



Ministério da  
**Ciência, Tecnologia  
e Inovação**



sid.inpe.br/mtc-m19/2012/08.10.15.02-TDI

## **ANÁLISE MULTIRRESOLUÇÃO ADAPTATIVA NO CONTEXTO DA RESOLUÇÃO NUMÉRICA DE UM MODELO DE MAGNETOHIDRODINÂMICA IDEAL**

Anna Karina Fontes Gomes

Dissertação de Mestrado do Curso  
de Pós-Graduação em Computação  
Aplicada, orientada pela Dra. Mar-  
garete Oliveira Domingues, apro-  
vada em 13 de setembro de 2012.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/3CE6FSE>>

INPE  
São José dos Campos  
2012

## **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:**

Ivone Martins - Serviço de Informação e Documentação (SID)



Ministério da  
**Ciência, Tecnologia  
e Inovação**



sid.inpe.br/mtc-m19/2012/08.10.15.02-TDI

## **ANÁLISE MULTIRRESOLUÇÃO ADAPTATIVA NO CONTEXTO DA RESOLUÇÃO NUMÉRICA DE UM MODELO DE MAGNETOHIDRODINÂMICA IDEAL**

Anna Karina Fontes Gomes

Dissertação de Mestrado do Curso  
de Pós-Graduação em Computação  
Aplicada, orientada pela Dra. Mar-  
garete Oliveira Domingues, apro-  
vada em 13 de setembro de 2012.

URL do documento original:

<<http://urlib.net/8JMKD3MGP7W/3CE6FSE>>

INPE  
São José dos Campos  
2012

Dados Internacionais de Catalogação na Publicação (CIP)

---

Gomes, Anna Karina Fontes.

G585a Análise multirresolução adaptativa no Contexto da resolução numérica de um Modelo de magnetohidrodinâmica ideal / Anna Karina Fontes Gomes. – São José dos Campos : INPE, 2012. xxviii+171 p. ; (sid.inpe.br/mtc-m19/2012/08.10.15.02-TDI)

Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2012.

Orientadora : Dra. Margarete Oliveira Domingues.

1. magnetohidrodinâmica. 2. análise multirresolução. 3. adaptativa. 4. volumes finitos I.Título.

CDU 519.6

---

Copyright © 2012 do MCT/INPE. Nenhuma parte desta publicação pode ser reproduzida, armazenada em um sistema de recuperação, ou transmitida sob qualquer forma ou por qualquer meio, eletrônico, mecânico, fotográfico, reprográfico, de microfilmagem ou outros, sem a permissão escrita do INPE, com exceção de qualquer material fornecido especificamente com o propósito de ser entrado e executado num sistema computacional, para o uso exclusivo do leitor da obra.

Copyright © 2012 by MCT/INPE. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, microfilming, or otherwise, without written permission from INPE, with the exception of any material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use of the reader of the work.

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

Dr. Elbert Einstein Nehrer Macau



---

Presidente / INPE / São José dos Campos - SP

Dra. Margarete Oliveira Domingues



---

Orientador(a) / INPE / SJC Campos - SP

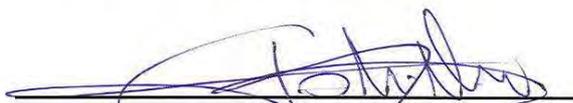
Dr. Odim Mendes Júnior



---

Membro da Banca / INPE / SJC Campos - SP

Dr. José Eduardo Castilho



---

Convidado(a) / UNB / Brasília - DF

Dr. Erico Luiz Rempel



---

Convidado(a) / ITA / SJC Campos - SP

*Este trabalho foi aprovado por:*

maioria simples

unanimidade

Aluno (a): **Anna Karina Fontes Gomes**

São José dos Campos, 13 de Setembro de 2012



*“Not where you stand, nor where you are.  
It matters only where you look.”*

STRATOVARIUS  
*“Learning to Fly”*, 2003



*Dedico este trabalho à minha família,  
aos meus bons amigos e  
ao heavy metal.*



## AGRADECIMENTOS

À Profa. Dra. Margarete Oliveira Domingues, meu muito obrigado pela orientação, amizade e paciência ao longo destes anos.

Aos Professores Dr. Odim Mendes Jr., José Eduardo Castilho, Érico Rempel e Elbert Macau, pelas discussões enriquecedoras, conselhos e apoio.

Ao engenheiro Varlei Menconi, pelo auxílio técnico e boa vontade sempre que foi preciso.

Ao Dr. Olivier Roussel e Bastien Di Pierro, pela disponibilização dos seus códigos.

Aos amigos Vitor Moura e Cledenilson, pela ajuda inicial neste trabalho, amizade e pelos bons momentos desta jornada.

À secretária Caroline Silva, por toda a sua boa vontade, carinho e paciência.

Aos Professores e amigos Dr. Augusto C. Barbosa, Dra. Cláudia Concordido e Dr. Carlos Frederico Vasconcelos, por sempre acreditarem em mim ao longo desses anos, pelo incentivo e amizade.

Aos meus pais Julio Gomes e Márcia Fontes por todo amor e apoio que me deram ao longo da minha vida.

Aos grandes amigos Luiz Gustavo Diniz, Sirléa A. Rosa, Amarísio Araújo, Fernando Pereira, Rodolfo Lotte, Giselle Lourenço e Patrícia Garcia, por fazerem parte dos melhores momentos desta jornada.

À bibliotecária do INPE, Yolanda Ribeiro Souza, pela revisão bibliográfica.

Ao CNPq pelo apoio financeiro (bolsa de mestrado processo 132045/2010-9 e projeto de pesquisa edital universal 483226/2011-4).

A todos que, direta ou indiretamente, contribuíram para a realização deste trabalho.



## RESUMO

A simulação de fenômenos do plasma espacial se dá, basicamente, por três tipos de modelagem: partículas, híbrida e magnetohidrodinâmica. Esses tipos de modelagem diferem basicamente quanto à simplificação do problema e à abrangência da área de estudo. Neste trabalho, estuda-se o modelo magnetohidrodinâmico (MHD) ideal conservativo com multiplicadores de Lagrange generalizados estendidos (EGLM). Esse modelo considera o plasma como um fluido não-colisional e utiliza correções de divergência livre para que a equação de fluxo magnético seja bem resolvida numericamente. Em várias aplicações desse tipo de modelo, como, por exemplo, nos estudos das ciências espaciais, ocorrem fenômenos como descontinuidades e choques localizados na solução. Para que esses tipos de comportamentos sejam bem modelados, é necessário um refinamento maior da malha de simulação numérica nos locais onde ocorrem. Nos métodos tradicionais, esse refinamento é realizado globalmente na malha, *i.e.*, refina-se toda a malha, mesmo onde não há necessidade de refinamento. Por outro lado, nos métodos adaptativos, utiliza-se uma malha que se adapta automaticamente à solução que se deseja obter, em cada passo de tempo. Ao utilizar a técnica de análise multirresolução adaptativa, basicamente, os coeficientes wavelet são utilizados como indicadores de regularidade local da solução numérica e definem como se dará os refinamentos hierárquicos da malha. Ao mesmo tempo, ao utilizar essa técnica é possível obter, em qualquer momento, a solução do modelo no nível mais refinado da malha, utilizando os algoritmos de reconstrução no contexto da análise de multirresolução. Neste trabalho, é utilizada a abordagem de multirresolução adaptativa, combinada ao método dos volumes finitos, para simular numericamente o modelo MHD com multiplicador de Lagrange generalizado estendido, a fim de verificar essa nova abordagem nesse contexto físico.



# ADAPTIVE MULTIREOLUTION ANALYSIS IN THE CONTEXT OF NUMERICAL RESOLUTION OF IDEAL MAGNETOHYDRODYNAMICS MODEL

## ABSTRACT

The space plasma simulation is basically related to three models: particles-in-cell, hybrid and magnetohydrodynamics. The main difference between these models is the scale of the physical phenomena. In this work, we study the conservative ideal magnetohydrodynamics model (MHD) using the extended generalized Lagrange multiplier (EGLM). In this model, the plasma is considered a non-collisional fluid and it has divergence-free corrections for the magnetic flux equation, because we need this equation to be correctly-solved numerically. In several applications of this model, e.g., the space sciences studies, there are different types of phenomena, such as discontinuities and shocks. Thus, it is even more important to refine the simulation mesh to well-model those types of behaviour locally. On the traditional methods of refining, the mesh refinement is done globally, *i.e.*, it refines the entire mesh, even where it does not need to be refined. On the other hand, in adaptive methods it uses a mesh that adapts itself to the solution we want to obtain, in every time step. Using the adaptive multiresolution analysis technique, the wavelet coefficients are basically used as indicators of local regularity of the numerical solution and they define how the hierarchical refinement will be done. Moreover, when using this technique, it is possible to obtain, at any moment, the solution in a more refined level by using the multiresolution algorithms. In this work, we use the multiresolution approach, with the finite volume method, to represent the solution of the extended generalized Lagrangian multiplier MHD model and we want to verify this new approach in this physical context.



## LISTA DE FIGURAS

	<u>Pág.</u>
3.1 Leque de Riemann . . . . .	14
3.2 Condição inicial tipo Riemann . . . . .	17
3.3 Solução exata de um modelo MHD ideal . . . . .	18
3.4 Solução exata de um modelo MHD ideal . . . . .	19
4.1 Células em uma malha unidimensional . . . . .	22
4.2 Médias celulares e o fluxo intercelular. . . . .	23
4.3 Esquema das células em uma malha . . . . .	26
5.1 Estrutura da solução de um problema de Riemann . . . . .	30
5.2 Ondas elementares como solução de um problema de Riemann . . . . .	31
5.3 Leque de Riemann para o fluxo HLL. . . . .	33
5.4 Leque de Riemann para o fluxo HLLD. . . . .	35
5.5 Comparação de uma solução numérica para diferentes fluxos . . . . .	41
6.1 Algoritmo MHD-FV . . . . .	43
6.2 Condição inicial tipo Riemann 1D: esquema das variações das variáveis. . . . .	45
6.3 Resultados de Miyoshi e Kusano (2005) para o Caso 1 . . . . .	46
6.4 Variáveis $\rho$ e $v_y$ estimadas com o fluxo HLL . . . . .	47
6.5 Variáveis $\rho$ e $v_y$ estimadas com o fluxo HLLD . . . . .	48
6.6 Variáveis $\rho$ e $v_y$ estimadas com o fluxo HLL e HLLD . . . . .	49
6.7 Variáveis $\rho$ e $p$ : comparação entre a solução exata e aproximada. . . . .	50
6.8 Variáveis $v_x$ , $v_y$ e $v_z$ : comparação entre a solução exata e aproximada. . . . .	51
6.9 Variáveis $B_y$ e $B_z$ : comparação entre a solução exata e aproximada. . . . .	52
6.10 Domínio para uma condição 2D . . . . .	54
6.11 Solução numérica do problema tipo Riemann 2D, como em Dedner et al (2002) . . . . .	55
6.12 Variáveis $\rho$ e $p$ para os fluxos HLL e HLLD . . . . .	56
6.13 Variáveis $v_x$ , $v_y$ e $v_z$ para os fluxos HLL e HLLD . . . . .	57
6.14 Variáveis $B_x$ , $B_y$ e $B_z$ para os fluxos HLL e HLLD . . . . .	58
7.1 Hierarquia de partições de $\Omega$ . . . . .	64
7.2 Ilustração dos operadores de projeção e predição . . . . .	65
7.3 Operação de truncamento . . . . .	71
7.4 Hierarquia de malhas aninhadas . . . . .	72

7.5	Exemplo de uma malha adaptativa obtida com o truncamento dos coeficientes wavelet de um campo de densidade . . . . .	73
9.1	Algoritmo CARMEN . . . . .	79
9.2	Variáveis $\rho$ e $p$ aproximadas com o fluxo HLL para o Caso 1. . . . .	81
9.3	Variáveis $v_x$ , $v_y$ e $v_z$ aproximadas com o fluxo HLL para o Caso 1. . . . .	83
9.4	Variáveis $B_y$ e $B_z$ aproximadas com o fluxo HLL para o Caso 1. . . . .	84
9.5	Variáveis $\rho$ e $p$ aproximadas com o fluxo HLLD para o Caso 1. . . . .	85
9.6	Variáveis $v_x$ , $v_y$ e $v_z$ aproximadas com o fluxo HLLD para o Caso 1. . . . .	86
9.7	Variáveis $B_y$ e $B_z$ aproximadas com o fluxo HLLD para o Caso 1. . . . .	87
9.8	Malhas adaptativas referentes ao Caso 1 - HLL . . . . .	88
9.9	Malhas adaptativas referentes ao Caso 1 - HLLD . . . . .	89
9.10	Variável $\rho$ aproximada com o fluxo HLL para o Caso 2. . . . .	94
9.11	Variável $p$ aproximada com o fluxo HLL para o Caso 2. . . . .	95
9.12	Variável $v_x$ aproximada com o fluxo HLL para o Caso 2. . . . .	96
9.13	Variável $v_y$ aproximada com o fluxo HLL para o Caso 2. . . . .	97
9.14	Variável $v_z$ aproximada com o fluxo HLL para o Caso 2. . . . .	98
9.15	Variável $B_x$ aproximada com o fluxo HLL para o Caso 2. . . . .	99
9.16	Variável $B_y$ aproximada com o fluxo HLL para o Caso 2. . . . .	100
9.17	Variável $B_z$ aproximada com o fluxo HLL para o Caso 2. . . . .	101
9.18	Variável $\rho$ aproximada com o fluxo HLLD para o Caso 2. . . . .	102
9.19	Variável $p$ aproximada com o fluxo HLLD para o Caso 2. . . . .	103
9.20	Variável $v_x$ aproximada com o fluxo HLLD para o Caso 2. . . . .	104
9.21	Variável $v_y$ aproximada com o fluxo HLLD para o Caso 2. . . . .	105
9.22	Variável $v_z$ aproximada com o fluxo HLLD para o Caso 2. . . . .	106
9.23	Variável $B_x$ aproximada com o fluxo HLLD para o Caso 2. . . . .	107
9.24	Variável $B_y$ aproximada com o fluxo HLLD para o Caso 2. . . . .	108
9.25	Variável $B_z$ aproximada com o fluxo HLLD para o Caso 2. . . . .	109
9.26	Malhas adaptativas referentes ao Caso 2 - HLL . . . . .	110
9.27	Malhas adaptativas referentes ao Caso 2 - HLLD . . . . .	111
F.1	Variáveis $\rho$ e $p$ para os fluxos HLL e HLLD . . . . .	142
F.2	Variáveis $v_x$ , $v_y$ e $v_z$ para os fluxos HLL e HLLD . . . . .	143
F.3	Variáveis $B_x$ , $B_y$ e $B_z$ para os fluxos HLL e HLLD . . . . .	144
F.4	Variável $\rho$ aproximada com o fluxo HLL para o Caso 2. . . . .	145
F.5	Variável $p$ aproximada com o fluxo HLL para o Caso 2. . . . .	146
F.6	Variável $v_x$ aproximada com o fluxo HLL para o Caso 2. . . . .	147
F.7	Variável $v_y$ aproximada com o fluxo HLL para o Caso 2. . . . .	148
F.8	Variável $v_z$ aproximada com o fluxo HLL para o Caso 2. . . . .	149

F.9	Variável $B_x$ aproximada com o fluxo HLL para o Caso 2. . . . .	150
F.10	Variável $B_y$ aproximada com o fluxo HLL para o Caso 2. . . . .	151
F.11	Variável $B_z$ aproximada com o fluxo HLL para o Caso 2. . . . .	152
F.12	Variável $\rho$ aproximada com o fluxo HLLD para o Caso 2. . . . .	153
F.13	Variável $p$ aproximada com o fluxo HLLD para o Caso 2. . . . .	154
F.14	Variável $v_x$ aproximada com o fluxo HLLD para o Caso 2. . . . .	155
F.15	Variável $v_y$ aproximada com o fluxo HLLD para o Caso 2. . . . .	156
F.16	Variável $v_z$ aproximada com o fluxo HLLD para o Caso 2. . . . .	157
F.17	Variável $B_x$ aproximada com o fluxo HLLD para o Caso 2. . . . .	158
F.18	Variável $B_y$ aproximada com o fluxo HLLD para o Caso 2. . . . .	159
F.19	Variável $B_z$ aproximada com o fluxo HLLD para o Caso 2. . . . .	160
G.1	Malha diádica em um intervalo . . . . .	161
G.2	Predição utilizando a interpolação linear . . . . .	163
G.3	Coefficiente wavelet como sendo o erro de predição no caso linear . . . . .	164
G.4	Estrutura do algoritmo de análise . . . . .	164
G.5	Estrutura do algoritmo de síntese . . . . .	165
G.6	Decomposição bi-nível para análise multirresolução interpoladora . . . . .	168
G.7	A função $f(x)$ e a posição dos coeficientes wavelet significantes . . . . .	169
G.8	Malhas diádicas uniformes . . . . .	171



## LISTA DE TABELAS

	<u>Pág.</u>
3.1 Condição inicial tipo Riemann. . . . .	17
6.1 Erro na norma $L^1$ calculada para o Caso 1 . . . . .	53
6.2 Condição inicial tipo Riemann 1D - Direção $y$ . . . . .	53
6.3 Condição inicial tipo Riemann 2D . . . . .	54
6.4 Erro na norma $L^1$ calculada para o Caso 2 . . . . .	59
9.1 Tempos de CPU, compressão de CPU e memória para o Caso 1 MR . . .	90
9.2 Erro na norma $L^1$ calculada para o Caso 1 MR . . . . .	92
9.3 Tempos de CPU, compressão de CPU e memória para o Caso 2 MR . . .	112
9.4 Erro na norma $L^1$ calculada para o Caso 2 MR . . . . .	113



## LISTA DE ABREVIATURAS E SIGLAS

1D	–	Uma Dimensão
2D	–	Duas Dimensões
CARMEN	–	Código com algoritmo de multirresolução adaptativa para EDP evolutivas
CFL	–	Courant-Friedrich-Lewy
DW	–	Dai e Woodward
EDO	–	Equações Diferenciais Ordinárias
EDP	–	Equações Diferenciais Parciais
EGLM	–	Multiplicador de Lagrange Generalizado Estendido
FV	–	Volumes Finitos, do inglês <i>finite volumes</i>
GLM	–	Multiplicador de Lagrange generalizado
HLL	–	Harten-Lax-Van Leer
HLLD	–	Harten-Lax-Van Leer-Descontinuidades
HLLEM	–	Harten-Lax-Van Leer-Einfeldt-Modificado
L	–	esquerda, do inglês <i>left</i>
MHD	–	Magnetohidrodinâmico
MHD-FV	–	Código que calcula solução do modelo EGLM-MHD em volumes finitos
MR	–	Multirresolução
R	–	direita, do inglês <i>right</i>



## LISTA DE SÍMBOLOS

$t$	– tempo físico
$\mathbf{U}$	– vetor das variáveis do modelo EGLM-MHD
$\mathbf{F}(\mathbf{U})$	– vetor fluxo do modelo EGLM-MHD
$\mathbf{S}(\mathbf{U})$	– vetor de termos fonte do modelo EGLM-MHD
$\rho$	– densidade de um fluido
$p$	– pressão de um fluido
$p_T$	– pressão total
$E$	– energia do fluido
$\psi$	– função potencial
$v_x$	– componente $x$ da velocidade
$v_y$	– componente $y$ da velocidade
$v_z$	– componente $z$ da velocidade
$B_x$	– componente $x$ do campo magnético
$B_y$	– componente $y$ do campo magnético
$B_z$	– componente $z$ do campo magnético
$\mathbf{u}$	– vetor velocidade
$\mathbf{B}$	– vetor campo magnético
$\nabla \cdot \mathbf{B}$	– divergência do campos magnético
$\mathbf{I}$	– tensor de ordem 2 unitário
$\mathbf{J}$	– vetor densidade corrente
$D$	– operador diferencial linear
$c_h$	– coeficiente da correção hiperbólica
$c_p$	– coeficiente da correção parabólica
$C_i$	– $i$ -ésima célula da malha
$Q_i^n$	– média celular da $i$ -ésima célula da malha
$\Delta t$	– passo de tempo
$\Delta x$	– passo na direção $x$
$\Delta y$	– passo na direção $y$
$c_{CFL}$	– constante CFL
$\gamma$	– constante adiabática
$S_i$	– velocidade de propagação
$\lambda_i$	– autovalores do modelo EGLM-MHD
$c_f$	– velocidade magneto-acústica rápida
$c_a$	– velocidade de Alfvén
$c_s$	– velocidade magneto-acústica lenta
$f_{j,k}$	– amostras da função $f$ em um nível $j$ e posição $k$
$\mathbf{f}_j$	– conjunto de amostras discretas $f_{j,k}$ em um nível $j$ e posição $k$
$d_{j,k}$	– erros (ou detalhes) de aproximação em um nível $j$ e posição $k$

$\mathbf{d}_j$	– conjunto de coeficientes wavelet $d_{j,k}$ em um nível $j$ e posição $k$
$V_j$	– espaço gerado pelas funções escala em um nível $j$
$W_j$	– espaço gerado pelas funções wavelet em um nível $j$
$\Phi_{j,k}$	– funções escala em um nível $j$ e posição $k$
$\Psi_{j,k}$	– funções wavelet em um nível $j$ e posição $k$
$X_j$	– hierarquia de um intervalo unitário de malhas diáticas em um nível $j$
$\mathcal{D}_j$	– operador de discretização em um nível $j$
$P_{j+1 \rightarrow j}$	– operador de projeção do nível $j + 1$ para $j$
$\epsilon$	– limiar de truncamento
$\mathbf{R}$	– operador de refinamento
$\mathbf{E}_{\text{MR}}$	– operador de evolução em multirresolução adaptativa
$\mathcal{T}_\epsilon$	– operador de truncamento da malha adaptativa
$\Gamma_n$	– malha esparsa no tempo $t_n$

## SUMÁRIO

	<u>Pág.</u>
<b>1 INTRODUÇÃO</b> . . . . .	<b>1</b>
<b>2 MODELO MAGNETOHIDRODINÂMICO</b> . . . . .	<b>5</b>
2.1 Modelo Magnetohidrodinâmico Ideal . . . . .	5
2.2 Correções Numéricas de Divergência Nula . . . . .	6
2.2.1 Correção Parabólica-Hiperbólica . . . . .	8
2.2.2 Correção Elíptica . . . . .	8
2.2.3 Correção Parabólica . . . . .	9
2.2.4 Correção Hiperbólica . . . . .	9
2.3 Multiplicador de Lagrange Generalizado . . . . .	9
2.4 Extensão do Multiplicador de Lagrange Generalizado . . . . .	9
<b>3 LEIS DE CONSERVAÇÃO</b> . . . . .	<b>11</b>
3.1 Modelo EGLM-MHD . . . . .	11
3.2 Forma Primitiva . . . . .	12
3.2.1 Autovalores . . . . .	13
3.3 Solução Exata e Verificação . . . . .	16
3.3.1 Condição Inicial tipo Riemann . . . . .	16
3.3.2 Solução Exata . . . . .	16
<b>4 MÉTODO DOS VOLUMES FINITOS (VF)</b> . . . . .	<b>21</b>
4.1 Formulação Geral do Método . . . . .	21
4.2 A Condição Courant-Friedrichs-Lewy . . . . .	25
4.3 Fluxos . . . . .	25
4.4 Evolução Temporal . . . . .	26
<b>5 FLUXOS NUMÉRICOS</b> . . . . .	<b>29</b>
5.1 Ondas Elementares como Solução de um Problema de Riemann . . . . .	29
5.1.1 Onda de choque . . . . .	30
5.1.2 Onda de contato . . . . .	30
5.1.3 Onda de rarefação . . . . .	31
5.2 O Problema de Riemann e o Fluxo Godunov . . . . .	32

5.2.1	Fluxo Harten-Lax-Van Leer . . . . .	32
5.2.1.1	Algoritmo HLL . . . . .	34
5.2.2	Fluxo Harten-Lax-Van Leer-Descontinuidades . . . . .	35
5.2.2.1	Algoritmo HLLD . . . . .	40
<b>6</b>	<b>FV: VERIFICAÇÃO E RESULTADOS . . . . .</b>	<b>43</b>
6.1	Caso 1: Condição Inicial tipo Riemann 1D . . . . .	44
6.1.1	Verificação . . . . .	44
6.1.2	Resultados . . . . .	48
6.2	Caso 2: Condição Inicial Tipo Riemann 2D . . . . .	54
<b>7</b>	<b>MULTIRRESOLUÇÃO (MR) . . . . .</b>	<b>61</b>
7.1	MR para médias celulares . . . . .	63
7.1.1	Predição de médias celulares . . . . .	64
7.1.2	Transformada MR . . . . .	66
7.1.2.1	Análise . . . . .	66
7.1.2.2	Síntese . . . . .	66
7.1.3	Contexto funcional . . . . .	67
7.1.4	Indicador wavelet de regularidade local . . . . .	69
7.1.5	Compressão de dados . . . . .	70
7.1.5.1	Estabilidade . . . . .	70
7.1.6	Malhas cartesianas em dimensões superiores . . . . .	72
<b>8</b>	<b>VF e MR: COMBINANDO OS MÉTODOS . . . . .</b>	<b>75</b>
<b>9</b>	<b>MR ADAPTATIVA: VERIFICAÇÃO E RESULTADOS . . . . .</b>	<b>79</b>
9.1	Condição Inicial tipo Riemann 1D . . . . .	80
9.2	Condição Inicial tipo Riemann 2D . . . . .	93
9.2.1	Resultados: Fluxo HLL . . . . .	94
9.2.2	Resultados: Fluxo HLLD . . . . .	102
<b>10</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS . . . . .</b>	<b>115</b>

<b>REFERÊNCIAS BIBLIOGRÁFICAS</b> . . . . .	<b>117</b>
<b>APÊNDICE A - TERMOS DO SISTEMA EGLM-MHD</b> . . . . .	<b>.121</b>
<b>APÊNDICE B - EGLM-MHD: AUTOVALORES E AUTOVE-</b> <b>TORES</b> . . . . .	<b>.123</b>
B.1 Cálculo simbólico . . . . .	123
<b>APÊNDICE C - CONDIÇÃO DE CONTORNO</b> . . . . .	<b>.127</b>
C.1 Condição de Contorno . . . . .	127
<b>APÊNDICE D - CARMEN: EVOLUÇÃO TEMPORAL.</b> . . . . .	<b>.129</b>
<b>APÊNDICE E - IMPLEMENTAÇÕES</b> . . . . .	<b>.131</b>
E.1 Código MHD-FV . . . . .	131
E.2 Código CARMEN . . . . .	134
E.2.1 Fluxo HLL . . . . .	136
E.2.2 Fluxo HLLD . . . . .	137
E.2.3 Evolução temporal . . . . .	138
<b>APÊNDICE F - SOLUÇÃO DE REFERÊNCIA PARA O CASO 2</b> .141	
F.1 Solução de Referência - MHD-FV . . . . .	142
F.2 Solução de Referência - CARMEN . . . . .	145
F.2.1 Resultados: Fluxo HLL . . . . .	145
F.2.2 Resultados: Fluxo HLLD . . . . .	153
<b>APÊNDICE G - MR PARA VALORES PONTUAIS</b> . . . . .	<b>.161</b>
G.0.3 Interpolando Predições . . . . .	162
G.1 Transformada MR . . . . .	163
G.1.1 Análise . . . . .	164
G.1.2 Síntese . . . . .	165
G.2 Indicador wavelet de regularidade local . . . . .	165
G.3 Contexto funcional . . . . .	166
G.4 Compressão de dados . . . . .	168
G.5 Estabilidade . . . . .	170
G.5.1 Análise MR para valores pontuais em dimensões superiores . . . . .	170



# 1 INTRODUÇÃO

No contexto da resolução numérica de modelos físicos, é comum a utilização de uma abordagem adaptativa. Ao ser utilizada em um problema físico, esse tipo de abordagem pode diminuir o custo computacional de uma simulação, pois esse tipo de método evita cálculos e o uso de memória desnecessários. A modelagem adaptativa tem sido bastante utilizada na áreas de pesquisas relacionadas às Ciências Espaciais, em diferentes abordagens. A adaptatividade auxilia na resolução de problemas em escala local e, ao mesmo tempo, interage com escalas globais de forma consistente. Neste trabalho, opta-se pela utilização da abordagem adaptativa no contexto da análise multirresolução, aplicando-a a um modelo da magnetohidrodinâmica (MHD). Esse tipo de modelo é utilizado na modelagem numérico-computacional de fenômenos relacionados à física do plasma espacial.

