, que utiliza os coeficientes dados na Tabela 5.5, tem uma formulação compacta como a proposta no Teorema 4.

**Teorema 4.** *Considerando a Equação 5.1 com os coeficientes do método RK de terceira ordem e três estágios, utilizando os coeficientes dados na Tabela 5.5. A evolução temporal pode ser executada utilizando os três passos:*

1.  $\bar{Q}^* = \bar{Q}^m + \Delta t f(t^m, \bar{Q}^m)$
2.  $\bar{Q}^{**} = \frac{3}{4}\bar{Q}^m + \frac{1}{4}\bar{Q}^* + \frac{1}{4}\Delta t f(t^m + \Delta t, \bar{Q}^*)$
3.  $\bar{Q}^{m+1} = \frac{1}{3}\bar{Q}^m + \frac{2}{3}\bar{Q}^{**} + \frac{2}{3}\Delta t f(t^m + \frac{1}{2}\Delta t, \bar{Q}^{**})$

*Demonstração.* Dada a equação de evolução temporal:

$$\bar{Q}^{m+1} = \bar{Q}^m + \frac{k_1}{6} + \frac{k_2}{6} + \frac{2k_3}{3} \quad (5.10)$$

Substituindo  $k_1$  e  $k_2$  conforme a Equação 5.2, e utilizando as constantes obtidas na

Tabela 5.5:

$$\bar{Q}^{m+1} = \bar{Q}^m + \frac{\Delta t^m f(t^m, \bar{Q}^m)}{6} + \frac{\Delta t^m f(t^m + \Delta t^m, \bar{Q}^m + \Delta t^m f(t^m, \bar{Q}^m))}{6} + \frac{2k_3}{3}. \quad (5.11)$$

Definindo a variável auxiliar  $\bar{Q}^*$ :

$$\bar{Q}^* := \bar{Q}^m + \Delta t^m f(t^m, \bar{Q}^m) \quad (5.12)$$

e isolando o termo  $\Delta t^m f(t^m, \bar{Q}^m)$ , obtém-se:

$$\Delta t^m f(t^m, \bar{Q}^m) = \bar{Q}^* - \bar{Q}^m. \quad (5.13)$$

Substituindo as Equações 5.12 e 5.13 na Equação 5.11:

$$\bar{Q}^{m+1} = \bar{Q}^m + \frac{\bar{Q}^* - \bar{Q}^m}{6} + \frac{\Delta t^m f(t^m + \Delta t^m, \bar{Q}^*)}{6} + \frac{2k_3}{3} \quad (5.14)$$

Definindo a segunda variável auxiliar:

$$\bar{Q}^{**} := \bar{Q}^m + \frac{1}{4}k_1 + \frac{1}{4}k_2. \quad (5.15)$$

O termo  $k_3 = \Delta t f\left(t^m + \frac{1}{2}\Delta t^m, \bar{Q}^m + \frac{1}{4}k_1 + \frac{1}{4}k_2\right)$  pode ser reescrito como:

$$k_3 = \Delta t^m f\left(t^m + \frac{1}{2}\Delta t^m, \bar{Q}^{**}\right) \quad (5.16)$$

Substituindo  $k_3$  na Equação 5.14:

$$\bar{Q}^{m+1} = \bar{Q}^m + \frac{\bar{Q}^* - \bar{Q}^m}{6} + \frac{\Delta t^m f(t^m + \Delta t^m, \bar{Q}^*)}{6} + \frac{2\Delta t^m f\left(t^m + \frac{1}{2}\Delta t^m, \bar{Q}^{**}\right)}{3}. \quad (5.17)$$

Como  $k_1 = \Delta t^m f(t^m, \bar{Q}^m)$  e  $k_2 = \Delta t^m f(t^m + \Delta t^m, \bar{Q}^*)$ , ao substituir a Equação 5.13 na Equação 5.15, o termo  $\bar{Q}^{**}$  pode ser reescrito como:

$$\bar{Q}^{**} = \frac{3}{4}\bar{Q}^m + \frac{1}{4}\bar{Q}^* + \frac{1}{4}\Delta t^m f(t^m + \Delta t^m, \bar{Q}^*) \quad (5.18)$$

Isolando o termo  $\Delta t^m f(t^m + \Delta t^m, \bar{Q}^*)$ , obtém-se:

$$\Delta t^m f(t^m + \Delta t^m, \bar{Q}^*) = 4\bar{Q}^{**} - 3\bar{Q}^m - \bar{Q}^* \quad (5.19)$$



Substituindo na Equação 5.17:

$$\bar{Q}^{m+1} = \bar{Q}^m + \frac{\bar{Q}^* - \bar{Q}^m}{6} + \frac{4\bar{Q}^{**} - 3\bar{Q}^m - \bar{Q}^*}{6} + \frac{2\Delta t^m f\left(t^m + \frac{1}{2}\Delta t^m, \bar{Q}^{**}\right)}{3} \quad (5.20)$$

Rearranjando os termos, obtém-se:

$$\bar{Q}^{m+1} = \frac{\bar{Q}^m}{3} + \frac{2\bar{Q}^{**}}{3} + \frac{2\Delta t^m f\left(t^m + \frac{1}{2}\Delta t^m, \bar{Q}^{**}\right)}{3} \quad (5.21)$$

Desta forma, o método RK de terceira ordem e três estágios, cujos coeficientes são dados na Tabela 5.5, pode ser executado utilizando os três passos dados pelas Equações 5.12, 5.6 e 5.21.  $\square$

Para efeito de simplificação, os passos da formulação compacta do método de Runge-Kutta serão expressos pela operação:

$$\bar{Q}^{m+1} = TimeEvolution(\bar{Q}^m, \Delta t^m) \quad (5.22)$$

**Corolário 1.** *O uso da formulação compacta do método RK3 proposto reduz o número de variáveis a serem armazenadas pela metade.*

*Demonstração.* A formulação compacta não necessita que os valores  $k_i$  sejam armazenados; mas sim o valor das variáveis auxiliares  $\bar{Q}^*$  e  $\bar{Q}^{**}$ , que podem ser armazenadas no mesmo endereço de memória sem causar perdas, pois estes valores apenas serão necessários para calcular o próximo passo, podendo ser descartados em seguida. Neste caso, o número de valores armazenados por célula para realizar a evolução temporal é reduzido de 2 ( $k_1, k_2$ ) para 1 ( $\bar{Q}^*$  e  $\bar{Q}^{**}$ , que podem ocupar o mesmo endereço de memória), gerando uma economia de memória de 50%. O valor  $k_3$  não precisa ser armazenado, pois ao ser calculado, a evolução temporal pode ser realizada imediatamente.  $\square$

### 5.3 Métodos de Runge–Kutta Contínuos

O uso dos métodos de Runge-Kutta Contínuos (RKC) foi proposto por Zennaro (1986). Os métodos RKC geram uma aproximação polinomial para cada intervalo

entre dois pontos calculados por um método RK. Os métodos de RKC têm a forma:

$$\bar{Q}(t^m + \theta \Delta t^m) = \bar{Q}^m + \sum_{i=1}^s \beta_i(\theta) k_i \quad (5.23)$$

Em que  $\theta \in (0, 1]$  e os pesos  $\beta_i(\theta)$  são polinômios. Os valores de  $k_i$  são obtidos da mesma maneira como os RKs tradicionais. Os polinômios  $\beta_i(\theta)$  são obtidos para métodos de segunda e terceira ordem conforme os Teoremas 5 e 6, respectivamente. Estes polinômios são calculados em função dos parâmetros  $a_{ij}$ ,  $b_i$  e  $c_i$  usados no método RK escolhido para evoluir a solução do instante  $t^m$  para  $t^{m+1}$ . É importante observar que, como o método RKC depende dos valores  $k_i$ , ele só pode ser executado após a evolução temporal.

Assim como os métodos RK tradicionais, os métodos RKC possuem uma ordem de convergência máxima que depende do número de estágios do método RK usado para criar a aproximação polinomial. Em Zennaro (1991) é calculado o número mínimo de estágios necessários para criar uma aproximação polinomial de diversas ordens de convergência. Estes resultados são apresentados na Tabela 5.6.

Tabela 5.6 - Ordem de convergência máxima para um método RKC de  $s$  estágios.

Número de estágios ( $s$ )	1	2	3	4	5	6	7	8
Melhor ordem ( $p$ ) possível	1	2	2	3	3	4	4	5

**Teorema 5.** *Os seguintes polinômios podem ser utilizados como pesos  $\beta_i(\theta)$ ,  $i = 1, 2$ , para o método de RKC de segunda ordem e dois estágios:*

$$\beta_1(\theta) = (b_1 - 1) \theta^2 + \theta \quad (5.24a)$$

$$\beta_2(\theta) = b_2 \theta^2 . \quad (5.24b)$$

*Demonstração.* A construção dos polinômios  $\beta_i(\theta)$  é realizada de maneira análoga a obtenção dos coeficientes  $b_i$  de um método RK tradicional. Considere a expansão em série de Taylor do termo  $Q(t^m + \theta \Delta t^m)$  na Equação 5.25a e a equação da forma geral do método RKC de segunda ordem na Equação 5.25b.

$$Q(t^m + \theta \Delta t^m) = Q^m + \theta \Delta t^m \frac{dQ}{dt}(t^m) + \frac{(\theta \Delta t^m)^2}{2} \frac{d^2Q}{dt^2}(t^m) + O((\Delta t^m)^3) . \quad (5.25a)$$

$$\bar{Q}(t^m + \theta \Delta t^m) = \bar{Q}^m + \beta_1(\theta) k_1 + \beta_2(\theta) k_2 . \quad (5.25b)$$

Para que a aproximação  $\bar{Q}(t^m + \theta \Delta t^m)$  tenha segunda ordem de convergência, ela deve satisfazer a condição:

$$\|Q(t^m + \theta \Delta t^m) - \bar{Q}(t^m + \theta \Delta t^m)\| = O((\Delta t^m)^3). \quad (5.26)$$

Realizando um processo análogo ao realizado no Apêndice A, obtém-se o sistema de equações sobre os polinômios  $\beta_i(\theta)$  que tornam a Equação 5.26 válida:

$$\beta_1(\theta) + \beta_2(\theta) = \theta, \quad (5.27a)$$

$$\beta_2(\theta) c_2 = \frac{\theta^2}{2}, \quad (5.27b)$$

$$\beta_2(\theta) a_{2,1} = \frac{\theta^2}{2}. \quad (5.27c)$$

Do sistema usado para calcular os coeficientes de um método RK de segunda ordem no Apêndice A, obtém-se que  $c_2 = \frac{1}{2b_2}$  e  $b_1 + b_2 = 1$ . Substituindo  $c_2$  na Equação 5.27b, obtem-se a Equação 5.24b.

Utilizando o polinômio  $\beta_2(\theta)$  na Equação 5.27a:

$$\beta_1(\theta) = \theta - b_2 \theta^2. \quad (5.28)$$

Como  $b_1 + b_2 = 1$ , obtém-se a Equação 5.24a. □

**Teorema 6.** *Seja o método RKC de três estágios utilizando os coeficientes do método RK de terceira ordem apresentados na Seção 5.1, Os polinômios  $\beta_i(\theta)$  utilizados como pesos são dados como:*

$$\beta_i(\theta) = w_i \theta^2 + (b_i - w_i) \theta, \quad i = 1, 2, 3. \quad (5.29)$$

Os valores  $w_i$  são dados em função de um parâmetro  $\alpha \in \mathbb{R}$ :

$$w_1 = \frac{-\alpha(c_3 - c_2) - c_2}{2c_2c_3}, \quad (5.30a)$$

$$w_2 = \frac{\alpha}{2c_2}, \quad (5.30b)$$

$$w_3 = \frac{1 - \alpha}{2c_3}. \quad (5.30c)$$

*Esta aproximação, em que se utiliza o método RKC, gera uma aproximação de segunda ordem de convergência.*

*Demonstração.* A construção dos polinômios  $\beta_i(\theta)$  é realizada de maneira semelhante ao realizado no Teorema 5. Considere a expansão em série de Taylor do termo  $Q(t^m + \theta\Delta t^m)$  na Equação 5.31a e a equação da forma geral do método RKC de três estágios na Equação 5.31b.

$$Q(t^m + \theta\Delta t^m) = Q^m + \theta\Delta t^m \frac{dQ}{dt}(t^m) + \frac{(\theta\Delta t^m)^2}{2} \frac{d^2Q}{dt^2}(t^m) + O((\Delta t^m)^3) \quad (5.31a)$$

$$\bar{Q}(t^m + \theta\Delta t^m) = \bar{Q}^m + \beta_1(\theta) k_1 + \beta_2(\theta) k_2 + \beta_3(\theta) k_3 \quad (5.31b)$$

A expansão em séries de Taylor é realizada até o termo de segunda ordem, pois o método RK3 usado neste trabalho possui três estágios e, conforme a Tabela 5.6, a maior ordem de convergência possível para a extensão contínua do método RK de três estágios é dois. Para que a aproximação  $\bar{Q}(t^m + \theta\Delta t^m)$  tenha segunda ordem de convergência, ela deve satisfazer a condição:

$$\|Q(t^m + \theta\Delta t^m) - \bar{Q}(t^m + \theta\Delta t^m)\| = O((\Delta t^m)^3) \quad (5.32)$$

Realizando um processo análogo ao realizado no Apêndice A, obtém-se o sistema de equações sobre os polinômios  $\beta_i(\theta)$  que tornam a Equação 5.32 válida:

$$\beta_1(\theta) + \beta_2(\theta) + \beta_3(\theta) = \theta \quad (5.33a)$$

$$\beta_2(\theta) c_2 + \beta_3(\theta) c_3 = \frac{\theta^2}{2} \quad (5.33b)$$

$$\beta_2(\theta) a_{2,1} + \beta_3(\theta) a_{3,1} + \beta_3(\theta) a_{3,2} = \frac{\theta^2}{2} \quad (5.33c)$$

em que os coeficientes  $a_{i,j}$  e  $c_i$  são os mesmos do método RK de três estágios. As Equações 5.33 possuem infinitas soluções, que podem ser escritas, de forma geral, conforme a Equação 5.29.  $\square$

Para o caso do RKC de terceira ordem, foi construída uma aproximação de segunda ordem pois o sistema obtido ao aplicar a técnica usada para obter o sistema das Equações 5.33 não possui soluções que satisfaçam a terceira ordem de convergência.

**Corolário 2.** *O método RKC de segunda ordem com os coeficientes do método RK proposto tem a forma:*

$$\beta_1(\theta) = -\frac{1}{2}\theta^2 + \theta, \quad (5.34a)$$

$$\beta_2(\theta) = \frac{1}{2}\theta^2. \quad (5.34b)$$

**Corolário 3.** *O método RKC de terceira ordem com os coeficientes do método RK proposto e  $\alpha = 0$ , tem a forma:*

$$\beta_1(\theta) = -\theta^2 + \frac{7}{6}\theta \quad (5.35a)$$

$$\beta_2(\theta) = \frac{1}{6}\theta \quad (5.35b)$$

$$\beta_3(\theta) = \theta^2 - \frac{1}{3}\theta. \quad (5.35c)$$

Assim, para obter a aproximação da solução em um instante dentro do intervalo  $(t^m, t^m + \Delta t^m)$ , utilizam-se os valores  $k_i$  calculados para realizar a evolução de  $t^m$  para  $t^m + \Delta t^m$  utilizando um método de Runge-Kutta e os polinômios  $\beta_i(\theta)$  correspondentes ao método RK.

### 5.3.1 RKC: Formulação compacta

A aplicação dos métodos RKC neste trabalho é a de calcular o valor da evolução temporal nos instantes  $t^m + \frac{\Delta t^m}{2}$ . Para realizar esta aproximação, são necessários os valores dos polinômios  $\beta_i(\theta)$  com  $\theta = \frac{1}{2}$  e os valores  $k_i$ . Os valores  $\beta_i$  para um método RK2 podem ser obtidos substituindo o valor  $\theta = \frac{1}{2}$  nas Equações 5.34:

$$\beta_1\left(\frac{1}{2}\right) = \frac{3}{8}, \quad \beta_2\left(\frac{1}{2}\right) = \frac{1}{8}. \quad (5.36)$$

Então, a formulação do método RKC de segunda ordem para o instante  $t^m + \frac{\Delta t^m}{2}$  tem a forma:

$$\bar{Q}\left(t^m + \frac{\Delta t^m}{2}\right) = \bar{Q}^m + \frac{3}{8}k_1 + \frac{1}{8}k_2. \quad (5.37)$$

Para o método RK3 é realizado um processo análogo, utilizando a Equação 5.35:

$$\beta_1\left(\frac{1}{2}\right) = \frac{1}{3}, \beta_2\left(\frac{1}{2}\right) = \frac{1}{12}, \beta_3\left(\frac{1}{2}\right) = \frac{1}{12}. \quad (5.38)$$

Portanto, a formulação do método RKC de terceira ordem para o instante  $t^m + \frac{\Delta t^m}{2}$  tem a forma:

$$\bar{Q}\left(t^m + \frac{\Delta t^m}{2}\right) = \bar{Q}^m + \frac{1}{3}k_1 + \frac{1}{12}k_2 + \frac{1}{12}k_3. \quad (5.39)$$

Entretanto, devido a formulação compacta dos métodos RK usados neste trabalho, os valores  $k_i$  não são armazenados, então é necessário realizar o método em etapas

que são executadas simultaneamente com a formulação compacta.

- Método RK2:

1.  $\bar{Q}_{RKC}^* = \bar{Q}^m + \frac{3}{8}\Delta t^m f(t^m, \bar{Q}^m)$
2.  $\bar{Q}^{m+1} = \bar{Q}_{RKC}^* + \frac{1}{8}\Delta t^m f(t^m + \Delta t, \bar{Q}^*)$

- Método RK3:

1.  $\bar{Q}_{RKC}^* = \bar{Q}^m + \frac{1}{3}\Delta t^m f(t^m, \bar{Q})$
2.  $\bar{Q}_{RKC}^{**} = \bar{Q}_{RKC}^* + \frac{1}{12}\Delta t^m f(t^m + \Delta t, \bar{Q}^*)$
3.  $\bar{Q}(t^m + \frac{\Delta t}{2}) = \bar{Q}_{RKC}^{**} + \frac{1}{12}\Delta t^m f(t^m + \frac{1}{2}\Delta t, \bar{Q}^{**})$

O Algoritmo 4 descreve a evolução temporal utilizando os métodos de Runge–Kutta compactos.

---

**Algoritmo 4** Evolução temporal com métodos compactos.

---

**Require:** Seja  $s$  o número de estágios do método RK compacto utilizado.

**for**  $q = 1 : q \leq s : q = q + 1$  **do**

**for** Para todas as células da malha adaptada **do**

        Cálculo dos fluxos numéricos;

        Realizar a etapa  $q$  do método RK compacto escolhido;

**end for**

**end for**

---

As variáveis  $\bar{Q}_{RKC}^*$ ,  $\bar{Q}_{RKC}^{**}$  e  $\bar{Q}(t^m + \frac{\Delta t}{2})$  podem ser armazenadas no mesmo endereço de memória. Esta evolução para o instante  $t^m + \frac{\Delta t}{2}$  será importante para a execução da técnica de tempo local, descrita no próximo capítulo.

## 6 TÉCNICA DE TEMPO LOCAL

As técnicas adaptativas espaciais obtêm um grande ganho em desempenho em relação as abordagens tradicionais. Visando melhorar ainda mais o desempenho destes métodos, são propostos os métodos adaptativos temporais, que adequam o passo temporal de forma que minimize o número de evoluções temporais para o problema. Neste trabalho é proposto o uso do método de tempo local para este propósito. Este tipo de técnica é relevante devido a necessidade de simulações cada vez mais rápidas e mais complexas, como em aplicações na área espacial e em engenharia.

A técnica de tempo local (LT) só tem sentido em métodos adaptativos. Ela consiste em realizar a evolução temporal de cada célula do domínio discretizado de forma independente, utilizando, em cada célula, um passo  $\Delta t_{L-\ell}^m := 2^{L-\ell} \Delta t^m$ , em que  $L$  é o nível mais refinado da malha e  $\ell \in \mathbb{N}$  é o nível de refinamento da célula. Este passo temporal é escolhido de modo que possa manter a estabilidade numérica do problema.