No modelo MHD, o plasma é descrito por um sistema de quatro equações diferenciais parciais: três de fluido e uma de campo magnético. Essas equações, quando estão em sua forma conservativa, descrevem a conservação de massa, momento, energia e a evolução do campo magnético. O modelo MHD é uma extensão da dinâmica de fluidos, a qual descreve o comportamento de fluidos eletricamente condutores. No caso do modelo MHD, tem-se a inclusão dos efeitos da força eletromagnética. Esse sistema de equações descreve a evolução de quantidades macroscópicas e, por isso, é utilizado quando não há interesse no movimento individual de partículas. Para outros propósitos, existem outros tipos de modelos que descrevem o comportamento do plasma. Uma revisão desse tema encontra-se em Ledvina et al (2008).

O objeto de estudo deste trabalho é um modelo MHD Ideal em sua forma conservativa. Para encontrar a solução desse modelo é necessário conhecer o fluxo em questão. O fluxo depende da solução do modelo, a qual ainda não é conhecida. A fim de aproximá-lo, foram desenvolvidos diferentes métodos numéricos, os quais são chamadas de *fluxos numéricos*. Neste trabalho são testados dois diferentes tipos de fluxos numéricos: o Harten-Lax-Van Leer (HLL) e o Harten-Lax-Van Leer-Descontinuidades (HLLD) (HARTEN et al., 1983; MIYOSHI; KUSANO, 2005). Fisicamente, existe uma restrição relacionada ao campo magnético, a qual age no comportamento do plasma. Essa restrição garante que o campo magnético seja livre de divergência, *i.e.*, livre comportamentos como fontes e sorvedouros. No entanto, no contexto numérico-computacional, essa restrição pode não ser obedecida, necessitando, assim, de um ajuste para que ela seja satisfeita. Esse tipo de ajuste é chamado

correção de divergência livre e, neste trabalho, é utilizada uma correção chamada parabólica-hiperbólica (DEDNER et al., 2002).

O modelo MHD ideal, juntamente com a correção de divergência livre parabólica-hiperbólica, tem uma discretização temporal explícita. Esse tipo de discretização permite o uso do algoritmo de multirresolução adaptativa por médias celulares, juntamente com o método de volumes finitos. No algoritmo adaptativo é definido o nível de refinamento local que deve ser utilizado para representar a solução do modelo. Diferentemente do *mesh refinement*, o algoritmo adaptativo de multirresolução utiliza estruturas mais localizadas, a fim de reproduzir com mais precisão o comportamento da solução, mais detalhes em Deiterding et al (2009).

Para este trabalho, implementa-se um código para simular o modelo MHD com multiplicador de Lagrange generalizado estendido (EGLM), utilizando o método dos volumes finitos, chamando MHD-FV. Neste mesmo código, também são implementados os fluxos HLL e HLLD. Esse mesmo modelo é implementado no código adaptativo CARMEN, também com os fluxos HLL e HLLD. No Apêndice E, são apresentados detalhes dessas implementações.

### **Contribuições Originais da Dissertação**

A principal contribuição desse trabalho é testar se o modelo EGLM-MHD, discretizado por um esquema de volumes finitos, pode ser representado eficientemente por uma forma adaptativa no contexto wavelet de multirresolução.

### **Desafios da Dissertação**

Os principais desafios desse trabalho são a implementação e verificação das equações MHD, juntamente com a correção de divergência livre parabólica-hiperbólica em um modelo de volumes finitos para referência, com os dois fluxos numéricos HLL e HLLD, num código adaptativo. Nesse contexto, a abordagem adaptativa é focada no problema bidimensional do modelo EGLM-MHD.

### **Organização dos Capítulos**

No Capítulo 2, é descrito o modelo MHD utilizado nesse trabalho. Já no Capítulo 3, introduzem-se os conceitos de leis de conservação hiperbólicas e a solução exata de um modelo MHD. No Capítulo 4 o método dos volumes finitos é apresentado e ex-

plicitado. Os fluxos numéricos são apresentados no Capítulo 5 e os resultados para a abordagem de volumes finitos são apresentados no Capítulo 6. No Capítulo 7 é apresentada a análise multiresolução adaptativa por médias celulares e, no Capítulo 8, a combinação desse algoritmo com o método dos volumes finitos. No Capítulo 9 são apresentados os resultados obtidos utilizando o algoritmo adaptativo de multirresolução. No Capítulo 10, as conclusões e trabalhos futuros.



## 2 MODELO MAGNETOHIDRODINÂMICO

Existem diferentes tipos de modelo MHD, os quais foram utilizados para descrever diferentes tipos de fenômenos que envolvem o plasma. Um desses modelos é o MHD Ideal, quando um fluido não é resistivo. Também existem modelos mais sofisticados como, por exemplo, o MHD Multi-espécie, quando cada espécie possui sua própria equação de continuidade, e MHD Multi-fluido, quando cada espécie é tratada como um fluido diferente. Maiores detalhes sobre os tipos de modelo MHD são encontrados no paper de revisão de Ledvina et al (2008).

### 2.1 Modelo Magnetohidrodinâmico Ideal

As equações que governam o modelo MHD Ideal formam um conjunto de equações diferenciais parciais, as quais determinam inteiramente a física de um fluido condutor. Em particular, o modelo é apenas aplicado quando são assumidas as seguintes condições:

1. os componentes do plasma não estão distantes do equilíbrio termodinâmico local;
2. o plasma possui uma função de distribuição Maxwelliana;
3. o fluxo de calor não é importante;
4. a condição de neutralidade da carga é válida;
5. a componente de alta frequência do campo elétrico pode ser negligenciada.

Há duas formas de escrever as variáveis em um modelo MHD: a forma primitiva, que utiliza diretamente as variáveis originais ou primitivas como pressão e velocidade, e a forma conservativa, que utiliza variáveis conservativas, como energia e momento. A abordagem primitiva não é adequada para simulações quando há descontinuidades nas variáveis, podendo gerar oscilações ou mesmo alterar a física do problema em estudo. Na abordagem conservativa é possível desenvolver esquemas numéricos conservativos da energia total do sistema para que, com isso, seja possível obter informações corretas em relação a choques e saltos nas descontinuidades (LEVEQUE, 1990). Nesse trabalho, é utilizada a forma conservativa do modelo MHD ideal. Do ponto de vista numérico, a forma conservativa das equações do modelo MHD ideal é utilizada para o cálculo do fluxo numérico e, também, para evoluir a solução no tempo.

O modelo MHD ideal, na sua forma conservativa (BITTENCOURT, 2004), é descrito pelo sistema de equações diferenciais parciais

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = 0, \quad \text{Continuidade} \quad (2.1a)$$

$$\frac{\partial E}{\partial t} + \nabla \cdot \left[ \left( E + p + \frac{\mathbf{B} \cdot \mathbf{B}}{2} \right) \mathbf{u} - \mathbf{B} (\mathbf{u} \cdot \mathbf{B}) \right] = 0, \quad \text{Energia/pressão} \quad (2.1b)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot \left[ \rho \mathbf{u} \mathbf{u} + \mathbf{I} \left( p + \frac{\mathbf{B} \cdot \mathbf{B}}{2} \right) - \mathbf{B} \mathbf{B} \right] = 0, \quad \text{Momento} \quad (2.1c)$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{u} \mathbf{B} - \mathbf{B} \mathbf{u}) = 0, \quad \text{Campo Magnético} \quad (2.1d)$$

em que  $\rho$  é a densidade,  $p$  é a pressão,  $\mathbf{u}$  é o vetor velocidade,  $\mathbf{B}$  é o vetor campo magnético e  $\mathbf{I}$  é o tensor unitário de ordem 2. A energia  $E$  dada pela lei constitutiva

$$E = \frac{p}{\gamma - 1} + \rho \frac{\mathbf{u} \cdot \mathbf{u}}{2} + \frac{\mathbf{B} \cdot \mathbf{B}}{2}, \quad (2.2)$$

em que  $\gamma$  é uma constante adiabática.

Com adição da restrição física

$$\nabla \cdot \mathbf{B} = 0, \quad \text{Eq. Fluxo Magnético}, \quad (2.3)$$

garante-se que o sistema será livre de divergência. Essa restrição não é naturalmente obtida. Em geral, nas simulações numéricas, pode-se gerar resultados discordantes da física representada. Esse problema é bem abordado em Brackbill e Barnes (1980). Há várias possíveis abordagens para esse problema, algumas delas estão relacionadas às correções do campo magnético. Neste trabalho utiliza-se este tipo de abordagem, mais detalhes estão na próxima seção.

## 2.2 Correções Numéricas de Divergência Nula

A divergência é uma propriedade de um campo físico, que pode ser obtida por um operador vetorial que mede a magnitude de comportamentos como fonte ou sorvedouro em um campo vetorial. Fisicamente, o modelo MHD é um sistema livre de divergência no campo magnético, não possuindo esses tipos de comportamento. Normalmente, nas simulações numéricas,  $\nabla \cdot \mathbf{B}$  não é exatamente zero à medida que o tempo evolui. Com isso, erros são introduzidos no sistema, fazendo com que

a condição inicial em cada iteração seja violada e causando problemas relacionados à estabilidade da simulação e à obtenção de resultados não-físicos (TóTH, 2000; DEDNER et al., 2002).

Para evitar que esses erros numéricos ocorram, deve-se utilizar técnicas que introduzam correções ao modelo. No contexto deste trabalho, para fazer a correção da restrição imposta pela Equação de Fluxo Magnético 2.3, introduz-se uma nova função escalar  $\psi$  ao problema. Isto é feito inserindo  $\psi$  na Equação 2.1d, o que resulta na correção de divergência livre da seguinte forma

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{B} - \mathbf{B}\mathbf{u}) + \nabla\psi = 0, \quad (2.4)$$

$$D(\psi) + \nabla \cdot \mathbf{B} = 0, \quad (2.5)$$

em que  $D$  é um operador diferencial linear.

Observa-se que, dessa forma, se a divergência nula é válida, tem-se  $D(\psi) \equiv 0$ , implicando em  $\psi$  constante. Assim, obtém-se  $\nabla\psi \equiv 0$ , ou seja, o modelo físico não é alterado. Aplicando os operadores de divergência ( $\nabla \cdot$ ), ( $D\nabla \cdot$ ) na Equação 2.4, de evolução temporal ( $\frac{\partial}{\partial t}$ ) e o Laplaciano ( $\Delta$ ) na Equação 2.5, obtém-se as seguintes equações

$$\frac{\partial}{\partial t} \nabla \cdot \mathbf{B} + \Delta\psi = 0, \quad (2.6a)$$

$$\frac{\partial}{\partial t} D(\nabla \cdot \mathbf{B}) + \Delta D(\psi) = 0, \quad (2.6b)$$

$$\frac{\partial}{\partial t} D(\psi) + \frac{\partial}{\partial t} \nabla \cdot \mathbf{B} = 0, \quad (2.6c)$$

$$\Delta D(\psi) + \Delta \nabla \cdot \mathbf{B} = 0. \quad (2.6d)$$

Substituindo as Equações 2.6d em 2.6b e 2.6c em 2.6a, tem-se

$$\frac{\partial}{\partial t} D(\nabla \cdot \mathbf{B}) - \Delta \nabla \cdot \mathbf{B} = 0, \quad (2.7)$$

$$\frac{\partial}{\partial t} D(\psi) - \Delta\psi = 0. \quad (2.8)$$

Conclui-se, então, que  $\nabla \cdot \mathbf{B}$  e  $\psi$  satisfazem a mesma equação, independente da escolha de  $D$ , como pode ser observado nas Equações 2.7 e 2.8. A escolha desse ope-

rador diferencial dá origem a quatro tipos de correção: elíptica, parabólica, hiperbólica e parabólica-hiperbólica (DEDNER et al., 2002). As mais populares entre essas correções são as correções parabólica-hiperbólica e elíptica. A seguir são descritas resumidamente essas quatro correções.

### 2.2.1 Correção Parabólica-Hiperbólica

A correção parabólica-hiperbólica proposta em Dedner et al (2002) é feita escolhendo-se o seguinte operador  $D$

$$D(\psi) := \frac{1}{c_h^2} \frac{\partial \psi}{\partial t} + \frac{1}{c_p^2} \psi \quad (2.9)$$

o qual é uma combinação dos operadores parabólico e hiperbólico. Em que  $c_h$  e  $c_p$  são os coeficientes associados aos operadores hiperbólico e parabólico, respectivamente. Substituindo o operador dado pela Equação 2.9 em 2.5, obtém-se

$$\frac{\partial \psi}{\partial t} + c_h^2 \nabla \cdot \mathbf{B} = -\frac{c_h^2}{c_p^2} \psi \quad \text{Correção Parabólica-hiperbólica} \quad (2.10)$$

A correção parabólica-hiperbólica possui tanto a propagação, como a dissipação de erros ao longo do tempo.

Nas correções parabólica e hiperbólica há uma propagação de erros numéricos, tornando-as menos populares, porém com baixo custo computacional. Com o intuito de obter uma correção mais acurada do que essas, é introduzida a correção parabólica-hiperbólica como proposta em Dedner et al (2002). Essa correção pode ser vista como sendo um caminho do meio entre as correções elípticas, parabólicas e hiperbólicas, tanto do ponto de vista de custo computacional como de precisão numérica. Além disso, essa correção parabólica-hiperbólica permite que se utilize esquemas explícitos temporais para a resolução numérica desse sistema, o que os qualifica para o uso de métodos de multirresolução adaptativos, como são descritos no Capítulo 7, que são de interesse nesta dissertação.

### 2.2.2 Correção Elíptica

A correção elíptica garante que  $\nabla \cdot \mathbf{B} = 0$ , o que a torna a correção mais popular entre as citadas neste capítulo. Entretanto, para a resolução de um problema elíptico é necessário o uso de esquemas implícitos, o que implica em um alto custo

computacional e impossibilita a utilização do método adaptativo de multirresolução.

### 2.2.3 Correção Parabólica

A correção parabólica é baseada na conhecida Equação do Calor. Se condições de contorno adequadas são utilizadas, a correção parabólica faz com que os erros locais de divergência sejam dissipados e suavizados.

### 2.2.4 Correção Hiperbólica

A correção hiperbólica é baseada na conhecida como Equação de Propagação (ou Equação da Onda). Com a correção hiperbólica, os erros locais de divergência são propagados para a fronteira com velocidade finita  $c_h > 0$ .

## 2.3 Multiplicador de Lagrange Generalizado

A correção parabólica-hiperbólica, juntamente com o Sistema de Equações 2.1, formam o chamado *Multiplicador de Lagrange Generalizado das Equações de MHD* (GLM-MHD):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = 0, \quad (2.11a)$$

$$\frac{\partial E}{\partial t} + \nabla \cdot \left( (E + p + \frac{\mathbf{B} \cdot \mathbf{B}}{2}) \mathbf{u} - \mathbf{B}(\mathbf{u} \cdot \mathbf{B}) \right) = 0, \quad (2.11b)$$

$$\frac{\partial \psi}{\partial t} + c_h^2 \nabla \cdot \mathbf{B} = -\frac{c_h^2}{c_p^2} \psi, \quad (2.11c)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot \left( \rho \mathbf{u} \mathbf{u} + \mathbf{I}(p + \frac{\mathbf{B} \cdot \mathbf{B}}{2}) - \mathbf{B} \mathbf{B} \right) = 0, \quad (2.11d)$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{u} \mathbf{B} - \mathbf{B} \mathbf{u} + \psi \mathbf{I}) = 0, \quad (2.11e)$$

Para correções mistas, como a correção parabólica-hiperbólica considerada neste estudo, o sistema GLM-MHD é hiperbólico e suas variáveis  $\rho$ ,  $E$ ,  $\rho \mathbf{u}$  e  $\mathbf{B}$  se mantêm conservadas.

## 2.4 Extensão do Multiplicador de Lagrange Generalizado

Uma outra formulação do modelo MHD ideal com correções de divergência livre pode ser obtida a partir das equações do modelo GLM-MHD e de Euler. Nas equações de fluido, a influência do campo magnético aparece como sendo uma força externa

chamada *força de Lorentz*. A força de Lorentz é descrita como

$$\mathbf{F}_L = \mathbf{J} \times \mathbf{B}, \quad (2.12a)$$

$$= (\nabla \times \mathbf{B}) \times \mathbf{B}, \quad (2.12b)$$

$$= \nabla \cdot \left[ \mathbf{B}\mathbf{B} - \frac{1}{2}\mathbf{B}^2\mathbf{I} \right] - \mathbf{B}(\nabla \cdot \mathbf{B}), \quad (2.12c)$$

em que  $\mathbf{J}$  é o vetor densidade de corrente elétrica. Considerando a Equação de Euler de conservação de momento, juntamente com a força de Lorentz, tem-se

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u} + \mathbf{I}p) = \mathbf{J} \times \mathbf{B}, \quad (2.13)$$

Substituindo  $\mathbf{J} \times \mathbf{B}$  pelo termo obtido na Equação 2.12c, obtém-se

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \mathbf{u} + \mathbf{I}p) = \nabla \cdot \left[ \mathbf{B}\mathbf{B} - \frac{1}{2}\mathbf{B}^2\mathbf{I} \right] - \mathbf{B}(\nabla \cdot \mathbf{B}), \quad (2.14)$$

a qual é a Equação de conservação de momento do modelo em questão. De maneira similar, obtém-se o termo fonte  $-\mathbf{B} \cdot (\nabla \psi)$  para a Equação de Energia 2.1b (DEDNER et al., 2002). Dessa forma, tem-se o seguinte sistema MHD em forma conservativa

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{u} = 0, \quad (2.15a)$$

$$\frac{\partial E}{\partial t} + \nabla \cdot \left( (E + p + \frac{\mathbf{B} \cdot \mathbf{B}}{2}) \mathbf{u} - \mathbf{B}(\mathbf{u} \cdot \mathbf{B}) \right) = -\mathbf{B} \cdot (\nabla \psi), \quad (2.15b)$$

$$\frac{\partial \psi}{\partial t} + c_h^2 \nabla \cdot \mathbf{B} = -\frac{c_h^2}{c_p^2} \psi, \quad (2.15c)$$

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot \left( \rho \mathbf{u} \mathbf{u} + \mathbf{I}(p + \frac{\mathbf{B} \cdot \mathbf{B}}{2}) - \mathbf{B}\mathbf{B} \right) = -\mathbf{B}(\nabla \cdot \mathbf{B}), \quad (2.15d)$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{B} - \mathbf{B}\mathbf{u} + \psi\mathbf{I}) = 0, \quad (2.15e)$$

o qual é chamado de *Extensão do Sistema GLM-MHD* (EGLM). Essa formulação do modelo MHD ideal foi a escolhida para ser utilizada neste trabalho. No próximo capítulo, introduzem-se os conceitos de leis de conservação hiperbólicas, expandindo a discussão para o caso EGLM-MHD.

### 3 LEIS DE CONSERVAÇÃO

Leis de conservação podem ser traduzidas por sistemas de Equações Diferenciais Parciais (EDPs) evolutivas, com condições iniciais e condições de contorno, em que existem variáveis que se mantêm conservadas na evolução temporal. Essas equações são normalmente não-lineares e com uma estrutura particularmente simples. Essas leis são baseadas em princípios físicos como, por exemplo, conservação de momento, massa e energia. Neste trabalho, tem-se interesse em leis de conservação hiperbólicas (LEVEQUE, 1990). De forma genérica, uma lei de conservação hiperbólica pode ser escrita como

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = \mathbf{S}(\mathbf{U}), \quad (3.1)$$

em que  $\mathbf{U}$  é o vetor de variáveis em sua forma conservativa,  $\mathbf{F}(\mathbf{U}) = (F(\mathbf{U})_x, F(\mathbf{U})_y)$  é o fluxo físico e  $\mathbf{S}(\mathbf{U})$  é um vetor de termos de fonte.

Normalmente, as funções de fluxo são não-lineares, fazendo com que o sistema de EDPs também seja não-linear. Geralmente, não é possível obter soluções exatas desses tipos de equações, apenas soluções numéricas aproximadas por esquemas. Neste trabalho, o sistema de leis de conservação hiperbólica de grande importância é o modelo EGLM-MHD.

#### 3.1 Modelo EGLM-MHD

O modelo EGLM-MHD, dado como nas Equações 2.15, apresentado na pág. 10, é uma lei de conservação hiperbólica. Nesse caso, os vetores  $\mathbf{U}$ ,  $\mathbf{F}(\mathbf{U})$  e  $\mathbf{S}(\mathbf{U})$ , de acordo com a Equação A.1, são descritos a seguir.

$$\mathbf{U} = \begin{pmatrix} \rho \\ E \\ \psi \\ \rho \mathbf{u} \\ \mathbf{B} \end{pmatrix}, \quad \mathbf{S}(\mathbf{U}) = \begin{pmatrix} 0 \\ -\mathbf{B} \cdot (\nabla \psi) \\ -\frac{c_h^2}{c_p^2} \psi \\ -\mathbf{B}(\nabla \cdot \mathbf{B}) \\ \mathbf{0} \end{pmatrix}, \quad (3.2)$$

$$\mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho \mathbf{u} \\ (E + p + \frac{\mathbf{B} \cdot \mathbf{B}}{2}) \mathbf{u} - \mathbf{B}(\mathbf{u} \cdot \mathbf{B}) \\ c_h^2 \mathbf{B} \\ \rho \mathbf{u} \mathbf{u} + \mathbf{I}(p + \frac{\mathbf{B} \cdot \mathbf{B}}{2}) - \mathbf{B} \mathbf{B} \\ \mathbf{u} \mathbf{B} - \mathbf{B} \mathbf{u} + \psi \mathbf{I} \end{pmatrix}. \quad (3.3)$$

A forma expandida desses vetores são apresentadas no Apêndice A. Utilizando a forma matricial do modelo MHD, pode-se calcular os autovalores do sistema. Para calculá-los, é preciso transformar o sistema da forma conservativa para a forma primitiva. Ou seja, ao invés de utilizar as variáveis conservadas momento  $\rho \mathbf{u}$  e energia  $E$ , utilizam-se as variáveis primitivas velocidade  $\mathbf{u}$  e pressão  $p$ . Além disso, o sistema não permanece mais na forma de leis de conservação. Na próxima seção, esse modelo MHD é escrito em sua forma primitiva.

### 3.2 Forma Primitiva

Neste trabalho, é de interesse o estudo dos autovalores do modelo EGLM-MHD. Para isso, conforme apresentado em Dedner et al (2002), utiliza-se o modelo GLM-MHD, como descrito pelo Sistema 2.11 na Seção 2.3. Ao omitir o termo fonte  $-\frac{c_h^2}{c_p^2} \psi$ , referente à equação da correção de divergência parabólica-hiperbólica, é possível escrevê-lo como uma equação diferencial homogênea. Em duas dimensões no espaço, pode-se escrever o sistema GLM-MHD homogêneo em sua forma primitiva como

$$\frac{\partial \mathbf{W}}{\partial t} + A_x(\mathbf{W}) \frac{\partial \mathbf{W}}{\partial x} + A_y(\mathbf{W}) \frac{\partial \mathbf{W}}{\partial y} = 0 \quad (3.4)$$

em que  $\mathbf{W} = (\rho, p, \psi, v_x, v_y, v_z, B_x, B_y, B_z)$  é o vetor de variáveis primitivas e as matrizes são

$$A_x(\mathbf{W}) = \begin{pmatrix} v_x & 0 & 0 & \rho & 0 & 0 & 0 & 0 & 0 \\ 0 & v_x & (1-\gamma)B_x & \gamma p & 0 & 0 & (\gamma-1)\mathbf{u} \cdot \mathbf{B} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & c_h^2 & 0 & 0 \\ 0 & \frac{1}{\rho} & 0 & v_x & 0 & 0 & -\frac{B_x}{\rho} & \frac{B_y}{\rho} & \frac{B_z}{\rho} \\ 0 & 0 & 0 & 0 & v_x & 0 & -\frac{B_y}{\rho} & -\frac{B_x}{\rho} & 0 \\ 0 & 0 & 0 & 0 & 0 & v_x & -\frac{B_z}{\rho} & 0 & -\frac{B_x}{\rho} \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & B_y & -B_x & 0 & -v_y & v_x & 0 \\ 0 & 0 & 0 & B_z & 0 & -B_x & -v_z & 0 & v_x \end{pmatrix} \quad (3.5)$$

e

$$A_y(\mathbf{W}) = \begin{pmatrix} v_y & 0 & 0 & 0 & \rho & 0 & 0 & 0 & 0 \\ 0 & v_y & (1-\gamma)B_y & 0 & \gamma p & 0 & 0 & (\gamma-1)\mathbf{u} \cdot \mathbf{B} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_h^2 & 0 \\ 0 & 0 & 0 & v_y & 0 & 0 & -\frac{B_y}{\rho} & -\frac{B_x}{\rho} & 0 \\ 0 & \frac{1}{\rho} & 0 & 0 & v_y & 0 & \frac{B_x}{\rho} & -\frac{B_y}{\rho} & \frac{B_z}{\rho} \\ 0 & 0 & 0 & 0 & 0 & v_y & 0 & -\frac{B_z}{\rho} & -\frac{B_y}{\rho} \\ 0 & 0 & 0 & -B_y & B_x & 0 & v_y & -v_x & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & B_z & -B_y & 0 & -v_z & v_y \end{pmatrix}. \quad (3.6)$$

A partir das matrizes  $A_x(\mathbf{W})$  e  $A_y(\mathbf{W})$ , é possível calcular os autovalores referente ao sistema GLM-MHD.

### 3.2.1 Autovalores

Ao calcular os autovalores da matriz  $A_x$  ou  $A_y$  (DEDNER et al., 2002), obtém-se nove autovalores

$$\begin{aligned} \lambda_{h-} &= -c_h, & \lambda_{f-} &= v_k - c_f, & \lambda_{a-} &= v_k - c_a, \\ \lambda_{s-} &= v_k - c_s, & \lambda_e &= v_k, & \lambda_{s+} &= v_k + c_s, \\ \lambda_{a+} &= v_k + c_a, & \lambda_{f+} &= v_k + c_f, & \lambda_{h+} &= c_h, \end{aligned} \quad (3.7)$$

em que  $k = x$  ou  $k = y$  e as velocidades  $c_s \leq c_a \leq c_f$  são dadas por

$$c_a = |b_k|, \quad (3.8a)$$

$$c_f = \sqrt{\frac{1}{2}(a^2 + b^2 + \sqrt{(a^2 + b^2)^2 - 4a^2b_k^2})}, \quad (3.8b)$$

$$c_s = \sqrt{\frac{1}{2}(a^2 + b^2 - \sqrt{(a^2 + b^2)^2 - 4a^2b_k^2})}, \quad (3.8c)$$

com

$$b_k := \sqrt{\frac{B_k^2}{\rho}}, \quad b^2 := \frac{|B|^2}{\rho}, \quad a := \sqrt{\frac{\gamma p}{\rho}}. \quad (3.9)$$

Os autovalores  $\lambda_a$ ,  $\lambda_s$  e  $\lambda_f$  representam, respectivamente, as ondas de *Alfvén* e as ondas magneto-acústicas lenta (*slow*, em inglês) e rápida (*fast*, em inglês). Os termos  $c_a$ ,  $c_s$  e  $c_f$  são as velocidades de Alfvén, magneto-acústica lenta e rápida, e estão associadas a esses autovalores. Já o autovalor  $\lambda_e$ , está relacionado à onda de entropia. Essas ondas são conhecidas como curvas características. Essas curvas descrevem a propagação através do plasma e são associadas aos autovalores do sistema. As curvas características são de grande importância neste contexto e são descritas no Capítulo 5. Na Figura 3.1, essas curvas características são apresentadas em uma estrutura de leque de Riemann (*Riemann fan*, em inglês).

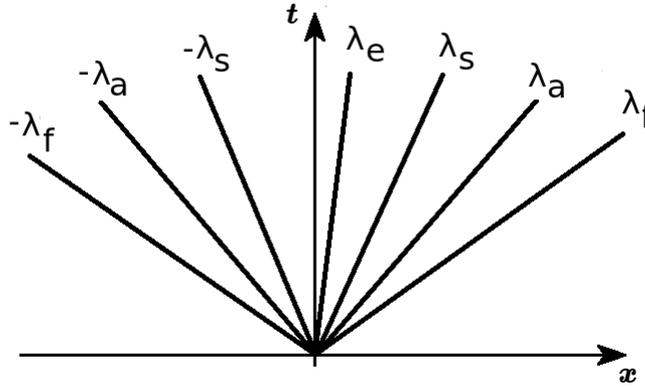


Figura 3.1 - Leque de Riemann: Curvas características referentes aos autovalores do sistema GLM-MHD. Em que  $\lambda_f$ ,  $\lambda_a$ ,  $\lambda_s$  são, respectivamente, as ondas magneto-acústicas rápidas, de Alfvén, magneto-acústicas lentas e  $\lambda_e$  é a onda relacionada à entropia

Para calcular os autovalores do modelo EGLM-MHD, deve-se criar uma analogia ao

que foi feito para o modelo GLM-MHD, ou seja, também deve-se escrevê-lo como uma equação diferencial homogênea. A diferença, nesse caso, é que o modelo EGLM-MHD possui termos fonte nas equações de energia e momento, ou seja, o vetor  $\mathbf{S}(\mathbf{U})$ .

Deve-se supor, então, que existam duas matrizes  $B_x(\mathbf{W})$  e  $B_y(\mathbf{W})$ , as quais representam o vetor  $\mathbf{S}(\mathbf{U})$  nas direções  $x$  e  $y$ , tais que o sistema EGLM-MHD possa ser escrito como uma equação diferencial homogênea. Então, omitindo o termo fonte  $-\frac{c_p^2}{c_p^2}\psi$ , referente à equação da correção de divergência parabólica-hiperbólica, a forma primitiva desse modelo pode ser escrita como

$$\frac{\partial \mathbf{W}}{\partial t} + (A_x(\mathbf{W}) - B_x(\mathbf{W})) \frac{\partial \mathbf{W}}{\partial x} + (A_y(\mathbf{W}) - B_y(\mathbf{W})) \frac{\partial \mathbf{W}}{\partial y} = 0, \quad (3.10)$$

em que

$$B_x(\mathbf{W}) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & (1-\gamma)B_x & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{B_x}{\rho} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{B_y}{\rho} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{B_z}{\rho} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (3.11)$$

$$B_y(\mathbf{W}) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & (1-\gamma)B_y & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{B_x}{\rho} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{B_y}{\rho} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{B_z}{\rho} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (3.12)$$

As matrizes  $(A_x(\mathbf{W}) - B_x(\mathbf{W}))$  e  $(A_y(\mathbf{W}) - B_y(\mathbf{W}))$  possuem os mesmos autovalores obtidos a partir das matrizes  $A_x(\mathbf{W})$  e  $A_y(\mathbf{W})$ , conforme observado em Dedner et al (2002). Conclui-se, então, que o modelo EGLM-MHD possui as mesmas curvas características que o modelo GLM-MHD.

O cálculo dos autovetores do modelo EGLM-MHD pode ser indispensável, dependendo do esquema numérico que se deseja utilizar para calcular o vetor  $\mathbf{F}(\mathbf{U})$ , como, por exemplo, o esquema numérico HLLEM (WESENBERG, 2003). Neste trabalho, como pode ser visto no Capítulo 5, não é necessária a utilização dos autovetores associados aos autovalores desse sistema, pois são utilizados os esquemas numéricos HLL e HLLD.

### 3.3 Solução Exata e Verificação

Nessa seção é apresentada a solução exata de um modelo MHD. Para fins de teste, a condição inicial utilizada nesse exemplo é uma condição tipo Riemann, a qual é baseada na formulação de um problema de Riemann.

#### 3.3.1 Condição Inicial tipo Riemann

Em meados do século XIX, Riemann estudou um problema de um tubo com dois gases. Se o efeito das paredes do tubo são desprezadas, esse problema torna-se essencialmente unidimensional. Este tipo de problema de valor inicial ficou então conhecido como Problema de Riemann. Matematicamente, chama-se Problema de Riemann 1D associado à uma lei de conservação  $\frac{\partial u}{\partial t} + A \frac{\partial u}{\partial x} = 0$  ao seguinte problema de valor inicial

$$u(x, 0) = \begin{cases} u_l, & \text{se } x \leq 0 \\ u_r, & \text{se } x > 0 \end{cases} \quad (3.13)$$

Análoga ao problema de Riemann, a condição utilizada neste exemplo é chamada condição inicial tipo Riemann (LEVEQUE, 1990).

#### 3.3.2 Solução Exata

Sob certas circunstâncias, a solução exata do modelo MHD ideal pode ser calculada utilizando um programa chamado *Exact Riemann Solver for Ideal MHD*<sup>1</sup>, que utiliza o algoritmo descrito em Torrilhon (2003). Um exemplo é a condição inicial tipo Riemann, descrita na Tabela 3.1. Essa condição inicial é similar à condição tipo Riemann 1D utilizada em Dai e Woodward (DAI; WOODWARD, 1995), exceto pela constante na variável  $B_z$ , que é definida por  $\sqrt{2\pi}$  ao invés de  $\sqrt{4\pi}$ . Essa condição inicial tipo Riemann 1D também é utilizada em Li (2005).

---

<sup>1</sup><https://web.mathcces.rwth-aachen.de/mhdsolver/>

Tabela 3.1 - Condição inicial tipo Riemann.

	$\rho$	$p$	$v_x$	$v_y$	$v_z$	$B_x$	$B_y$	$B_z$
$x \leq 0$	1.08	0.95	1.2	0.01	0.5	$2.0/\sqrt{4\pi}$	$3.6/\sqrt{4\pi}$	$2.0/\sqrt{2\pi}$
$x > 0$	1.0	1.0	0.0	0.0	0.0	$2.0/\sqrt{4\pi}$	$4.0/\sqrt{4\pi}$	$2.0/\sqrt{2\pi}$

A condição inicial tipo Riemann, em um intervalo  $[-0.5, 0.5]$ , é ilustrada na Figura 3.2.

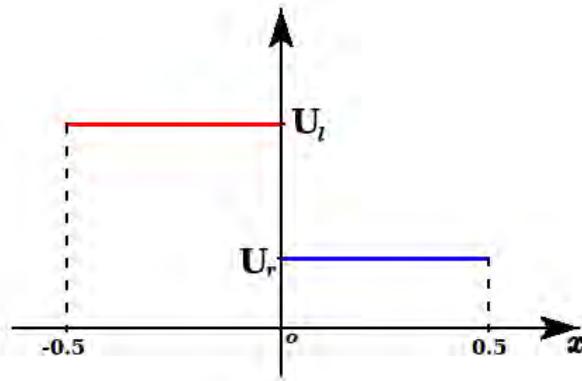


Figura 3.2 - Condição inicial tipo Riemann no intervalo  $[-0.5, 0.5]$ . Os valores  $U_l$  e  $U_r$  são, respectivamente, os valores à esquerda e à direita.

Nas Figuras 3.3 e 3.4, é apresentada a solução exata do modelo MHD ideal, conforme Torrilhon (2002), para  $t = 0.1$ , variáveis densidade ( $\rho$ ), pressão ( $p$ ), componentes  $x$ ,  $y$  e  $z$  da velocidade ( $v_x$ ,  $v_y$ ,  $v_z$ ) e componentes  $y$  e  $z$  do campo magnético ( $B_y$ ,  $B_z$ ), utilizando a condição inicial da Tabela 3.1. A solução exata também é descrita pelas Equações 3.14–3.21.

Esse resultado é muito importante no contexto deste trabalho, pois é utilizado na verificação dos resultados apresentados pelo modelo EGLM-MHD. Nesse contexto, para diferenciá-lo de um outro caso de Riemann que é utilizado, ele é denotado de condição inicial tipo Riemann 1D.

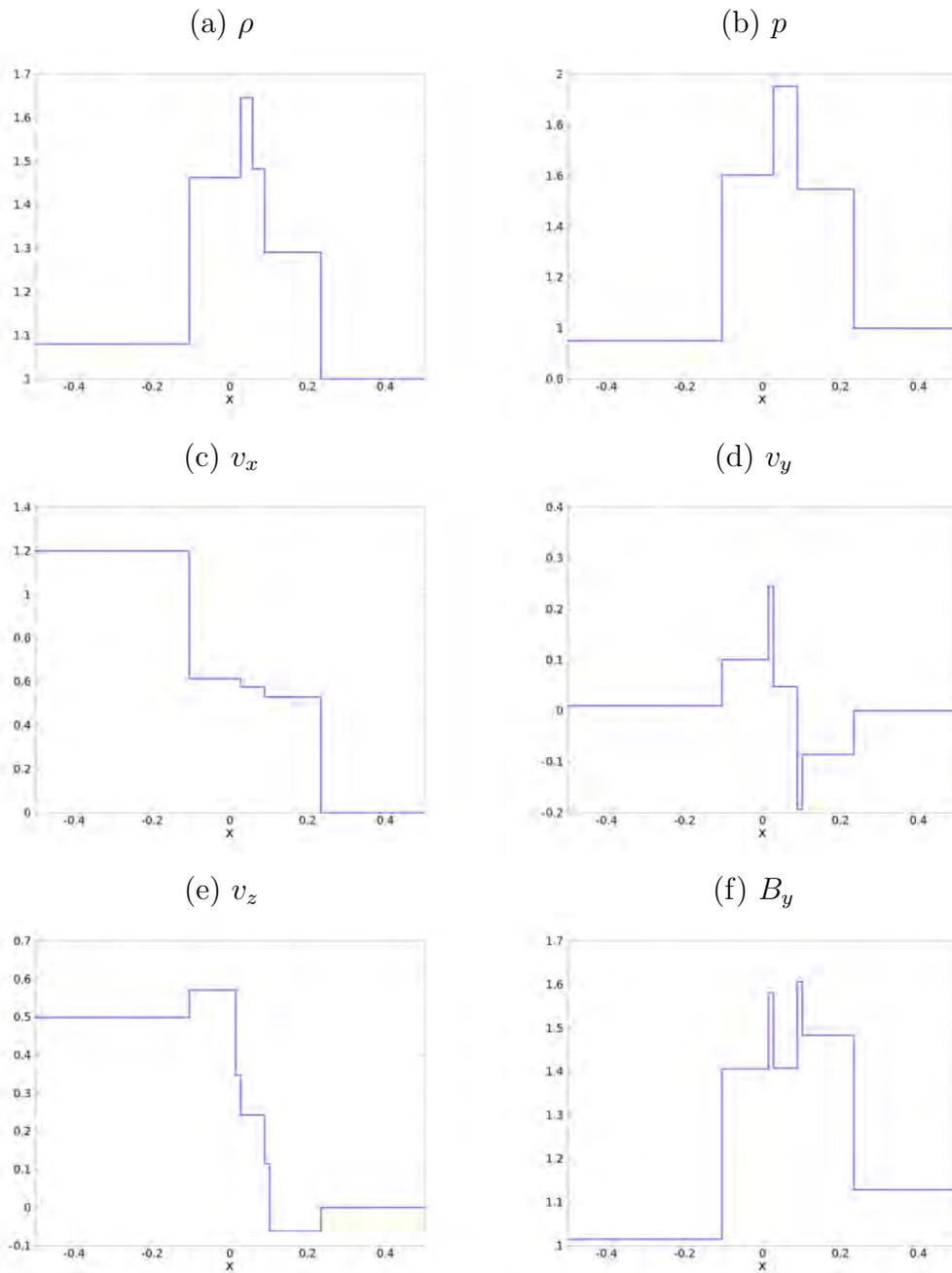


Figura 3.3 - Gráficos da solução exata de um modelo MHD ideal, para as variáveis densidade  $\rho$ , pressão  $p$ , componentes  $x$ ,  $y$ ,  $z$  da velocidade  $v_x$ ,  $v_y$ ,  $v_z$  e componente  $y$  do campo magnético  $B_y$ , no tempo  $t = 0.1$ , utilizando  $\gamma = 5/3$ .

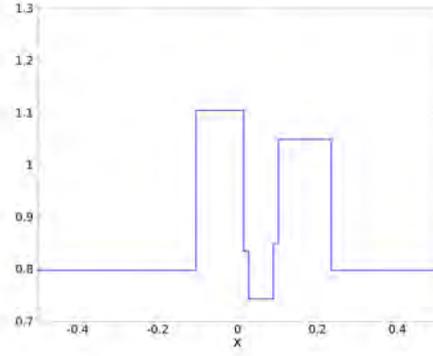


Figura 3.4 - Gráficos da solução exata de um modelo MHD ideal, para a variável componente  $z$  do campo magnético  $B_z$ , no tempo  $t = 0.1$ , utilizando  $\gamma = 5/3$ .

$$\rho = \begin{cases} 1.0800, & \text{se } -0.500 \leq x < -0.1040, \\ 1.4626, & \text{se } -0.1040 \leq x < 0.0276, \\ 1.6459, & \text{se } 0.0276 \leq x < 0.0576, \\ 1.4832, & \text{se } 0.0576 \leq x < 0.0888, \\ 1.2914, & \text{se } 0.0888 \leq x < 0.2348, \\ 1.0000, & \text{se } 0.2348 \leq x < 0.5000, \end{cases} \quad (3.14)$$

$$p = \begin{cases} 0.9500, & \text{se } -0.500 \leq x < -0.1040, \\ 1.6026, & \text{se } -0.1040 \leq x < 0.0276, \\ 1.9526, & \text{se } 0.0276 \leq x < 0.0888, \\ 1.5481, & \text{se } 0.0888 \leq x < 0.2348, \\ 1.0000, & \text{se } 0.2348 \leq x < 0.5000, \end{cases} \quad (3.15)$$

$$v_x = \begin{cases} 1.2000, & \text{se } -0.500 \leq x < -0.1040, \\ 0.6139, & \text{se } -0.1040 \leq x < 0.0276, \\ 0.5763, & \text{se } 0.0276 \leq x < 0.0888, \\ 0.5299, & \text{se } 0.0888 \leq x < 0.2348, \\ 0.0000, & \text{se } 0.2348 \leq x < 0.5000, \end{cases} \quad (3.16)$$

$$v_y = \begin{cases} 0.0100, & \text{se } -0.5000 \leq x < -0.1040, \\ 0.1011, & \text{se } -0.1040 \leq x < 0.0147, \\ 0.2460, & \text{se } 0.0147 \leq x < 0.0276, \\ 0.0482, & \text{se } 0.0276 \leq x < 0.0888, \\ -0.1938, & \text{se } 0.0888 \leq x < 0.1026, \\ -0.0854, & \text{se } 0.1026 \leq x < 0.2348, \\ 0.0000, & \text{se } 0.2348 \leq x < 0.5000, \end{cases} \quad (3.17)$$

$$v_z = \begin{cases} 0.5000, & \text{se } -0.5000 \leq x < -0.1040, \\ 0.5716, & \text{se } -0.1040 \leq x < 0.0147, \\ 0.3484, & \text{se } 0.0147 \leq x < 0.0276, \\ 0.2440, & \text{se } 0.0276 \leq x < 0.0888, \\ 0.1162, & \text{se } 0.0888 \leq x < 0.1026, \\ -0.0604, & \text{se } 0.1026 \leq x < 0.2348, \\ 0.0000, & \text{se } 0.2348 \leq x < 0.5000, \end{cases} \quad (3.18)$$

$$B_x = \frac{2}{\sqrt{4\pi}}, \quad (3.19)$$

$$B_y = \begin{cases} 1.0155, & \text{se } -0.5000 \leq x < -0.1040, \\ 1.4064, & \text{se } -0.1040 \leq x < 0.0147, \\ 1.5816, & \text{se } 0.0147 \leq x < 0.0276, \\ 1.4083, & \text{se } 0.0276 \leq x < 0.0888, \\ 1.6069, & \text{se } 0.0888 \leq x < 0.1026, \\ 1.4837, & \text{se } 0.1026 \leq x < 0.2348, \\ 1.1284, & \text{se } 0.2348 \leq x < 0.5000, \end{cases} \quad (3.20)$$

$$B_z = \begin{cases} 0.7979, & \text{se } -0.5000 \leq x < -0.1040, \\ 1.1050, & \text{se } -0.1040 \leq x < 0.0147, \\ 0.8351, & \text{se } 0.0147 \leq x < 0.0276, \\ 0.7436, & \text{se } 0.0276 \leq x < 0.0888, \\ 0.8485, & \text{se } 0.0888 \leq x < 0.1026, \\ 1.0491, & \text{se } 0.1026 \leq x < 0.2348, \\ 0.7979, & \text{se } 0.2348 \leq x < 0.5000, \end{cases} \quad (3.21)$$

No Capítulo 4 é apresentado o método dos volumes finitos, o qual é utilizado para discretizar o modelo EGLM-MHD e possibilitar sua simulação ao longo do tempo.

## 4 MÉTODO DOS VOLUMES FINITOS (VF)

Como visto no Capítulo 3, problemas de valor inicial e de contorno para leis de conservação não possuem, em geral, solução exata. Por isso, é preciso utilizar esquemas numéricos para calcular uma solução aproximada. Para utilizar esses esquemas é preciso discretizá-los utilizando métodos numéricos. Existem alguns métodos numéricos como, por exemplo, o método das diferenças finitas e o método dos volumes finitos.

Nesse trabalho, o interesse volta-se para o método dos volumes finitos. Esse método é baseado na forma integral das leis de conservação, ao contrário, por exemplo, do método das diferenças finitas, que é baseado em equações diferenciais. Melhor do que as aproximações por pontos de grade que pertencem ao domínio, o método dos volumes finitos utiliza células para representar uma quantidade  $\mathbf{U}$ . O domínio do problema é particionado em células de grade e o valor de  $\mathbf{U}$  é aproximado em cada célula pela média de  $\mathbf{U}$  na célula.

Na próxima seção, o método é exposto com maiores detalhes. Maiores explicações acerca do assunto e referências podem ser encontradas em Leveque (2002).

### 4.1 Formulação Geral do Método

A partir de uma lei de conservação hiperbólica unidimensional homogênea, o método dos volumes finitos é deduzido a seguir.

Considere a equação

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} = 0, \quad (4.1)$$

em que  $\mathbf{F}(\mathbf{U})$  é a função fluxo e  $\mathbf{U}$  é o vetor 9-dimensional de quantidades conservadas. Nesse estudo, o interesse é apenas nos vetores  $\mathbf{F}(\mathbf{U})$  e  $\mathbf{U}$  referentes ao modelo EGLM-MHD, como definidos na Seção 3.1, Capítulo 3. Aplicando a forma integral das leis de conservação na Equação 4.1, para quaisquer  $x_1$  e  $x_2$  do domínio,  $x_1 < x_2$ , tem-se

$$\frac{d}{dt} \int_{x_1}^{x_2} \mathbf{U} dx = \mathbf{F}(\mathbf{U})|_{x_1} - \mathbf{F}(\mathbf{U})|_{x_2}. \quad (4.2)$$

O princípio geral da forma integral das leis de conservação é relacionado à taxa de variação de  $\mathbf{U}$  em um volume (ou célula). Essa taxa é igual ao fluxo intercelular, ou seja, o fluxo que passa na fronteira entre duas células da malha. Em uma dimensão, as células  $C_i$  de um determinado domínio são segmentos de reta  $C_i = (x_{i-1/2}, x_{i+1/2})$ ,  $i \in \mathbb{N}$ , como esquematizado na Figura 4.1. Para cada  $C_i$  per-

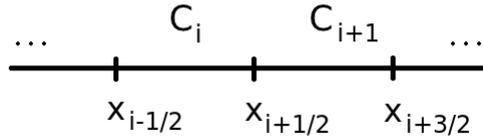


Figura 4.1 - Células pertencentes à uma malha unidimensional de volumes finitos.

tencente à malha, considera-se o valor  $Q_i^n$  como uma aproximação para o valor médio de  $\mathbf{U}$  no  $i$ -ésimo intervalo para o tempo  $t_n$ ,  $n \in \mathbb{N}$  e  $t_n < t_{n+1}$ . Considerando que a malha utilizada é regular, ou seja, todas as células pertencentes à ela possuem o mesmo tamanho, tem-se que

$$Q_i^n \approx \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} \mathbf{U} dx \equiv \frac{1}{\Delta x} \int_{C_i} \mathbf{U} dx. \quad (4.3)$$

Os valores  $Q_i$  são chamados *médias celulares*. Quando utilizam-se médias celulares, pode-se utilizar propriedades importantes das leis de conservação ao derivar um método numérico. Mais particularmente, é possível garantir que um método numérico seja conservativo no sentido de imitar a solução exata do problema. Isso acontece, porque o somatório  $\sum_{i=1}^N Q_i^n \Delta x$  aproxima a integral de  $\mathbf{U}$  e, caso seja utilizado um método que está na forma conservativa, esse somatório muda somente de acordo com o fluxo nas fronteiras. A massa total dentro do domínio computacional é preservada ou, pelo menos, varia corretamente desde que as condições de contorno sejam devidamente aplicadas. Dada a média celular  $Q_i^n$  no intervalo  $C_i$ , aproxima-se o valor de  $Q_i^{n+1}$  utilizando a forma integral descrita na Equação 4.2, ou seja,

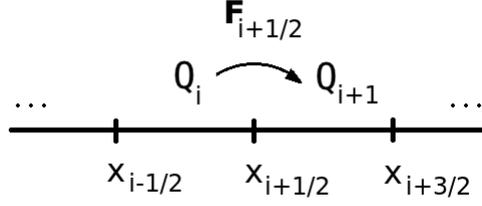


Figura 4.2 - Médias celulares e o fluxo intercelular.

$$\frac{d}{dt} \int_{C_i} \mathbf{U} dx = \mathbf{F}(\mathbf{U})|_{x_{i-1/2}} - \mathbf{F}(\mathbf{U})|_{x_{i+1/2}}. \quad (4.4)$$

Integrando a Equação 4.4 no intervalo entre  $t_n$  a  $t_{n+1}$ , rearranjando seus termos e dividindo-os por  $\Delta x$ , obtém-se

$$\begin{aligned} \frac{1}{\Delta x} \int_{C_i} \mathbf{U}|_{t_{n+1}} dx &= \frac{1}{\Delta x} \int_{C_i} \mathbf{U}|_{t_n} dx \\ &- \frac{1}{\Delta x} \left[ \int_{t_n}^{t_{n+1}} \mathbf{F}(\mathbf{U})|_{x_{i-1/2}} dt - \int_{t_n}^{t_{n+1}} \mathbf{F}(\mathbf{U})|_{x_{i+1/2}} dt \right]. \end{aligned} \quad (4.5)$$

Reescrevendo a Equação 4.5, utilizando a Equação 4.3, tem-se

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x} (F_{i-1/2}^n - F_{i+1/2}^n), \quad (4.6)$$

em que  $F_{i\pm 1/2}^n$  são as aproximações dos fluxos médios nas interfaces celulares quando  $x = x_{i\pm 1/2}$  no tempo  $t = t_n$ , ou seja,

$$F_{i\pm 1/2}^n \approx \frac{1}{\Delta x} \left[ \int_{t_n}^{t_{n+1}} \mathbf{F}(\mathbf{U})|_{x_{i\pm 1/2}} dt \right]. \quad (4.7)$$

Em problemas hiperbólicos, como o modelo MHD, a informação é propagada com velocidade finita. Dessa forma, é rasuável supor que os fluxos  $F_{i\pm 1/2}^n$  tratados neste

trabalho podem ser obtidos em função dos valores de  $Q_{i-1}^n$ ,  $Q_i^n$  e  $Q_{i+1}^n$ , *i.e.*,

$$\begin{aligned} F_{i+1/2}^n &= \mathcal{F}(Q_i^n, Q_{i+1}^n), \\ F_{i-1/2}^n &= \mathcal{F}(Q_{i-1}^n, Q_i^n), \end{aligned} \quad (4.8)$$

em que  $\mathcal{F}$  é uma função de fluxo numérico. Logo, a Equação 4.6 pode ser reescrita como

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x} [\mathcal{F}(Q_{i-1}^n, Q_i^n) - \mathcal{F}(Q_i^n, Q_{i+1}^n)]. \quad (4.9)$$

A Equação 4.9 se parece com a aproximação por diferenças finitas para a evolução temporal da lei de conservação descrita pela Equação 4.1. Isso acontece para esse caso particular em que utiliza-se uma lei de conservação unidimensional para dedução do método dos volumes finitos. No geral, esse tipo de situação não acontece. Ambos os métodos podem ser utilizados com o intuito de discretização de equações diferenciais, mas é importante ressaltar que, por mais que haja esse tipo de similaridade para o caso unidimensional, esses dois métodos são completamente distintos e possuem formulação bastante diferentes. O método dos volumes finitos possui uma formulação que funciona muito bem para o objetivo deste trabalho, já que é baseado em leis de conservação (LEVEQUE, 2002). Por isso, optou-se por sua utilização neste trabalho de mestrado.

Dois aspectos de extrema importância em análise numérica são a estabilidade e a consistência dos esquemas numéricos utilizados. Sem essa abordagem, não é possível garantir a convergência de um esquema numérico.

Seja  $\Delta x$  o comprimento de cada célula da malha na direção  $x$ . Para um método numérico ser convergente, é preciso que sejam satisfeitas duas condições quando  $\Delta x \rightarrow 0$ :

- **Consistência**, quando a discretização feita converge para a equação original;
- **Estabilidade**, no sentido de que os pequenos erros obtidos em cada espaço de tempo não cresçam rapidamente.