Supondo que o nível mais refinado da malha atende a condição de CFL, a Equação 2.22 pode ser reescrita como:

$$\sigma = \lambda \frac{\Delta t_L^m}{\Delta x_L} \quad (6.1)$$

em que:

$$\lambda = \sum_{n=1}^N \lambda_n \quad (6.2)$$

Seja  $l$  um nível mais grosseiro da malha, substituindo os valores  $\Delta t_L^m$  e  $\Delta x_L$  pelos valores correspondentes a um nível  $l$  qualquer, obtém-se:

$$\lambda \frac{\Delta t_{L-l}^m}{\Delta x_{L-l}} = \lambda \frac{2^{L-l} \Delta t_L^m}{2^{L-l} \Delta x_L} = \lambda \frac{\Delta t_L^m}{\Delta x_L} \quad (6.3)$$

Portanto, se o esquema numérico é estável no nível mais refinado, então ele é estável em todos os demais níveis.

Usando a técnica de tempo local, a evolução temporal da Equação 5.22 pode ser reescrita como:

$$\bar{Q}^{m+2^{L-l}} = TimeEvolution(\bar{Q}^m, \Delta t_{L-l}^m) \quad (6.4)$$

Desta forma, uma célula cuja resolução está um nível mais grosseira do que o nível mais refinado, evolui do instante  $t^m$  para o instante  $t^{m+2}$ , enquanto uma célula

no nível mais refinado evolui do instante  $t^m$  para o instante  $t^{m+1}$ . Para realizar a evolução temporal das células no nível mais refinado do instante  $t^{m+1}$  para o instante  $t^{m+2}$ , é preciso realizar uma segunda evolução temporal, para isso são necessários os valores de todas as células adjacentes no instante  $t^{m+1}$ , porém caso alguma célula adjacente seja um nível mais grosseira, o valor da mesma no instante  $t^{m+1}$  não está disponível, pois esta evolui do instante  $t^m$  para  $t^{m+2}$ . Este problema ocorre para todos os níveis da malha, exceto para o nível menos refinado da malha, pois nenhuma célula realizará um passo temporal maior do que o da mesma. A Figura 6.1 ilustra este problema. Para realizar a segunda evolução temporal das Células 1 e 2, que estão em um nível mais refinado, é necessário uma aproximação do valor da Célula 3 no instante  $t^{m+1} = t^m + \frac{1}{2}\Delta t_1^m$ . Com esta aproximação, é possível calcular todos os fluxos neste instante e enfim, realizar a segunda evolução temporal das Células 1 e 2.

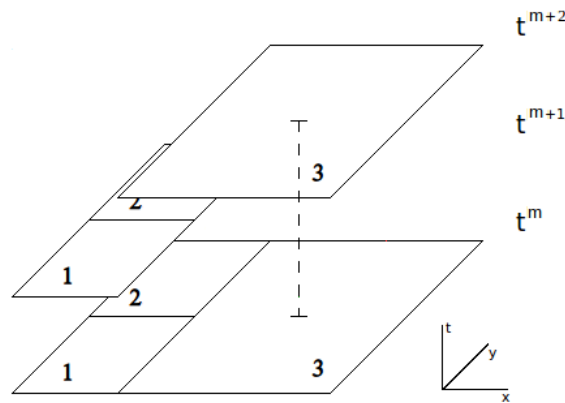


Figura 6.1 - Dessincronização da evolução temporal ao aplicar a técnica do LT.

A execução do LT, quando duas células adjacentes estão em níveis de refinamento diferentes, pode ser ilustrada conforme a Figura 6.2. Inicialmente, a evolução temporal deve ser feita nas células mais refinadas, no caso, as Célula 1 e 2 da Figura 6.2a, para isso deve ser calculado os fluxos numéricos entre estas células e a Célula 3 e o fluxo numérico entre as Células 1 e 2. Calculados estes fluxos, é feita a evolução temporal das Células 1 e 2 com um passo de tempo  $\Delta t_{L-\ell}^m$  (Figura 6.2b) conforme a técnica representada pela Equação 6.4. Depois deve ser feito o mesmo processo com as células no nível menos refinado, no exemplo, a Célula 3. Os fluxos entre esta célula e as Células 1 e 2 já foram calculados, sendo necessário apenas trocar o sinal destes valores (Figura 6.2c). Feito isso, realiza-se a evolução temporal da Célula 3 com um



passo de tempo  $\Delta t_{L-\ell+1}^m = 2\Delta t_{L-\ell}^m$  (Figura 6.2d). Com isso, as Células 1 e 2 precisam realizar outra evolução temporal para alcançar o mesmo instante que a Célula 3, para isso, é necessário uma estimativa do valor da Célula 3 no instante  $t^{m+1}$  para que sejam calculados os fluxos numéricos para a segunda evolução (Figura 6.2e). O próximo passo é calcular os fluxos entre as Células 1 e 2, além dos fluxos entre cada uma destas e a Célula 3 no instante intermediário (Figura 6.2f). O último passo é a segunda evolução das células mais refinadas, conforme a Figura 6.2g. Então, obtém-se o valor da malha em todas as células no instante  $t^{m+2}$  (Figura 6.2h).

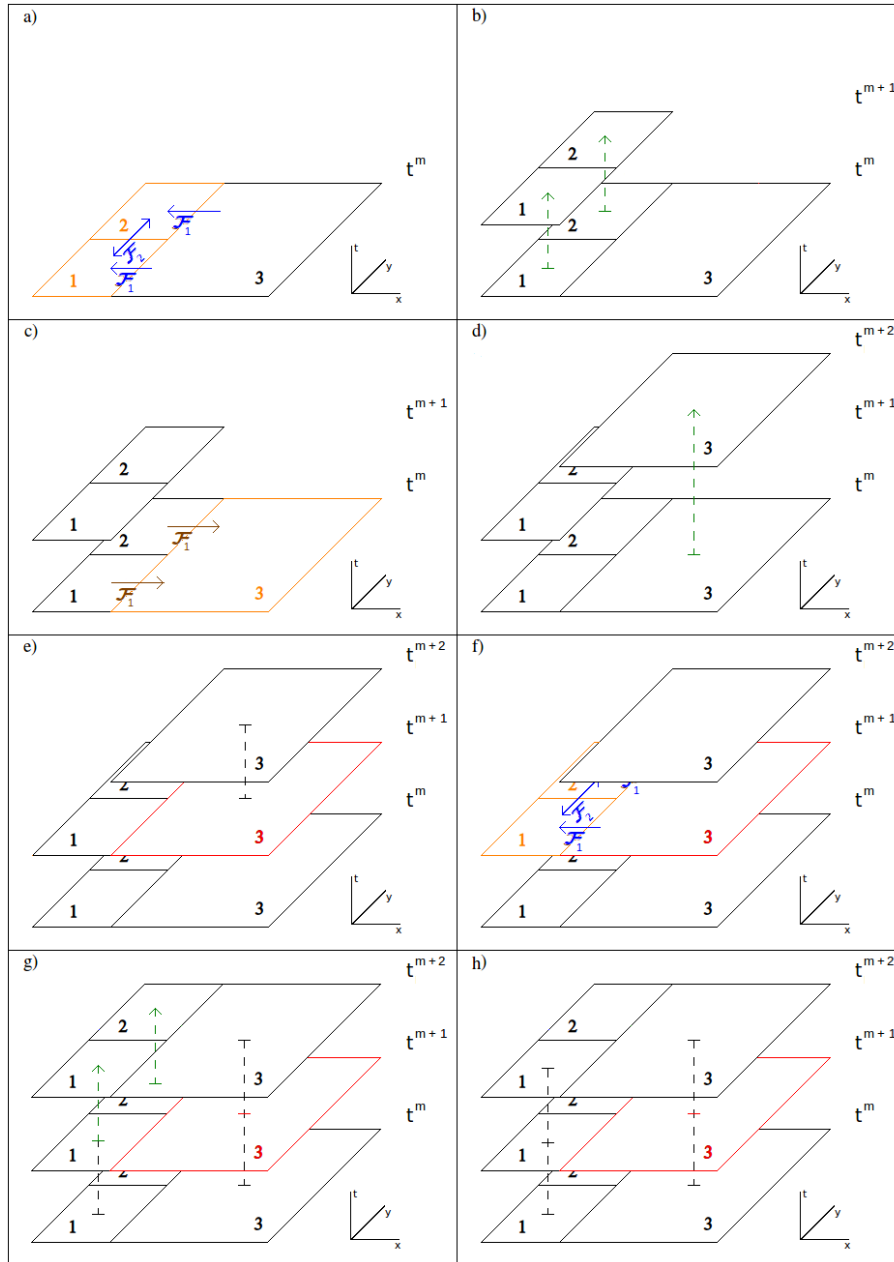


Figura 6.2 - Passos para a execução do LT. a) Início da evolução das Células 1 e 2 (em laranja) é necessário calcular os fluxos (em azul) entre as células a serem evoluída e suas células adjacentes; b) Evolução temporal (em verde) das Células 1 e 2 para o instante  $t^{m+1}$ ; c) Início da evolução da Célula 3, menos refinada. Os fluxos (em marrom) entre a Célula 3 e as Células 1 e 2 já foram calculados em a); d) Evolução temporal da Célula 3 para o instante  $t^{m+2}$ ; e) Cálculo de uma aproximação da Célula 3 para o instante  $t^{m+1}$ ; f) Início da segunda evolução das Células 1 e 2, é necessário o cálculo do fluxo entre as Células 1 e 2, também é necessário o cálculo dos fluxos entre a aproximação da Célula 3 e as Células 1 e 2; g) Evolução temporal das Células 1 e 2 para o instante  $t^{m+2}$ ; h) Fim da evolução para o instante  $t^{m+2}$ .

## 6.1 Abordagem tradicional

Tradicionalmente, a aproximação nos instantes intermediários da evolução temporal, necessária para a execução da técnica de tempo local é obtida utilizando os cálculos intermediários do método RK. Em Domingues et al. (2008), esta aproximação é feita para o caso de uma evolução temporal com um método RK2 conforme a expressão:

$$\bar{Q} \left( t^m + \frac{1}{2} \Delta t_{L-l}^m \right) \approx \bar{Q}^m + \frac{1}{2} k_1. \quad (6.5)$$

A Figura 6.3 ilustra a sincronização obtida ao utilizar a Equação 6.5 para calcular o valor da célula menos refinada no instante  $t^m + \frac{1}{2} \Delta t^m$ . Esta aproximação é baseada em uma série de Taylor truncada no termo de primeira ordem, o que produz uma aproximação de primeira ordem de convergência.

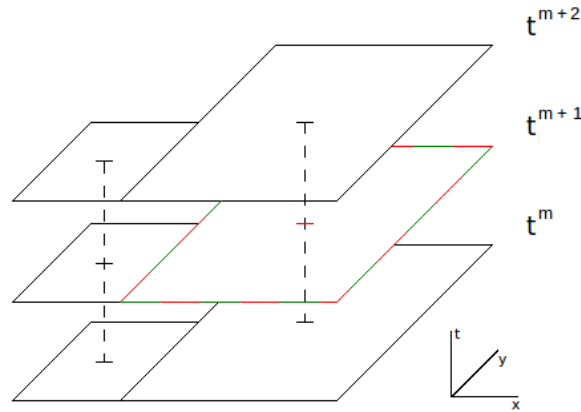


Figura 6.3 - Evolução temporal de duas células adjacentes utilizando o RK2.

## 6.2 Abordagem proposta

A abordagem tradicional para o cálculo dos valores intermediários não satisfaz ordens superiores, para isso, este trabalho propõe como alternativa, o uso dos métodos de Runge-Kutta contínuos, discutidos na Seção 5.3, para gerar a aproximação no instante  $t^m + \frac{1}{2} \Delta t_{L-l}^m$  de todas as células em que este valor for necessário. Este trabalho é pioneiro ao apresentar o uso de métodos RK de ordens superiores no contexto das técnicas de tempo local.

A vantagem do método proposto neste trabalho, para a obtenção dos valores da solução em instantes intermediários em relação ao até então utilizado para métodos

de ordem 2, é que não ocorre uma perda na ordem de convergência no valor aproximado. Enquanto para métodos de terceira ordem, o uso do RKC diminui a ordem da aproximação em 1, conforme discutido na Seção 5.3, apesar disso o uso do RKC de três estágios ainda gera uma aproximação superior à feita na Equação 6.5.

**Definição 8.** *Ciclo da evolução temporal é a sequência de níveis em que o algoritmo de evolução temporal deve percorrer para realizar a evolução de todas as células de um instante  $t^m$  até  $t^m + t^{m+2^{L-\ell_{\min}}}$ , em que  $\ell_{\min}$  é o nível menos refinado da malha.*

De forma geral, o Algoritmo 5 descreve o processo proposto para realizar um ciclo de evolução temporal em malhas com diversos níveis. Para este algoritmo, é necessário o uso de aproximações de primeira ordem para instantes futuros das células. Tais aproximações são realizadas como:

$$\bar{Q}^{m+2^{L-\ell-1}} = \bar{Q}^m + 2^{L-\ell-1} \frac{k_1}{2} \quad (6.6)$$

---

**Algoritmo 5** Execução do LT - Ciclo da evolução temporal.

---

**Require:** Seja  $\ell_{\min} \in \mathbb{N}$  o nível mais grosseiro da malha adaptada.

Execute a Subrotina 1, descrita no Algoritmo 6, com um parâmetro de entrada  $j = \ell_{\min}$ ;

---

Conforme apresentado na Seção 4.2, tanto a malha adaptativa quanto o passo temporal  $\Delta t^m$  podem ser atualizados a cada iteração do método numérico, mas para o caso do LT, estas atualizações devem ser feitas apenas no início de um ciclo da evolução temporal, pois corre-se o risco das células não terminarem o ciclo da evolução temporal no mesmo instante da discretização temporal.

---

**Algoritmo 6** Execução do LT - Subrotina 1.

---

**Require:** Seja  $j \in \mathbb{N}$  um parâmetro de entrada do algoritmo.

**for**  $\ell = L : \ell > j : \ell = \ell - 1$  **do**

    Etapa 1 do método RK compacto para as células do nível  $\ell$ ;

**end for**

**if**  $j \neq \ell_{\min}$  **then**

    Aproximação das células do nível  $j$  no instante intermediário, conforme a Equação 6.6;

    Etapa 2 do método RK compacto para as células do nível  $j - 1$ ;

    Etapa 3 do método RK compacto para as células do nível  $j - 1$ ;

    Obter aproximação do RKC para as células no nível  $j - 1$  com  $\theta = 0,5$ ;

**end if**

Etapa 1 do método RK compacto para as células do nível  $j$ ;

**if**  $j \neq L$  **then**

**for**  $p = L : p \geq j + 1 : p = p - 1$  **do**

**if**  $p = L$  **then**

            Aproximação das células do nível  $p - 1$  no instante intermediário, conforme a Equação 6.6;

            Etapa 2 do método RK compacto para as células do nível  $p$ ;

            Etapa 3 do método RK compacto para as células do nível  $p$ ;

**end if**

        Subrotina 1, descrita no Algoritmo 6, com um parâmetro de entrada  $p$ ;

**end for**

**else**

    Etapa 2 do método RK compacto para as células do nível  $L$ ;

    Etapa 3 do método RK compacto para as células do nível  $L$ ;

**end if**

Nota: No caso de uma execução com método RK2, apenas ignore os passos da Etapa 3.

---

### 6.3 Varredura da malha computacional

O procedimento para a realização de técnicas de tempo local requer que se faça uma varredura por níveis da malha computacional, o que no caso dos tipos malha utilizadas neste trabalho, árvore binária para 1D ou árvore quaternária para 2D, acarreta em um procedimento de busca em largura na árvore. Este procedimento possui uma complexidade computacional maior do que um procedimento de busca em profundidade, que é utilizado nos métodos adaptativos espaciais sem técnicas de tempo local.

#### 6.3.1 Busca em profundidade

A busca em profundidade é um algoritmo usado para percorrer uma estrutura de árvore a partir de sua raiz. Este tipo de busca é considerado o mais eficiente para percorrer todos os nós de uma árvore. Porém, para a aplicação no contexto de uma árvore representando uma estrutura multi-nível, a busca em profundidade não faz buscas por níveis, ou seja, esta busca percorre células de diferentes níveis. O Algoritmo 7 descreve o processo de busca em profundidade, enquanto a Figura 6.4 ilustra a ordem de células percorridas pelo algoritmo. Esta busca é ideal para a execução do método MR/VF tradicional pois este método não exige nenhuma ordenação nas células a serem evoluídas.

---

**Algoritmo 7** Busca em profundidade sobre a árvore de armazenamento das células.

---

Realize os cálculos referentes à célula (nó) atual, se necessário;  
**while** O nó possui algum filho não percorrido **do**  
    Execute recursivamente o Algoritmo 7 utilizando este filho como raiz;  
**end while**

---

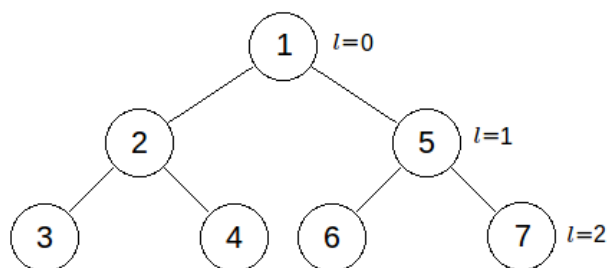


Figura 6.4 - Busca em profundidade: percorrimento das células de uma árvore.

### 6.3.2 Busca em largura

A busca em largura é um algoritmo usado para percorrer uma estrutura de árvore a partir de sua raiz. Este tipo de busca é considerado pouco eficiente para percorrer todos os nós de uma árvore. Para a aplicação no contexto de uma árvore representando uma estrutura multi-nível, ao contrário da busca em profundidade, a busca em largura faz buscas por níveis, ou seja, esta busca percorre células do mesmo nível. O Algoritmo 8 descreve o processo de busca em largura, enquanto a Figura 6.5 ilustra a ordem de células percorridas pelo algoritmo. Esta busca é ideal para a execução do método MR/LT/VF proposto pois este método exige uma ordenação nas células a serem evoluídas.

---

**Algoritmo 8** Busca em largura sobre a árvore de armazenamento das células.

---

**Require:** Seja  $\ell$  um parâmetro de entrada que representa o nível a ser percorrido e  $j$  o nível da célula percorrida.

**if**  $\ell > j$  **then**

**while** O nó possui algum filho não percorrido **do**

        Execute recursivamente o Algoritmo 8 utilizando este filho como raiz;

**end while**

**end if**

**if**  $\ell = j$  **then**

    Realize os cálculos referentes à célula (nó) atual;

**end if**

---

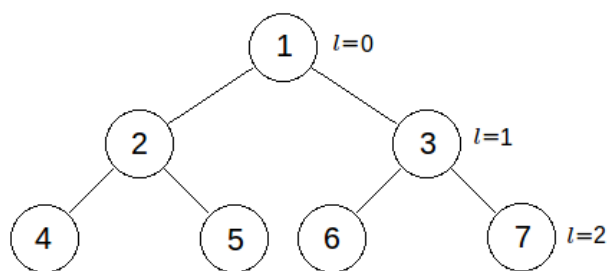


Figura 6.5 - Busca em largura: percorrimento das células de uma árvore.

Do ponto de vista da complexidade computacional, a busca em largura apresenta uma grande desvantagem em relação a busca em profundidade. Enquanto uma varredura com uma busca em profundidade percorre todas as células em um único procedimento, a busca em largura é executada uma vez para cada nível da malha

que deve ser evoluído.

O uso de uma malha implementada como uma árvore causa dois problemas que aumentam a complexidade do algoritmo do método MR/LT:

- a) O método LT requer uma busca em largura na árvore para cada célula a ser evoluída;
- b) A busca pelas células adjacentes pode ser complicada, exigindo o uso de vários condicionais.

Para estes problemas são propostas duas soluções:

- a) Indexar as células de cada nível em uma Tabela Hash, esta estrutura permite organizar ponteiros para as células da malha na forma de listas encadeadas, em que cada lista contém ponteiros para as células do mesmo nível de refinamento. Entretanto, tal estrutura consome uma quantidade significativa de memória computacional, pois para cada célula da malha computacional, é necessário um ponteiro que aponte para esta.
- b) Utilizar uma árvore em que cada filho possui dois ponteiros que apontam para os dois vizinhos que não são seus irmãos, simplificando a busca por células adjacentes.