## 4.2 A Condição Courant-Friedrichs-Lewy

A condição Courant-Friedrichs-Lewy (CFL) está intimamente ligada com a estabilidade de um esquema numérico. Ela foi uma consequência dos trabalhos de Courant, Friedrichs e Lewy, como pode ser visto em Courant et al (1967), sobre métodos das diferenças finitas para EDPs, a fim de provar a existência de soluções. A idéia era definir sequências de soluções aproximadas dessas equações, provar a sua convergência à medida que a malha era refinada e mostrar que seu limite satisfazia as EDPs. Nesse processo de demonstração da convergência dessa sequência, eles notaram que é necessária uma condição de estabilidade para qualquer que seja o método numérico.

Essa condição é necessária e deve ser satisfeita por todo método dos volumes finitos, caso seja esperada a estabilidade e a convergência de uma solução de uma equação diferencial em que sua malha é refinada. Ao ser satisfeita, ela condiciona que um método deve ser utilizado no sentido de que há chances da informação ser propagada com as velocidades físicas corretas, como determinadas pelos autovalores do sistema.

Matematicamente, a condição CFL para o caso bidimensional pode ser expressa pela desigualdade

$$\nu \leq C \tag{4.10}$$

em que  $\nu$  é chamado *número de Courant* e  $C$  é uma constante adimensional que depende da equação que será resolvida. Em um sistema hiperbólico, existe, geralmente, um conjunto de  $m$  velocidades de onda dadas pelos autovalores  $\lambda_1, \lambda_2, \dots, \lambda_m$ . Nesse caso, o número de Courant é definido como

$$\nu = \frac{\Delta t}{\min\{\Delta x, \Delta y\}} \max_p |\lambda_p|,$$

em que  $p \in \{1, \dots, m\}$  e  $\Delta t$  é o passo de tempo da respectiva iteração.

É importante ressaltar que a condição CFL é necessária, porém não suficiente para garantir a estabilidade de um método. Existem esquemas numéricos que obedecem a essa condição e, ainda assim, são instáveis.

## 4.3 Fluxos

O fluxo  $\mathbf{F}(\mathbf{U})$  é calculado de uma maneira intercelular, ou seja, utiliza-se células vizinhas e calcula-se o fluxo entre essas células. No caso bidimensional, esse fluxo intercelular é calculado nas direções  $x$  e  $y$ . Na Figura 4.3, tem-se as células  $C, C_L, C_R$ ,

$C_U$  e  $C_D$ , as quais correspondem, respectivamente, à célula que está sendo calculada e às células à esquerda, direita, acima e abaixo dela. O fluxo é calculado na interface entre a célula  $C$  e suas adjacentes.

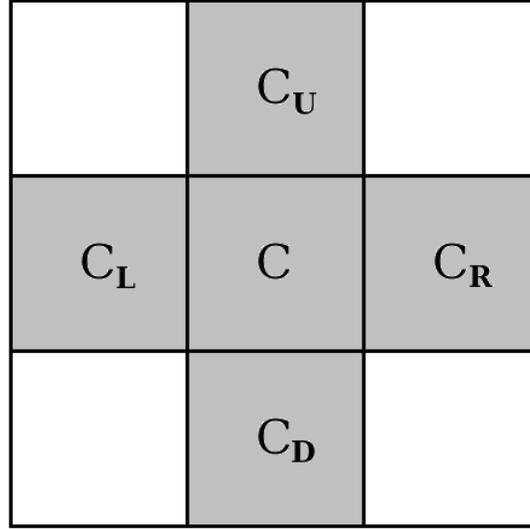


Figura 4.3 - Célula  $C$ , para a qual está sendo calculado o fluxo, e suas células vizinhas  $C_L$ ,  $C_R$ ,  $C_U$  e  $C_D$ . O fluxo intercelular é calculado a partir da célula  $C$  e suas vizinhas, dependendo da direção desejada.

Para o fluxo na direção  $x$  (horizontal), utiliza-se as médias celulares de  $C_L$  e  $C_R$ . Analogamente, na direção  $y$  (vertical) utiliza-se as médias celulares das células  $C_U$  e  $C_D$ . No caso bidimensional, como neste trabalho, o fluxo é calculado em ambas as direções  $x$  e  $y$ , independente da condição inicial utilizada. Esses valores dos fluxos direcionais são utilizados na evolução temporal da solução do modelo EGLM-MHD.

#### 4.4 Evolução Temporal

Para aproximar a solução  $\mathbf{U}$  no tempo  $t_{n+1}$ , utiliza-se uma evolução temporal explícita, a qual depende da solução aproximada e dos fluxos intercelulares no tempo  $t_n$ . No contexto de volumes finitos deste trabalho, para o caso bidimensional, a evolução temporal é feita utilizando o método de Euler de primeira ordem, isto é,

$$\mathbf{U}^{t_{n+1}} = \mathbf{U}^{t_n} - \frac{\Delta t}{\Delta x} (\mathbf{F}_x(\mathbf{U}_R, \mathbf{U}) - \mathbf{F}_x(\mathbf{U}, \mathbf{U}_L)) - \frac{\Delta t}{\Delta y} (\mathbf{F}_y(\mathbf{U}_D, \mathbf{U}) - \mathbf{F}_y(\mathbf{U}, \mathbf{U}_U)), \quad (4.11)$$

em que  $\mathbf{F}_x$ ,  $\mathbf{F}_y$  e  $\Delta x$ ,  $\Delta y$  são, respectivamente, os fluxos intercelulares e os espaçamentos das células nas direções  $x$  e  $y$ , e  $\Delta t$  é o passo de tempo. Como foi apresentada, a evolução temporal da solução utilizada neste trabalho depende diretamente do fluxo intercelular para calcular o valor da solução aproximada, para um tempo  $t_{n+1}$  qualquer. Portanto, quanto mais numericamente eficiente a aproximação desse fluxo é, melhor a aproximação da solução a ser obtida. Neste trabalho, os seguintes fluxos são testados ao longo da implementação: o Harten-Lax-Van Leer (HLL) ([HARTEN et al., 1983](#)) e o Harten-Lax-Van Leer-Descontinuidade (HLLD) ([MIYOSHI; KUSANO, 2005](#)). Esses fluxos numéricos são apresentados e discutidos detalhadamente no Capítulo 5.



## 5 FLUXOS NUMÉRICOS

Os Fluxos numéricos, ou *Riemann solvers*, são conhecidos como sendo uma importante ferramenta no contexto da resolução de esquemas numéricos para leis de conservação. Por esse motivo, com o passar do tempo muitas atenções se voltaram para a comunidade de dinâmica de fluido computacional com o objetivo de desenvolver Riemann solvers mais eficientes. Para o modelo MHD foram desenvolvidos diversos tipos de Riemann solvers (WESENBERG, 2003). Em particular, o Harten-Lax-Van Leer (HLL) (HARTEN et al., 1983) e o HLL-Discontinuities (HLLD) (MIYOSHI; KUSANO, 2005) são os utilizados neste trabalho.

### 5.1 Ondas Elementares como Solução de um Problema de Riemann

Um problema de Riemann para um sistema hiperbólico não-linear  $m \times m$ , com dados  $\mathbf{U}_L$  e  $\mathbf{U}_R$ , é um problema de valor inicial

$$\begin{cases} \mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0, \\ \mathbf{U}(x, 0) = \mathbf{U}^{(0)}(x) = \begin{cases} \mathbf{U}_L, & x < 0, \\ \mathbf{U}_R, & x > 0. \end{cases} \end{cases} \quad (5.1)$$

A solução de similaridade  $\mathbf{U}(x, t)$  da Equação 5.1 consiste em  $m+1$  estados constantes separados por  $m$  ondas, como representado pelo gráfico  $x-t$  na Figura 5.1. Para cada autovalor  $\lambda_i$ , existe uma família de ondas. Para sistemas lineares com coeficientes constantes, cada onda é uma descontinuidade com velocidade  $S_i = \lambda_i$  e define um campo linearmente degenerado.

Para sistemas não-lineares, as ondas podem ser descontínuas, como ondas de choque e de contato, ou ondas suaves de transição, como as de rarefação. Os tipos possíveis de ondas presentes na solução de um problema de Riemann dependem crucialmente das condições de fechamento. Supondo que estados iniciais  $\mathbf{U}_L$ ,  $\mathbf{U}_R$  sejam conectados por uma única onda, ou seja, a solução do problema de Riemann consiste em uma única, e não-trivial, onda. A suposição é inteiramente justificada pelo fato de sempre ser possível resolver um problema de Riemann com dados genéricos e, então, selecionar estados constantes em cada um dos lados de uma determinada onda como sendo os dados iniciais para um problema de Riemann. Se a onda é descontínua, então ela é uma onda de choque ou uma onda de contato.

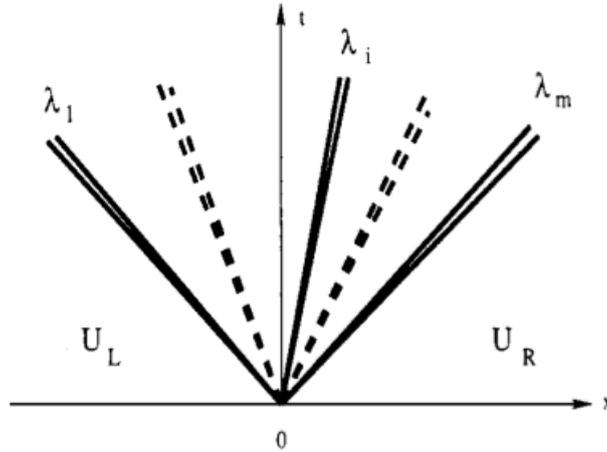


Figura 5.1 - Estrutura da solução de um problema de Riemann para um sistema linear de leis de conservação.  
 Fonte: Toro (1999).

### 5.1.1 Onda de choque

Para uma onda de choque, os dois estados constantes  $\mathbf{U}_L$  e  $\mathbf{U}_R$  são conectados por um único salto de descontinuidade em um *campo não-linear genuíno* e as seguintes condições procedem:

- As condições de Rankine-Hugoniot

$$\mathbf{F}(\mathbf{U}_R) - \mathbf{F}(\mathbf{U}_L) = S_i(\mathbf{U}_R - \mathbf{U}_L). \quad (5.2)$$

- A condição de entropia

$$\lambda_i(\mathbf{U}_L) > S_i > \lambda_i(\mathbf{U}_R). \quad (5.3)$$

A Figura 5.2-a representa uma onda de choque com velocidade  $S_i$ . A característica  $\frac{dx}{dt} = \lambda_i$  em ambos os lados da onda vai em direção à onda de choque, o que ilustra o caráter compressível de um choque.

### 5.1.2 Onda de contato

Para uma onda de contato, os dois estados  $\mathbf{U}_L$  e  $\mathbf{U}_R$  são conectados por um único salto de descontinuidade com velocidade  $S_i$ , em um campo linearmente degenerado

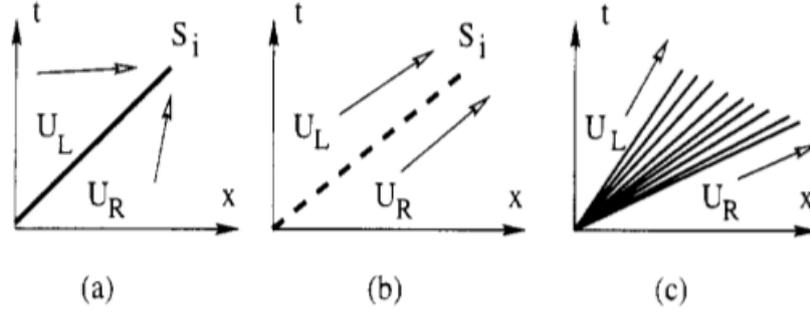


Figura 5.2 - Ondas elementares como soluao de um problema de Riemann: onda de choque da velocidade  $S_i$  (a), discontinuidade de contato da velocidade  $S_i$  (b) e onda de rarefaao (c).  
Fonte: Toro (1999).

$i$  e as seguintes condioes procedem:

- As condioes de Rankine-Hugoniot

$$\mathbf{F}(\mathbf{U}_R) - \mathbf{F}(\mathbf{U}_L) = S_i(\mathbf{U}_R - \mathbf{U}_L). \quad (5.4)$$

- A constancia dos invariantes de Riemann generalizados em toda a onda

$$\frac{d\omega_1}{\kappa_1^{(i)}} = \frac{d\omega_2}{\kappa_2^{(i)}} = \frac{d\omega_3}{\kappa_3^{(i)}} = \dots = \frac{d\omega_m}{\kappa_m^{(i)}}. \quad (5.5)$$

- A condiao da caracteristica paralela

$$\lambda_i(\mathbf{U}_L) = \lambda_i(\mathbf{U}_R) = S_i. \quad (5.6)$$

Na Figura 5.2-b  representada uma descontinuidade de contato. As caracteristicas em ambos os lados da onda so paralelas a ela.

### 5.1.3 Onda de rarefaao

Para uma onda de rarefaao, os dois estados  $\mathbf{U}_L$  e  $\mathbf{U}_R$  so conectados por uma transiao suave, em um campo genuinamente no-linear e as seguintes condioes procedem:

- A constância dos invariantes de Riemann generalizados em toda a onda

$$\frac{d\omega_1}{\kappa_1^{(i)}} = \frac{d\omega_2}{\kappa_2^{(i)}} = \frac{d\omega_3}{\kappa_3^{(i)}} = \dots = \frac{d\omega_m}{\kappa_m^{(i)}}. \quad (5.7)$$

- A divergência das características

$$\lambda_i(\mathbf{U}_L) < \lambda_i(\mathbf{U}_R). \quad (5.8)$$

Na Figura 5.2-c é representada uma onda de rarefação. As características nos lados direito e esquerdo da onda divergem, assim como as características na onda.

## 5.2 O Problema de Riemann e o Fluxo Godunov

Neste trabalho, o interesse volta-se para dois tipos de Riemann solver: HLL e HLLD. O HLL é um Riemann solver mais simples de ser implementado e, por esse motivo, é utilizado nesse trabalho com o intuito inicial de teste e verificação. Já o HLLD, o qual é baseado no HLL, é utilizado por ser um fluxo eficiente na obtenção da solução, pois faz uma aproximação mais acurada, principalmente onde há descontinuidades (MIYOSHI; KUSANO, 2005). A vantagem de utilizar tipos diferentes de Riemann solvers é que torna-se possível testar a eficiência de cada um para esse modelo MHD. Ambos os fluxos são comentados nas próximas seções.

### 5.2.1 Fluxo Harten-Lax-Van Leer

O fluxo numérico HLL foi, primeiramente, apresentado em Harten et al (1983). Eles apresentaram, neste trabalho, o seguinte Riemann solver (na direção  $x$ )

$$\tilde{\mathbf{U}}(x, t) = \begin{cases} \mathbf{U}_L, & \frac{x}{t} \leq S_L, \\ \mathbf{U}_{HLL}, & S_L \leq \frac{x}{t} \leq S_R, \\ \mathbf{U}_R, & \frac{x}{t} \geq S_R, \end{cases} \quad (5.9)$$

em que  $\mathbf{U}_{HLL}$  é o vetor de estado constante dado por

$$\mathbf{U}_{HLL} = \frac{S_R \mathbf{U}_R - S_L \mathbf{U}_L + F_L - F_R}{S_R - S_L}, \quad (5.10)$$

e  $S_L$  e  $S_R$  são as velocidades de propagação das ondas. A Figura 5.3 mostra a

estrutura da solução aproximada do problema de Riemann, chamado Riemann solver HLL. Nota-se que essa aproximação consiste em um único estado constante. Todos

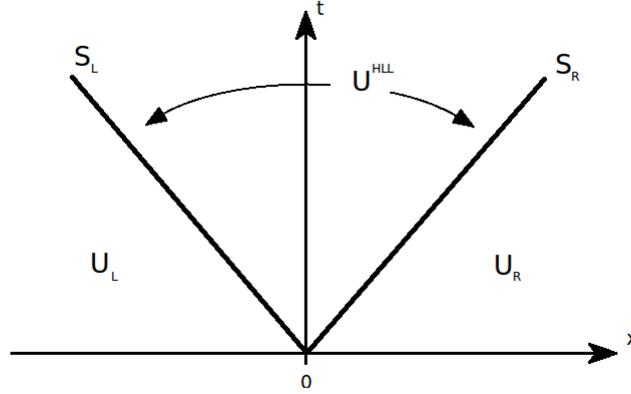


Figura 5.3 - *Leque de Riemann*: região estrela está limitada pelas duas ondas  $S_L$  e  $S_R$ . A solução dessa região consiste no estado  $\mathbf{U}^{HLL}$ .  
Fonte: Toro (1999).

os estados intermediários separados por ondas intermediárias são aglomerados em um único estado  $\mathbf{U}_{HLL}$ . O fluxo correspondente a esse estado é o  $\mathbf{F}_{HLL}$  e o caso de interesse para seu cálculo é caso subsônico, em que  $S_L \leq 0 \leq S_R$ . Tem-se, então,

$$\mathbf{F}_{HLL} = \mathbf{F}_L + S_L(\mathbf{U}_{HLL} - \mathbf{U}_L), \quad (5.11)$$

ou

$$\mathbf{F}_{HLL} = \mathbf{F}_R + S_R(\mathbf{U}_{HLL} - \mathbf{U}_R). \quad (5.12)$$

Utilizando as Equações 5.9, 5.11 e 5.12, obtém-se o fluxo HLL

$$\mathbf{F}_{HLL} = \frac{S_R \mathbf{F}_L - S_L \mathbf{F}_R + S_R S_L (\mathbf{U}_R - \mathbf{U}_L)}{S_R - S_L}, \quad (5.13)$$

O fluxo intercelular correspondente para o método de aproximação Godunov é dado por

$$\mathbf{F}_{HLL}^X = \begin{cases} \mathbf{F}_L, & \text{se } S_L > 0, \\ \mathbf{F}_{HLL}, & \text{se } S_L \leq 0 \leq S_R, \\ \mathbf{F}_R, & \text{se } 0 \geq S_R. \end{cases} \quad (5.14)$$

Dado um algoritmo para calcular as velocidades  $S_L$  e  $S_R$ , tem-se uma aproximação do fluxo intercelular para ser usado na forma conservativa. Neste trabalho, utiliza-se os seguintes valores

$$S_L = \min(u_L - c_L, u_R - c_R), \quad S_R = \max(u_L + c_L, u_R + c_R), \quad (5.15)$$

os quais são conhecidos como *Limitadores de Davis*. Os valores  $u_L, u_R$  representam as velocidades na direção  $x$  e  $c_L, c_R$  velocidades magnetoacústicas. Nesse caso,  $c_L, c_R$  são iguais a  $c_f$ , como definido na Equação 3.8c, pág. 14. Existem outras expressões que podem ser utilizadas para calcular esses valores, além dos limitadores de Davis. Uma outra forma é apresentada para o Riemann solver HLLD.

#### 5.2.1.1 Algoritmo HLL

- a) Calcular as velocidades de propagação  $S_L$  e  $S_R$  de acordo com a Equação 5.15.
- b) Calcular o fluxo HLL de acordo com a Equação 5.14, utilizando as variáveis em sua forma conservativa.

De maneira análoga ao que foi apresentado, assim como discutido no Capítulo 4, é possível obter o fluxo  $\mathbf{F}_{HLL}^Y$ , referente à direção  $y$ . Nesse caso, utiliza-se os valores acima e a baixo da célula em questão.

Uma desvantagem do Riemann solver HLL é o fato dele possuir uma configuração de duas ondas. Esse tipo de configuração funciona bem para sistemas hiperbólicos de duas equações. No entanto, para sistemas com mais equações como o modelo EGLM-MHD, essa configuração de duas ondas não aproxima a solução da mesma maneira, como consequência das descontinuidades de contato e, também, ondas de choque.

### 5.2.2 Fluxo Harten-Lax-Van Leer-Descontinuidades

O Riemann solver HLLD, descrito em Miyoshi e Kusano (2005), pode ser visto como uma extensão do Riemann solver HLL. Ao contrário do HLL, onde utiliza-se uma configuração de duas ondas (ou um estado intermediário), o HLLD é formulado utilizando uma configuração de cinco ondas para ser calculado, retornando uma solução aproximada não linear, de forma que se obtenha uma aproximação mais precisa da solução desejada. Dessa forma, tem-se quatro estados intermediários  $\mathbf{U}_L^*$ ,  $\mathbf{U}_L^{**}$ ,  $\mathbf{U}_R^*$  e  $\mathbf{U}_R^{**}$ , divididos pelas ondas  $S_L$ ,  $S_L^*$ ,  $S_M$ ,  $S_R^*$  e  $S_R$ . A escolha da onda  $S_M$  neste fluxo numérico é para estimar a velocidade normal média a partir do estado intermediário HLL, como na Equação 5.10,

$$S_M = \frac{(S_R - v_{x_R})\rho_R v_{x_R} - (S_L - v_{x_L})\rho_L v_{x_L} - p_{T_R} + p_{T_L}}{(S_R - v_{x_R})\rho_R - (S_L - v_{x_L})\rho_L}. \quad (5.16)$$

Na Figura 5.4, tem-se o *Leque de Riemann* com uma configuração de cinco ondas,  $S_L$ ,  $S_L^*$ ,  $S_M$ ,  $S_R^*$  e  $S_R$ , que limitam os quatro estados  $\mathbf{U}_L^*$ ,  $\mathbf{U}_L^{**}$ ,  $\mathbf{U}_R^*$  e  $\mathbf{U}_R^{**}$ . As ondas  $S_L$  e  $S_R$  estão relacionadas às velocidades  $c_f$ , assim como no HLL. Já as ondas  $S_L^*$ ,  $S_R^*$  e  $S_M$  estão relacionadas às velocidades de Afvén  $c_a$  e ao autovalor  $\lambda_e$ , apresentado na pág. 14, respectivamente.

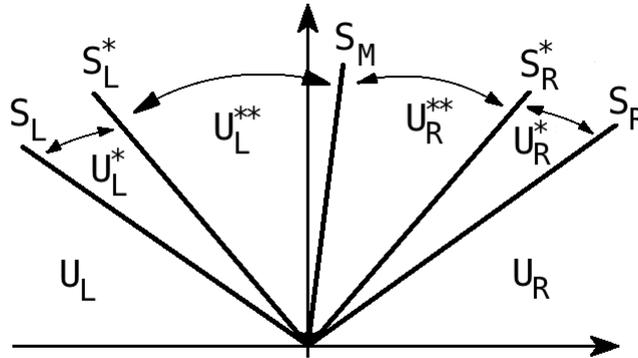


Figura 5.4 - As regiões  $\mathbf{U}_L^*$ ,  $\mathbf{U}_R^*$ ,  $\mathbf{U}_L^{**}$  e  $\mathbf{U}_R^{**}$  estão limitadas pelas duas ondas  $S_L$  e  $S_R$ .  
Fonte Adaptada: Miyoshi e Kusano (2005).

Em todo o Leque de Riemann, a velocidade é assumida como constante, ou seja,

$$v_{x_L}^* = v_{x_L}^{**} = v_{x_R}^{**} = v_{x_R}^* = S_M. \quad (5.17)$$

Em adição à Equação 5.17, a pressão total é mantida como constante

$$p_{T_L}^* = p_{T_L}^{**} = p_{T_R}^{**} = p_{T_R}^* = p_T^*. \quad (5.18)$$

A partir da escolha da onda  $S_M$ , obtém-se a igualdade como na Equação 5.18. De forma explícita,  $p_T^*$  pode ser escrita como

$$p_T^* = \frac{(S_R - v_{x_R})\rho_R p_{T_L} - (S_L - v_{x_L})\rho_L p_{T_R} + \rho_L \rho_R (S_R - v_{x_R})(S_L - v_{x_L})(v_{x_R} - v_{x_L})}{(S_R - v_{x_R})\rho_R - (S_L - v_{x_L})\rho_L}. \quad (5.19)$$

Dados  $S_M$  e  $p_T^*$ , os estados  $\mathbf{U}_\alpha^* = (\rho_\alpha^*, p_\alpha^*, v_{x_\alpha}^*, v_{y_\alpha}^*, v_{z_\alpha}^*, B_{x_\alpha}^*, B_{y_\alpha}^*, B_{z_\alpha}^*)$ , com  $\alpha = L$  ou  $\alpha = R$ , vizinhos aos estados  $\mathbf{U}_\alpha$ , são obtidos a partir dos saltos ao longo de  $S_\alpha$ ,

$$S_\alpha \begin{pmatrix} \rho_\alpha^* \\ E_\alpha^* \\ \rho_\alpha^* S_M \\ \rho_\alpha^* v_{y_\alpha}^* \\ \rho_\alpha^* v_{z_\alpha}^* \\ B_{y_\alpha}^* \\ B_{y_\alpha}^* \end{pmatrix} - \begin{pmatrix} \rho_\alpha^* S_M \\ (E_\alpha^* + p_T^*) S_M - B_x(\mathbf{u}_\alpha^* \cdot \mathbf{B}_\alpha^*) \\ \rho_\alpha^* S_M^2 + p_T^* - B_x^2 \\ \rho_\alpha^* v_{y_\alpha}^* S_M - B_x B_{y_\alpha} \\ \rho_\alpha^* v_{z_\alpha}^* S_M - B_x B_{z_\alpha} \\ B_{y_\alpha}^* S_M - B_x v_{y_\alpha}^* \\ B_{y_\alpha}^* S_M - B_x v_{z_\alpha}^* \end{pmatrix} = \begin{pmatrix} \rho_\alpha \\ E_\alpha \\ \rho_\alpha S_M \\ \rho_\alpha v_{y_\alpha} \\ \rho_\alpha v_{z_\alpha} \\ B_{y_\alpha} \\ B_{y_\alpha} \end{pmatrix} - \begin{pmatrix} \rho_\alpha v_{x_\alpha} \\ (E_\alpha + p_T) v_{x_\alpha} - B_x(\mathbf{u}_\alpha \cdot \mathbf{B}_\alpha) \\ \rho_\alpha v_{x_\alpha}^2 + p_T - B_x^2 \\ \rho_\alpha v_{y_\alpha} S_M - B_x B_{y_\alpha} \\ \rho_\alpha v_{z_\alpha} S_M - B_x B_{z_\alpha} \\ B_{y_\alpha} v_{x_\alpha} - B_x v_{y_\alpha} \\ B_{y_\alpha} v_{x_\alpha} - B_x v_{z_\alpha} \end{pmatrix}. \quad (5.20)$$

A partir da Equação 5.20, deriva-se as variáveis dos estados  $\mathbf{U}_\alpha^*$ .

$$\rho_\alpha^* = \rho_\alpha \frac{S_\alpha - v_{x_\alpha}}{S_\alpha - S_M}, \quad (5.21a)$$

$$v_{y_\alpha}^* = v_{y_\alpha} - B_x B_{y_\alpha} \frac{S_M - v_{x_\alpha}}{\rho_\alpha (S_\alpha - v_{x_\alpha})(S_\alpha - S_M) - B_x^2}, \quad (5.21b)$$

$$v_{z_\alpha}^* = v_{z_\alpha} - B_x B_{z_\alpha} \frac{S_M - v_{x_\alpha}}{\rho_\alpha (S_\alpha - v_{x_\alpha})(S_\alpha - S_M) - B_x^2}, \quad (5.21c)$$

$$B_{y_\alpha}^* = B_{y_\alpha} \frac{\rho_\alpha (S_\alpha - v_{x_\alpha})^2 - B_x^2}{\rho_\alpha (S_\alpha - v_{x_\alpha})(S_\alpha - S_M) - B_x^2}, \quad (5.21d)$$

$$B_{z_\alpha}^* = B_{z_\alpha} \frac{\rho_\alpha (S_\alpha - v_{x_\alpha})^2 - B_x^2}{\rho_\alpha (S_\alpha - v_{x_\alpha})(S_\alpha - S_M) - B_x^2}, \quad (5.21e)$$

Finalmente deduz-se a equação referente a  $E_\alpha^*$ , a qual depende das variáveis dos estados  $\mathbf{U}_\alpha^*$  deduzidas nas Equações em 5.21,

$$E_\alpha^* = \frac{(S_\alpha - v_{x_\alpha})E_\alpha - p_{T_\alpha} v_{x_\alpha} + p_T^* S_M + B_x (\mathbf{u}_\alpha \cdot \mathbf{B}_\alpha - \mathbf{u}_\alpha^* \cdot \mathbf{B}_\alpha^*)}{S_\alpha - S_M}. \quad (5.22)$$

De forma similar, é possível obter as equações referentes aos estados  $\mathbf{U}_\alpha^{**} = (\rho_\alpha^{**}, p_\alpha^{**}, v_{x_\alpha}^{**}, v_{y_\alpha}^{**}, v_{z_\alpha}^{**}, B_{x_\alpha}^{**}, B_{y_\alpha}^{**}, B_{z_\alpha}^{**})$ . A partir da condição de salto da equação de continuidade ao longo de um arbitrário  $S$ , em que  $S_L < S < S_M$  ou  $S_M < S < S_R$

$$\rho_\alpha^{**} = \rho_\alpha^*, \quad (5.23)$$

por causa da Relação 5.17.

As velocidades de propagação das ondas de Alfvén nos estados intermediários são estimadas por

$$S_L^* = S_M - \frac{|B_x|}{\sqrt{\rho_L^*}} \quad S_R^* = S_M - \frac{|B_x|}{\sqrt{\rho_R^*}}. \quad (5.24)$$

Deve-se considerar as condições de salto para as componentes tangenciais da veloci-

dade e do campo magnético, ao longo de  $S_M$

$$S_M \begin{pmatrix} \rho_L^* v_{y_L}^{**} \\ \rho_L^* v_{z_L}^{**} \\ B_{y_L}^{**} \\ B_{y_L}^{**} \end{pmatrix} - \begin{pmatrix} \rho_L^* v_{y_L}^{**} S_M - B_x B_{y_L}^{**} \\ \rho_L^* v_{z_L}^{**} S_M - B_x B_{z_L}^{**} \\ B_{y_L}^{**} S_M - B_x v_{y_L}^{**} \\ B_{y_L}^{**} S_M - B_x v_{z_L}^{**} \end{pmatrix} = S_M \begin{pmatrix} \rho_R^* v_{y_R}^{**} \\ \rho_R^* v_{z_R}^{**} \\ B_{y_R}^{**} \\ B_{y_R}^{**} \end{pmatrix} - \begin{pmatrix} \rho_R^* v_{y_R}^{**} S_M - B_x B_{y_R}^{**} \\ \rho_R^* v_{z_R}^{**} S_M - B_x B_{z_R}^{**} \\ B_{y_R}^{**} S_M - B_x v_{y_R}^{**} \\ B_{y_R}^{**} S_M - B_x v_{z_R}^{**} \end{pmatrix}. \quad (5.25)$$

A partir da Equação 5.25, se  $B_x \neq 0$ , obtém-se as seguintes relações

$$v_{y_L}^{**} = v_{y_R}^{**} \equiv v_y^{**}, \quad (5.26a)$$

$$v_{z_L}^{**} = v_{z_R}^{**} \equiv v_z^{**}, \quad (5.26b)$$

$$B_{y_L}^{**} = B_{y_R}^{**} \equiv B_y^{**}, \quad (5.26c)$$

$$B_{z_L}^{**} = B_{z_R}^{**} \equiv B_z^{**}. \quad (5.26d)$$

Caso  $B_x = 0$ , é impossível resolver a Equação 5.25. Essas relações mostram que  $v_y^{**}$ ,  $v_z^{**}$ ,  $B_y^{**}$  e  $B_z^{**}$  por três estados intermediários, como pode ser visto na Figura 5.4. Portanto, substituindo as Equações 5.23, 5.24, 5.26b e 5.26c em leis de conservação integrais ao longo do leque de Riemann,

$$(S_R - S_R^*)\mathbf{U}_R^* + (S_R^* - S_M)\mathbf{U}_R^{**} + (S_M - S_L^*)\mathbf{U}_L^{**} + (S_L^* - S_L)\mathbf{U}_L^* - S_R \mathbf{U}_R + S_L \mathbf{U}_L + \mathbf{F}_R - \mathbf{F}_L = 0, \quad (5.27)$$

deriva-se que

$$v_y^{**} = \frac{v_{y_L}^* + \sqrt{\rho_R^*} v_{y_R}^* + (B_{y_R}^* - B_{y_L}^*) \text{sign}(B_x)}{\sqrt{\rho_L^*} + \sqrt{\rho_R^*}}, \quad (5.28a)$$

$$v_z^{**} = \frac{\sqrt{\rho_L^*} v_{z_L}^* + \sqrt{\rho_R^*} v_{z_R}^* + (B_{z_R}^* - B_{z_L}^*) \text{sign}(B_x)}{\sqrt{\rho_L^*} + \sqrt{\rho_R^*}}, \quad (5.28b)$$

$$B_y^{**} = \frac{\sqrt{\rho_L^*} B_{y_R}^* + \sqrt{\rho_R^*} B_{y_L}^* + \sqrt{\rho_L^* \rho_R^*} (v_{y_R}^* - v_{y_L}^*) \text{sign}(B_x)}{\sqrt{\rho_L^*} + \sqrt{\rho_R^*}}, \quad (5.28c)$$

$$B_z^{**} = \frac{\sqrt{\rho_L^*} B_{z_R}^* + \sqrt{\rho_R^*} B_{z_L}^* + \sqrt{\rho_L^* \rho_R^*} (v_{z_R}^* - v_{z_L}^*) \text{sign}(B_x)}{\sqrt{\rho_L^*} + \sqrt{\rho_R^*}}, \quad (5.28d)$$

em que  $\text{sign}(B_x)$  é 1, para  $B_x > 0$ , e  $-1$ , para  $B_x < 0$ . Por fim, tem-se a energia no estado  $\mathbf{U}^{**}$ ,

$$E_\alpha^{**} = E_\alpha^* \mp \sqrt{\rho_\alpha^*}(\mathbf{u}_\alpha^* \cdot \mathbf{B}_\alpha^* - \mathbf{u}_\alpha^{**} \cdot \mathbf{B}_\alpha^{**})\text{sign}(B_x). \quad (5.29)$$

Dessa forma, deriva-se os estados  $\mathbf{U}_\alpha^*$  e  $\mathbf{U}_\alpha^{**}$ , possibilitando o cálculo dos fluxos correspondentes  $\mathbf{F}_\alpha^*$  e  $\mathbf{F}_\alpha^{**}$ .

O fluxo HLLD, em uma célula da malha, é dado pela equação

$$\mathbf{F}_{HLLD}^X = \begin{cases} \mathbf{F}_L, & \text{se } S_L > 0, \\ \mathbf{F}_L^*, & \text{se } S_L \leq 0 \leq S_L^*, \\ \mathbf{F}_L^{**}, & \text{se } S_L^* \leq 0 \leq S_M, \\ \mathbf{F}_R^{**}, & \text{se } S_M \leq 0 \leq S_R^*, \\ \mathbf{F}_R^*, & \text{se } S_R^* \leq 0 \leq S_R, \\ \mathbf{F}_R, & \text{se } S_R < 0, \end{cases}, \quad (5.30)$$

em que  $\mathbf{F}_K$  é o fluxo físico e

$$\begin{aligned} \mathbf{F}_K^* &= \mathbf{F}_K + S_K (\mathbf{U}_K^* - \mathbf{U}_K) \\ \mathbf{F}_K^{**} &= \mathbf{F}_K + S_K^* \mathbf{U}_K^{**} - (S_K^* - S_K) \mathbf{U}_K^* - S_K \mathbf{U}_K, \end{aligned} \quad (5.31)$$

com  $K = R$  ou  $K = L$ , denotando direita e esquerda (do inglês *left* e *right*). Além disso, tem-se os limitadores

$$S_M = \frac{(S_R - v_{x_R})\rho_R v_{x_R} - (S_L - v_{x_L})\rho_L v_{x_L} - p_{T_L} + p_{T_R}}{(S_R - v_{x_R})\rho_R - (S_L - v_{x_L})\rho_L}, \quad (5.32)$$

$$S_L = \min(u_L, u_R) - \min(c_L, c_R), \quad S_R = \max(u_L, u_R) + \max(c_L, c_R), \quad (5.33a)$$

em que  $p_T$  é a pressão total, dada por  $p_T = p + \frac{\mathbf{B} \cdot \mathbf{B}}{2}$ . Esses valores relacionados ao fluxo HLLD são calculados dessa forma para a direção  $x$ . Para as outras direções, o cálculo do fluxo é feito de forma análoga. Detalhes da construção desse fluxo numérico são encontrados em Miyoshi e Kusano (2005), onde são especificadas as variáveis dos estados intermediários e a obtenção das mesmas. O fluxo numérico HLLD também é calculado direcionalmente neste trabalho. Essa generalização é análoga à feita para o fluxo HLL.



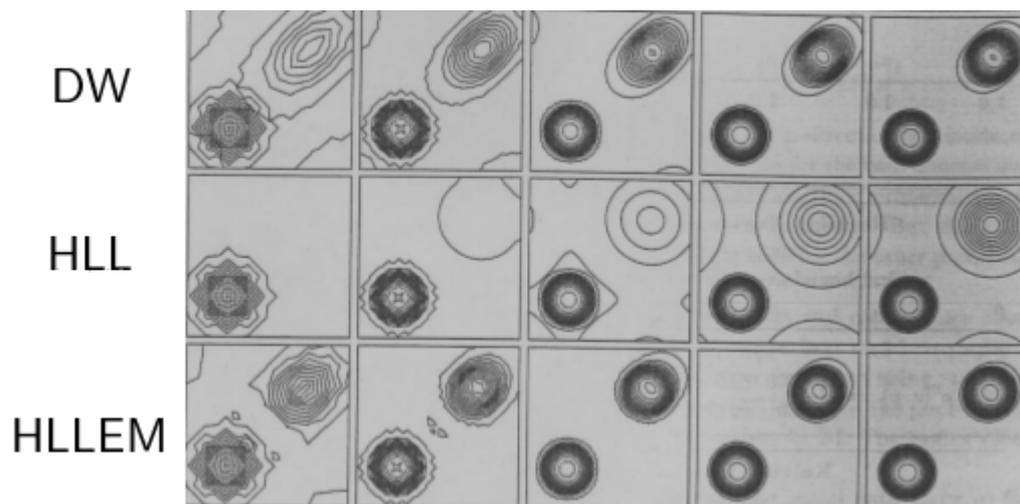


Figura 5.5 - Componente  $z$  do campo magnético, com  $0.0 < t \leq 1.0$ , para os fluxos numéricos DW, HLL e HLLEM (de cima para baixo), com malhas triangulares variando de 256 a 65536 triângulos (da esquerda para a direita).

Fonte: Wesenberg (2003).

Nos próximos capítulos são apresentados a abordagem multirresolução adaptativa utilizada e os resultados obtidos com e sem essa abordagem, utilizando os Riemann solvers HLL e HLLD, como descritos nesse capítulo.



## 6 FV: VERIFICAÇÃO E RESULTADOS

O código utilizado neste trabalho chama-se MHD-FV, o qual foi implementado em  $C^{++}$  ao longo deste trabalho, e inspirado no código desenvolvido em Fortran90 por Pierro (2009), baseado nas idéias apresentadas em Dedner et al (2002). O código MHD-FV calcula o modelo EGLM-MHD utilizando uma discretização por volumes finitos. Também foram implementados nesse código, os Riemann solvers HLL e HLLD. Para que seja possível obter a solução do modelo, utiliza-se um algoritmo como apresentado na Figura 6.1.

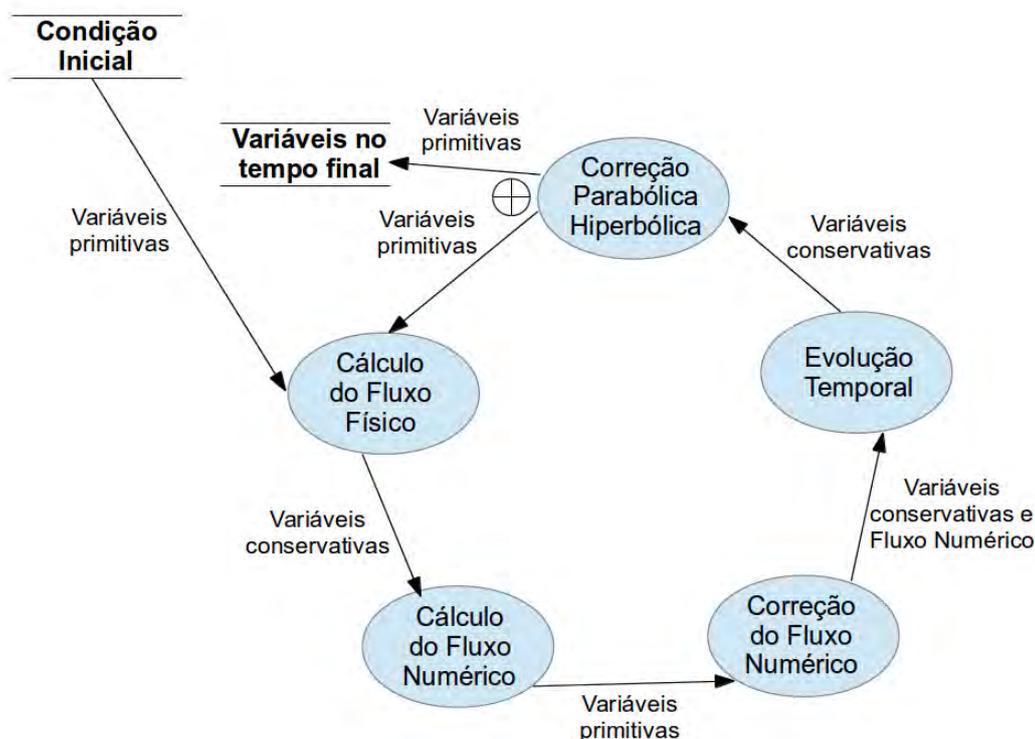


Figura 6.1 - Algoritmo utilizado para obter a solução do modelo com o código MHD-FV.

A Figura 6.1 descreve de forma simples como o algoritmo utilizado funciona para um determinado tempo  $t_n$ . A condição inicial, é a entrada necessária para realizar todos

os cálculos. Os círculos em azul são os processos necessários para o cálculo da solução numérica. As setas, em preto, indicam o fluxo de dados que está entrando dentro de um determinado processo. O símbolo  $\oplus$  representa um *ou exclusivo*, ou seja, um dos dois fluxos de dados será escolhido, mas nunca os dois. Ao final do algoritmo, obtém-se os valores das variáveis do modelo para o tempo  $t_{n+1}$ . Enquanto o tempo final não for atingido, os valores obtidos entram no processo *Cálculo do Fluxo Físico* e todo o algoritmo é repetido novamente, como é possível observar na figura.

Após a implementação do código, com os dois diferentes Riemann solvers HLL e HLLD, iniciou-se a etapa de verificação, para garantir que a solução obtida numericamente está se aproximando corretamente da solução esperada. Nessa etapa, são testadas duas condições iniciais conhecidas como tipo Riemann 1D e tipo Riemann 2D, como as utilizadas em Li (2005) e Dedner et al (2002). A CPU utilizada tem as seguintes especificações: processador Intel<sup>(R)</sup> Core<sup>(TM)</sup> i7 2.67GHz; 8Mb de memória cache; e 8Gb de memória RAM. Já o ambiente operacional utilizado é o GNU/LINUX (Ubuntu 12.04) e o compilador  $g^{++}$  4.6.3.

## 6.1 Caso 1: Condição Inicial tipo Riemann 1D

### 6.1.1 Verificação

Dois testes básicos são utilizados para a verificação do funcionamento dos fluxos. Em um deles, obtém-se a solução do modelo EGLM-MHD na direção  $x$ , mantendo constante as variáveis na direção  $y$ . No outro, faz-se um procedimento similar considerando o eixo  $y$  com as variações e mantendo-se as variáveis constantes em  $x$ . Um esquema das variações das variáveis nesse caso é apresentado na Figura 6.2.

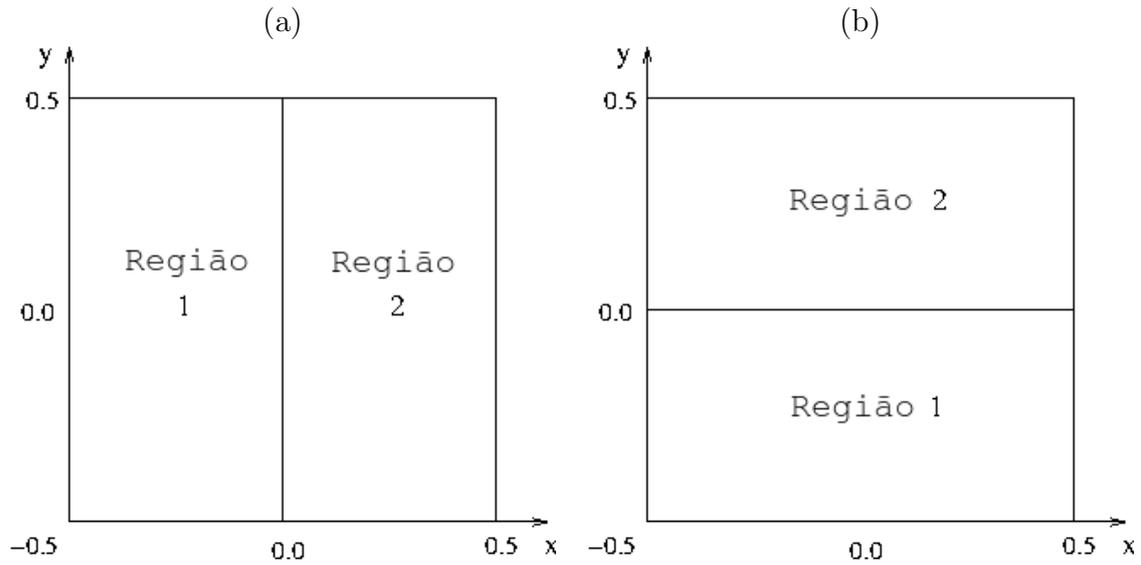


Figura 6.2 - Condição inicial tipo Riemann 1D: esquema das variações das variáveis. (a) variando em  $x$  e constante em  $y$ ; (b) constante em  $x$  e variando em  $y$ .

No caso da condição inicial tipo Riemann 1D, fez-se a verificação comparando a solução aproximada obtida com uma solução exata<sup>1</sup> (TORRILHON, 2003), como a apresentada no Capítulo 3. É importante ressaltar que, mesmo utilizando condições iniciais tipo Riemann 1D, o código calcula a solução bidimensional do modelo, ou seja, o fluxo é calculado nas direções  $x$  e  $y$ . Neste experimento, utiliza-se a condição inicial tipo Riemann 1D proposta em Li (2005) (Tabela 3.1), juntamente com a condição de contorno *Absorption Neumann*, como descrita no Apêndice C.

Para fins de ilustração, na Figura 6.3 são apresentadas as soluções aproximadas obtidas com os fluxos HLL, HLLD e Roe em Miyoshi e Kusano (2005). Neste exemplo, é utilizada uma condição inicial tipo Riemann 1D similar à utilizada neste trabalho, como na Tabela 3.1, exceto que lá os valores  $B_x = 4/\sqrt{4\pi}$  e  $B_z = 2/\sqrt{4\pi}$ , a malha unidimensional possui 800 pontos,  $c_{CFL} = 0.8$  e  $t = 0.2$ .

<sup>1</sup><https://web.mathcces.rwth-aachen.de/mhdsolver/>

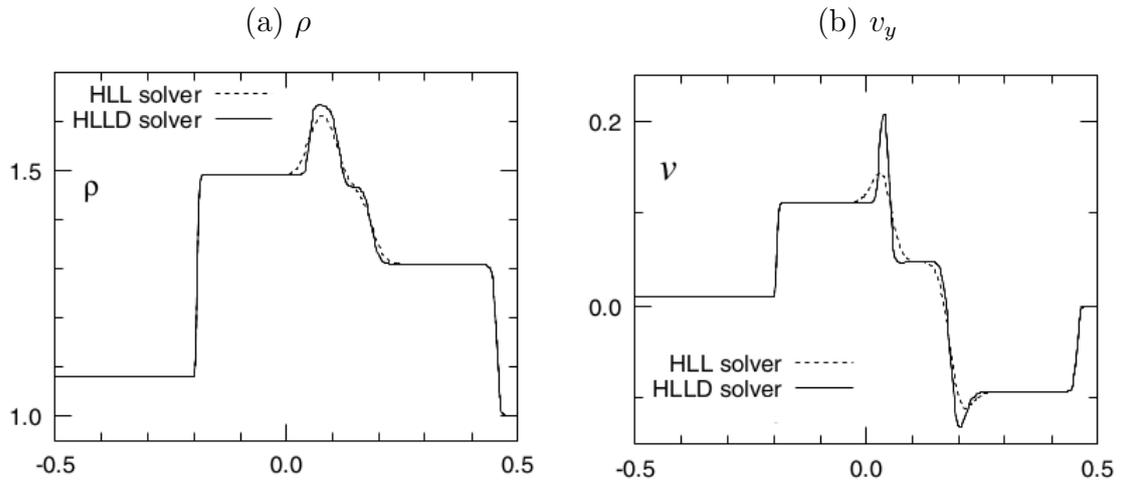


Figura 6.3 - Resultados do Caso 1 unidimensional obtidos para uma condição inicial tipo Riemann 1D com os fluxos HLL e HLLD. As variáveis densidade (a) e componente  $y$  da velocidade (b) foram estimadas com uma malha de 800 pontos de grade no intervalo  $[-0.5, 0.5]$ , com  $c_{CFL} = 0.8$  e  $t = 0.2$ . Fonte: Miyoshi e Kusano (2005).

A condição inicial utilizada em Miyoshi e Kusano (2005) difere por duas variáveis e há uma diferença de parâmetros, mas existe uma similaridade com a solução aproximada obtida neste trabalho, como pode ser visto nas Figuras 6.4 e 6.5. O maior interesse nessa comparação é o comportamento dos Riemann solvers para essas variáveis.

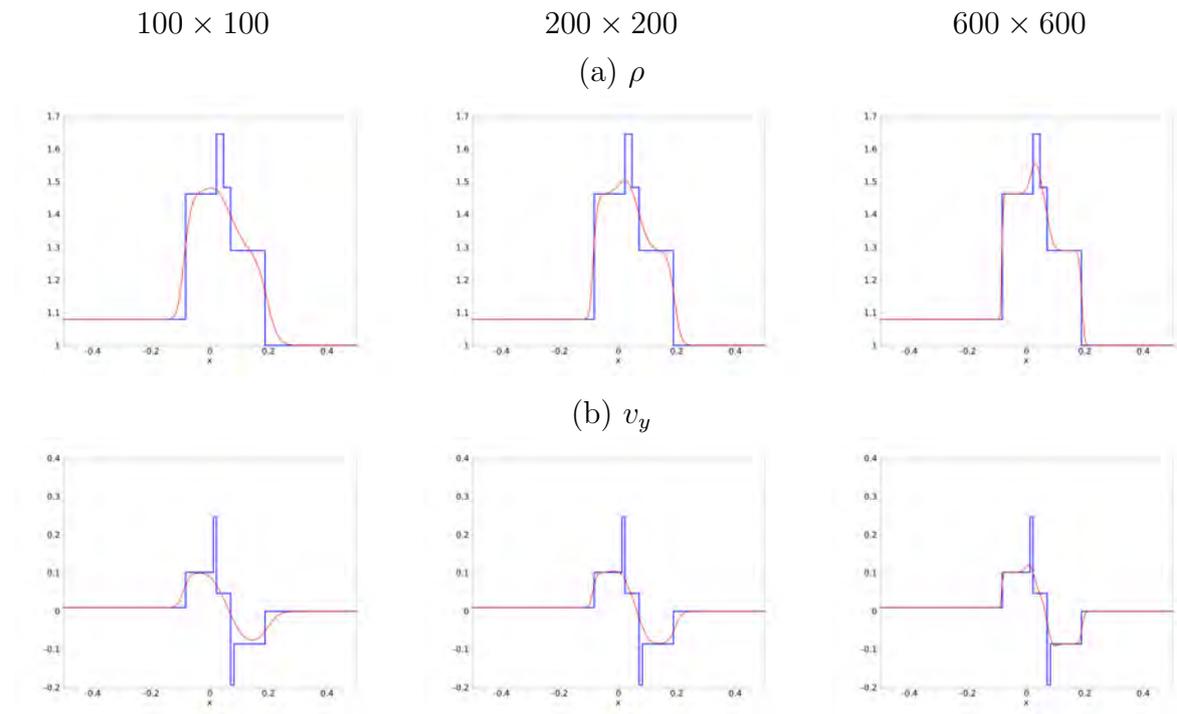


Figura 6.4 - Variáveis densidade (a) e componente  $y$  da velocidade (b) estimadas com o fluxo HLL para o tempo  $t = 0.08$ ,  $c_{CFL} = 0.3$  e para malhas de  $100 \times 100$ ,  $200 \times 200$  e  $600 \times 600$  células. Soluções exata (azul) e aproximada (vermelho).

Nas figuras apresentadas, tem-se a solução aproximada (vermelho) e exatas (azul) das variáveis densidade ( $\rho$ ) e componente  $y$  da velocidade ( $v_y$ ), calculadas com malhas regulares com número de células diferentes ( $100 \times 100$ ,  $200 \times 200$  e  $600 \times 600$  células.) para os fluxos numéricos HLL e HLLD.