Ambas as soluções propostas resolvem o problema da alta complexidade com o uso de mais memória computacional, o que nem sempre é desejável no contexto das simulações numéricas. Tais propostas serão analisadas em trabalhos posteriores.



---

**Algoritmo 9** Execução do método proposto.

---

**Require:** Condição inicial  $Q^0$  e condições de contorno.

**Require:** Tempo final:  $t_f$ .

**Require:** Condição de CFL  $\sigma$ .

**Require:** Valor de *threshold*  $\epsilon$ .

**for**  $t^m = 0: t^m \leq t_f$  **do**

    Conversão para variáveis primitivas;

    Obtenção dos autovalores;

    Cálculo do passo temporal, conforme a Equação 2.23;

    Conversão para variáveis conservativas;

    Construção da malha adaptativa, conforme o Algoritmo 3;

**if** O método usa a técnica de tempo local **then**

        Evolução temporal, conforme o Algoritmo 5, utilizando a busca em árvore do Algoritmo 8;

$t^m = t^m + 2^{L-\ell_{\min}} \Delta t^m$ ;

**else**

        Evolução temporal, conforme o Algoritmo 4, utilizando a busca em árvore do Algoritmo 7;

$t^m = t^m + \Delta t^m$ ;

**end if**

    Reconstrução da malha, conforme o Algoritmo 2

**end for**

Imprimir solução;

---



## 7 APLICAÇÕES

Neste capítulo é apresentado uma introdução aos modelos físicos simulados com a técnica proposta neste trabalho.

### 7.1 Equação de Burgers

A Equação de Burgers é uma EDP não linear que representa um modelo simples de turbulência, esta equação é dada por:

$$\frac{\partial U(x, t)}{\partial t} + \frac{\partial \left(\frac{1}{2}U^2(x, t)\right)}{\partial x} = \mu \frac{\partial^2 U(x, t)}{\partial x^2} \quad (7.1)$$

em que  $t > 0$  e  $\mu = 0,003183099$  é o coeficiente de viscosidade. Comparando a Equação de Burgers com a Equação 2.1 obtém-se o valor do fluxo físico  $F_1 = \frac{1}{2}U^2$  na direção  $x$ .

Para o caso unidimensional, este modelo é resolvido numericamente utilizando as seguintes condições iniciais e de contorno:

$$U(0, t) = U(1, t) = 0 \quad (7.2a)$$

$$U(x, 0) = \sin(2\pi x) \quad (7.2b)$$

A Figura 7.1 ilustra a condição inicial usada para este caso. A solução numérica obtida é comparada com a solução exata do problema, descrita em (HOLMSTRÖM, 1997) como:

$$U(x, t) = \frac{\int_{-\infty}^{\infty} \sin(2\pi(x-y)) \exp\left\{\frac{1}{4\mu}\left[\frac{\cos(2\pi(x-y))}{\pi} - \frac{y^2}{t}\right]\right\} dy}{\int_{-\infty}^{\infty} \exp\left\{\frac{1}{4\mu}\left[\frac{\cos(2\pi(x-y))}{\pi} - \frac{y^2}{t}\right]\right\} dy} \quad (7.3)$$

Para este caso, a simulação é realizada até o instante 0,4, utilizando a constante CFL:  $\sigma = 0,5$ .

Para o caso bidimensional estudado, a condição inicial é:

$$U(x, y, 0) = \sin(2\pi x) \sin(2\pi y), \quad 0 \leq x \leq 1, 0 \leq y \leq 1 \quad (7.4)$$

em que  $\mu = 0$ . A Figura 7.2 ilustra a condição inicial usada para este caso. Para este caso, a simulação é realizada até o instante 0,9, utilizando a constante CFL:  $\sigma = 0,5$ .

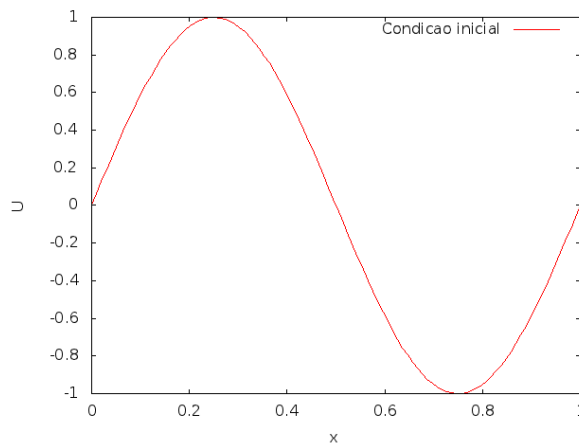


Figura 7.1 - Condição inicial do problema proposto da Equação de Burgers 1D.

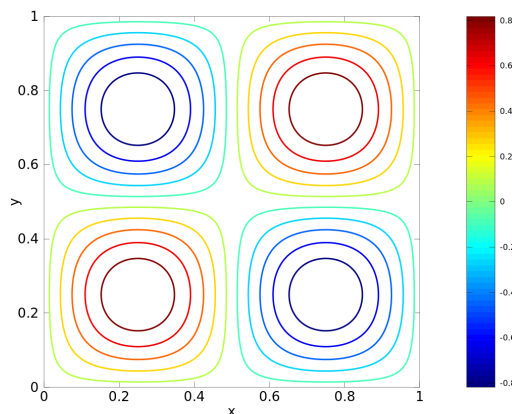


Figura 7.2 - Condição inicial do problema proposto da Equação de Burgers 2D.

## 7.2 Problema de chamas

O segundo problema abordado neste trabalho consiste na simulação da formação de uma bola de fogo a partir de uma faísca em um ambiente com gás inflamável. Considerando que ocorram tanto para gás queimado, quanto para gás não queimado temperaturas iguais para partículas na mesma situação, além de uma igual difusão de massa (ROUSSEL, 2003; DOMINGUES et al., 2008).

### 7.2.1 Caso unidimensional

Para o caso unidimensional, este problema pode ser modelado utilizando a equação:

$$\frac{\partial T}{\partial t} + v_f \frac{\partial T}{\partial x} = \frac{\partial^2 T}{\partial x^2} + \omega(T) \quad (7.5)$$

em que a função  $T(x, t)$  é a temperatura adimensional normalizada entre 0 (gás ainda não queimado) e 1 (gás queimado) e  $v_f = \int \omega dx$ . O termo  $\omega(T)$ , denominado taxa de reação química, é dado pela equação:

$$\omega(T) = \frac{Ze^2}{2}(1 - T)e^{\frac{Ze(1-T)}{\tau(1-T)^{-1}}} \quad (7.6)$$

sendo  $Ze$  é um número adimensional de energia de ativação, conhecido como número de Zeldovich, e  $\tau$  é uma taxa de conversão de gás não queimado para gás queimado.

Nas simulações do caso unidimensional, são utilizadas as seguintes condições iniciais:

$$T(x, 0) = 1, \quad \|x\| \leq 1 \quad (7.7a)$$

$$T(x, 0) = e^{1-x}, \quad \|x\| > 1 \quad (7.7b)$$

as condições de contorno são:

$$\frac{\partial T}{\partial x}(-15, t) = 0 \quad (7.8a)$$

$$T(15, t) = 0 \quad (7.8b)$$

São utilizados os parâmetros  $Ze = 10$ ,  $\tau = 0,8$  e  $\sigma = 0,5$ . Neste caso as simulações são realizadas até o instante  $t = 0,5$ . Para o caso unidimensional, a concentração de gás não queimado  $Y$  é definida como  $Y = 1 - T$ .

### 7.2.2 Caso tridimensional

Para casos multidimensionais, a equação do problema de chamas pode ser escrita como:

$$\frac{\partial T}{\partial t} = \nabla^2 T + \omega - s \quad (7.9a)$$

$$\frac{\partial Y}{\partial t} = \frac{1}{Le} \nabla^2 Y - \omega \quad (7.9b)$$

em que a taxa de reação  $\omega$  é dada por:

$$\omega(T, Y) = \frac{Ze^2}{2Le} Y e^{\frac{Ze(T-1)}{1+\tau(T-1)}} \quad (7.10)$$

De acordo com a lei de Stefan-Boltzmann, a perda de calor  $s$  causada pela radiação é dada por:

$$s(T) = \kappa [(T + \tau^{-1} - 1)^4 - (\tau^{-1} - 1)^4] \quad (7.11)$$

em que  $\kappa$  é um coeficiente de radiação adimensional. Para estas simulações, é usado o valor  $\kappa = 0, 1$ .

Neste problema são usadas as seguintes condições iniciais descritas utilizando coordenadas esféricas:

$$T(r, 0) = 1, \quad \text{se } r \leq r_0 \quad (7.12a)$$

$$T(r, 0) = e^{1-\frac{r}{r_0}}, \quad \text{se } r > r_0 \quad (7.12b)$$

$$Y(r, 0) = 0, \quad \text{se } r \leq r_0 \quad (7.12c)$$

$$Y(r, 0) = 1 - e^{Le(1-\frac{r}{r_0})}, \quad \text{se } r > r_0 \quad (7.12d)$$

em que  $r_0 = 1$  é o raio inicial da bola de fogo.

O valor  $r$  é dado como:

- Para o caso bidimensional:

$$r = \sqrt{\frac{X^2}{a^2} + \frac{Y^2}{b^2}}, \quad a = 2, b = 1 \quad (7.13)$$

em que:

$$X = x \cos(\theta) - y \sin(\theta) \quad (7.14a)$$

$$Y = -x \sin(\theta) + y \cos(\theta) \quad (7.14b)$$

- Para o caso tridimensional:

$$r = \sqrt{\frac{X^2}{a^2} + \frac{Y^2}{b^2} + \frac{Z^2}{c^2}}, \quad a = 1, 5, b = 1, 5, c = 3 \quad (7.15)$$

em que:

$$X = x \cos(\theta) - y \sin(\theta) \quad (7.16a)$$

$$Y = [x \sin(\theta) + y \cos(\theta)] \cos(\phi) - z \sin(\phi) \quad (7.16b)$$

$$Z = [x \sin(\theta) + y \cos(\theta)] \sin(\phi) + z \cos(\phi) \quad (7.16c)$$

As condições de contorno adotadas são as de Neumann. Nestas simulações são utilizados os parâmetros  $Ze = 10$ ,  $\tau = 0, 64$ ,  $Le = 0, 3$ ,  $\theta = \frac{\pi}{3}$ ,  $\phi = \frac{\pi}{4}$  e  $\sigma = 0, 5$ .

### 7.3 Equações de Euler

As Equações de Euler modelam a dinâmica de um gás não ionizado, este modelo é representado pelas equações:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0 \quad (7.17a)$$

$$\frac{\partial \rho \vec{v}}{\partial t} + \nabla \cdot (\rho \vec{v}) \vec{v} + \nabla p = 0 \quad (7.17b)$$

$$\frac{\partial E}{\partial t} + \nabla \cdot (\vec{v}(E + p)) = 0 \quad (7.17c)$$

em que  $p$  é a pressão,  $\vec{v}$  é a velocidade e  $\rho$  é a densidade do fluido. A energia por unidade de massa  $E$  é dada por:

$$E = \rho e + \frac{\rho \|\vec{u}\|^2}{2} \quad (7.18)$$

Este sistema é completado pela equação de estado de um gás ideal:

$$p = \frac{\rho T}{\gamma Ma^2} \quad (7.19)$$

em que  $\gamma$  é a taxa de calor específico,  $T$  é a temperatura e  $Ma$  é o número de Mach.

Para o caso bidimensional, as Equações 7.17 podem ser expressas na forma da Equação 2.1 utilizando os valores:

$$U = \begin{bmatrix} \rho \\ \rho v_{x_1} \\ \rho v_{x_2} \\ E \end{bmatrix} \quad (7.20a)$$

$$F_1 = \begin{bmatrix} \rho v_{x_1} \\ p + \rho v_{x_1}^2 \\ \rho v_{x_1} v_{x_2} \\ v_{x_1}(E + p) \end{bmatrix} \quad (7.20b)$$

$$F_2 = \begin{bmatrix} \rho v_{x_2} \\ \rho v_{x_1} v_{x_2} \\ p + \rho v_{x_2}^2 \\ v_{x_2}(E + p) \end{bmatrix} \quad (7.20c)$$

em que a velocidade é dada pelo vetor  $\vec{v} = (v_{x_1}, v_{x_2})$ .

Este problema é simulado até o instante  $t = 0,5$ , utilizando a condição de fronteira de Neumann, número de CFL  $\sigma = 0,5$  e os parâmetros físicos  $Ma = 1$  e  $\gamma = 1,4$ , dentro de um domínio  $[-1; 1] \times [-1; 1]$ .

A condição inicial usada neste trabalho é conhecida como configuração #6 de Lax-Liu. O domínio é dividido em quatro quadrantes cujos valores iniciais estão contidos na Tabela 7.1.

Tabela 7.1 - Configuração da condição inicial utilizada para a Equação de Euler.

Variáveis	Quadrante			
	1 <sup>o</sup>	2 <sup>o</sup>	3 <sup>o</sup>	4 <sup>o</sup>
Densidade - $\rho$	1,0	2,0	1,0	3,0
Pressão - $p$	1,0	1,0	1,0	1,0
Velocidade no eixo $x_1 - v_{x_1}$	0,75	0,75	-0,75	-0,75
Velocidade no eixo $x_2 - v_{x_2}$	-0,5	0,5	0,5	-0,5

Nota: 1<sup>o</sup> quadrante - superior direito; 2<sup>o</sup> quadrante - superior esquerdo;  
3<sup>o</sup> quadrante - inferior esquerdo; 4<sup>o</sup> quadrante - inferior direito.



## 8 RESULTADOS

Neste capítulo são apresentados os resultados obtidos com a técnica de tempo local proposta (MR/LT/RK3), a abordagem adaptativa espacial dada em Harten (1993), Roussel (2003), Domingues et al. (2008) (MR/RK3) e o método VF tradicional. Os detalhes de cada simulação são dados no Capítulo 7. A avaliação do método MR/LT proposto em relação ao método MR é realizada utilizando a equação:

$$gain = 1 - \frac{erro_{MR/LT}}{erro_{MR}} \cdot \frac{tempo_{MR/LT}}{tempo_{MR}} \quad (8.1)$$

em que os valores *erro* e *tempo* são os erros médios e tempo de CPU dos respectivos métodos. O valor *gain* indica, em %, o ganho do método proposto em relação ao MR.

### 8.1 Equação de Burgers

Nesta seção são apresentados os resultados obtidos para a Equação de Burgers, cujos parâmetros foram apresentados na Seção 7.1, para os casos 1D e 2D.

#### 8.1.1 Caso unidimensional

A Figura 8.1 apresenta a solução obtida ao aplicar o método VF/RK3 para o problema da Equação de Burgers unidimensional apresentado na Seção 7.1 e a solução exata do problema conforme apresentada na mesma seção. As simulações da Equação de Burgers para o caso unidimensional foram realizadas utilizando o netbook Chameleon, especificado no Apêndice B.

Ao usar malhas adaptativas com valor de *threshold*  $\epsilon = 10^{-3}$  com o mesmo número de níveis de refinamento, aplicando os métodos MR/RK3 e MR/LT/RK3, obtém-se os erros absolutos e as malhas adaptadas conforme apresentados na Figura 8.2.

A distribuição de células por nível de refinamento das malhas adaptadas obtidas para a Equação de Burgers unidimensional, utilizando os métodos MR/RK3 e LT/MR/RK3 é apresentada na Figura 8.3. Analisando as Figuras 8.2 e 8.3, é possível verificar uma maior concentração de células em níveis menos refinados, o que indica uma possível suavização da solução ao aplicar o método proposto.

Os resultados comparativos de tempo computacional, memória utilizada e erro médio obtido são apresentados na Tabela 8.1, estes resultados são obtidos para malhas com diversos números de níveis de refinamento. Para o melhor caso,  $L = 13$ , o método

proposto apresentou um valor de *gain* de 44%. O método se mostrou mais eficiente a medida em que são utilizadas malhas com mais níveis de refinamento

Conforme apresentado na Tabela 8.1, o método proposto apresentou o mesmo custo em memória e um menor tempo computacional em relação ao método MR/RK3, enquanto não houve alteração na ordem do erro obtido. A ordem deste erro não é alterada pois o erro é controlado utilizando o valor  $\epsilon$  pré-determinado, esta constante limita o erro máximo que é aceitável para todas as células do domínio.

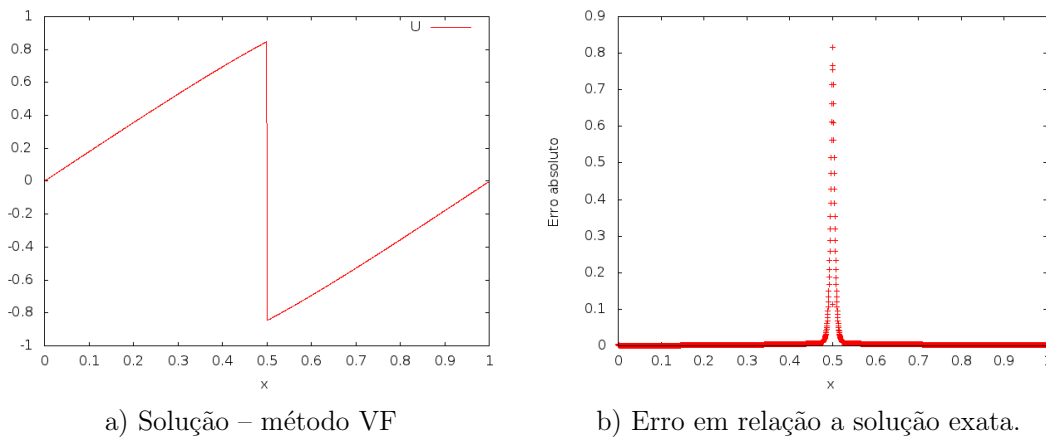


Figura 8.1 - Solução obtida para a Equação de Burgers unidimensional utilizando o método VF e erro em relação a solução exata, com uma malha de 11 níveis de refinamento (2048 células). Tempo final:  $t_f = 0,4$ .

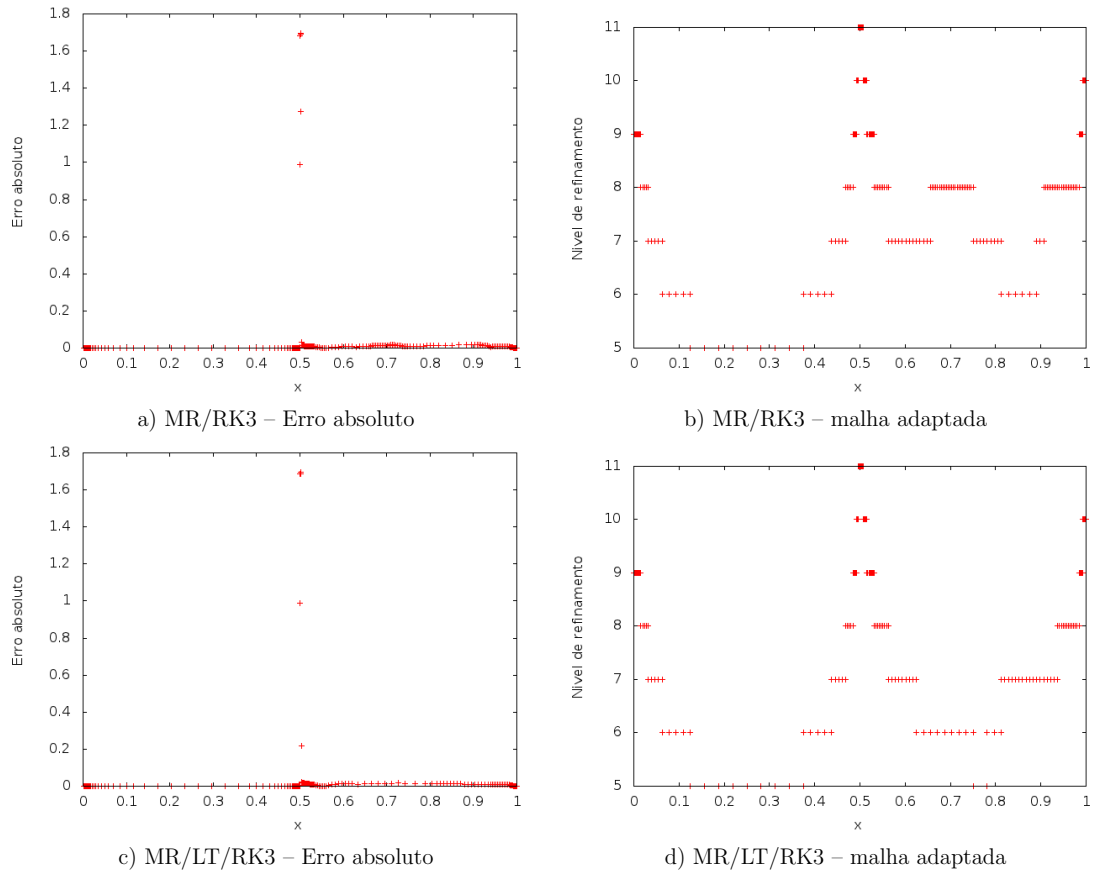


Figura 8.2 - Erro absoluto das soluções obtidas para a Equação de Burgers unidimensional e níveis de resolução das células da malha adaptada para os métodos MR/RK3 e MR/LT/RK3. Malha:  $L = 11$  (2048 células). Tempo final:  $t_f = 0,4$ .

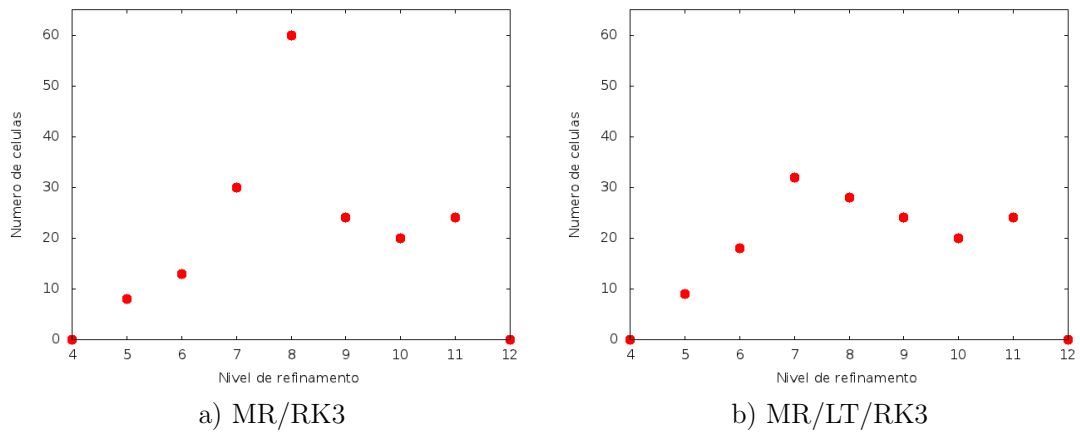


Figura 8.3 - Equação de Burgers 1D – Distribuição do número de células por nível de refinamento utilizando os métodos MR/RK3 e MR/LT/RK3.

Tabela 8.1 - Comparativo dos erros, tempos de processamento e economia de memória para a Equação de Burgers unidimensional, obtidas utilizando os métodos VF/RK3, MR/RK3 e a técnica MR/LT/RK3 proposta.  $\epsilon = 10^{-3}$ . Tempo final:  $t_f = 0,4$ .

Método		Erro médio ( $\times 10^{-1}$ )	CPU (%)	
			Tempo	Memória
$L = 11$	VF/RK3		100	100
	MR/RK3	0,56	35	23
	MR/LT/RK3	0,68	30	23
$L = 12$	VF/RK3		100	100
	MR/RK3	1,00	18	12
	MR/LT/RK3	1,10	12	12
$L = 13$	VF/RK3		100	100
	MR/RK3	1,50	10	6
	MR/LT/RK3	1,70	5	6

Nota: Computador utilizado: Netbook Chameleon. Tempos computacionais obtidos pelo método VF/RK3: 6,4 min ( $L = 11$ ); 53,1 min ( $L = 12$ ); 398,8 min ( $L = 13$ ).

### 8.1.2 Caso bidimensional

A Figura 8.4 apresenta a solução obtida ao aplicar o método VF/RK3 para o problema da Equação de Burgers bidimensional. As simulações da Equação de Burgers para este caso foram realizadas utilizando o servidor Jacaranda para as evoluções com RK2 e o desktop Gladstein para as evoluções com RK3, essas máquinas são especificadas no Apêndice B.

Ao usar malhas adaptativas com valor de *threshold*  $\epsilon = 10^{-3}$  e o mesmo número de níveis de refinamento, ao aplicar os métodos MR/RK3 e MR/LT/RK3, obtém-se os erros absolutos e as malhas adaptadas conforme apresentados na Figura 8.5.

A distribuição de células por nível de refinamento das malhas adaptadas obtidas para a Equação de Burgers unidimensional, utilizando os métodos MR/RK3 e LT/MR/RK3 é apresentada na Figura 8.6. Analisando as Figuras 8.5 e 8.6, é possível verificar uma leve diferenciação das malhas adaptadas e que os erros cometidos em relação ao método FV são bastante similares.

Os resultados comparativos de tempo computacional, memória utilizada e erro médio obtido são apresentados nas Tabelas 8.2 e 8.3. A Tabela 8.2 apresenta resultados utilizando as abordagens de tempo local para o método RK2 (MR/LT/RK2 dada em (DOMINGUES et al., 2008) e MR/LT/RKC2 proposta neste trabalho). A Tabela 8.3 compara os resultados obtidos pelos métodos de multirresolução adaptativa com e sem o uso de tempo local utilizando RK3 para evolução temporal.

Estes resultados são obtidos para malhas com diversos números de níveis de refinamento. Para o caso  $L = 11$ , o método proposto apresentou um valor de *gain* de 8% no caso do RK2 e 17% no caso do RK3, isso indica que método se mostrou mais eficiente a medida em que são utilizadas malhas com mais níveis de refinamento. O ganho obtido no caso RK2 foi calculado sobre a técnica de tempo local dada em (DOMINGUES et al., 2008).

Conforme apresentado na Tabela 8.3, o método proposto apresentou o mesmo custo em memória e um menor tempo computacional em relação ao método MR/RK3, enquanto não houve alteração na ordem do erro obtido.

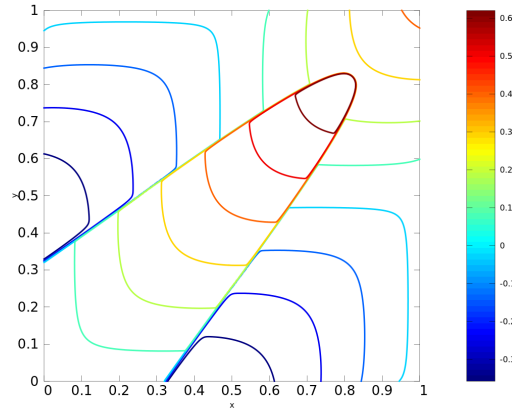


Figura 8.4 - Solução obtidas para a Equação de Burgers bidimensional utilizando o método VF com uma malha de 10 níveis de refinamento ( $1024 \times 1024$  células). Tempo final:  $t_f = 0,9$ .

Tabela 8.2 - Comparativo dos erros, tempos de processamento e economia de memória para a Equação de Burgers bidimensional, obtidas utilizando os métodos MR/LT/RK2, proposta em Domingues et al. (2008) e a técnica MR/LT/RKC2 proposta.  $\epsilon = 10^{-3}$ . Tempo final:  $t_f = 0,9$ .

Método		Erro médio ( $\times 10^{-3}$ )	CPU (%)	
			Tempo	Memória
$L = 10$	MR/LT/RK2	4,17	39	17
	MR/LT/RKC2	4,14	36	17
$L = 11$	MR/LT/RK2	4,78	13	7
	MR/LT/RKC2	4,65	13	7

Nota: Computador utilizado: Servidor Jacaranda. Tempos computacionais obtidos pelo método VF/RK2: 5,7h ( $L = 10$ ); 25,8h ( $L = 11$ ).

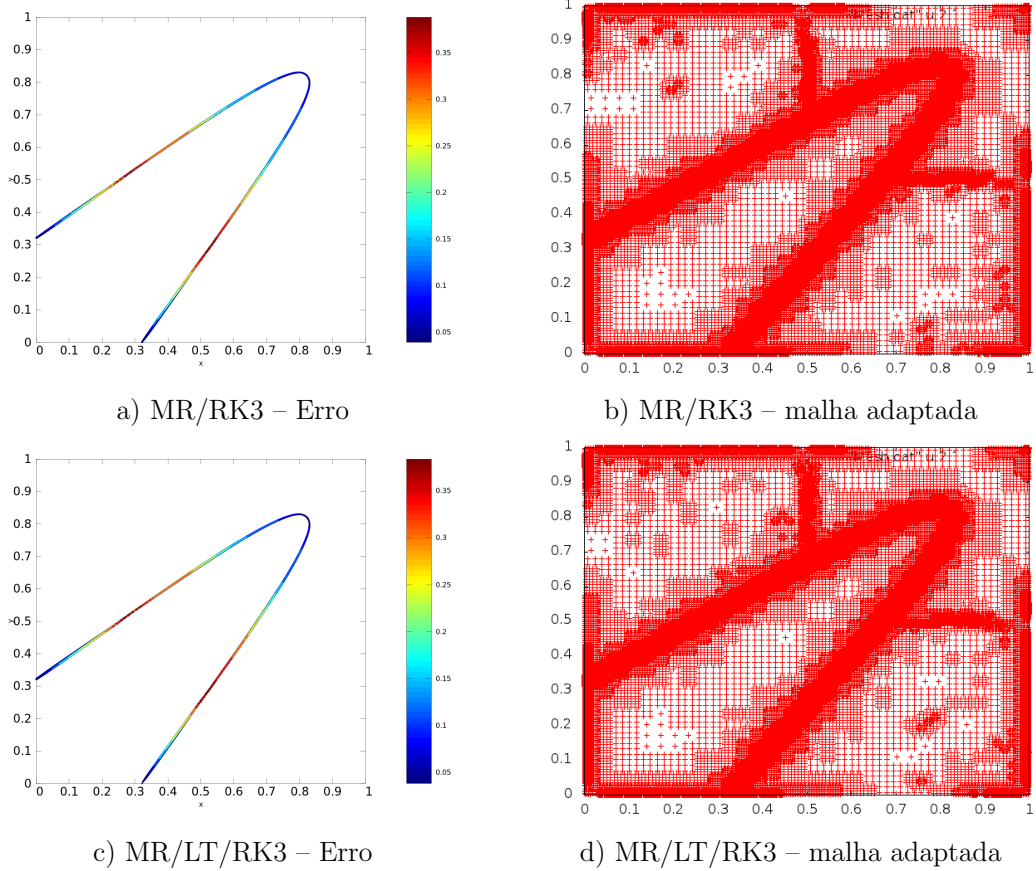


Figura 8.5 - Erro absoluto das soluções obtidas para a Equação de Burgers bidimensional e níveis de resolução das células da malha adaptada utilizando os métodos MR/RK3 e MR/LT/RK3. Malha:  $L = 10$  níveis (1024 células). Tempo final:  $t_f = 0,9$ .

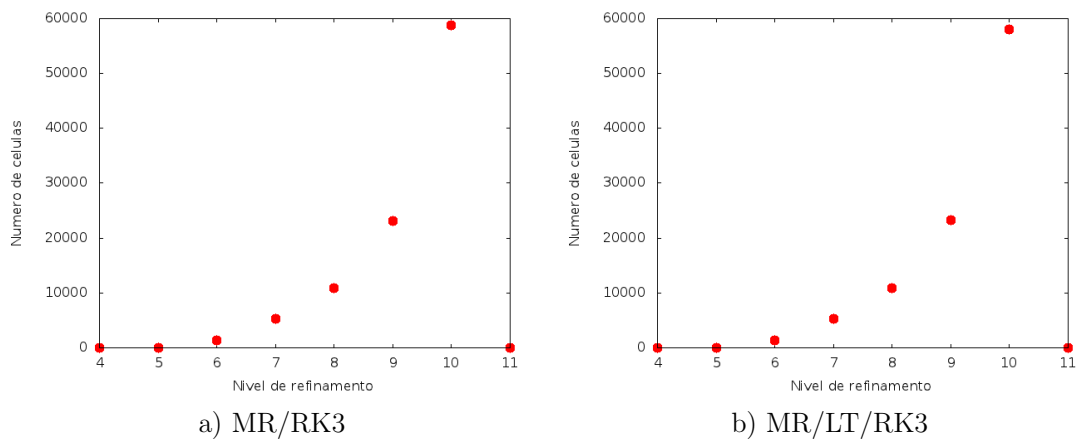


Figura 8.6 - Equação de Burgers 2D – Distribuição do número de células por nível de refinamento utilizando os métodos MR/RK3 e MR/LT/RK3.

Tabela 8.3 - Comparativo dos erros, tempos de processamento e economia de memória para a Equação de Burgers bidimensional, obtidas utilizando os métodos MR/RK3 e a técnica MR/LT/RK3 proposta.  $\epsilon = 10^{-3}$ . Tempo final:  $t_f = 0,9$ .

	Método	Erro médio ( $\times 10^{-3}$ )	CPU (%)	
			Tempo	Memória
$L = 10$	MR/RK3	4,12	62	17
	MR/LT/RK3	4,05	61	18
$L = 11$	MR/RK3	4,51	27	7
	MR/LT/RK3	4,55	22	7

Nota: Computador utilizado: Desktop Gladshein. O método VF foi executado utilizando outra máquina (servidor Jacaranda). As porcentagens do tempo computacional foram obtidas a partir de projeções realizadas pelo software carmen. Tempos computacionais projetados para o método VF/RK3: 8,4h ( $L = 10$ ); 66,7h ( $L = 11$ ).



## 8.2 Problema de chamadas

Nesta seção são apresentados os resultados obtidos para o problema de chamadas, cujos parâmetros foram apresentados na Seção 7.2, para os casos 1D e 3D.

### 8.2.1 Caso unidimensional

A solução obtida para o caso unidimensional ao utilizar o método VF/RK3 é apresentada na Figura 8.7. As simulações do problema de chamadas unidimensional foram realizadas utilizando o netbook Chameleon, especificado no Apêndice B.

Ao usar malhas adaptativas com valor de *threshold*  $\epsilon = 10^{-3}$  e o mesmo número de níveis de refinamento, ao aplicar os métodos MR/RK3 e MR/LT/RK3, obtém-se os erros absolutos para as variáveis  $T$ ,  $Y$  e  $\omega$ , apresentados na Figura 8.9. As respectivas malhas adaptadas e distribuição de células por nível de refinamento são apresentadas na Figura 8.8. Analisando a Figura 8.8, é possível verificar que para este caso as malhas permaneceram iguais ao aplicar ambos os métodos.

Os resultados comparativos de tempo computacional, memória utilizada e erro médio obtido são apresentados na Tabela 8.4, estes resultados são obtidos para malhas com diversos números de níveis de refinamento. Para o caso  $L = 13$ , o método proposto apresentou valores de *gain* de 85% em relação à variável  $\omega$ , que foi o pior caso, e 87% em relação à variável  $T$ , que foi o melhor caso. Isso indica que para o problema de chamadas unidimensional, o método com MR/LT/RK3 apresentou uma melhora a medida em que se usa malhas mais refinadas, tanto na qualidade da solução, a ponto de superar o método MR, quanto em tempo computacional.

Conforme apresentado na Tabela 8.4, o método proposto apresentou o mesmo custo em memória e um menor tempo computacional em relação ao método MR/RK3, enquanto não houve alteração na ordem do erro obtido. A precisão da solução numérica pode ser associada a constante  $\epsilon$ . Isto ocorre devido ao fato do erro cometido ao representar as células em níveis menos refinados ser controlado por esta constante.

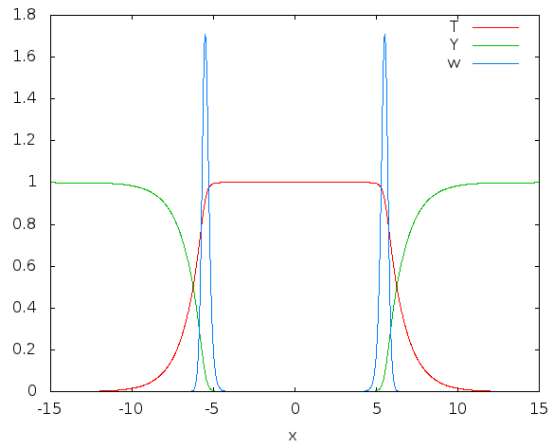
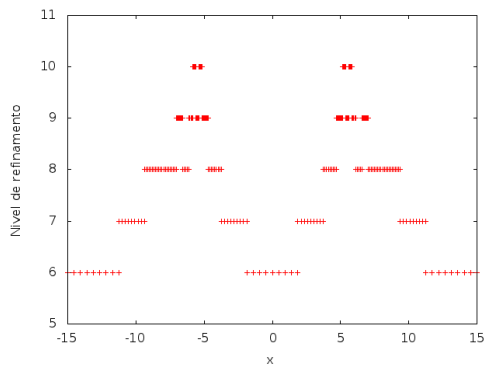


Figura 8.7 - Solução para o problema de chamas unidimensional utilizando o método VF/RK3 em uma malha de 11 níveis de resolução. Tempo final:  $t_f = 0,5$ .

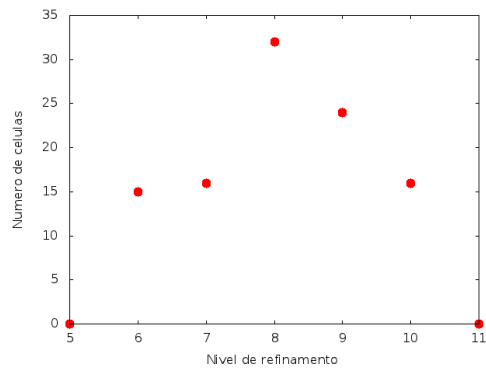
Tabela 8.4 - Comparativo dos erros, tempos de processamento e economia de memória, para o problema de chamas unidimensional, obtidas utilizando os métodos VF/RK3, MR/RK3 e a técnica MR/LT/RK3 proposta.  $\epsilon = 10^{-3}$ . Tempo final:  $t_f = 0,5$ .

	Método	Erro médio ( $\times 10^{-3}$ )			CPU (%)	
		$T$	$Y$	$\omega$	Tempo	Memória
$L = 11$	VF/RK3				100	100
	MR/RK3	0,30	0,30	1,80	46	21
	MR/LT/RK3	1,60	1,60	9,50	27	21
$L = 12$	VF/RK3				100	100
	MR/RK3	0,32	0,32	1,90	23	10
	MR/LT/RK3	0,63	0,63	3,80	9	10
$L = 13$	VF/RK3				100	100
	MR/RK3	0,32	0,32	1,90	11	5
	MR/LT/RK3	0,15	0,15	1,00	3	5

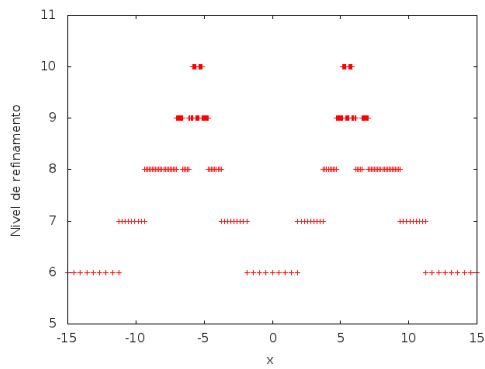
Nota: Computador utilizado: Netbook Chameleon. Tempos computacionais obtidos pelo método VF/RK3: 9,4min ( $L = 11$ ); 1,2h ( $L = 12$ ); 9,9h ( $L = 13$ ).



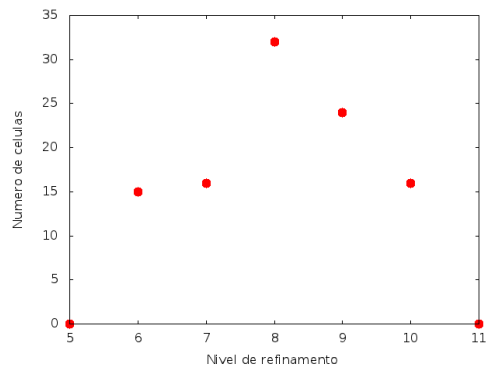
a) MR/RK3 – malha adaptada



b) MR/RK3 – distribuição de células por nível



c) MR/LT/RK3 – malha adaptada



d) – MR/LT/RK3 distribuição de células por nível

Figura 8.8 - Malhas adaptadas à solução e distribuição de células por nível de refinamento, para o problema de chamas unidimensional, obtidas ao aplicar, em uma malha de 11 níveis de resolução, os métodos MR/RK3 e MR/LT/RK3.