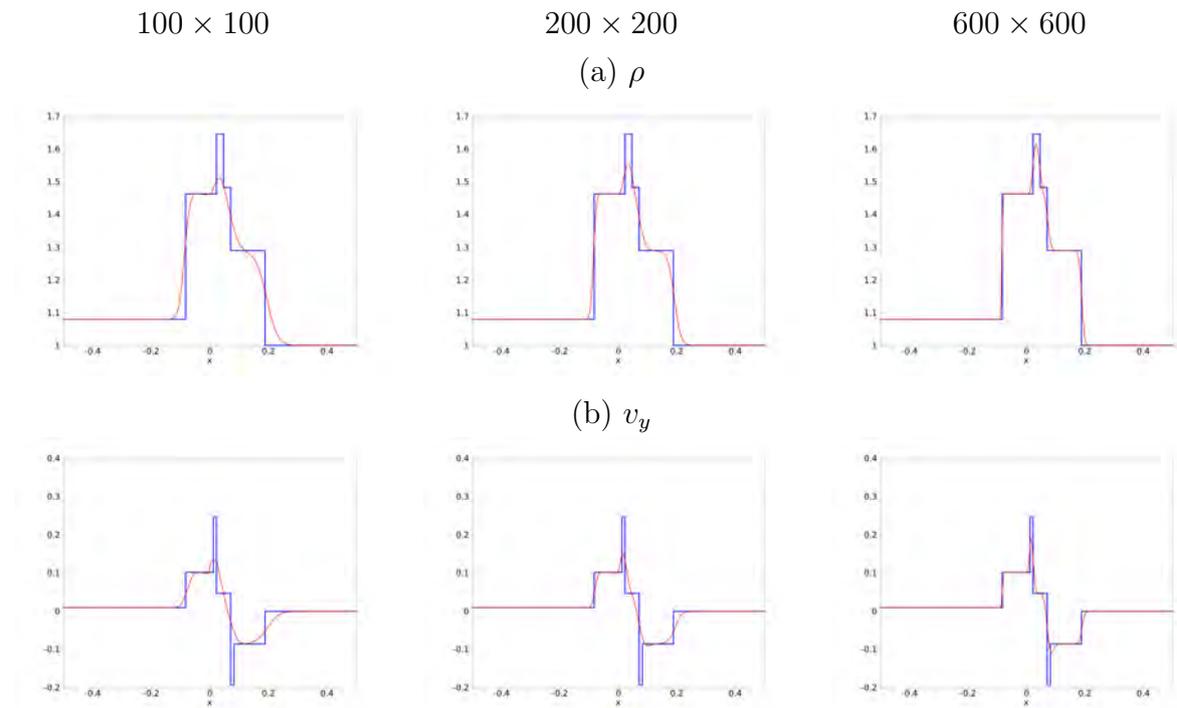


Figura 6.5 - Variáveis densidade (a) e componente  $y$  da velocidade (b) estimadas com o fluxo HLLD para o tempo  $t = 0.08$ ,  $c_{CFL} = 0.3$  e para malhas de  $100 \times 100$ ,  $200 \times 200$  e  $600 \times 600$  células. Soluções exata (azul) e aproximada (vermelho).

Pelas Figuras 6.4 e 6.5 é possível notar as diferenças entre a solução das variáveis densidade  $\rho$  e componente  $y$  da velocidade  $v_y$  para diferentes tamanhos de malha. À medida que a malha é mais refinada, pode-se concluir que a solução tende a ser melhorada. No entanto, a qualidade de uma aproximação numérica não está apenas relacionada ao refinamento da malha, mas, também, ao esquema numérico que é utilizado para aproximar a solução. Para ilustrar esse tipo de abordagem, utilizando os exemplos anteriores, compara-se as duas variáveis para os dois diferentes tipos de fluxos numéricos implementados.

### 6.1.2 Resultados

Na Figura 6.6 é possível visualizar de forma clara o comportamento de ambos os fluxos numéricos para diferentes tamanhos de malhas. Observa-se, então, que mesmo em uma malha bem menos refinada, o fluxo numérico HLLD se mostra mais eficiente, no sentido de aproximação da solução numérica do modelo MHD. Desse forma, fica

claro que a escolha do esquema numérico é de grande importância, devendo ser feita de acordo com o problema e o objetivo em questão.

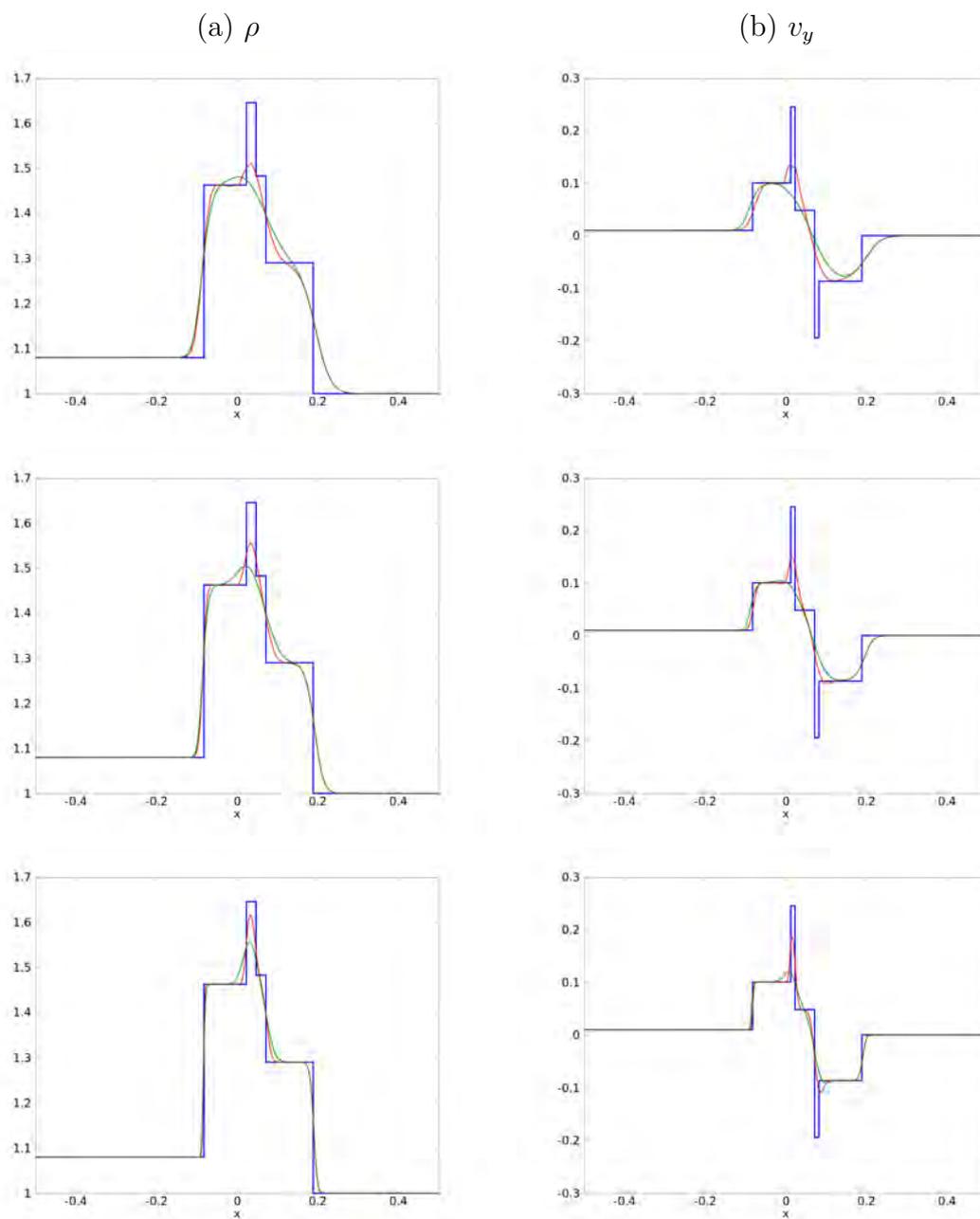


Figura 6.6 - Variáveis densidade (a) e componente  $y$  da velocidade (b) para malhas de  $100 \times 100$  (superior),  $200 \times 200$  (centro) e  $600 \times 600$  (inferior) células no tempo  $t = 0.08$  e  $c_{CFL} = 0.3$ . Soluções exata (azul), aproximada com o fluxo HLL (verde) e aproximada com o fluxo HLLD (vermelho).

Nas Figuras 6.7, 6.8 e 6.9, apresenta-se a solução numérica do modelo EGLM-MHD obtida com os fluxos HLL e HLLD para todas as variáveis no caso da condição inicial do tipo Riemann 1D. Essa solução aproximada é obtida para uma malha regular de  $512 \times 512$  células, em um tempo  $t = 0.1$  e  $c_{CFI} = 0.3$ .

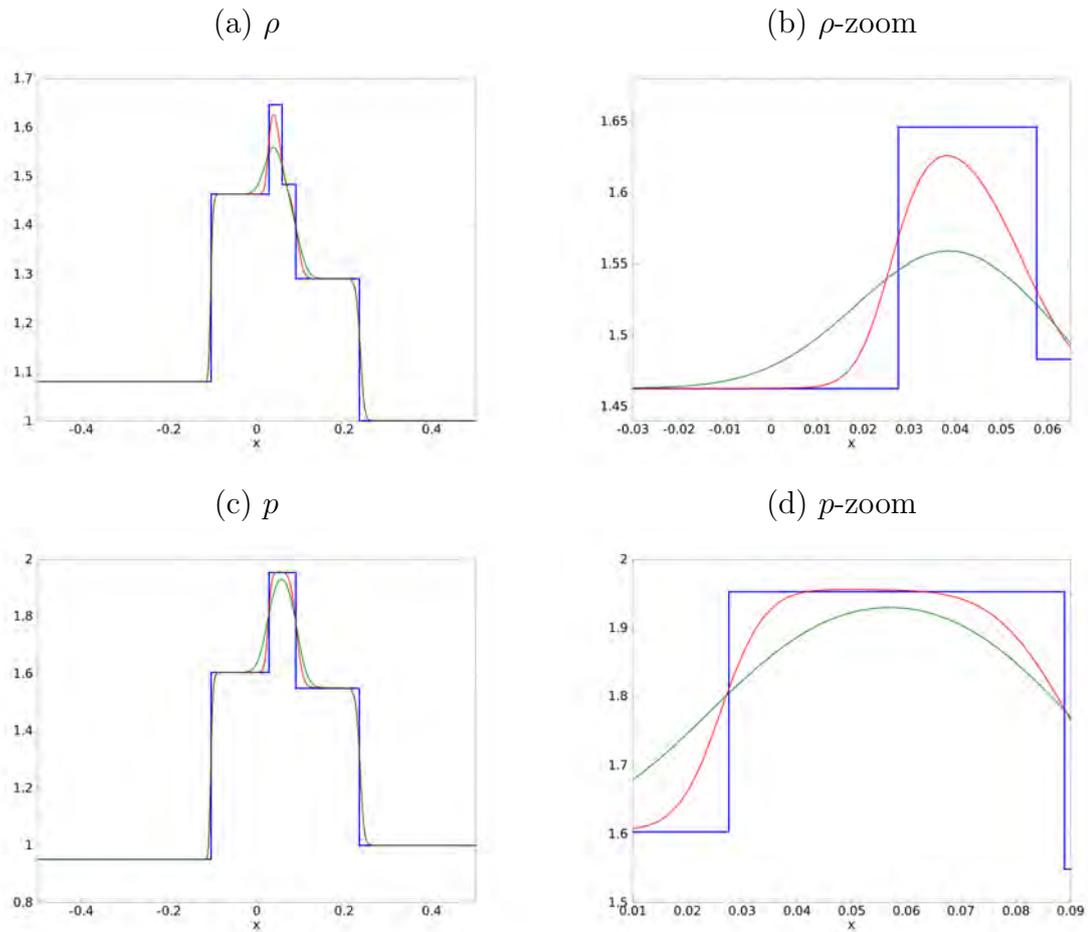


Figura 6.7 - Densidade (a) e pressão (c), e seus respectivos zoom's (b) e (d): solução exata (azul), HLL (verde) e HLLD (vermelho) para o tempo  $t = 0.08$ ,  $c_{CFI} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e uma malha com  $512 \times 512$  células.

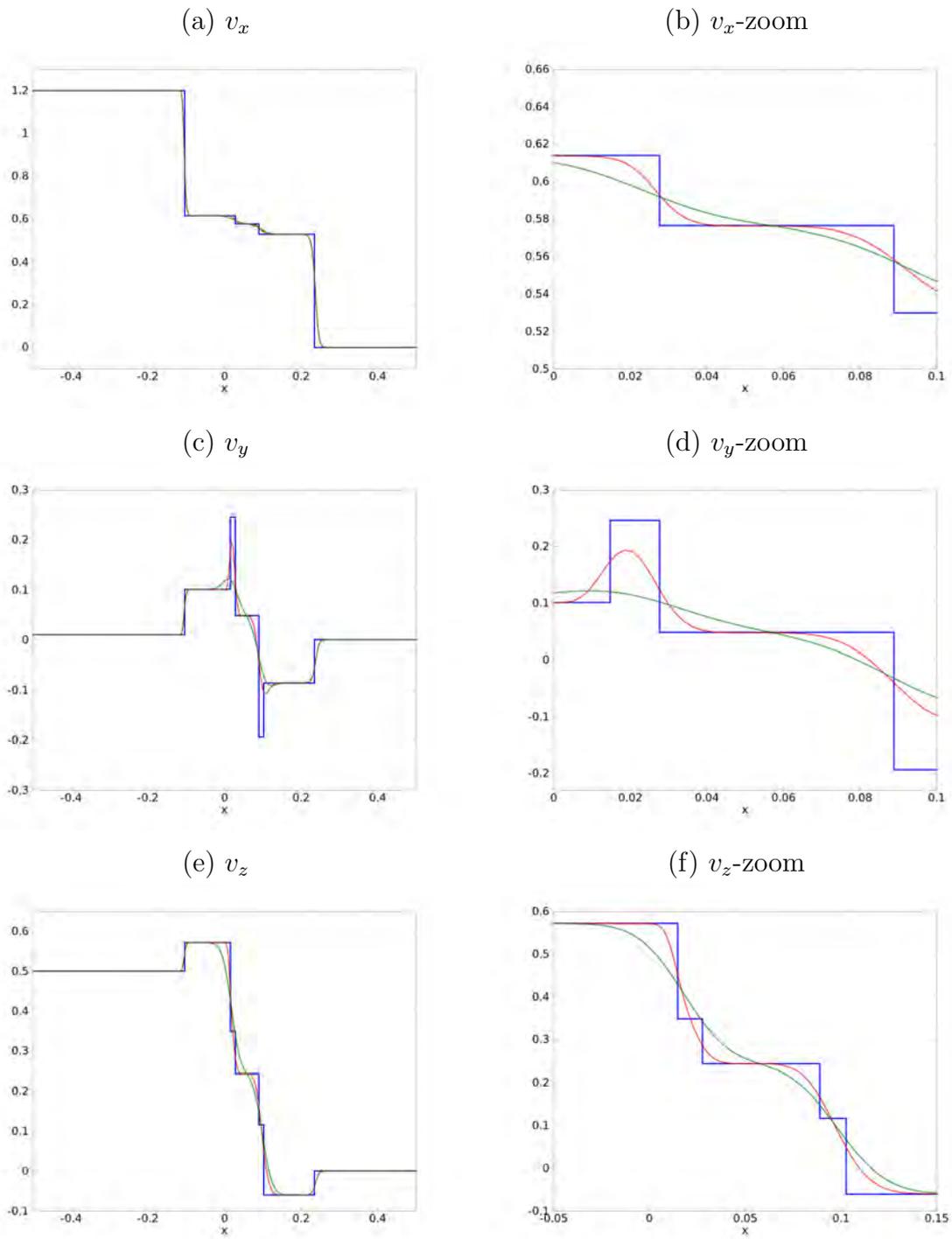


Figura 6.8 - Componente  $x$  (a),  $y$  (c) e  $z$  (f) da velocidade e seus respectivos zoom's (b), (d) e (f): solução exata (azul), HLL (verde) e HLLD (vermelho) para o tempo  $t = 0.08$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e uma malha com  $512 \times 512$  células.

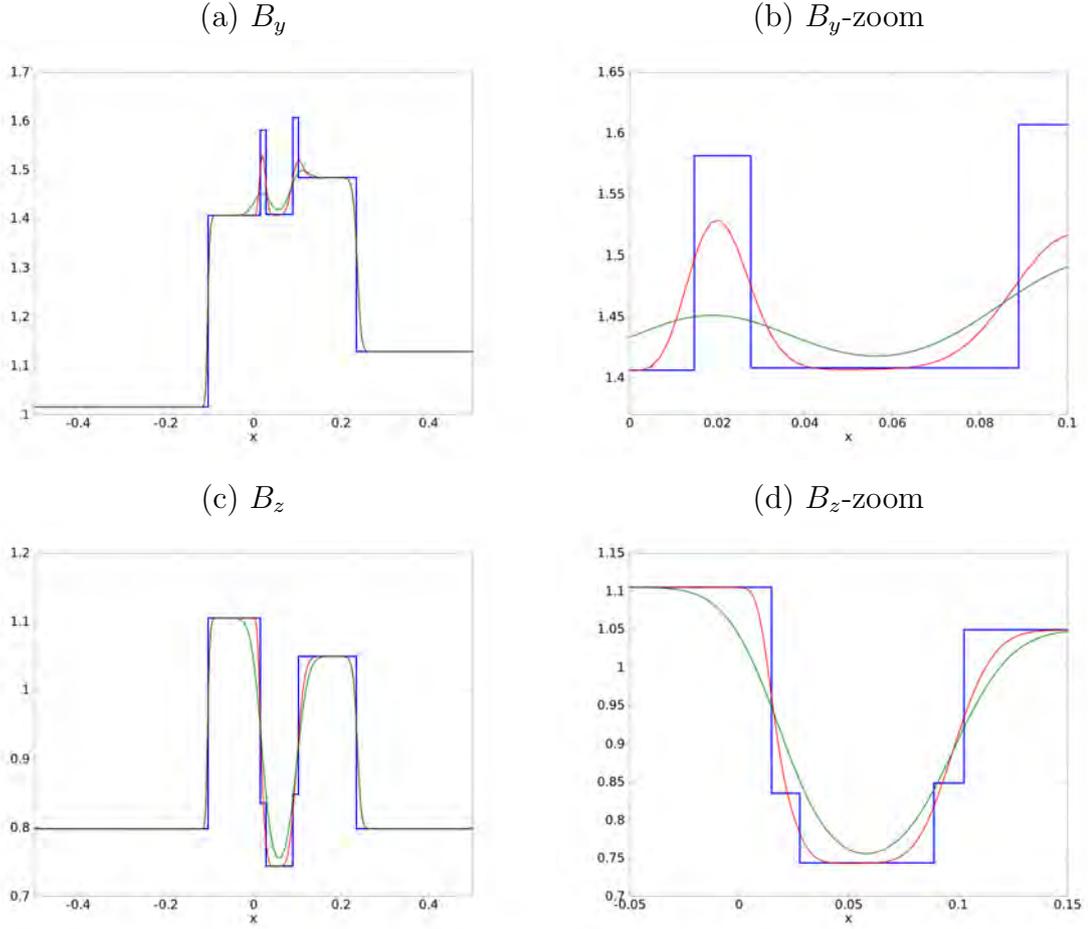


Figura 6.9 - Componentes  $y$  (a) e  $z$  (c) do campo magnético, e seus respectivos zoom's (b) e (d): solução exata (azul), HLL (verde) e HLLD (vermelho) para o tempo  $t = 0.08$ ,  $c_{CF L} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e uma malha com  $512 \times 512$  células.

A Tabela 6.1 mostra o erro na norma  $L^1$  calculada entre as soluções exata e aproximada, para os fluxos HLL e HLLD. O erro na norma  $L^1$  é calculado como na Equação 6.1.

$$e_{L^1} = \frac{1}{N_x N_y} D_x D_y \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} (\mathbf{U}_{ref} - \mathbf{U}_{num}), \quad (6.1)$$

em que  $N_x$  e  $N_y$  são as quantidades de células nas direções  $x$  e  $y$ ;  $D_x$  e  $D_y$  são os domínios  $x$  e  $y$ ; e  $\mathbf{U}_{ref}$  e  $\mathbf{U}_{num}$  são as soluções de referência e numérica.

Analisando os dados da tabela, percebe-se uma melhora no erro de aproximação para todas as variáveis, exceto para a variável  $B_x$ , a qual é constante. Constatando,

numericamente, a eficiência de cada fluxo ao aproximar a solução.

Tabela 6.1 - Erro na norma  $L_1$  calculada entre as soluções exata e aproximada, para os fluxos HLL e HLLD.

Fluxo	Variáveis			
	$\rho$	$p$	$v_x$	$v_y$
HLL	$10.3 \cdot 10^{-03}$	$17.2 \cdot 10^{-03}$	$6.70 \cdot 10^{-03}$	$7.08 \cdot 10^{-03}$
HLLD	$7.43 \cdot 10^{-03}$	$11.5 \cdot 10^{-03}$	$5.99 \cdot 10^{-03}$	$5.28 \cdot 10^{-03}$
	$v_z$	$B_x$	$B_y$	$B_z$
HLL	$7.49 \cdot 10^{-03}$	$4.16 \cdot 10^{-07}$	$9.91 \cdot 10^{-03}$	$10.5 \cdot 10^{-03}$
HLLD	$4.01 \cdot 10^{-03}$	$4.16 \cdot 10^{-07}$	$7.76 \cdot 10^{-03}$	$6.26 \cdot 10^{-03}$

Uma importante etapa de verificação do código MHD-FV é a aproximação da solução na direção  $y$ , como ilustrado na Figura 6.2. A condição inicial para esse caso é descrita na Tabela 6.2 e utilizada de acordo com os valores de  $y$

Tabela 6.2 - Condição inicial tipo Riemann 1D - Direção  $y$

	$\rho$	$p$	$v_x$	$v_y$	$v_z$	$B_x$	$B_y$	$B_z$
$y \leq 0$	1.08	0.95	0.01	1.2	0.5	$3.6/\sqrt{4\pi}$	$2.0/\sqrt{4\pi}$	$2.0/\sqrt{2\pi}$
$y > 0$	1.0	1.0	0.0	0.0	0.0	$4.0/\sqrt{4\pi}$	$2.0/\sqrt{4\pi}$	$2.0/\sqrt{2\pi}$

e os valores  $v_x$ ,  $v_y$  e  $B_x$ ,  $B_y$  trocam de posição entre si ( $v_x$  por  $v_y$  e  $B_x$  por  $B_y$ ), para que seja possível verificar se os resultados na direção  $y$  são iguais aos da direção  $x$ . Feito isso, os resultados obtidos são idênticos, verificando, assim, o código MHD em volumes finitos.

A verificação do código MHD-FV em ambas as direções  $x$  e  $y$  para o caso da condição inicial tipo Riemann 1D, possibilitou o início da etapa de verificação e resultados para a condição inicial tipo Riemann 2D.

## 6.2 Caso 2: Condição Inicial Tipo Riemann 2D

Após as validações do código para a condição inicial 1D, inicia-se os testes para obtenção da solução do modelo EGLM-MHD para a condição inicial tipo Riemann 2D, o qual é um caso mais interessante. Para fins de comparação e verificação, a condição inicial utilizada foi a proposta em Dedner et al (2002).

Tabela 6.3 - Condição inicial tipo Riemann 2D

Quadrante	$\rho$	$p$	$v_x$	$v_y$	$v_z$	$B_x$	$B_y$	$B_z$
1	1.0304	2.2874	1.4127	-1.0146	-1.0691	0.3501	0.5078	0.1576
2	1.0000	2.4323	1.7500	-1.0000	0.0000	0.5642	0.5078	0.2539
3	1.8887	7.6110	0.1236	-0.9224	0.0388	0.5642	0.9830	0.4915
4	0.9308	2.1583	1.5639	-0.4977	0.0618	0.3501	0.9830	0.3050

A Tabela 6.3 da condição inicial tipo Riemann 2D foi adaptada para os valores das quantidades primitivas, ao contrário da tabela apresentada em Dedner et al (2002), que apresenta os valores na forma conservativa. O conceito da condição inicial tipo Riemann 2D é análogo ao da condição inicial tipo Riemann 1D, mas nas duas direções  $x$  e  $y$ , particionando-se o domínio. Na Figura 6.10, tem-se uma ilustração para a condição inicial tipo Riemann 2D.

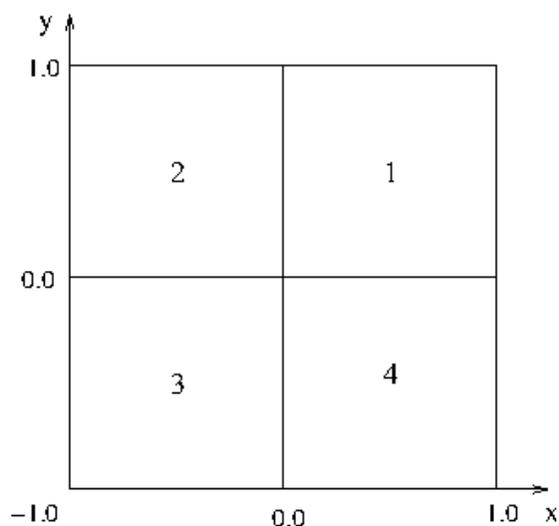


Figura 6.10 - Condição inicial tipo Riemann 2D em um domínio  $[-1, 1] \times [-1, 1]$ . Os valores 1, 2, 3 e 4 denotam os quadrantes, os quais são especificados na Tabela 6.3

A solução de referência é calculada utilizando-se uma malha com  $2048 \times 2048$  células. Para ilustrar a solução encontrada em Dedner et al (2002), é apresentada na Figura 6.11 a variável densidade.

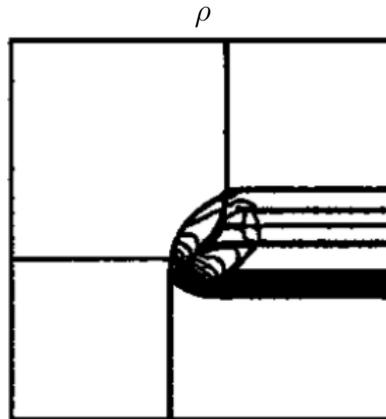


Figura 6.11 - Solução para uma condição inicial tipo Riemann 2D, para a variável densidade (a), no tempo  $t = 0.1$ . A solução foi aproximada utilizando o Riemann solver DW (DAI; WOODWARD, 1995) em um domínio  $[-1, 1] \times [-1, 1]$ . Fonte: Dedner et al (2002).

Nas Figuras 6.12, 6.13 e 6.14, utiliza-se uma malha com  $512 \times 512$  células, para simular a solução para o Caso 2. Ambos os fluxos numéricos HLL e HLLD são utilizados. O valores  $c_{CFL} = 0.3$  e  $\gamma = 5/3$  também são definidos nesta simulação.