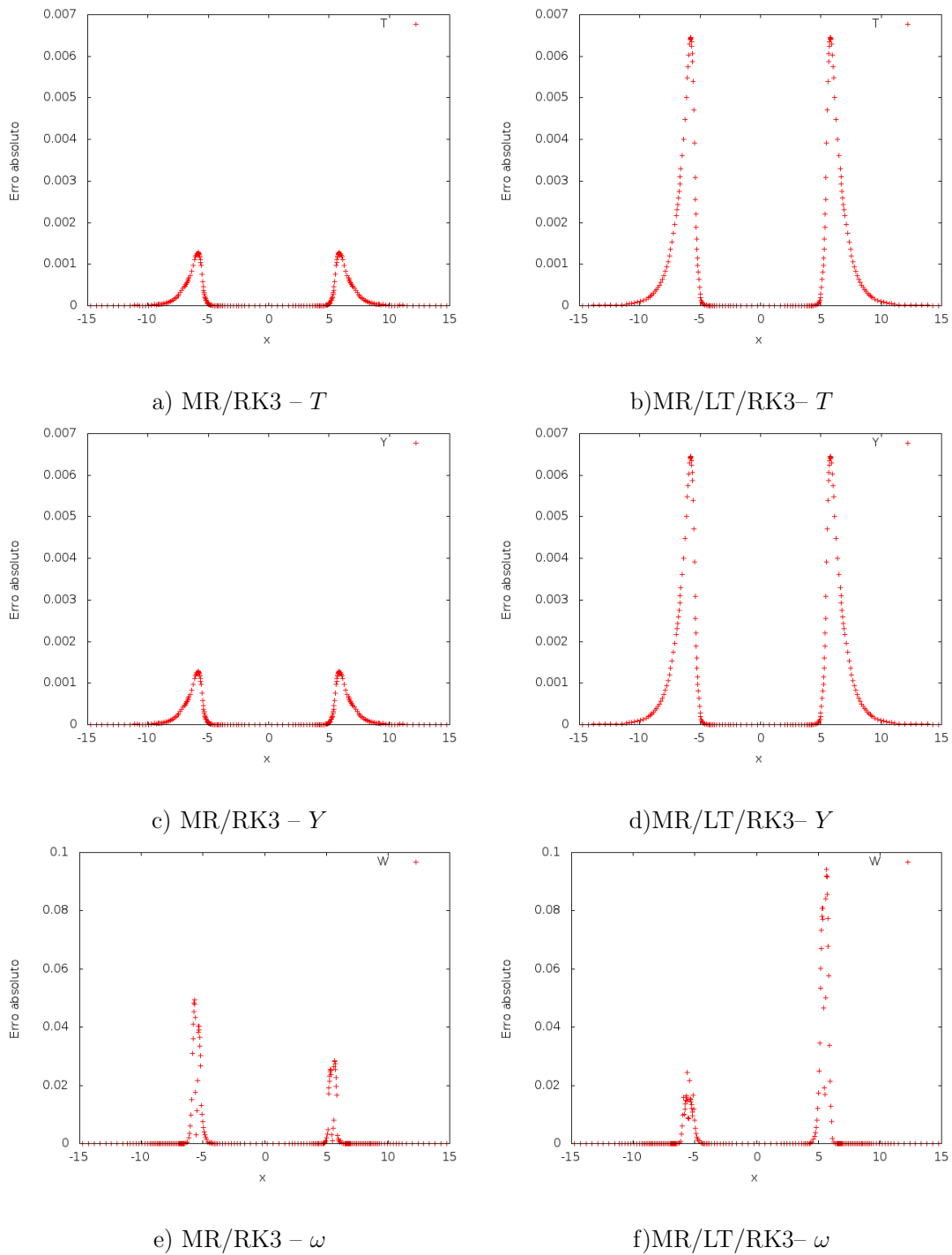


Figura 8.9 - Erro absoluto cometido em relação à solução obtida utilizando o método VF/RK3 para o problema de chamas unidimensional ao aplicar os métodos MR/RK3 e MR/LT/RK3, utilizando uma malha de 11 níveis de resolução. Variáveis: Temperatura ( $T$ ), Concentração ( $Y$ ), Taxa de reação ( $\omega$ ). Tempo final:  $t_f = 0,5$ .

### 8.2.2 Caso tridimensional

A simulações do problema de chamas tridimensional foram realizadas utilizando o servidor Walkjrjas , especificado no Apêndice B.

Ao usar malhas adaptativas com valor de *threshold*  $\epsilon = 10^{-2}$  e o mesmo número de níveis de refinamento, ao aplicar os métodos MR/RK3 e MR/LT/RK3, obtém-se os erros absolutos para as variáveis  $T$ ,  $Y$  e  $\omega$ , apresentados na Figura 8.9. A distribuição de células por nível de refinamento ao aplicar os métodos MR/RK3 e MR/LT/RK3 é apresentada na Figura 8.10. As simulações foram realizadas até o tempo final:  $t_f = 15$ .

Os resultados comparativos de tempo computacional, memória utilizada e erro médio obtido são apresentados na Tabela 8.5, estes resultados são obtidos para malhas com diversos números de níveis de refinamento. Para o caso  $L = 7$ , o método proposto apresentou valores de *gain* de 9% em relação à variável  $\omega$ , que foi o pior caso, e 43% para a variável  $Y$ , o melhor caso. Espera-se obter melhores resultados ao aplicar em casos com malhas com mais níveis de refinamento, pois a malha com  $L = 7$  é considerada pouco refinada.

Conforme apresentado na Tabela 8.5, o método proposto apresentou o mesmo custo em memória e um menor tempo computacional em relação ao método MR/RK3, enquanto não houve alteração na ordem do erro obtido.

O comportamento da solução do problema de chamas tridimensional é ilustrado nas Figuras 8.11. Essas figuras mostram a evolução do problema, uma estrutura de chamas que se replica e expande ao longo do tempo.

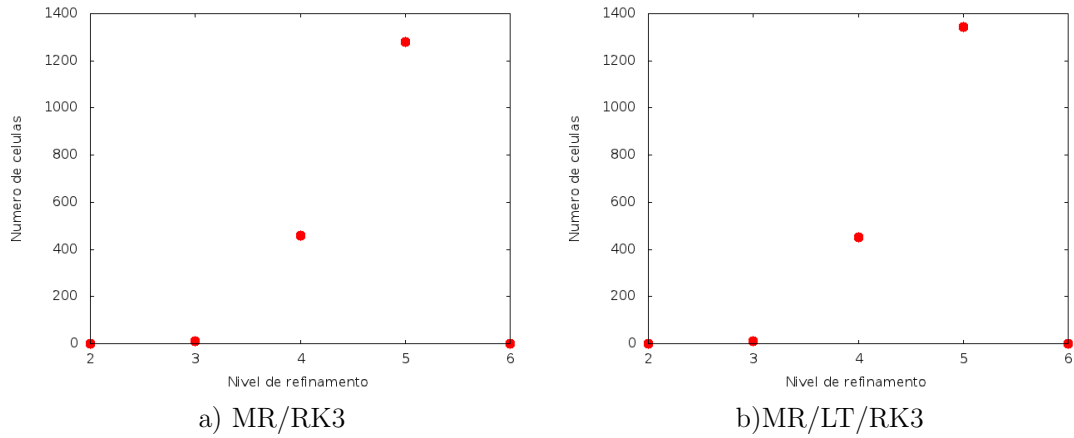


Figura 8.10 - Problema de chamadas 3D – Distribuição do número de células por nível de refinamento utilizando os métodos MR/RK3 e MR/LT/RK3.

Tabela 8.5 - Comparativo dos erros, tempos de processamento e economia de memória, para o problema de chamadas tridimensional, obtidas utilizando os métodos VF/RK3, MR/RK3 e a técnica MR/LT/RK3 proposta.  $\epsilon = 10^{-2}$ . Tempo final:  $t_f = 15$ .

	Método	Erro médio( $\times 10^{-4}$ )			CPU (%)	
		$T$	$Y$	$\omega$	Tempo	Memória
$L = 6$	VF/RK3				100	100
	MR/RK3	0,67	2,22	$1,23 \times 10^{-7}$	64	26
	MR/LT/RK3	1,41	4,61	$3,07 \times 10^{-7}$	52	26
$L = 7$	VF/RK3				100	100
	MR/RK3	0,72	2,63	$1,27 \times 10^{-7}$	14	6
	MR/LT/RK3	0,98	2,33	$1,79 \times 10^{-7}$	9	6

Nota: Computador utilizado: Servidor Walkjrjas. Tempos computacionais obtidos pelo método VF/RK3: 33,7min ( $L = 6$ ); 20,7h ( $L = 7$ ).

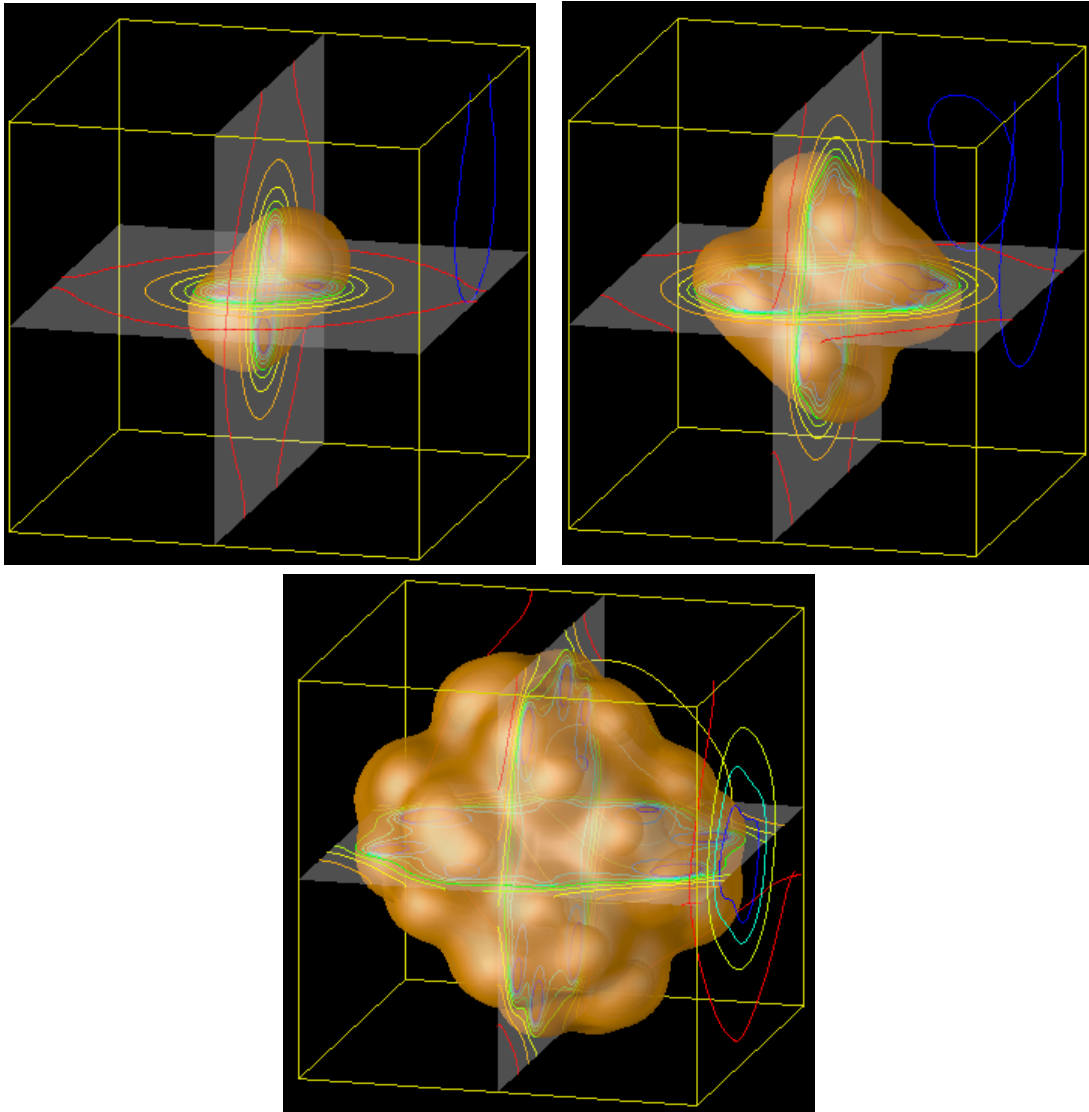


Figura 8.11 - Solução do problema de chamas 3D em três instantes distintos.

### 8.3 Equações de Euler bidimensionais

A solução do problema das equações de Euler, com os parâmetros apresentados na Seção 7.3 e uma malha de  $L = 9$  níveis, é apresentada na Figura 8.12. A simulações do problema das equações de Euler foram realizadas utilizando o Desktop Gadshein, especificado no Apêndice B.

Ao usar malhas adaptativas com valor de *threshold*  $\epsilon = 10^{-3}$  e o mesmo número de níveis de refinamento, ao aplicar os métodos MR/RK3 e MR/LT/RK3, obtém-se os erros absolutos para as variáveis  $\rho$ ,  $p$  e  $T$ , exibidos na Figura 8.14, e para as variáveis  $E$ ,  $v_x$  e  $v_y$ , exibidos na Figura 8.15. As respectivas malhas adaptadas e distribuição de células por nível de refinamento são apresentadas na Figura 8.13. Analisando as Figuras 8.9 e 8.8, é possível verificar que a malha referente ao método MR/LT/RK3 é levemente menos refinada do que a obtida pelo método MR/RK3. Quanto aos erros, o método MR/LT/RK3 apresentou pequenas estruturas extras em relação ao método MR/RK3 nas variáveis  $p$ ,  $E$ ,  $v_x$  e  $v_y$ .

Os resultados comparativos de tempo computacional, memória utilizada e erro médio obtido são apresentados na Tabela 8.6, estes resultados são obtidos para malhas com diversos números de níveis de refinamento. Para o caso  $L = 11$ , o método proposto apresentou valores de *gain* de  $-12\%$  em relação à variável  $v_y$ , que foi o pior caso, e  $-1\%$  para a variável  $\rho$ , o melhor caso. Espera-se melhores resultados ao aplicar o método em malhas com mais níveis de refinamento.

Conforme apresentado na Tabela 8.6, o método proposto apresentou o mesmo custo em memória e um menor tempo computacional em relação ao método MR/RK3, enquanto não houve alteração na ordem do erro obtido.



Tabela 8.6 - Comparativo dos erros, tempos de processamento e economia de memória, para as Equações de Euler bidimensionais, obtidas utilizando os métodos VF/RK3, MR/RK3 e a técnica MR/LT/RK3 proposta.  $\epsilon = 10^{-3}$ . Tempo final:  $t_f = 0,25$ .

	Método	Erro médio( $\times 10^{-3}$ )						CPU (%)	
		$\rho$	$p$	$T$	$E$	$v_x$	$v_y$	Tempo	Memória
$L = 9$	VF/RK3							100	100
	MR/RK3	0,77	0,78	0,27	1,80	0,38	0,29	71	40
	MR/LT/RK3	1,00	1,00	0,33	2,50	0,53	0,44	68	40
$L = 10$	VF/RK3							100	100
	MR/RK3	0,96	0,94	0,32	2,10	0,41	0,37	38	21
	MR/LT/RK3	1,20	1,20	0,40	2,80	0,60	0,55	36	21
$L = 11$	VF/RK3							100	100
	MR/RK3	1,96	1,60	0,68	3,30	0,89	0,73	20	10
	MR/LT/RK3	2,20	1,90	0,76	4,00	1,10	0,91	18	10

Nota: Computador utilizado: Desktop Gldshein. Tempos computacionais obtidos pelo método VF/RK3: 51,4min ( $L = 9$ ); 6,9h ( $L = 10$ ); 54,6h ( $L = 11$ ).

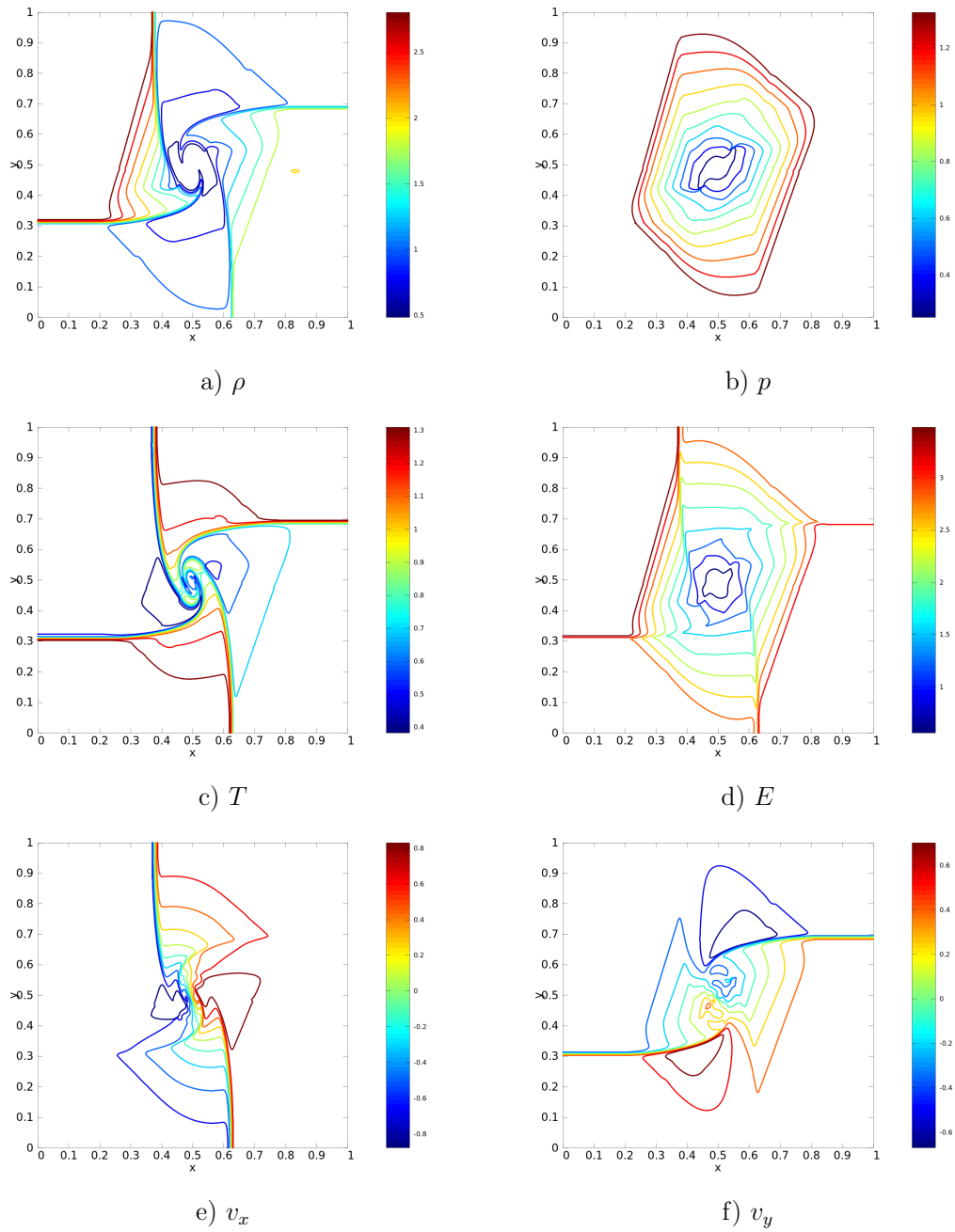
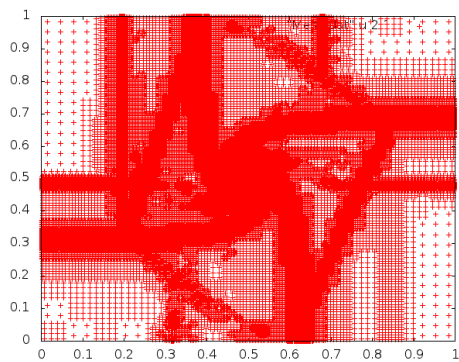
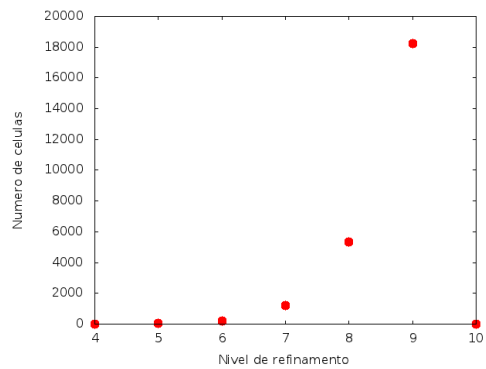


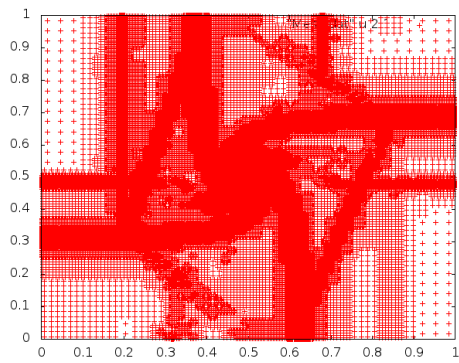
Figura 8.12 - Solução obtida para o problema das Equações de Euler bidimensionais utilizando o método VF/RK3 utilizando uma malha com  $L = 9$  níveis de resolução, ou seja, uma malha de  $512 \times 512$  células. Variáveis: a) densidade ( $\rho$ ); b) pressão ( $p$ ); c) temperatura ( $T$ ); d) energia ( $E$ ); e) velocidade no eixo  $x$  ( $v_x$ ); f) velocidade no eixo  $y$  ( $v_y$ ). Tempo final:  $t_f = 0,25$ .



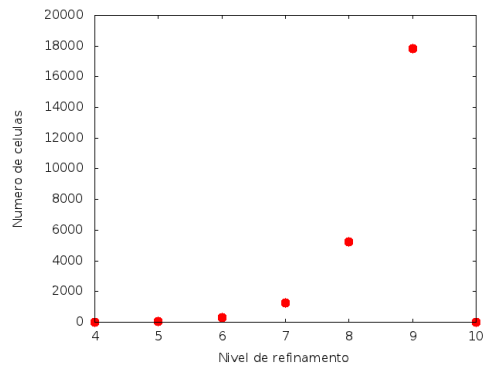
a) MR/RK3 – Malha adaptada



b) MR/RK3 – Distribuição de células por nível

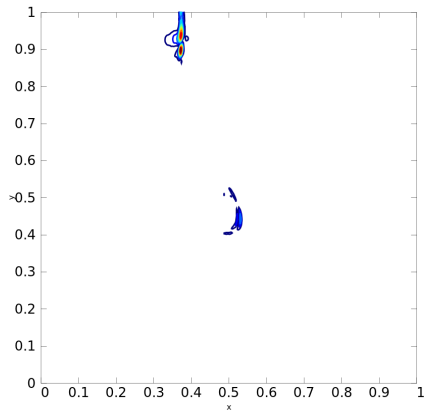


c) MR/RK3/LT – Malha adaptada

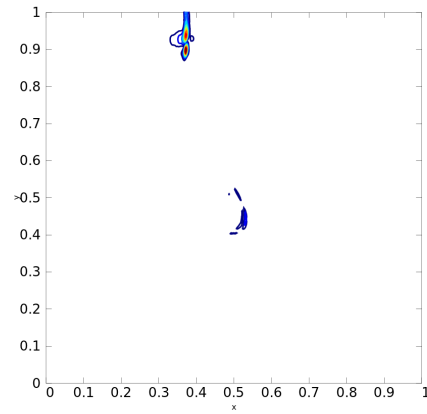


d) MR/RK3/LT – Distribuição de células por nível

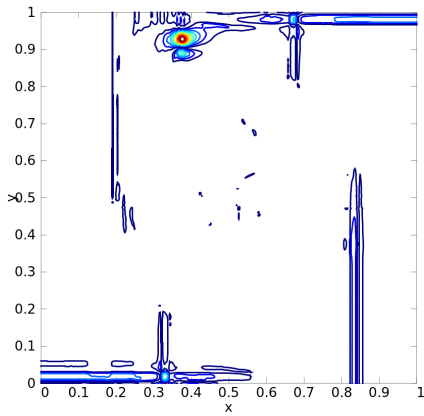
Figura 8.13 - Malhas adaptadas à solução e distribuição de células por nível de refinamento, para o problema das Equações de Euler bidimensional, obtidas ao aplicar, em uma malha com  $L = 9$  níveis de resolução, os métodos MR/RK3 e MR/LT/RK3.



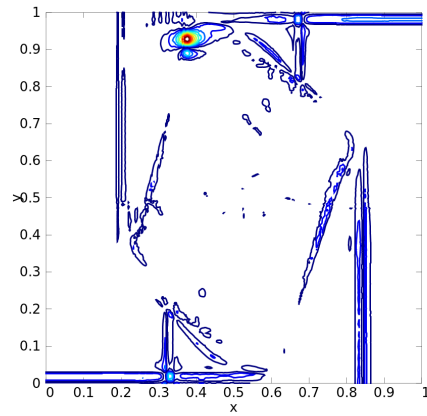
a) MR/RK3 -  $\rho$



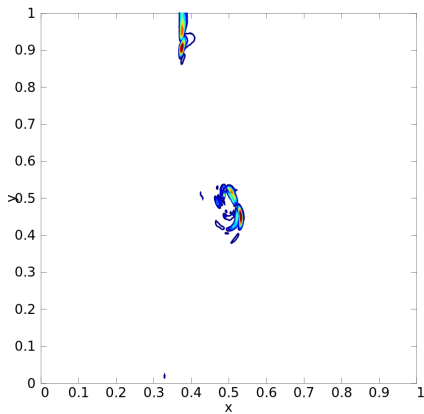
b) MR/RK3/LT -  $\rho$



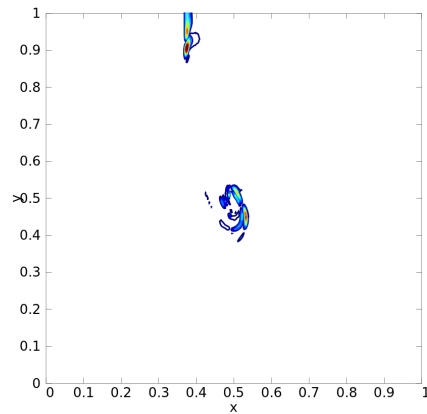
c) MR/RK3 -  $p$



d) MR/RK3/LT -  $p$

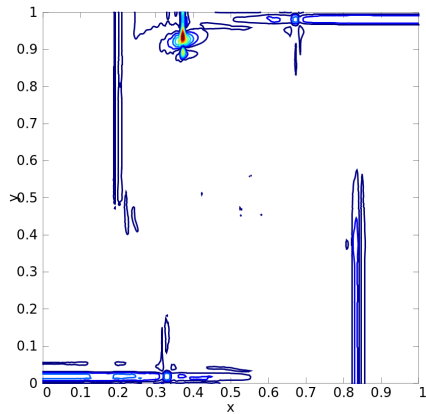


e) MR/RK3 -  $T$

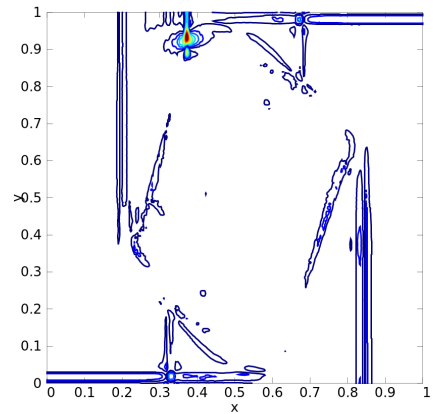


f) MR/RK3/LT -  $T$

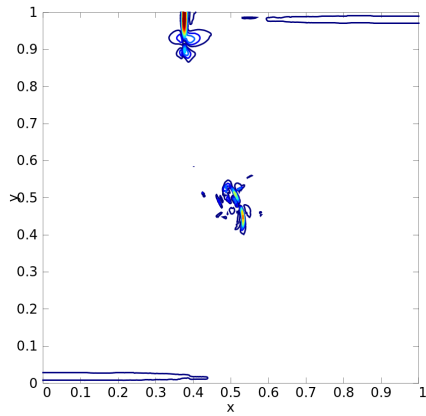
Figura 8.14 - Erro absoluto cometido em relação à solução obtida utilizando o método VF/RK3 para o problema das Equações de Euler bidimensionais ao aplicar os método MR/RK3 e MR/LT/RK3, utilizando uma malha de  $L = 9$  níveis de resolução, ou seja, em uma malha de  $512 \times 512$  células. Variáveis: a) densidade ( $\rho$ ); b) pressão ( $p$ ); c) temperatura ( $T$ ). Tempo final:  $t_f = 0,25$ .



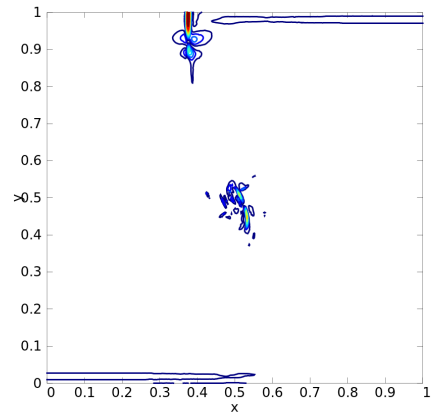
a) MR/RK3 –  $E$



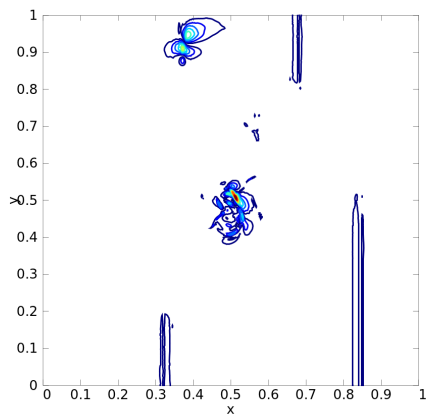
b) MR/RK3/LT –  $E$



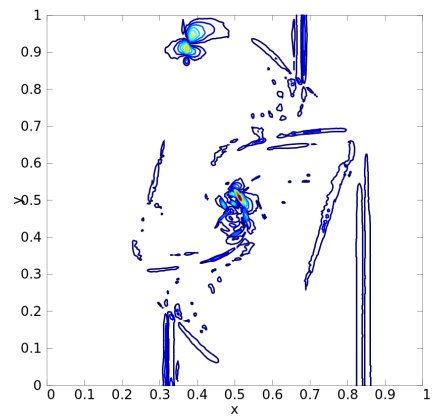
c) MR/RK3 –  $v_x$



d) MR/RK3/LT –  $v_x$



e) MR/RK3 –  $v_y$



f) MR/RK3/LT –  $v_y$

Figura 8.15 - Erro absoluto cometido em relação à solução obtida utilizando o método VF/RK3 para o problema das Equações de Euler bidimensionais ao aplicar os método MR/RK3 e MR/LT/RK3, utilizando uma malha de  $L = 9$  níveis de resolução, ou seja, com  $512 \times 512$  células. Variáveis: Energia ( $E$ ); velocidade no eixo  $x$  ( $v_x$ ); c) velocidade no eixo  $y$  ( $v_y$ ). Tempo final:  $t_f = 0,25$ .



## 9 CONCLUSÕES

As pesquisas em ciências espaciais e da atmosfera demandam, na atualidade, cada vez mais o auxílio de modelagens numérico-computacionais, que propiciam ferramentas para completar a compreensão dos fenômenos ou mesmo propor aspectos inéditos desses. O INPE tem esforços importantes de estudos e desenvolvimentos nessas áreas de atuação científica. Motivado por essas necessidades, este trabalho de mestrado ocupou-se do desenvolvimento de métodos para resolução numérica de equações diferenciais evolutivas, que são base das modelagens em hidrodinâmica e magneto-hidrodinâmica.

A abordagem diferencial desenvolvida é a utilização da análise multirresolução adaptativa em que se propõe uma técnica para ajuste do passo de tempo considerando a análise da regularidade espacial da solução.

Dessa forma, como maneira de condução à resolução, propôs-se e desenvolveu-se uma nova abordagem, com a aplicação do método de Runge–Kutta contínuo (RKC) de ordens superiores no contexto de uma evolução temporal utilizando técnicas de tempo local para métodos de multirresolução adaptativa. Explora-se a propriedade de que a multirresolução (MR) gera uma malha ajustada espacialmente à solução do problema estudado. Este trabalho, assim, contribui efetivamente para uma abordagem adaptativa tanto espacial quanto temporal.

Como resultados, para problemas com estruturas bem localizadas, o método proposto apresentou vantagens em termos de tempo computacional quando comparado com o método de volumes finitos (VF) tradicional e também o método com adaptatividade espacial (MR) sem o uso da técnica de tempo local. Conclui-se, assim, que a técnica de MR utilizando a técnica de tempo local (MR/LT) mostrou-se como uma alternativa competitiva para a simulação desses problemas. Entretanto, para problemas sem estruturas localizadas, o método proposto apresenta menos desempenho, apesar do ganho em tempo computacional.

Com o sucesso obtido neste trabalho, a técnica será aplicada em trabalhos futuros para problemas de magnetohidrodinâmica com aplicações em plasma espacial em que existam estruturas localizadas. Também pretende-se estudar novas abordagens para a implementação da malha computacional, visando aprimorar as buscas realizadas e reduzir a quantidade de memória computacional alocada nas simulações.

Além disto, pretende-se estudar técnicas para a paralelização dos processos envol-

vidos nas simulações, novamente visando melhorar o desempenho das resoluções numéricas para problemas evolutivos com estruturas localizadas.



## REFERÊNCIAS BIBLIOGRÁFICAS

- BIHARI, B. L.; HARTEN, A. Multiresolution schemes for the numerical solution of 2-D conservation laws i. **SIAM J. Sci. Comput.**, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, v. 18, n. 2, p. 315–354, mar. 1997. ISSN 1064-8275. Disponível em:  
<<http://dx.doi.org/10.1137/S1064827594278848>>. 21
- BOTHMER, V.; DAGLIS, I. A. **Space weather - physics and effects**. Chichester: Springer-Verlag, 2007. 1
- BÜCHNER, J.; DUM, C. T.; SCHOLER, M. **Space plasma simulation**. Berlin: Springer-Verlag, 2003. 1
- BURDEN, R. L.; FAIRES, D. J. **Análise numérica**. [S.l.]: Cengage Learning, 2008. 33, 34, 35, 89, 91
- CHOUDHURI, A. R. **The physics of fluids and plasmas - an introduction for astrophysicists**. Cambridge: Cambridge University Press, 2004. 1
- COHEN, A. **Numerical analysis of wavelet methods**. Amsterdam: JAI Press, 2003. 26
- DEITERDING, R. Block-structured adaptive mesh refinement - theory, implementation and application. **Esaim: Proc.**, SMAI, Institut Henri Poincaré, Paris; EDP Sciences, Les Ulis, v. 34, p. 97–150, 2011. ISSN 1270-900X/e. 1
- DOMINGUES, M. O.; GOMES, S. M.; ROUSSEL, O.; SCHNEIDER, K. An adaptive multiresolution scheme with local time stepping for evolutionary PDEs. **J. Comput. Phys.**, v. 227, n. 8, p. 3758–3780, abr. 2008. Disponível em:  
<<http://dx.doi.org/10.1016/j.jcp.2007.11.046>>. 2, 49, 58, 63, 67, 68
- \_\_\_\_\_. Adaptive multiresolution methods. **Esaim: Proc.**, v. 34, p. 1–96, 2011. 1, 29
- HARTEN, A. Multiresolution algorithms for the numerical solution of hyperbolic conservation laws. **CAM Report**, 1993. 1, 2, 21, 26, 29, 63
- \_\_\_\_\_. Adaptive multiresolution schemes for shock computations. **J. Comput. Phys.**, v. 115, p. 319–338, 1994. ISSN 1270-900X/e. 2
- HOLMSTRÖM, M. **Wavelet based methods for time dependent PDEs**. Tese (Doutorado) — University of Uppsala, 1997. 57

- JARDIN, S. **Computational methods in plasma physics**. New York: CRC Press, 2010. 1
- KIVELSON, M. G.; RUSSELL, C. T. **Introduction to space physics**. Cambridge: Cambridge University Press, 1996. 1
- LEVEQUE, R. J. **Finite-volume methods for hyperbolic problems**. Cambridge: Cambridge University Press, 2004. 5
- MALLAT, S. **A wavelet tour of signal processing**. Burlington: Elsevier, 1998. 2, 13, 29
- MALLAT, S. G. A theory for multiresolution signal decomposition: The wavelet representation. **Transactions on Pattern Analysis and Machine Intelligence**, 1989. 1
- MIYOSHI, T.; KUSANO, K. A multi-state HLL approximate Riemann solver for ideal magnetohydrodynamics. **J. Comput. Phys.**, v. 208, p. 315–344, 2005. 11
- MOURA, C. A. de; KUBRUSLY, C. S. **The Courant-Friedrichs-Lewy (CFL) condition: 80 years after its discovery**. [S.l.]: Birkhäuser, 2012. 9
- MÜLLER, S. **Adaptive multiscale schemes for conservation laws**. [S.l.]: Springer, Heidelberg, 2003. 2
- ROUSSEL, O. **Developpement d'un algorithme multiresolution adaptatif tridimensionnel pour la resolution des equations aux derivees patielles paraboliques: application aux instabilites thermodiffusives de flamme**. Tese (Doutorado) — Universite de la mediterranee, 2003. 18, 22, 23, 36, 58, 63
- ROUSSEL, O.; SCHNEIDER, K.; TSIGULIN, A.; BOCKHORN, H. A conservative fully adaptive multiresolution algorithm for parabolic PDEs. **J. Comput. Phys.**, v. 188, p. 493–523, 2003. 2
- TORO, E. F. **Riemann solvers and numerical methods for fluid dynamics: a practical introduction**. [S.l.]: Springer, 1999. 1, 10, 11
- ZENNARO, M. Natural continuous extensions of runge-kutta methods. **Mathematics of Computation**, v. 46, p. 119–133, 1986. 2, 39
- \_\_\_\_\_. Order barriers for continuous explicit Runge-Kutta methods. **Mathematics of Computation**, v. 56, p. 645–661, 1991. 40

## APÊNDICE A – COEFICIENTES PARA MÉTODOS RK

Neste apêndice é apresentada uma técnica para a obtenção dos coeficientes  $a_{i,j}$ ,  $b_i$  e  $c_i$  usados nos métodos de Runge–Kutta tradicionais e explícitos. A técnica apresentada para a obtenção de um método RK de ordem  $p$  e  $s$  estágios consiste em comparar a equação do método RK de  $s$  estágios (Equação A.6b) com uma série de Taylor truncada no termo de ordem  $O((\Delta t^m)^{p+1})$ .