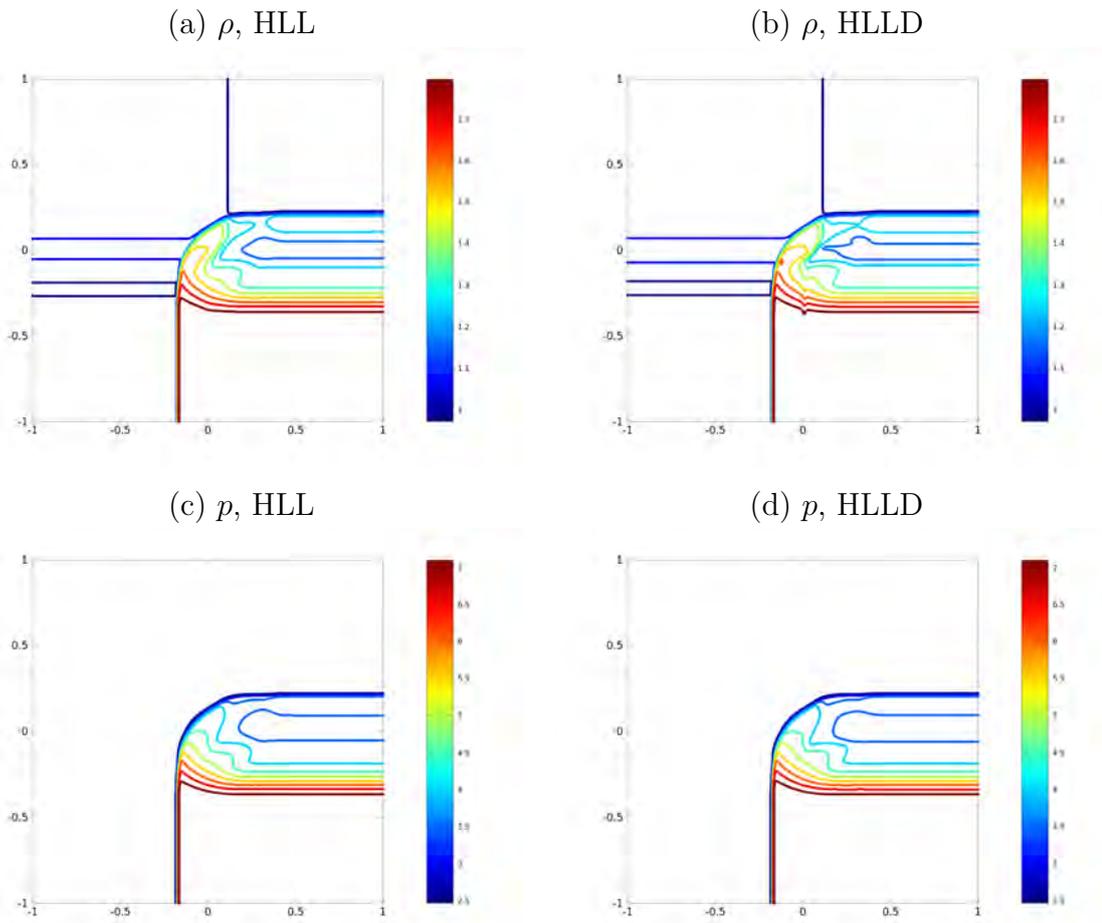


Figura 6.12 - Variáveis densidade e pressão para os fluxos HLL (a,c) e HLLD (b,d), respectivamente, em uma malha com  $512 \times 512$  células, para tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $c_{CFL} = 0.3$ .

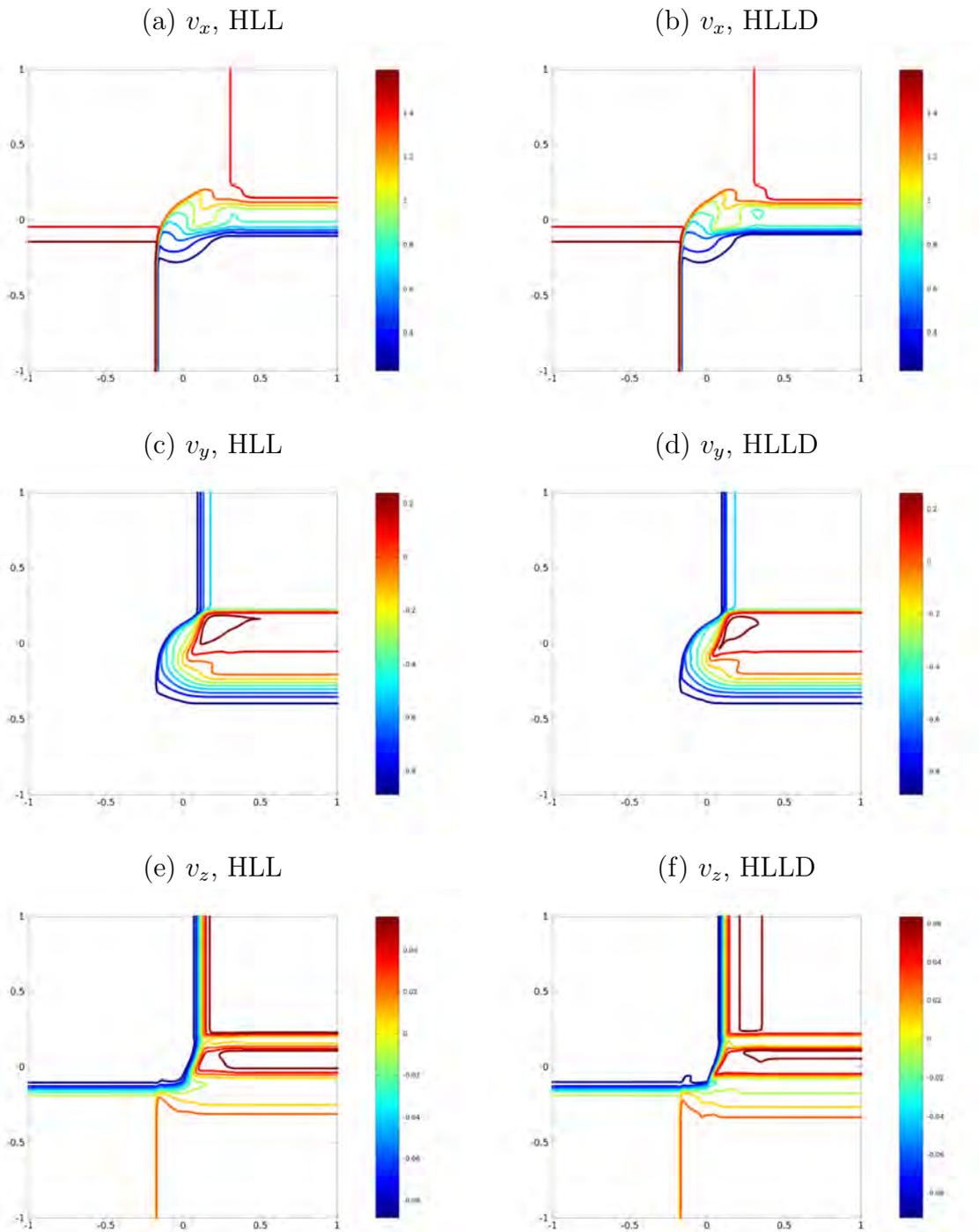


Figura 6.13 - Componentes  $x$ ,  $y$ ,  $z$  da velocidade para os fluxos HLL (a,c,e) e HLLD (b,d,f), respectivamente, em uma malha com  $512 \times 512$  células, para tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $c_{CFL} = 0.3$ .

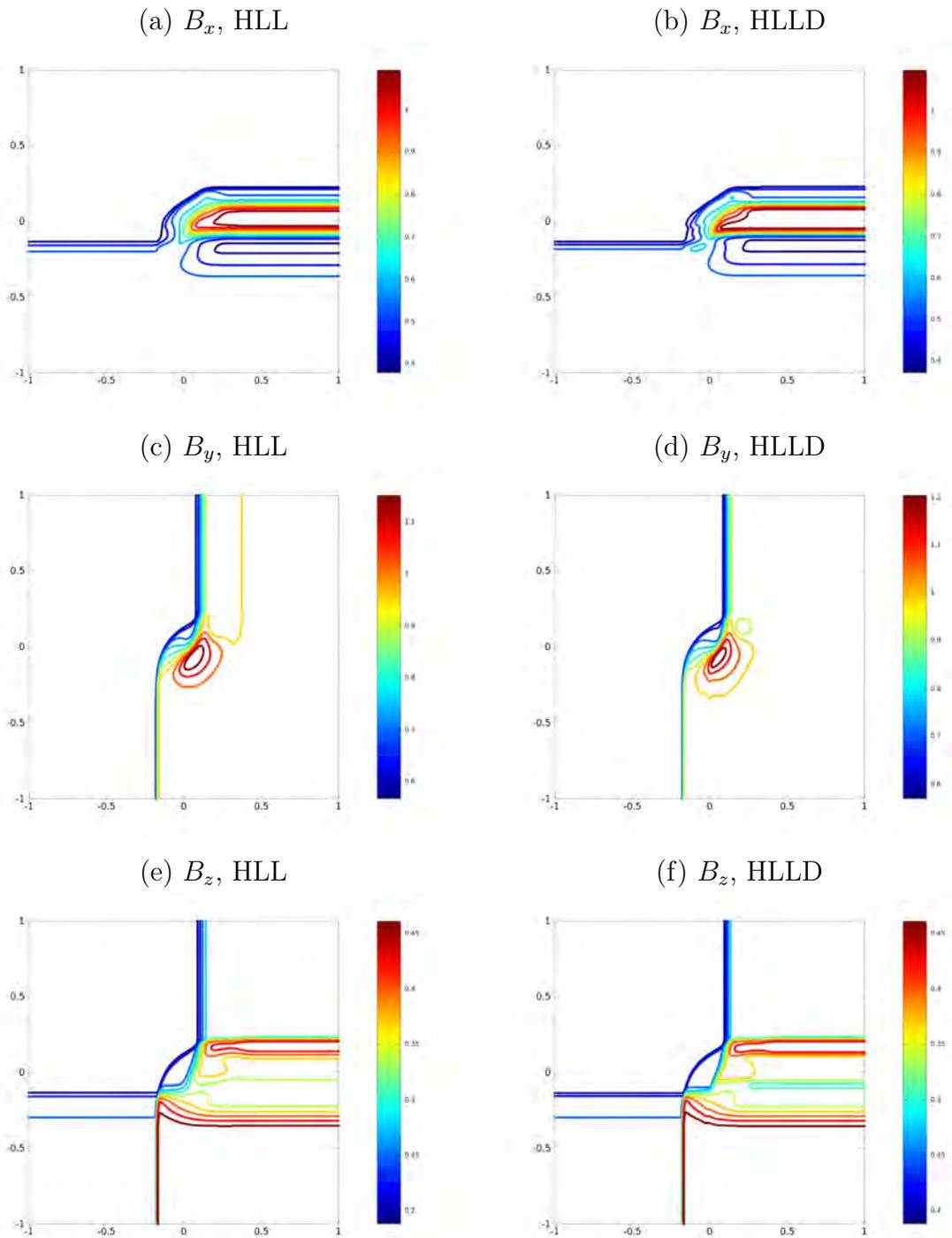


Figura 6.14 - Componentes  $x$ ,  $y$ ,  $z$  do campo magnético para os fluxos HLL (a,c,e) e HLLD (b,d,f), respectivamente, em uma malha com  $512 \times 512$  células, para tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $CCFL = 0.3$ .

Na Tabela 6.4, tem-se o erro na norma  $L^1$  calculada pela diferença dos valores de teste e de referência. Para obter os valores de referência, simula-se a solução do modelo EGLM-MHD para Caso 2, com o código MHD-FV, para uma malha de  $2048 \times 2048$  células, e utiliza-se essa solução para obter uma aproximação dos valores para uma malha de  $512 \times 512$  células.

Tabela 6.4 - Erro na norma  $L_1$  calculada entre as soluções de referência e aproximada, para os fluxos HLL e HLLD. A solução de referência é calculada com o fluxo HLLD.

Fluxo	Variáveis			
	$\rho$	$p$	$v_x$	$v_y$
HLL	$2.17 \cdot 10^{-02}$	$10.35 \cdot 10^{-02}$	$3.67 \cdot 10^{-02}$	$2.83 \cdot 10^{-02}$
HLLD	$1.77 \cdot 10^{-02}$	$8.33 \cdot 10^{-02}$	$2.12 \cdot 10^{-02}$	$2.27 \cdot 10^{-02}$
	$v_z$	$B_x$	$B_y$	$B_z$
HLL	$8.12 \cdot 10^{-03}$	$3.19 \cdot 10^{-02}$	$11.55 \cdot 10^{-03}$	$11.23 \cdot 10^{-03}$
HLLD	$5.12 \cdot 10^{-03}$	$1.67 \cdot 10^{-02}$	$8.04 \cdot 10^{-03}$	$8.02 \cdot 10^{-03}$

A solução obtida em Dedner et al (2002) para essa condição inicial tipo Riemann 2D foi calculada com um fluxo numérico diferente do utilizado nesse trabalho, mas no mesmo tempo  $t = 0.1$ , com as mesmas condições iniciais e parâmetros. Além disso, não foi especificada pelo autor as isolinhas utilizadas na visualização da solução. Por esses motivos, a solução obtida neste trabalho não é, necessariamente, idêntica à solução obtida em Dedner et al (2002). No entanto, existe uma grande similaridade entre as diferentes soluções, o que é um ponto importante para a verificação preliminar do código MHD-FV para o caso 2D. As diferenças existentes entre as soluções obtidas neste trabalho e em Dedner et al (2002) podem ser relacionadas à escolha do fluxo numérico para obtenção da solução e, também, ao tipo de malha utilizada, bem como as isolinhas utilizadas na visualização.

No próximo capítulo são apresentados os resultados obtidos utilizando um algoritmo de multirresolução adaptativa para os casos 1D e 2D e, também, um comparativo entre as soluções simuladas com e sem esse algoritmo.



## 7 MULTIRRESOLUÇÃO (MR)

Num contexto multi-escala, a informação é organizada em diferentes níveis de escalas  $0 \leq j \leq J$ . De uma forma geral, em um certo nível  $j$ , o dado discreto  $\mathbf{f}_j = (f_{j,k})$  é um conjunto de amostras  $f_{j,k}$  da função  $f$ . Na análise wavelet, os detalhes de uma representação em dois níveis consecutivos  $j$  e  $j + 1$  são guardados nos coeficientes wavelet  $\mathbf{d}_j = (d_{j,k})$ . O índice  $k$  está associado com a localização espacial, onde as amostras ou os detalhes são obtidos. Por exemplo, essas amostras podem ser obtidas por pontos (ou por médias celulares) em malhas hierárquicas e os detalhes  $d_{j,k}$  são os erros obtidos no nível  $j + 1$ , utilizando uma previsão dos valores  $\mathbf{f}_j$  de um nível menos refinado  $j$ . Nesse contexto, a principal ferramenta de multirresolução é uma transformada um-para-um entre níveis

$$\mathbf{f}_{j+1} \Leftrightarrow (\mathbf{f}_j, \mathbf{d}_j).$$

Aplicando a transformada direta em uma informação  $\mathbf{f}_{j+1}$  em um nível mais refinado  $j + 1$ , tem-se a informação no nível menos refinado a seguir  $\mathbf{f}_j$  e os detalhes (ou erros de aproximação)  $\mathbf{d}_j$ .

Nesta seção, é apresentado um resumo sobre o método de análise multirresolução no contexto de interesse deste trabalho, baseado na revisão deste método apresentada em Domingues et al (2011). Mais detalhes referentes à teoria de Wavelets são encontrados em Castilho et al (2012) Essa transformada satisfaz três propriedades fundamentais:

1. **Localização:** Localização significa que os cálculos computacionais de uma entrada  $\mathbf{f}_j$  ou  $\mathbf{d}_j$  requerem o conhecimento de um número finito de informações vizinhas de  $\mathbf{f}_{j+1}$ .
2. **Cancelamento de Polinômios:** De forma a se ter uma boa compressão de dados, os detalhes  $\mathbf{d}_j$  devem ser zero para polinômios até um certo grau. Isso significa que, em regiões suaves, as amplitudes dos coeficientes wavelets são desprezíveis e podem ser desconsideradas. Por outro lado, próximos a singularidades, os detalhes são importantes e assim esses coeficientes devem ser mantidos.
3. **Estabilidade:** Esta é uma propriedade crucial que garante que pequenas perturbações introduzidas nos dados transformados em qualquer escala não são ampli-

ficadas sem controle e aplicações iterativas da transformada multi-nível

$$\mathbf{f}_J \Leftrightarrow \mathbf{f}_J^{MR} = (\mathbf{f}_0, \mathbf{d}_0, \dots, \mathbf{d}_{J-1}).$$

Na análise wavelet, as transformadas MR direta e inversa são normalmente associadas à representação de funções em espaços funcionais da forma

$$V_J = V_0 + \sum_{j=0}^{J-1} W_j, \quad (7.1)$$

em que  $V_j$  denota o espaço gerado pelas funções escalas e  $W_j$  o espaço gerados pelas funções wavelets. Funções  $f \in V_J$  podem ser expandidas, ou como

$$f(x) = \sum_k f_{J,k} \Phi_{J,k}(x),$$

com a função escala, que é uma representação em um nível, ou com as funções wavelets, que é uma representação em multi-escala

$$f(x) = \sum_k f_{0,k} \Phi_{0,k}(x) + \sum_{j=0}^{J-1} \sum_k d_{j,k} \Psi_{j,k}(x).$$

Os coeficientes wavelet e escala são dados por

$$f_{j,k} = \langle f, \Phi_{j,k}^* \rangle, \quad d_{j,k} = \langle f, \Psi_{j,k}^* \rangle,$$

sendo definidos por produtos internos com funções duais escala e wavelet,  $\Phi_{j,k}^*$  e  $\Psi_{j,k}^*$ , respectivamente.

Nos métodos clássicos, os valores de  $\mathbf{f}_j$  são obtidos de uma informação local de amostras igualmente espaçadas de uma grade Cartesiana  $X_j$ , e as informações entre níveis das transformações são obtidas de convoluções com filtros passa-baixa e passa-alta frequências (mais detalhes em Mallat (1999) e Cohen (2000)).

Neste trabalho considera-se uma análise multirresolução usando uma outra abordagem desenvolvida por Harten (1996) e Abgrall e Harten (1998). A formulação proposta por Harten é uma análise multirresolução baseada na representação de dados, com conceitos de discretização, projeção e predição. Esses conceitos são mais

detalhados neste capítulo, em que uma discretização por médias celulares é considerada para uma malha uniforme em um intervalo e em malhas bidimensionais Cartesianas. No Apêndice F, é apresentada uma discretização por valores pontuais.

Para a aplicação dessa dissertação, o ingrediente fundamental desse método é que os coeficientes wavelet podem ser usados como indicadores da regularidade local, que uma consequência direta do cancelamento de polinômios. Neste contexto, o princípio é representar funções apenas pelas suas médias celulares correspondentes aos coeficientes wavelet significantes. Para uma representação de funções com singularidades localizadas, é esperado ser necessário poucas células na malha adaptada nas regiões suaves, e mais células nas regiões próximas às singularidades. Gera-se, então, uma representação esparsa estruturada que é usada na solução numérica de Equações Diferenciais Parciais (EDP) Evolutivas. Neste capítulo, um resumo dos principais ingredientes desse método de representação é apresentado, mais detalhes podem ser obtidos em Domingues et al (2010).

No caso unidimensional, considera-se um intervalo unitário  $\Omega = [0, 1]$  em malhas diádicas. Num contexto mais geral, é possível estender essas idéias para dimensões superiores.

## 7.1 MR para médias celulares

Considere uma hierarquia de um intervalo unitário de malhas diádicas uniformes  $X_j$ . As malhas diádicas  $X_j$  formam uma hierarquia de partições do intervalo  $[0, 1] = \cup_k \Omega_{j,k}$  pelas células disjuntas  $\Omega_{j,k} = (k2^{-j}, (k+1)2^{-j})$ ,  $k = 0, \dots, 2^j - 1$ , como ilustrado na Figura 7.1. Considera-se, agora, a configuração onde ao operador de discretização  $\mathcal{D}_j : f \rightarrow \mathbf{f}_j$  é definido por funções absolutamente integráveis  $f$  em  $[0, 1]$  pelas médias celulares na partição do nível  $j$

$$f_{j,k} = 2^j \int_{\Omega_{j,k}} f(x) dx.$$

Como cada célula  $\Omega_{j,k}$  é formada por duas células filhas, de mesmo tamanho, no nível  $j+1$ , ou seja,

$$\Omega_{j,k} = \Omega_{j+1,2k} \cup \Omega_{j+1,2k+1},$$

segue-se, então, que o operador projeção  $P_{j+1 \rightarrow j} : \mathbf{f}_{j+1} \rightarrow \mathbf{f}_j$ , o qual fornece as médias celulares exatas em um nível  $j$ , a partir de valores em um nível  $j+1$ , é



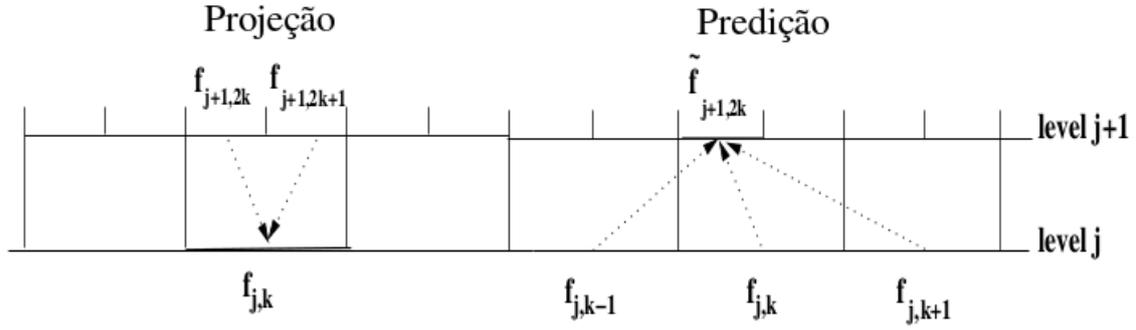


Figura 7.2 - Ilustração dos operadores de projeção e predição.  
Fonte: Domingues et al (2011).

Similarmente ao tratamento da fronteira no caso dos valores pontuais, na primeira célula do lado esquerdo, tem-se

$$\begin{aligned}\tilde{f}_{j+1,1} &= \frac{11}{8}f_{j,0} - \frac{1}{2}f_{j,1} + \frac{1}{8}f_{j,2}, \\ \tilde{f}_{j+1,0} &= \frac{5}{8}f_{j,0} + \frac{1}{2}f_{j,1} - \frac{1}{8}f_{j,2},\end{aligned}$$

e, no lado direito,

$$\begin{aligned}\tilde{f}_{j+1,2^{j+1}-1} &= -\frac{1}{8}f_{j,2^j-3} + \frac{1}{2}f_{j,2^j-2} + \frac{5}{8}f_{j,2^j-1}, \\ \tilde{f}_{j+1,2^{j+1}-2} &= \frac{1}{8}f_{j,2^j-3} - \frac{1}{2}f_{j,2^j-2} + \frac{11}{8}f_{j,2^j-1}.\end{aligned}$$

Pode-se verificar que esse operador de predição é derivado da interpolação polinomial da média celular, no sentido de

$$\begin{aligned}\tilde{f}_{j+1,2k+1} &= 2^{j+1} \int_{\Omega_{j+1,2k+1}} p(x) dx \\ \tilde{f}_{j+1,2k} &= 2^{j+1} \int_{\Omega_{j+1,2k}} p(x) dx\end{aligned}$$

em  $p(x)$  é o polinômio quadrático do qual as médias celulares das três células mais próximas do estêncil  $\Omega_{j,m}$ , com  $|m - k| \leq 1$ , coincidem com  $f_{j,m}$

$$f_{j,m} = 2^j \int_{\Omega_{j,m}} p(x) dx.$$

Esse procedimento pode ser estendido para derivar predições de ordem superior, usando estêncis maiores, os quais incluem  $\Omega_{j,k}$  e os  $2L$  vizinhos mais próximos  $\Omega_{j,m}$ ,  $|m - k| < L$ ,  $L > 1$ . Primeiramente, define-se  $p(x)$  como o polinômio de grau  $2L$ , tal que as médias celulares no estêncil  $2L$  de células  $\Omega_{j,m}$  coincidam com  $f_{j,m}$  e, então, define-se  $\tilde{f}_{j+1,2k}$  e  $\tilde{f}_{j+1,2k+1}$  como as médias celulares de  $p(x)$  nas células filhas  $\Omega_{j+1,2k}$  e  $\Omega_{j+1,2k+1}$ . Tal procedimento reproduz polinômios de grau  $M = 2L$ .

### 7.1.2 Transformada MR

Assim como no caso de discretização por valores pontuais, para construir a análise multirresolução para médias celulares, deve-se manter os detalhes entre um nível de escala  $j$  e o próximo nível acima  $j + 1$  como sendo o erro de predição em uma das células filha. Por exemplo,

$$d_{j,k} = f_{j+1,2k+1} - \tilde{f}_{j+1,2k+1}.$$

Dados  $\mathbf{f}_j$  e  $\mathbf{d}_j$ ,  $\mathbf{f}_{j+1}$  pode ser recuperado, como indicado nos próximos algoritmos.

#### 7.1.2.1 Análise - $\mathbf{f}_J \rightarrow \mathbf{f}_J^{MR} = (\mathbf{f}_0, \mathbf{d}_0, \dots, \mathbf{d}_{J-1})$

- Para  $j = J - 1, \dots, 0$ , dado  $\mathbf{f}_{j+1}$ :  
 Faça a projeção:  $P_{j+1 \rightarrow j} : \mathbf{f}_{j+1} \rightarrow \mathbf{f}_j$   
 Faça a predição:  $P_{j \rightarrow j+1} : \mathbf{f}_j \rightarrow \tilde{\mathbf{f}}_{j+1}$   
 Calcule os erros de predição:  $d_{j,k} = f_{j+1,2k+1} - \tilde{f}_{j+1,2k+1}$

#### 7.1.2.2 Síntese - $\mathbf{f}_J^{MR} = (\mathbf{f}_0, \mathbf{d}_0, \dots, \mathbf{d}_{J-1}) \rightarrow \mathbf{f}_J$

- Para  $j = 0, \dots, J - 1$ , dados  $\mathbf{f}_j, \mathbf{d}_j$ :  
 Faça a predição:  $P_{j \rightarrow j+1} : \mathbf{f}_j \rightarrow \tilde{\mathbf{f}}_{j+1}$   
 Calcule  $\mathbf{f}_{j+1}$ :

$$\begin{aligned} f_{j+1,2k+1} &= d_{j,k} + \tilde{f}_{j+1,2k+1} \\ f_{j+1,2k} &= 2f_{j,k} - \tilde{f}_{j+1,2k+1} \end{aligned}$$

### 7.1.3 Contexto funcional

No presente caso de discretização de médias celulares, pode-se, também, definir a decomposição de multirresolução como nas Equações G.4 e G.5. Sendo

$$f_{j,k} = 2^j \int_{\Omega_{j,k}} f(x) dx = \langle f, \Phi_{j,k}^* \rangle,$$

em que  $\Phi_{j,k}^* = 2^j \chi_{\Omega_{j,k}}$ . O termo  $\chi_C$  representa a função característica do conjunto  $C$ , a qual possui o valor 1, para  $x \in C$ , e 0 caso contrário. A relação de escala mantém-se

$$\Phi_{j,k}^* = \frac{1}{2} [\Phi_{j+1,2k}^* + \Phi_{j+1,2k+1}^*]$$

Conclui-se, então, que as wavelet duais  $\Psi_{j,k}^*$ , tais que

$$d_{j,k} = f_{j+1,2k+1} - \tilde{f}_{j+1,2k+1} = \langle f, \Psi_{j,k}^* \rangle$$

são dadas por

$$\begin{aligned} \Psi_{j,k}^* &= 2^{j+1} \chi_{\Omega_{j+1,2k+1}} - 2^j \sum_m h_m^k \chi_{\Omega_{j,m}}, \\ &= \Phi_{j+1,2k+1}^* - \sum_m h_m^k \Phi_{j,m}^*, \end{aligned}$$

em que  $h_m^k$  são os coeficientes de predição, tais que

$$\tilde{f}_{j+1,2k+1} = \sum_m h_m^k f_{j,m}.$$

A definição das principais funções básicas  $\Phi_{j,k}$  e  $\Psi_{j,k}$  baseia-se na convergência do operador de predição.