### A.1 Série de Taylor

A série de Taylor aproxima a vizinhança, ao redor de um ponto de valor conhecido, de uma função utilizando a soma de infinitos termos que dependem das derivadas da função e da distância entre o ponto a ser aproximado e o ponto conhecido. A expansão em série de Taylor para uma função  $f(t, Q)$  na direção de  $t$  é escrita como:

$$f(t + \Delta t, Q + \Delta Q) = f(t, Q) + \sum_{k=1}^{\infty} \frac{\Delta t^k}{k!} \frac{d^k f(t, Q + \Delta Q)}{dt^k} \quad (\text{A.1})$$

Expandindo os termos  $\frac{d^k f(t, Q + \Delta Q)}{dt^k}$  em séries de Taylor na direção  $Q$  obtêm-se a expansão em série de Taylor para uma função de duas variáveis:

$$f(t + \Delta t, Q + \Delta Q) = \sum_{p=0}^{\infty} \frac{1}{p!} \sum_{q=0}^p C_q^p \Delta t^{p-q} \Delta Q^q \frac{\partial^p f(t, Q)}{\partial t^{p-q} \partial Q^q} \quad (\text{A.2})$$

em que

$$C_q^p = \frac{p!}{q!(p-q)!} \quad (\text{A.3})$$

### A.2 Ordem de convergência

A ordem de um método numérico está associada à grandeza do erro esperado em cada iteração. O método RK é dito de ordem de convergência  $p$  se o erro de discretização do método RK for da ordem de  $\Delta t^{p+1}$ . O erro de discretização é definido como o erro cometido em uma evolução do método numérico ( $\bar{Q}^{m+1}$ ) em relação ao valor exato da função no ponto estudado ( $Q^{m+1}$ ), ou seja:

$$\|Q^{m+1} - \bar{Q}^{m+1}\| = \sum_{k=p+1}^{\infty} \sum_{q=0}^k \bar{c}_{pq} \Delta t^{k-q} \Delta Q^q \frac{\partial^k f(t, Q)}{\partial t^{k-q} \partial Q^q} = O(\Delta t^{p+1}) \quad (\text{A.4})$$

em que os valores  $\bar{c}_{pq}$  são constantes arbitrárias. Segundo o teorema de equivalência de Lax (BURDEN; FAIRES, 2008), a convergência de um método numérico depende

de duas condições:

- **Consistência:** O erro cometido em cada aproximação se aproxima de zero a medida que o passo temporal tende a zero;

$$\lim_{\Delta t_m \rightarrow 0} \|Q^{m+1} - \bar{Q}^{m+1}\| = 0 \quad (\text{A.5})$$

- **Estabilidade:** Os erros cometidos durante o processo não são amplificados. Dependendo da escolha do esquema numérico, podem ocorrer três casos: O método ser absolutamente estável, o que implica que independente dos parâmetros  $\Delta t$  e  $\Delta x$ , o método não propagará erros de forma que estes sobreponham a solução; O método ser instável, isto é, a solução não irá convergir para a solução exata do problema; O método ser condicionalmente estável, o que implica que os erros cometidos pelas aproximações não serão propagados de forma a dominar o sistema desde que a razão  $\frac{\Delta t}{\Delta x}$  seja menor do que uma constante CFL  $\sigma$  que depende do problema a ser analisado.

Ou seja, se estas condições não forem satisfeitas, o resultado obtido com o método numérico não será coerente com a solução do modelo.

O uso de métodos explícitos está associado a esquemas numéricos condicionalmente estáveis, isso será utilizado para determinar o valor  $\Delta t^m$  de cada evolução temporal em função dos já pré-determinados  $\Delta x$  e  $\sigma$ .

### A.3 Obtenção dos coeficientes de um método RK

Os coeficientes do método RK de ordem  $p$  são obtidos ao realizar a expansão dos dois termos da Equação A.4, de forma que a igualdade continue válida. Na prática, este método consiste em encontrar os coeficientes que tornam o método de Runge-Kutta de  $s$  estágios da Equação A.6b idêntico à expansão em série de Taylor até o termo de derivada de ordem  $p$ .

$$Q^{m+1} = Q^m + \sum_{k=1}^{\infty} \frac{\Delta t^k}{k!} \frac{d^k Q}{dt^k} \quad (\text{A.6a})$$

$$\bar{Q}^{m+1} = \bar{Q}^m + \sum_{i=1}^s b_i k_i \quad (\text{A.6b})$$

Além das duas condições que garantam a convergência do método, são necessárias mais duas condições de existência e unicidade da solução a ser obtida (BURDEN; FAIRES, 2008):

- $f(t, Q)$  deve ser contínua;
- $f(t, Q)$  deve atender a condição de Lipschitz na variável  $Q$  para alguma constante  $L$ :

$$\|f(t, Q_1) - f(t, Q_2)\| \leq L|Q_1 - Q_2|, \quad L \in \mathbb{R} \quad (\text{A.7})$$

#### A.4 Métodos RK de primeira ordem

Para construir um método RK de primeira ordem, utilizam-se as seguintes expansões:

$$Q^{m+1} = Q^m + \Delta t \frac{dQ}{dt}(t^m) + O(\Delta t^2) \quad (\text{A.8a})$$

$$\bar{Q}^{m+1} = \bar{Q}^m + b_1 k_1 \quad (\text{A.8b})$$

A expansão em série de Taylor é realizada até o termo de ordem  $\Delta t$  pois deseja-se um método de primeira ordem. O método terá um estágio pois é, conforme a Tabela 5.1, o menor número de estágios necessários para se construir um método RK de primeira ordem. Substituindo as Equações A.8b e A.8a na Equação A.4, obtém-se:

$$\left\| Q^m + \Delta t \frac{dQ}{dt} + O(\Delta t^2) - \bar{Q}^m - b_1 k_1 \right\| = O(\Delta t^2). \quad (\text{A.9})$$

Por hipótese  $|Q^m - \bar{Q}^m| = O(\Delta t^2)$ , então:

$$\left\| \Delta t \frac{dQ}{dt} - b_1 k_1 + O(\Delta t^2) \right\| = O(\Delta t^2). \quad (\text{A.10})$$

Como  $\frac{dQ}{dt} = f(t^m, Q^m)$  e  $k_1 = \Delta t f(t^m, \bar{Q}^m)$ :

$$\left\| \Delta t f(t^m, Q^m) - b_1 \Delta t f(t^m, \bar{Q}^m) + O(\Delta t^2) \right\| = O(\Delta t^2) \quad (\text{A.11})$$

Por hipótese, a função  $f$  deve atender à condição de Lipschitz.

$$|f(t^m, Q^m) - f(t^m, \bar{Q}^m)| \leq L|Q^m - \bar{Q}^m| = LO(\Delta t^2) = O(\Delta t^2) \quad (\text{A.12})$$

Portanto  $f$  pode ser reescrita como:

$$f(t^m, \bar{Q}^m) = f(t^m, Q^m) + O(\Delta t^2). \quad (\text{A.13})$$

Substituindo  $f(t^m, \bar{Q}^m)$  na Equação A.11:

$$\left\| \Delta t f(t^m, Q^m) - b_1 \Delta t [f(t^m, Q^m) + O(\Delta t^2)] + O(\Delta t^2) \right\| = O(\Delta t^2). \quad (\text{A.14})$$

Distribuindo o termo  $b_1 \Delta t$ :

$$\left\| \Delta t f(t^m, Q^m) - b_1 \Delta t f(t^m, Q^m) + O(\Delta t^3) + O(\Delta t^2) \right\| = O(\Delta t^2). \quad (\text{A.15})$$

Reordenando os termos obtém-se:

$$\left\| \Delta t f(t^m, Q^m)(1 - b_1) + O(\Delta t^2) \right\| = O(\Delta t^2). \quad (\text{A.16})$$

Para que a igualdade da Equação A.16 seja válida, o termo  $\Delta t f(t^m, Q^m)(1 - b_1)$  deve ser nulo para todos os valores de  $f(t^m, Q^m)$ . Com isso obtém-se o valor do coeficiente  $b_1 = 1$ . Com isso o método de Runge-Kutta de 1ª ordem e um estágio pode ser escrito de forma única como:

$$\bar{Q}^{m+1} = \bar{Q}^m + \Delta t f(t^m, \bar{Q}^m). \quad (\text{A.17})$$

Este método é conhecido como método de Euler, sua representação em uma tabela de Butcher é dada na Tabela 5.3.

## A.5 Métodos RK de segunda ordem

Para construir um método RK de segunda ordem, utilizam-se as seguintes expansões:

$$Q^{m+1} = Q^m + \Delta t \frac{dQ}{dt}(t^m) + \frac{\Delta t^2}{2} \frac{d^2 Q}{dt^2}(t^m) + O(\Delta t^3) \quad (\text{A.18a})$$

$$\bar{Q}^{m+1} = \bar{Q}^m + b_1 k_1 + b_2 k_2 \quad (\text{A.18b})$$

Substituindo as Equações A.18b e A.18a em A.4, obtém-se:

$$\left\| Q^m + \Delta t \frac{dQ}{dt}(t^m) + \frac{\Delta t^2}{2} \frac{d^2 Q}{dt^2}(t^m) + O(\Delta t^3) - \bar{Q}^m - b_1 k_1 - b_2 k_2 \right\| = O(\Delta t^3). \quad (\text{A.19})$$

Como por hipótese  $|Q^m - \bar{Q}^m| = O(\Delta t^3)$ :

$$\left\| \Delta t \frac{dQ}{dt}(t^m) + \frac{\Delta t^2}{2} \frac{d^2Q}{dt^2}(t^m) - b_1 k_1 - b_2 k_2 + O(\Delta t^3) \right\| = O(\Delta t^3). \quad (\text{A.20})$$

Como  $\frac{dQ}{dt}(t^m) = f(t^m, Q^m)$ ,  $k_1 = \Delta t f(t^m, \bar{Q}^m)$  e utilizando as notações  $f := f(t^m, Q^m)$  e  $\bar{f} := f(t^m, \bar{Q}^m)$ :

$$\left\| \Delta t f + \frac{\Delta t^2}{2} \frac{df}{dt} - b_1 \Delta t \bar{f} - b_2 k_2 + O(\Delta t^3) \right\| = O(\Delta t^3). \quad (\text{A.21})$$

Seja o valor  $k_2$ , obtido conforme a Equação 5.2:

$$k_2 = \Delta t \bar{f} \left( t^m + c_2 \Delta t, \bar{Q}^m + a_{2,1} \Delta t \bar{f} \right) \quad (\text{A.22})$$

Expandindo o termo  $f(t^m + c_2 \Delta t, \bar{Q}^m + a_{2,1} \Delta t \bar{f})$  em séries de Taylor:

$$k_2 = \Delta t \left[ \bar{f} + c_2 \Delta t \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t \bar{f} \frac{\partial \bar{f}}{\partial Q} + O(\Delta t^2) \right] \quad (\text{A.23})$$

$$k_2 = \Delta t \bar{f} + c_2 \Delta t^2 \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t^2 \bar{f} \frac{\partial \bar{f}}{\partial Q} + O(\Delta t^3). \quad (\text{A.24})$$

Substituindo a Equação A.24 na Equação A.21:

$$\left\| \Delta t f + \frac{\Delta t^2}{2} \frac{df}{dt} - b_1 \Delta t \bar{f} - b_2 \left[ \Delta t \bar{f} + c_2 \Delta t^2 \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t^2 \bar{f} \frac{\partial \bar{f}}{\partial Q} + O(\Delta t^3) \right] + O(\Delta t^3) \right\| = O(\Delta t^3). \quad (\text{A.25})$$

Distribuindo o termo  $b_2$ , obtém-se:

$$\left\| \Delta t f + \frac{\Delta t^2}{2} \frac{df}{dt} - b_1 \Delta t \bar{f} - b_2 \Delta t \bar{f} - b_2 c_2 \Delta t^2 \frac{\partial \bar{f}}{\partial t} - b_2 a_{2,1} \Delta t^2 \bar{f} \frac{\partial \bar{f}}{\partial Q} + O(\Delta t^3) \right\| = O(\Delta t^3). \quad (\text{A.26})$$

De forma análoga ao que foi desenvolvido na Equação A.12, realiza-se a substituição:

$$f(t^m, \bar{Q}^m) = f(t^m, Q^m) + O(\Delta t^3). \quad (\text{A.27})$$

Expandindo o termo  $\frac{\partial \bar{f}}{\partial t}$  em série de Taylor em torno do ponto  $(t^m, Q^m)$ :

$$\frac{\partial \bar{f}}{\partial t} = \frac{\partial f}{\partial t} + |Q^m - \bar{Q}^m| \frac{\partial^2 f}{\partial t^2} + O(|Q^m - \bar{Q}^m|^2) \quad (\text{A.28})$$

Como por hipótese  $|Q^m - \bar{Q}^m| = O(\Delta t^3)$ :

$$\frac{\partial \bar{f}}{\partial t} = \frac{\partial f}{\partial t} + O(\Delta t^3) \quad (\text{A.29})$$

Realizando um procedimento análogo para  $\frac{\partial \bar{f}}{\partial Q}$ , obtém-se:

$$\frac{\partial \bar{f}}{\partial Q} = \frac{\partial f}{\partial Q} + O(\Delta t^3) \quad (\text{A.30})$$

Substituindo as Equações A.27, A.29 e A.30 na Equação A.26:

$$\begin{aligned} & \left\| \Delta t f + \frac{\Delta t^2}{2} \frac{df}{dt} - b_1 \Delta t [f + O(\Delta t^3)] - b_2 \Delta t [f + O(\Delta t^3)] \right. \\ & \left. - b_2 c_2 \Delta t^2 \left[ \frac{\partial f}{\partial t} + O(\Delta t^3) \right] - b_2 a_{2,1} \Delta t^2 [f + O(\Delta t^3)] \left[ \frac{\partial f}{\partial Q} + O(\Delta t^3) \right] \right. \\ & \left. + O(\Delta t^3) \right\| = O(\Delta t^3). \end{aligned} \quad (\text{A.31})$$

Reordenando os termos obtém-se:

$$\begin{aligned} & \left\| \Delta t f (1 - b_1 - b_2) + \Delta t^2 \left[ \frac{1}{2} \frac{df}{dt} - b_2 c_2 \frac{\partial f}{\partial t} - b_2 a_{2,1} f \frac{\partial f}{\partial Q} \right] \right. \\ & \left. + O(\Delta t^3) \right\| = O(\Delta t^3). \end{aligned} \quad (\text{A.32})$$

Aplicando a regra da cadeia no termo  $\frac{df}{dt}$ :

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial Q} \frac{dQ}{dt}. \quad (\text{A.33})$$

Como por definição  $\frac{dQ}{dt} = f$ :

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + f \frac{\partial f}{\partial Q}. \quad (\text{A.34})$$

Substituindo na Equação A.32:

$$\begin{aligned} & \left\| (1 - b_1 - b_2) \Delta t f + \Delta t^2 \left[ \frac{1}{2} \left( \frac{\partial f}{\partial t} + f \frac{\partial f}{\partial Q} \right) \right. \right. \\ & \left. \left. - b_2 c_2 \frac{\partial f}{\partial t} - b_2 a_{2,1} f \frac{\partial f}{\partial Q} \right] + O(\Delta t^3) \right\| = O(\Delta t^3). \end{aligned} \quad (\text{A.35})$$



Reordenando os termos obtém-se:

$$\left\| (1 - b_1 - b_2)\Delta t f + \Delta t^2 \left[ \left( \frac{1}{2} - b_2 c_2 \right) \frac{\partial f}{\partial t} + \left( \frac{1}{2} - b_2 a_{2,1} \right) f \frac{\partial f}{\partial Q} \right] + O(\Delta t^3) \right\| = O(\Delta t^3). \quad (\text{A.36})$$

A Equação A.36 só é válida se os coeficientes  $b_1$ ,  $b_2$ ,  $c_2$  e  $a_{2,1}$  satisfazem o sistema:

$$\begin{cases} b_1 + b_2 = 1 \\ b_2 c_2 = \frac{1}{2} \\ b_2 a_{2,1} = \frac{1}{2} \end{cases} \quad (\text{A.37})$$

O sistema de Equações A.37 tem várias soluções. Uma destas é:

$$b_1 = \frac{1}{2}, \quad b_2 = \frac{1}{2}, \quad c_2 = 1, \quad a_{2,1} = 1. \quad (\text{A.38})$$

Assim, um método de Runge-Kutta de 2ª ordem e dois estágios pode ser escrito como:

$$\bar{Q}^{m+1} = \bar{Q}^m + \frac{1}{2} f(t^m, \bar{Q}^m) + \frac{1}{2} f\left(t^m + \Delta t, \bar{Q}^m + f(t^m, \bar{Q}^m)\right) \quad (\text{A.39})$$

Este método é conhecido como método de Heun ou método de Euler aprimorado, sua representação em uma tabela de Butcher é dada na Tabela 5.4.

## A.6 Métodos RK de terceira ordem

Para construir um método RK de terceira ordem, utilizam-se as seguintes expansões:

$$Q^{m+1} = Q^m + \Delta t \frac{dQ}{dt}(t^m) + \frac{\Delta t^2}{2} \frac{d^2 Q}{dt^2}(t^m) + \frac{\Delta t^3}{6} \frac{d^3 Q}{dt^3}(t^m) + O(\Delta t^4) \quad (\text{A.40a})$$

$$\bar{Q}^{m+1} = \bar{Q}^m + b_1 k_1 + b_2 k_2 + b_3 k_3 \quad (\text{A.40b})$$

Substituindo as Equações A.40b e A.40a em A.4, obtém-se:

$$\left\| Q^m + \Delta t \frac{dQ}{dt}(t^m) + \frac{\Delta t^2}{2} \frac{d^2 Q}{dt^2}(t^m) + \frac{\Delta t^3}{6} \frac{d^3 Q}{dt^3}(t^m) + O(\Delta t^4) - \bar{Q}^m - b_1 k_1 - b_2 k_2 - b_3 k_3 \right\| = O(\Delta t^4) \quad (\text{A.41})$$

Como por hipótese  $|Q^m - \bar{Q}^m| = O(\Delta t^4)$ :

$$\left\| \Delta t \frac{dQ}{dt}(t^m) + \frac{\Delta t^2}{2} \frac{d^2 Q}{dt^2}(t^m) + \frac{\Delta t^3}{6} \frac{d^3 Q}{dt^3}(t^m) - b_1 k_1 - b_2 k_2 - b_3 k_3 + O(\Delta t^4) \right\| = O(\Delta t^4) \quad (\text{A.42})$$

Substituindo  $k_1 = f(t^m, \bar{Q}^m)$  e  $\frac{dQ}{dt} = f(t^m, Q^m)$ :

$$\left\| \Delta t f(t^m, Q^m) + \frac{\Delta t^2}{2} \frac{df}{dt}(t^m, Q^m) + \frac{\Delta t^3}{6} \frac{d^2 f}{dt^2}(t^m, Q^m) - b_1 \Delta t f(t^m, \bar{Q}^m) - b_2 k_2 - b_3 k_3 + O(\Delta t^4) \right\| = O(\Delta t^4) \quad (\text{A.43})$$

Para efeito de simplificação, será usada a notação  $f := f(t^m, Q^m)$  e  $\bar{f} := f(t^m, \bar{Q}^m)$ :

$$\left\| \Delta t f + \frac{\Delta t^2}{2} \frac{df}{dt} + \frac{\Delta t^3}{6} \frac{d^2 f}{dt^2} - b_1 \Delta t \bar{f} - b_2 k_2 - b_3 k_3 + O(\Delta t^4) \right\| = O(\Delta t^4) \quad (\text{A.44})$$

Decompondo o termo  $\frac{d^2 f}{dt^2}$ :

$$\frac{d^2 f}{dt^2} = \frac{d}{dt} \left( \frac{df}{dt} \right) \quad (\text{A.45})$$

Aplicando a regra da cadeia em  $\frac{df}{dt}$ , conforme realizado na Equação A.34:

$$\frac{d^2 f}{dt^2} = \frac{d}{dt} \left( \frac{\partial f}{\partial t} + \frac{\partial f}{\partial Q} \frac{dQ}{dt} \right) \quad (\text{A.46})$$

Aplicando a propriedade de linearidade do operador diferencial:

$$\frac{d^2 f}{dt^2} = \frac{d}{dt} \left( \frac{\partial f}{\partial t} \right) + \frac{d}{dt} \left( \frac{\partial f}{\partial Q} \frac{dQ}{dt} \right) \quad (\text{A.47})$$

Aplicando a regra da cadeia no primeiro termo e a propriedade de derivada do produto no segundo termo, obtém-se:

$$\frac{d^2 f}{dt^2} = \left( \frac{\partial^2 f}{\partial t^2} + \frac{\partial^2 f}{\partial t \partial Q} \frac{dQ}{dt} \right) + \left[ \frac{d}{dt} \left( \frac{\partial f}{\partial Q} \right) \frac{dQ}{dt} + \frac{\partial f}{\partial Q} \frac{d}{dt} \left( \frac{dQ}{dt} \right) \right] \quad (\text{A.48})$$

Por definição  $\frac{dQ}{dt} = f$ :

$$\frac{d^2 f}{dt^2} = \frac{\partial^2 f}{\partial t^2} + \frac{\partial^2 f}{\partial t \partial Q} f + \frac{d}{dt} \left( \frac{\partial f}{\partial Q} \right) f + \frac{\partial f}{\partial Q} \frac{df}{dt} \quad (\text{A.49})$$

Utilizando a regra da cadeia em  $\frac{d}{dt} \left( \frac{\partial f}{\partial Q} \right)$  e  $\frac{df}{dt}$ :

$$\frac{d^2 f}{dt^2} = \frac{\partial^2 f}{\partial t^2} + \frac{\partial^2 f}{\partial t \partial Q} f + \left( \frac{\partial^2 f}{\partial t \partial Q} + \frac{\partial^2 f}{\partial Q^2} \frac{dQ}{dt} \right) f + \frac{\partial f}{\partial Q} \left( \frac{\partial f}{\partial t} + \frac{\partial f}{\partial Q} \frac{dQ}{dt} \right) \quad (\text{A.50})$$

Substituindo  $\frac{dQ}{dt} = f$  e rearranjando os termos, obtém-se:

$$\frac{d^2 f}{dt^2} = \frac{\partial^2 f}{\partial t^2} + 2f \frac{\partial^2 f}{\partial t \partial Q} + \frac{\partial^2 f}{\partial Q^2} f^2 + \frac{\partial f}{\partial Q} \left( \frac{\partial f}{\partial t} + \frac{\partial f}{\partial Q} f \right) \quad (\text{A.51})$$

Substituindo as derivadas  $\frac{df}{dt}$  e  $\frac{d^2 f}{dt^2}$  da Equação A.44 pelos valores obtidos nas Equações A.34 e A.51:

$$\begin{aligned} & \left\| \Delta t f + \Delta t^2 \left( \frac{1}{2} \frac{\partial f}{\partial t} + \frac{1}{2} f \frac{\partial f}{\partial Q} \right) + \frac{\Delta t^3}{6} \left[ \frac{\partial^2 f}{\partial t^2} + 2f \frac{\partial^2 f}{\partial t \partial Q} + f^2 \frac{\partial^2 f}{\partial Q^2} \right. \right. \\ & \left. \left. + \frac{\partial f}{\partial Q} \left( \frac{\partial f}{\partial t} + f \frac{\partial f}{\partial Q} \right) \right] - b_1 \Delta t \bar{f} - b_2 k_2 - b_3 k_3 + O(\Delta t^4) \right\| = O(\Delta t^4) \end{aligned} \quad (\text{A.52})$$

Seja o valor  $k_3$ , obtido conforme a Equação 5.2:

$$k_3 = \Delta t f \left( t^m + c_3 \Delta t, \bar{Q}^m + a_{3,1} \Delta t \bar{f} + a_{3,2} k_2 \right) \quad (\text{A.53})$$

Expandindo o termo  $f \left( t^m + c_3 \Delta t, \bar{Q}^m + a_{3,1} \Delta t \bar{f} + a_{3,2} k_2 \right)$  em séries de Taylor:

$$\begin{aligned} k_3 = & \Delta t \left[ \bar{f} + c_3 \Delta t \frac{\partial \bar{f}}{\partial t} + \left( a_{3,1} \Delta t \bar{f} + a_{3,2} k_2 \right) \frac{\partial \bar{f}}{\partial Q} + \frac{c_3^2 \Delta t^2}{2} \frac{\partial^2 \bar{f}}{\partial t^2} \right. \\ & + c_3 \Delta t \left( a_{3,1} \Delta t \bar{f} + a_{3,2} k_2 \right) \frac{\partial^2 \bar{f}}{\partial t \partial Q} \\ & \left. + \frac{1}{2} \left( a_{3,1} \Delta t \bar{f} + a_{3,2} k_2 \right)^2 \frac{\partial^2 \bar{f}}{\partial Q^2} + O(\Delta t^3) \right] \end{aligned} \quad (\text{A.54})$$

Seja o valor  $k_2$ , obtido conforme a Equação 5.2:

$$k_2 = \Delta t f \left( t^m + c_2 \Delta t, \bar{Q}^m + a_{2,1} \Delta t \bar{f} \right) \quad (\text{A.55})$$

Expandindo  $f \left( t^m + c_2 \Delta t, \bar{Q}^m + a_{2,1} \Delta t \bar{f} \right)$  em série de Taylor:

$$k_2 = \Delta t \left[ \bar{f} + c_2 \Delta t \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t \bar{f} \frac{\partial \bar{f}}{\partial Q} + O(\Delta t^2) \right] \quad (\text{A.56})$$

Distribuindo o termo  $\Delta t$ :

$$k_2 = \left( \Delta t \bar{f} + c_2 \Delta t^2 \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t^2 \bar{f} \frac{\partial \bar{f}}{\partial Q} + O(\Delta t^3) \right) \quad (\text{A.57})$$

Substituindo a Equação A.57 na Equação A.54:

$$\begin{aligned} k_3 = & \Delta t \left\{ \bar{f} + c_3 \Delta t \frac{\partial \bar{f}}{\partial t} \right. \\ & + \left[ a_{3,1} \Delta t \bar{f} \right. \\ & + a_{3,2} \left( \Delta t \bar{f} + c_2 \Delta t^2 \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t^2 \bar{f} \frac{\partial \bar{f}}{\partial Q} + O(\Delta t^3) \right) \left. \right] \frac{\partial \bar{f}}{\partial Q} \\ & + \frac{c_3^2 \Delta t^2}{2} \frac{\partial^2 \bar{f}}{\partial t^2} \\ & + a_{3,1} c_3 \Delta t^2 \bar{f} \frac{\partial^2 \bar{f}}{\partial t \partial Q} \\ & + a_{3,2} c_3 \Delta t \left( \Delta t \bar{f} + c_2 \Delta t^2 \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t^2 \bar{f} \frac{\partial \bar{f}}{\partial Q} + O(\Delta t^3) \right) \frac{\partial^2 \bar{f}}{\partial t \partial Q} \\ & + \frac{1}{2} \left[ a_{3,1} \Delta t \bar{f} \right. \\ & + a_{3,2} \left( \Delta t \bar{f} + c_2 \Delta t^2 \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t^2 \bar{f} \frac{\partial \bar{f}}{\partial Q} + O(\Delta t^3) \right) \left. \right]^2 \frac{\partial^2 \bar{f}}{\partial Q^2} \\ & \left. + O(\Delta t^3) \right\} \end{aligned} \quad (\text{A.58})$$

Rearranjando os termos, obtém-se:

$$\begin{aligned} k_3 = & \Delta t \bar{f} \\ & + \Delta t^2 \left[ c_3 \frac{\partial \bar{f}}{\partial t} + (a_{3,1} + a_{3,2}) \bar{f} \frac{\partial \bar{f}}{\partial Q} \right] \\ & + \Delta t^3 \left[ + \frac{c_3^2}{2} \frac{\partial^2 \bar{f}}{\partial t^2} + \frac{a_{3,1}^2 \bar{f}^2}{2} + a_{3,1} a_{3,2} \bar{f}^2 + \frac{a_{3,2}^2 \bar{f}^2}{2} + a_{3,2} c_2 \frac{\partial \bar{f}}{\partial t} \frac{\partial \bar{f}}{\partial Q} \right. \\ & + a_{2,1} a_{3,2} \bar{f} \left( \frac{\partial \bar{f}}{\partial Q} \right)^2 + (a_{3,1} + a_{3,2}) c_3 \bar{f} \frac{\partial^2 \bar{f}}{\partial t \partial Q} \\ & \left. + a_{3,1} a_{3,2} c_2 \bar{f} \frac{\partial \bar{f}}{\partial t} + a_{2,1} a_{3,1} a_{3,2} \bar{f}^2 \frac{\partial \bar{f}}{\partial Q} + a_{3,2}^2 c_2 \bar{f} \frac{\partial \bar{f}}{\partial t} + a_{2,1} a_{3,2}^2 \bar{f}^2 \frac{\partial \bar{f}}{\partial Q} \right] \\ & + O(\Delta t^4) \end{aligned} \quad (\text{A.59})$$

Realizando uma análise análoga à realizada da Equação A.27 até a Equação A.31 e

reordenando os termos, obtém-se:

$$\begin{aligned}
k_3 = & \Delta t f \\
& + \Delta t^2 \left[ c_3 \frac{\partial f}{\partial t} + (a_{3,1} + a_{3,2}) f \frac{\partial f}{\partial Q} + \frac{c_3^2}{2} \frac{\partial^2 f}{\partial t^2} + \frac{a_{3,1}^2 f^2}{2} \right. \\
& \left. + a_{3,1} a_{3,2} f^2 + \frac{a_{3,2}^2 f^2}{2} \right] \\
& + \Delta t^3 \left[ a_{3,2} c_2 \frac{\partial f}{\partial t} \frac{\partial f}{\partial Q} + a_{2,1} a_{3,2} f \left( \frac{\partial f}{\partial Q} \right)^2 + (a_{3,1} + a_{3,2}) c_3 f \frac{\partial^2 f}{\partial t \partial Q} \right. \\
& \left. + a_{3,1} a_{3,2} c_2 f \frac{\partial f}{\partial t} + a_{2,1} a_{3,1} a_{3,2} f^2 \frac{\partial f}{\partial Q} + a_{3,2}^2 c_2 f \frac{\partial f}{\partial t} + a_{2,1} a_{3,2} f^2 \frac{\partial f}{\partial Q} \right] \\
& + O(\Delta t^4)
\end{aligned} \tag{A.60}$$

Seja o valor  $k_2$ , obtido conforme a Equação 5.2:

$$k_2 = \Delta t f \left( t^m + c_2 \Delta t, \bar{Q}^m + a_{2,1} \Delta t \bar{f} \right) \tag{A.61}$$

Expandindo  $f \left( t^m + c_2 \Delta t, \bar{Q}^m + a_{2,1} \Delta t \bar{f} \right)$  em série de Taylor, porém com um termo a mais do que a expansão da Equação A.56:

$$\begin{aligned}
k_2 = & \Delta t \left[ \bar{f} + c_2 \Delta t \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t \bar{f} \frac{\partial \bar{f}}{\partial Q} + \frac{c_2^2 \Delta t^2}{2} \frac{\partial^2 \bar{f}}{\partial t^2} + a_{2,1} c_2 \Delta t^2 \bar{f} \frac{\partial^2 \bar{f}}{\partial t \partial Q} \right. \\
& \left. + \frac{a_{2,1}^2 \Delta t^2 \bar{f}^2}{2} \frac{\partial^2 \bar{f}}{\partial Q^2} + O(\Delta t^3) \right]
\end{aligned} \tag{A.62}$$

Distribuindo o termo  $\Delta t$ :

$$\begin{aligned}
k_2 = & \Delta t \bar{f} + c_2 \Delta t^2 \frac{\partial \bar{f}}{\partial t} + a_{2,1} \Delta t^2 \bar{f} \frac{\partial \bar{f}}{\partial Q} + \frac{c_2^2 \Delta t^3}{2} \frac{\partial^2 \bar{f}}{\partial t^2} + a_{2,1} c_2 \Delta t^3 \bar{f} \frac{\partial^2 \bar{f}}{\partial t \partial Q} \\
& + \frac{a_{2,1}^2 \Delta t^3 \bar{f}^2}{2} \frac{\partial^2 \bar{f}}{\partial Q^2} + O(\Delta t^4)
\end{aligned} \tag{A.63}$$

Realizando uma análise análoga à realizada da Equação A.27 até a Equação A.31 e reordenando os termos, obtém-se:

$$\begin{aligned}
k_2 = & \Delta t f + c_2 \Delta t^2 \frac{\partial f}{\partial t} + a_{2,1} \Delta t^2 f \frac{\partial f}{\partial Q} + \frac{c_2^2 \Delta t^3}{2} \frac{\partial^2 f}{\partial t^2} + a_{2,1} c_2 \Delta t^3 f \frac{\partial^2 f}{\partial t \partial Q} \\
& + \frac{a_{2,1}^2 \Delta t^3 f^2}{2} \frac{\partial^2 f}{\partial Q^2} + O(\Delta t^4)
\end{aligned} \tag{A.64}$$

Substituindo as Equações A.60 e A.64 na Equação A.52:

$$\begin{aligned}
& \left\| \Delta t f + \Delta t^2 \left( \frac{1}{2} \frac{\partial f}{\partial t} + \frac{1}{2} f \frac{\partial f}{\partial Q} \right) + \frac{\Delta t^3}{6} \left[ \frac{\partial^2 f}{\partial t^2} + 2f \frac{\partial^2 f}{\partial t \partial Q} + f^2 \frac{\partial^2 f}{\partial Q^2} \right. \right. \\
& \left. \left. + \frac{\partial f}{\partial Q} \left( \frac{\partial f}{\partial t} + f \frac{\partial f}{\partial Q} \right) \right] - b_1 \Delta t \bar{f} \right. \\
& - b_2 \left[ \Delta t f + c_2 \Delta t^2 \frac{\partial f}{\partial t} + a_{2,1} \Delta t^2 f \frac{\partial f}{\partial Q} + \frac{c_2^2 \Delta t^3}{2} \frac{\partial^2 f}{\partial t^2} + a_{2,1} c_2 \Delta t^3 f \frac{\partial^2 f}{\partial t \partial Q} \right. \\
& \left. + \frac{a_{2,1}^2 \Delta t^3 f^2}{2} \frac{\partial^2 f}{\partial Q^2} + O(\Delta t^4) \right] \\
& - b_3 \left\{ \Delta t f \right. \\
& + \Delta t^2 \left[ c_3 \frac{\partial f}{\partial t} + (a_{3,1} + a_{3,2}) f \frac{\partial f}{\partial Q} + \frac{c_3^2}{2} \frac{\partial^2 f}{\partial t^2} + \frac{a_{3,1}^2 f^2}{2} + a_{3,1} a_{3,2} f^2 + \frac{a_{3,2}^2 f^2}{2} \right] \\
& + \Delta t^3 \left[ a_{3,2} c_2 \frac{\partial f}{\partial t} \frac{\partial f}{\partial Q} + a_{2,1} a_{3,2} f \left( \frac{\partial f}{\partial Q} \right)^2 + (a_{3,1} + a_{3,2}) c_3 f \frac{\partial^2 f}{\partial t \partial Q} \right. \\
& + a_{3,1} a_{3,2} c_2 f \frac{\partial f}{\partial t} + a_{2,1} a_{3,1} a_{3,2} f^2 \frac{\partial f}{\partial Q} + a_{3,2}^2 c_2 f \frac{\partial f}{\partial t} + a_{2,1} a_{3,2}^2 f^2 \frac{\partial f}{\partial Q} \left. \right] \\
& \left. + O(\Delta t^4) \right\} + O(\Delta t^4) \Big\| = O(\Delta t^4) \tag{A.65}
\end{aligned}$$

Reordenando os termos, obtém-se:

$$\begin{aligned}
& \left\| \Delta t \left[ f(1 - b_1 - b_2 - b_3) \right. \right. \\
& + \Delta t^2 \left[ \frac{\partial f}{\partial t} \left( \frac{1}{2} - b_2 c_2 - b_3 c_3 \right) \right. \\
& \left. + \frac{\partial f}{\partial Q} f \left( \frac{1}{2} - b_2 a_{2,1} - b_3 a_{3,1} - b_3 a_{3,2} \right) \right] \\
& + \Delta t^3 \left[ \frac{\partial^2 f}{\partial t^2} \left( \frac{1}{6} - \frac{b_2 c_2^2}{2} - \frac{b_3 c_3^2}{2} \right) \right. \\
& + f \frac{\partial^2 f}{\partial t \partial Q} \left( \frac{1}{3} - b_2 c_2 a_{2,1} - b_3 a_{3,1} c_3 - b_3 a_{3,2} c_3 \right) \\
& + \frac{\partial^2 f}{\partial Q^2} f^2 \left( \frac{1}{6} - \frac{b_2 a_{2,1}^2}{2} - \frac{b_3 a_{3,1}^2}{2} - b_3 a_{3,1} a_{3,2} - \frac{b_3 a_{3,2}^2}{2} \right) \\
& + \frac{\partial f}{\partial Q} \frac{\partial f}{\partial t} \left( \frac{1}{6} - b_3 a_{3,2} c_2 \right) \\
& \left. \left. + \left( \frac{\partial f}{\partial Q} \right)^2 f \left( \frac{1}{6} - b_3 a_{3,2} a_{2,1} \right) \right] + O(\Delta t^4) \right\| = O(\Delta t^4) \tag{A.66}
\end{aligned}$$

Para que a igualdade seja válida, seus coeficientes devem satisfazer o sistema:

$$\left\{ \begin{array}{l} b_1 + b_2 + b_3 = 1 \\ b_2 c_2 + b_3 c_3 = \frac{1}{2} \\ b_2 a_{2,1} + b_3 a_{3,1} + b_3 a_{3,2} = \frac{1}{2} \\ b_2 c_2^2 + b_3 c_3^2 = \frac{1}{3} \\ b_2 c_2 a_{2,1} + b_3 a_{3,1} c_3 + b_3 a_{3,2} c_3 = \frac{1}{3} \\ b_2 a_{2,1}^2 + b_3 a_{3,1}^2 + 2b_3 a_{3,1} a_{3,2} + b_3 a_{3,2}^2 = \frac{1}{3} \\ b_3 a_{3,2} c_2 = \frac{1}{6} \\ b_3 a_{3,2} a_{2,1} = \frac{1}{6} \end{array} \right. \quad (\text{A.67})$$

O sistema de Equações A.67 tem várias soluções. Uma destas soluções é:

$$\begin{aligned} b_1 &= \frac{1}{6}, & b_2 &= \frac{1}{6}, & b_3 &= \frac{2}{3}, \\ c_2 &= 1, & c_3 &= \frac{1}{2}, \\ a_{2,1} &= 1, & a_{3,1} &= \frac{1}{4}, & a_{3,2} &= \frac{1}{4}. \end{aligned} \quad (\text{A.68})$$

Assim, um método de Runge-Kutta de terceira ordem com três estágios pode ser escrito como:

$$\begin{aligned} \bar{Q}^{m+1} &= \bar{Q}^m + \frac{1}{6} f(t^m, \bar{Q}^m) \\ &+ \frac{1}{6} f\left(t^m + \Delta t, \bar{Q}^m + f(t^m, \bar{Q}^m)\right) \\ &+ \frac{2}{3} f\left[t^m + \frac{1}{2} \Delta t, \bar{Q}^m + \frac{1}{4} f(t^m, \bar{Q}^m) + \frac{1}{4} f\left(t^m + \Delta t, \bar{Q}^m + f(t^m, \bar{Q}^m)\right)\right] \end{aligned} \quad (\text{A.69})$$





## APÊNDICE B – COMPUTADORES UTILIZADOS

Tabela B.1 - Computadores utilizados para as simulações.

Nome	Tipo	Processador	Proprietário
Walkjrjas	Servidor	Quad-Core AMD Opteron(tm) Processor 8378	DGE/INPE
Jacaranda	Servidor	Intel(R) Xeon(R) CPU E5620 2.40GHz	LAC/INPE
Gladshain	Desktop	AMD Athlon(tm) 64 X2 Dual Core Processor 5000+ CPU 2,6GHz.	DGE/INPE
Chameleon	Netbook	Intel (R) Atom CPU 1,6GHz.	Autor



## PUBLICAÇÕES TÉCNICO-CIENTÍFICAS EDITADAS PELO INPE

### **Teses e Dissertações (TDI)**

Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

### **Manuais Técnicos (MAN)**

São publicações de caráter técnico que incluem normas, procedimentos, instruções e orientações.

### **Notas Técnico-Científicas (NTC)**

Incluem resultados preliminares de pesquisa, descrição de equipamentos, descrição e ou documentação de programas de computador, descrição de sistemas e experimentos, apresentação de testes, dados, atlas, e documentação de projetos de engenharia.

### **Relatórios de Pesquisa (RPQ)**

Reportam resultados ou progressos de pesquisas tanto de natureza técnica quanto científica, cujo nível seja compatível com o de uma publicação em periódico nacional ou internacional.

### **Propostas e Relatórios de Projetos (PRP)**

São propostas de projetos técnico-científicos e relatórios de acompanhamento de projetos, atividades e convênios.

### **Publicações Didáticas (PUD)**

Incluem apostilas, notas de aula e manuais didáticos.

### **Publicações Seriadas**

São os seriados técnico-científicos: boletins, periódicos, anuários e anais de eventos (simpósios e congressos). Contam destas publicações o Internacional Standard Serial Number (ISSN), que é um código único e definitivo para identificação de títulos de seriados.

### **Programas de Computador (PDC)**

São a seqüência de instruções ou códigos, expressos em uma linguagem de programação compilada ou interpretada, a ser executada por um computador para alcançar um determinado objetivo. Aceitam-se tanto programas fonte quanto os executáveis.

### **Pré-publicações (PRE)**

Todos os artigos publicados em periódicos, anais e como capítulos de livros.