A predição de uma média celular é convergente se, para qualquer sequência inicial  $s_j = (s_{j,k})$ , existe uma função integrável  $s_j(x)$ , tal que suas médias celulares nos níveis  $\ell \geq j$  coincidem com a saída obtida com a subdivisão iterativa de média celular  $s_\ell = P_{\ell-1 \rightarrow \ell} s_{\ell-1}$ . Se essa propriedade é mantida, as funções escala interpoladoras de médias celulares  $\Phi_{j,k}(x)$  podem ser definidas pelas médias celulares da subdivisão de sequências de delta de Dirac  $s_{j,m} = \delta(k - m)$ . Nos espaços funcionais,

$$V_j = \text{span} \{ \Phi_{j,k}(x), k = 0, \dots, 2^j - 1 \}$$

a expansão

$$f_j(x) = \sum_{k=0}^{2^j-1} f_{j,k} \Phi_{j,k}(x) \in V_j \quad (7.3)$$

corresponde à reconstrução por média celular de uma função integrável  $f$ , no sentido de que  $\mathcal{D}_j f_j(x) = \mathbf{f}_j$ , *i.e.*, as médias celulares de  $f$  e  $\mathbf{f}_j$  coincidem no nível  $j$ .

Utilizando os mesmos argumentos da definição de interpolação, obtém-se as relações escala

$$\Phi_{j,k}(x) = \sum_m c_k(m) \Phi_{j+1,m}(x),$$

em que  $c_k(m)$  são as médias celulares de  $\Phi_{j,k}$  no nível  $j+1$ , as quais são obtidas pelas predições da sequência delta  $\delta(k-m)$ . As seguintes propriedades são mantidas para essas funções escalas de médias celulares (DONOHO, 1993)

- a) Para  $N > 0$ ,  $\Phi_{j,k}$  é uniformemente contínua, com regularidade de Hölder  $s = s(N)$  aumentada com grau de interpolação  $N$ . Por exemplo,  $\Phi_{j,k}(x) \in C^s$  com  $s > 0.678$  e  $s > 1.272$  para  $N = 2$  e  $4$ , respectivamente (de fato, essa é uma consequência de uma fórmula de comutação ao expressar a derivada da interpolação de funções escala usando polinômios interpoladores de grau  $M-1$ , em termos das diferenças divididas das funções escala para média celular para  $N = M-2$ ).
- b) Distante das fronteiras,  $\Phi_{j,k}(x) = \phi(2^j x - k)$ , em que  $\phi = \varphi_{1,N+1}$  são as funções escala *spline* Cohen-Daubechies-Feauveau (COHEN et al., 1992).
- c) Possuem suporte compacto, com tamanho  $|\text{supp}(\Phi_{j,k})| = O(2^{-j})$ , aumentando com  $M$ .

Tendo-se que  $V_j \subset V_{j+1}$ , examina-se, então, espaços complementares  $W_j$ , tais que  $V_{j+1} = V_j + W_j$ , e wavelets  $\Psi_{j,k}$  que geram essas diferenças. Como tal, é esperado que a expansão multirresolução de  $\Psi_{j,k}$  corresponda a zero média celular no nível  $j$  e coeficientes wavelet  $d_{j,m} = \delta(k-m)$ . Portanto,  $\Psi_{j,k}$  deve ser dada pela expressão

$$\Psi_{j,k}(x) = \Phi_{j+1,2k+1}(x) - \Phi_{j+1,2k}(x).$$

Para o caso particular de predição por aproximação constante,  $\Phi_{j,k} = \chi_{\Omega_{j,k}}$  e os espaços  $V_j$  são formados por funções constante por partes, gerando uma análise

multirresolução de  $L^2(\Omega)$  tal que  $V_j \subset V_{j+1}$ ,  $\cup_{j \geq 0} V_j$  é denso em  $L^2(\Omega)$  e  $\{\Phi_{j,k}, k = 0, \dots, 2^j - 1\}$  é uma base ortogonal para  $V_j$  e a expansão (7.3) corresponde à projeção ortogonal de  $f$  em  $V_j$ . Os espaços wavelet  $W_j = \text{span}\{\Psi_{j,k}, k = 0, \dots, 2^j - 1\}$  são os complementos ortogonais de  $V_j$  em  $V_{j+1}$ , e  $\Psi_{j,k}(x)$  são as chamadas wavelets de Haar.

No geral, para predição de médias celulares por polinômios de ordem superior  $M = 2L$ , os espaços  $V_j$  também geram uma análise multirresolução de  $L^1(\Omega)$ , tal que  $V_j \subset V_{j+1}$ ,  $\cup_{j \geq 0} V_j$ , é denso em  $L^1[0, 1]$ . A sequência de aproximações  $f_j(x)$  corresponde às projeções biortogonais de  $f$  em  $V_j$  e

$$\begin{aligned} \|f_j\|_{L^1} &\leq C\|f\|_{L^1} \\ \|f - f_j\|_{L^1} &\rightarrow 0, \text{ as } j \rightarrow \infty. \end{aligned}$$

Uma referência sobre propriedades de convergência de aproximações multirresolução é encontrada em Cohen (2000).

#### 7.1.4 Indicador wavelet de regularidade local

No contexto de médias celulares, os coeficientes wavelet podem ser, também, utilizados como indicadores de regularidade local. Assumindo que a função  $f(x)$  tem suavidade  $C^s$ , para algum  $s \leq M$ , dentro de um intervalo  $I_{j,k}$  suportando  $\Psi_{j,k}^*$  (*i.e.*,  $I_{j,k}$  contém as células do estêncil utilizadas para a predição de  $\tilde{f}_{j+1,2k+1}$ ). Utilizando os resultados clássicos da aproximação local de polinômios por polinômios e a propriedade de cancelamento de polinômios, o coeficiente wavelet pode ser estimado por

$$\begin{aligned} |d_{j,k}| &\leq \inf_{q \in \Pi_{M-1}} \|f - q\|_{L^\infty(I_{j,k})} \|\Psi_{j,k}^*\|_{L^1} \\ &\leq C2^{-sj} |f|_{C^s(I_{j,k})}, \end{aligned}$$

em que  $\|\Psi_{j,k}^*\| = 1$  e o tamanho de  $I_{j,k}$  é  $O(2^{-j})$ . Portanto, para predições de alta precisão, o alto decaimento dos coeficientes wavelet é garantido em regiões suaves.

### 7.1.5 Compressão de dados

Dadas as médias celulares  $\mathbf{f}_J$  no nível mais refinado  $J$ , depois da iteração da transformada MR, podem ser representadas como

$$\mathbf{f}_J^{MR} = (\mathbf{f}_0, \mathbf{d}_0, \dots, \mathbf{d}_{J-1}).$$

A compressão de dados é obtida depois da etapa de truncamento

$$\mathcal{T}_{j,k}^\epsilon = \begin{cases} 0 & \text{se } |d_{j,k}| \leq \epsilon_j \\ d_{j,k} & \text{caso contrário.} \end{cases}$$

A Figura 7.3 mostra que a solução do modelo EGLM-MHD possui diferentes padrões de regularidade de acordo com o tempo. As malhas adaptativas para cada tempo ilustram suas caracterizações de regularidades locais.

Para as aplicações deste trabalho, no contexto do método dos volumes finitos, a idéia principal é utilizar os coeficientes wavelet em uma representação multirresolução das médias celulares da solução, para construir malhas adaptativas: utiliza-se células pouco refinadas nas regiões onde os coeficientes wavelet são pequenos (e a solução é suave) e refina-se onde esses coeficientes são significantes.

#### 7.1.5.1 Estabilidade

Para esquemas convergentes de subdivisão, pode-se utilizar a expansão

$$f_J(x) - f_J^\epsilon(x) = \sum_{j=0}^{J-1} \sum_k (d_{j,k} - d_{j,k}^\epsilon) \Psi_{j,k}(x).$$

Desde que o suporte de  $\Psi_{j,k}$  é uniformemente localizado em torno de  $C_{j,k}$ , tal que  $\|\Psi_{j,k}\|_{L^1} \leq C2^{-j}$ , o erro de truncamento  $L^1$  no nível  $j$  é

$$\begin{aligned} \left| \int_0^1 \sum_{k=0}^{2^j-1} (d_{j,k} - d_{j,k}^\epsilon) \Psi_{j,k}(x) dx \right| &\leq \epsilon_j \sum_{k=0}^{2^j-1} \|\Psi_{j,k}\|_{L^1} \\ &\leq C\epsilon_j. \end{aligned}$$

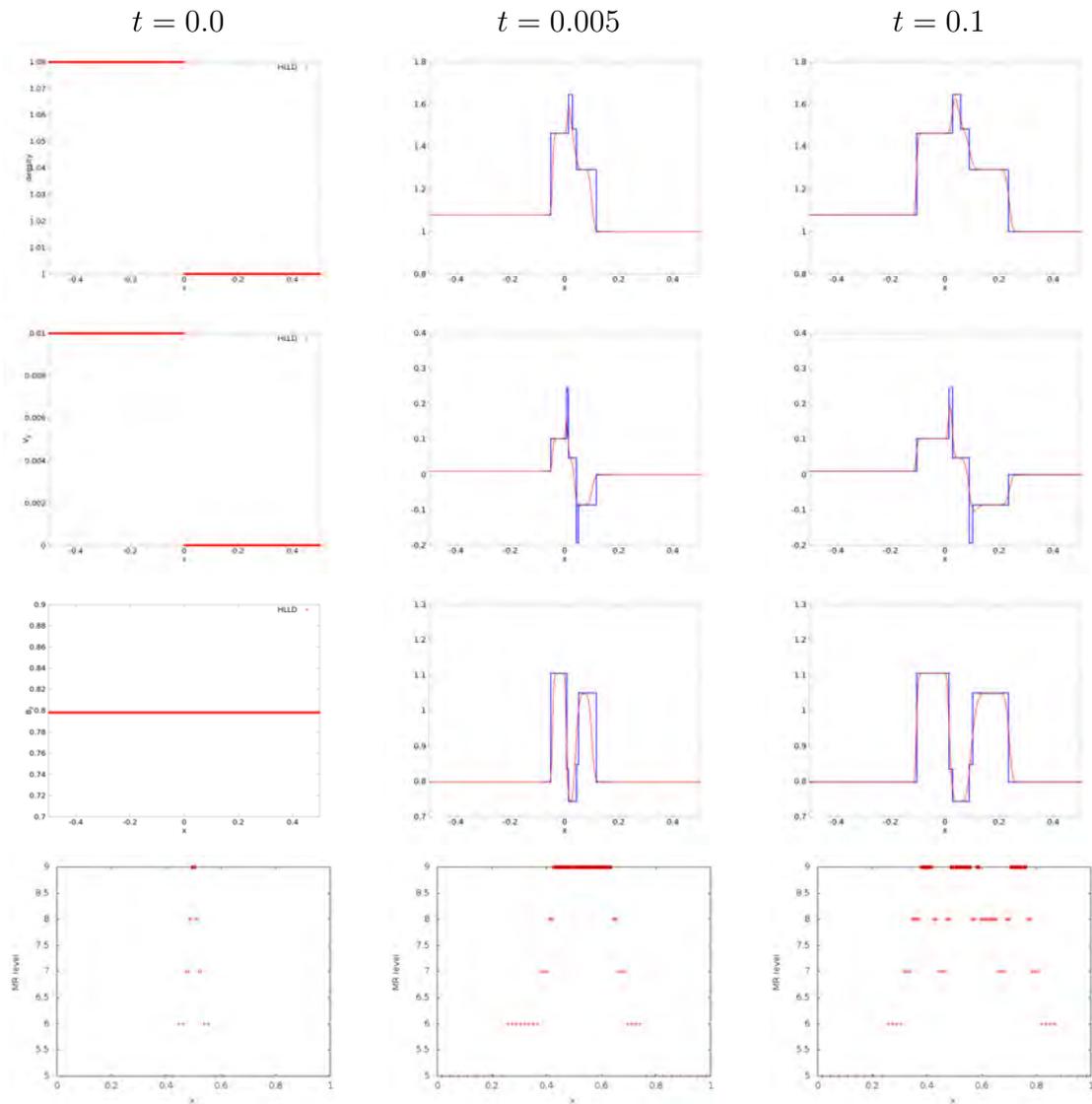


Figura 7.3 - Operação de truncamento. Variáveis  $\rho$ ,  $v_y$  e  $B_z$ , nos tempos  $t = 0.0$ ,  $t = 0.05$  e  $t = 0.1$ , e a malha adaptativa referente ao sistema MHD para esses tempos (de cima para baixo), para  $\epsilon = 0.005$ . Solução exata (azul) e aproximada (vermelho). Solução obtida no intervalo  $x \in [-0.5, 0.5]$ .

Consequentemente, a estabilidade  $L^1$  do erro de truncamento ao longo dos níveis de escala é estimado por

$$\|f_J - f_J^\epsilon\|_{L^1} \leq C \sum_{j=0}^{J-1} \epsilon_j.$$

Se  $\epsilon_{J-1} = \epsilon$ , e  $\epsilon_j = \frac{1}{2}\epsilon_{j+1} = \epsilon 2^{j-J}$ , conclui-se que o erro de truncamento é uniformemente limitado por  $C\epsilon$ , até uma mudança na constante  $C$ .

### 7.1.6 Malhas cartesianas em dimensões superiores

Em dimensões superiores, os esquemas MR para malhas cartesianas podem ser obtidos por produtos tensoriais de esquemas 1D. Por exemplo, na geometria 2D, considera-se a hierarquia de malhas cartesianas

$$G_j = \{\Omega_{j,\gamma}\}, \quad \gamma = (k, m) \in S_j, \quad \Omega = \cup_{\gamma \in S_j} \Omega_{j,\gamma}, \quad |\Omega_{j,\gamma}| \sim 2^{-2j},$$

em que  $\Omega_{j,\gamma} \in G_j$  é a união de quatro células em  $G_{j+1}$ . Uma ilustração é apresentada na Figura 7.4.

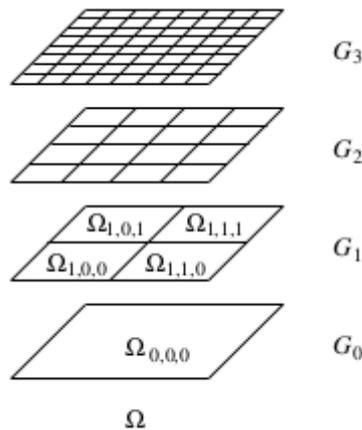


Figura 7.4 - Hierarquia de malhas aninhadas. Fonte: Domingues et al (2011).

Por exemplo, para que a predição seja exata para polinômios quadráticos, distante das fronteiras, as células vizinhas  $\Omega_{j,(k-1,m)}$ ,  $\Omega_{j,(k+1,m)}$ ,  $\Omega_{j,(k,m-1)}$  e  $\Omega_{j,(k,m+1)}$  são

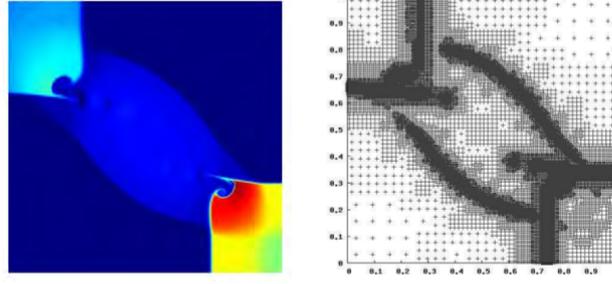


Figura 7.5 - Exemplo de uma malha adaptativa (direita) obtida com o truncamento dos coeficientes wavelet de um campo de densidade (esquerda).  
Fonte: Deiterding et al (2009).

consideradas e, para  $n, p \in \{0, 1\}$ , a formula de predicção (BIHARI; HARTEN, 1997) dá-se por

$$\begin{aligned} \tilde{f}_{j+1,(2k+n,2m+p)} &= f_{j,(k,m)} + \frac{1}{8}(-1)^p(f_{j,(k,m+1)} - f_{j,(k,m-1)}) + \frac{1}{8}(-1)^n(f_{j,(k+1,m)} - f_{j,(k-1,m)}) \\ &+ \frac{1}{64}(-1)^{np}[(f_{j,(k+1,m+1)} - f_{j,(k+1,m-1)}) - (f_{j,(k-1,m+1)} - f_{j,(k-1,m-1)})]. \end{aligned}$$

Agora, três coeficientes wavelet devem ser considerados e as transformações bi-nível são  $\mathbf{f}_{j+1} \leftrightarrow \{\mathbf{f}_j, \mathbf{d}_j^{(1)}, \mathbf{d}_j^{(2)}, \mathbf{d}_j^{(3)}\}$ , em que

$$d_{j,(k,m)}^{(1)} = f_{j+1,(2k,2m+1)} - \tilde{f}_{j+1,(2k,2m+1)},$$

$$d_{j,(k,m)}^{(2)} = f_{j+1,(2k+1,2m)} - \tilde{f}_{j+1,(2k+1,2m)},$$

$$d_{j,(k,m)}^{(3)} = f_{j+1,(2k+1,2m+1)} - \tilde{f}_{j+1,(2k+1,2m+1)}.$$

Utilizando esse esquema MR de terceira ordem para médias celulares, a Figura 7.5 mostra uma malha adaptativa para a solução numérica de um problema de Riemann, em que as células são escolhidas de acordo com os coeficientes wavelet com magnitude maior que um dado limiar de truncamento  $\epsilon$ . Mais detalhes são encontrados em Deiterding et al (2009).



## 8 VF e MR: COMBINANDO OS MÉTODOS

No conceito de adaptabilidade para EDP Evolutivas, os esquemas adaptativos de MR para solução numérica são escritos na forma

$$\frac{\partial u(x, t)}{\partial t} = \mathcal{D}u(x, t), \quad t > 0, x \in \Omega, \quad (8.1)$$

em que  $\mathcal{D}$  é um operador diferencial no espaço. A Equação 8.1 é completa quando tem-se condições inicial e de contorno para a mesma. Os métodos combinam esquemas tradicionais e estáveis, como volumes finitos ou diferenças finitas, com a habilidade de caracterização da regularidade local das funções pertencente aos coeficientes wavelet. Utilizando um ambiente apropriado, a idéia é representar soluções numéricas por médias celulares correspondentes aos coeficientes wavelet significativos. Em geral, pouca informação é necessária em cada passo de tempo. A malha adaptativa é menos refinada onde a solução é suave e mais refinada próximo às irregularidades. Em cada célula da malha adaptativa, a discretização do operador  $\mathcal{D}u(x, t)$  é modificada utilizando o tamanho do passo proporcional à escala local de cada célula.

Assume-se o esquema numérico de referência em uma formulação explícita

$$U^{n+1} = \mathbf{E}U^n,$$

em que os vetores  $U^n = U_J^n$  contêm a informação da solução numérica em uma malha uniforme no nível  $J$  e tempo  $t_n$ ,  $\mathbf{E}$  é o operador discreto de evolução que inclui a discretização das derivadas espaciais, a ação de um solver de EDO e a aplicação das condições de contorno.

Assume-se o esquema numérico de referência em uma formulação explícita, em que os vetores contêm as informações da solução numérica em uma malha uniforme no nível  $J$ , em que  $\mathbf{E}$  é o operador de evolução que inclui a discretização de derivadas espaciais, e a aplicação das condições de contorno. Em um método de multirresolução adaptativa, o objetivo é utilizar tal esquema de referência em uma abordagem mais eficiente, levando em conta a informação de regularidade local wavelet sobre a solução numérica. O primeiro passo nessa direção é a representação da solução numérica em uma malha esparsa. Para isso, três estágios básicos são utilizados

- a) **Refinamento:**  $U_{MR}^{n+} \leftarrow \mathbf{R}U_{MR}^n$
- b) **Evolução:**  $\check{U}_{MR}^{n+1} \leftarrow \mathbf{E}_{MR}U_{MR}^{n+}$
- c) **Reajuste da malha à solução:**  $U_{MR}^{n+1} \leftarrow \mathcal{T}_\epsilon \check{U}_{MR}^{n+1}$

O operador de refinamento  $\mathbf{R}$  é uma medida de precaução para prevenir que alguns gradientes se movam para as áreas da malha menos refinadas entre os passos de tempo. Como as regiões de suavidades ou irregularidades da solução podem mudar com o tempo,  $\Gamma^n$  pode não ser mais conveniente no próximo passo de tempo  $t_{n+1}$ . Por esse motivo, antes de fazer a evolução, a representação da solução deve ser estendida para a malha  $\Gamma^{n+}$ , a qual espera-se que contenha  $\Gamma^{n+1}$ . Então, o operador de evolução  $\mathbf{E}_{MR}$  é aplicado. O índice  $MR$  indica que a discretização das derivadas espaciais é modificada utilizando um passo de tamanho proporcional à escala local de cada célula. No último estágio, o operador de truncamento é aplicado com o intuito de reajustar algumas células em  $\Gamma^{n+}$ , onde o refinamento é desnecessário para uma representação precisa de  $U_{MR}^{n+1}$ .

Para a implementação desse algoritmo adaptativo, as ferramentas básicas são provenientes da análise multirresolução para médias celulares. Como descrito anteriormente, a informação discreta da solução é organizada em níveis ou resoluções. A cada nível  $j$ , os coeficientes wavelet  $\mathbf{d}_j$  são definidos como um erro de predição nas células filhas do determinado nível de discretização. Usando coeficientes wavelet como indicadores de regularidade, o principal ingrediente é o operador de truncamento  $\mathcal{T}_\epsilon$ : apenas a informação correspondente aos coeficientes wavelet com valores significantes, com magnitude maior que o limiar de truncamento  $\epsilon$ , devem ser mantidos na malha adaptativa. O operador de predição é, também, utilizado para calcular algumas informações necessárias que não estão presentes na malha adaptativa

computacional. Por exemplo, essas informações são necessárias no operador de refinamento, para aproximar a solução nos locais em  $\Gamma^{n+} \setminus \Gamma^n$  e, também, na aplicação de  $\mathbf{E}_{MR}$ , quando alguns estêncis necessários não estão presentes em  $\Gamma^{n+}$ . Detalhes sobre a estabilidade desse método podem ser obtidas em Domingues et al (2010).

No caso específico do modelo EGLM-MHD estudado, a malha adaptativa utilizada é refinada onde qualquer uma das variáveis necessitem de uma melhor representação, *i.e.*, a malha adaptativa utilizada é a união das malhas adaptativas individuais das variáveis desse modelo. Neste trabalho também optou-se, por simplicidade, por ter apenas um parâmetro  $\epsilon$  para todas as variáveis.



## 9 MR ADAPTATIVA: VERIFICAÇÃO E RESULTADOS

No contexto numérico da multirresolução adaptativa, como apresentada no Capítulo 7, utiliza-se um código com o algoritmo adaptativo implementado. O código utilizado chama-se CARMEN, implementado por Roussel (2003) na linguagem  $C^{++}$ . Já implementados no código CARMEN tem-se as equações de Navier-Stokes e mais cinco tipos de equações. Nesse código é possível calcular qualquer um desses seis modelos utilizando ou não um algoritmo de multirresolução adaptativa.

Na Figura 9.1, tem-se uma ilustração do funcionamento do código CARMEN.

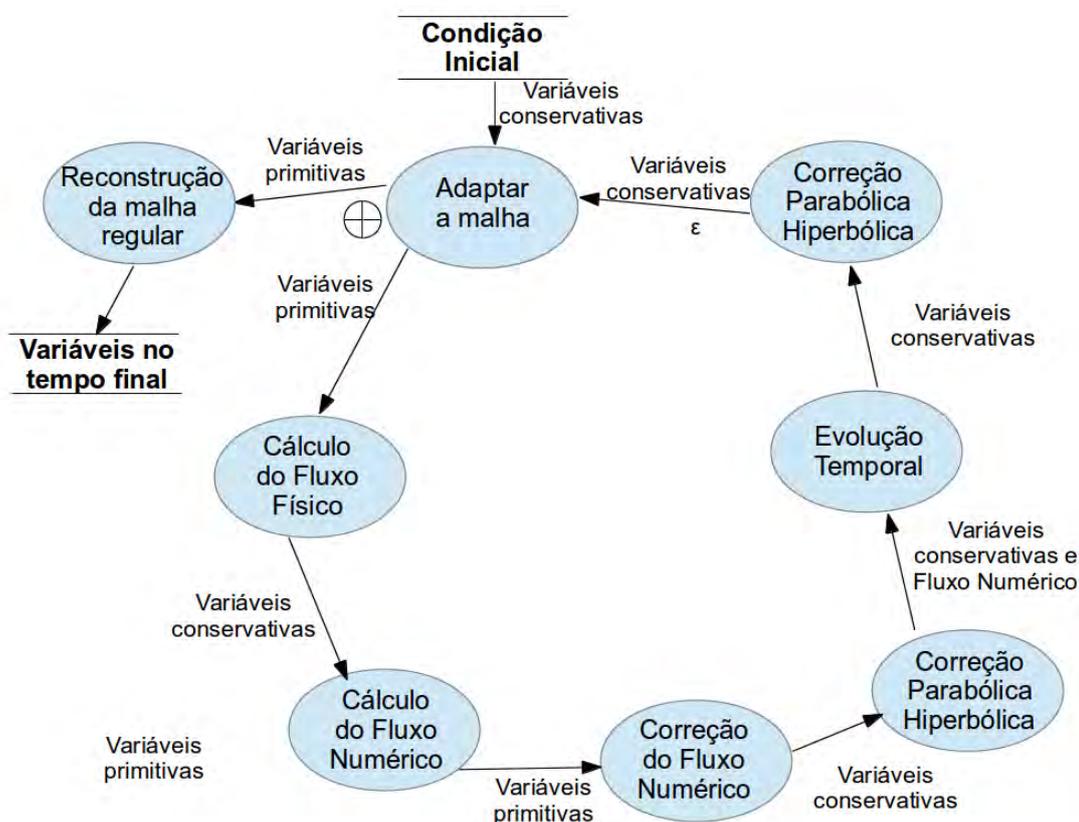


Figura 9.1 - Algoritmo utilizado para obter a solução do modelo com o código CARMEN.

O algoritmo de multirresolução implementado no código CARMEN utiliza uma estrutura de árvore com  $2^L$  nodos, em que  $L$  é o nível mais refinado da malha, para que

seja possível criar uma malha adaptativa com vários níveis. Para adaptar a malha, utiliza-se o parâmetro de truncamento  $\epsilon$ , a partir do qual decide-se onde é necessário o refinamento local ou não. À medida que o parâmetro  $\epsilon$  se aproxima de zero, o refinamento da malha torna-se mais rigoroso e, de maneira análoga, quanto mais esse  $\epsilon$  se afasta de zero, menos refinada a malha é. O parâmetro  $\epsilon$  deve ser escolhido levando em conta a representação da solução que está sendo obtida e, também, o tempo computacional que está sendo gasto para isso. Nesse contexto, é apresentada a solução do modelo EGLM-MHD para quatro diferentes valores de  $\epsilon$ .

Assim como para os resultados adaptativos, tem-se dois casos de estudo no contexto adaptativo. O Caso 1, em que utiliza-se a condição inicial tipo Riemann 1D e o Caso 2, em que utiliza-se a condição inicial tipo Riemann 2D. Para todos os casos utiliza-se uma malha de  $512 \times 512$  células,  $\gamma = 5/3$ ,  $c_{CFL} = 0.3$  e  $t = 0.1$ .

### **9.1 Condição Inicial tipo Riemann 1D**

Nesta seção são apresentados os resultados referentes à verificação desse método para o Caso 1, similarmente ao apresentado no Capítulo 6, pág. 43.

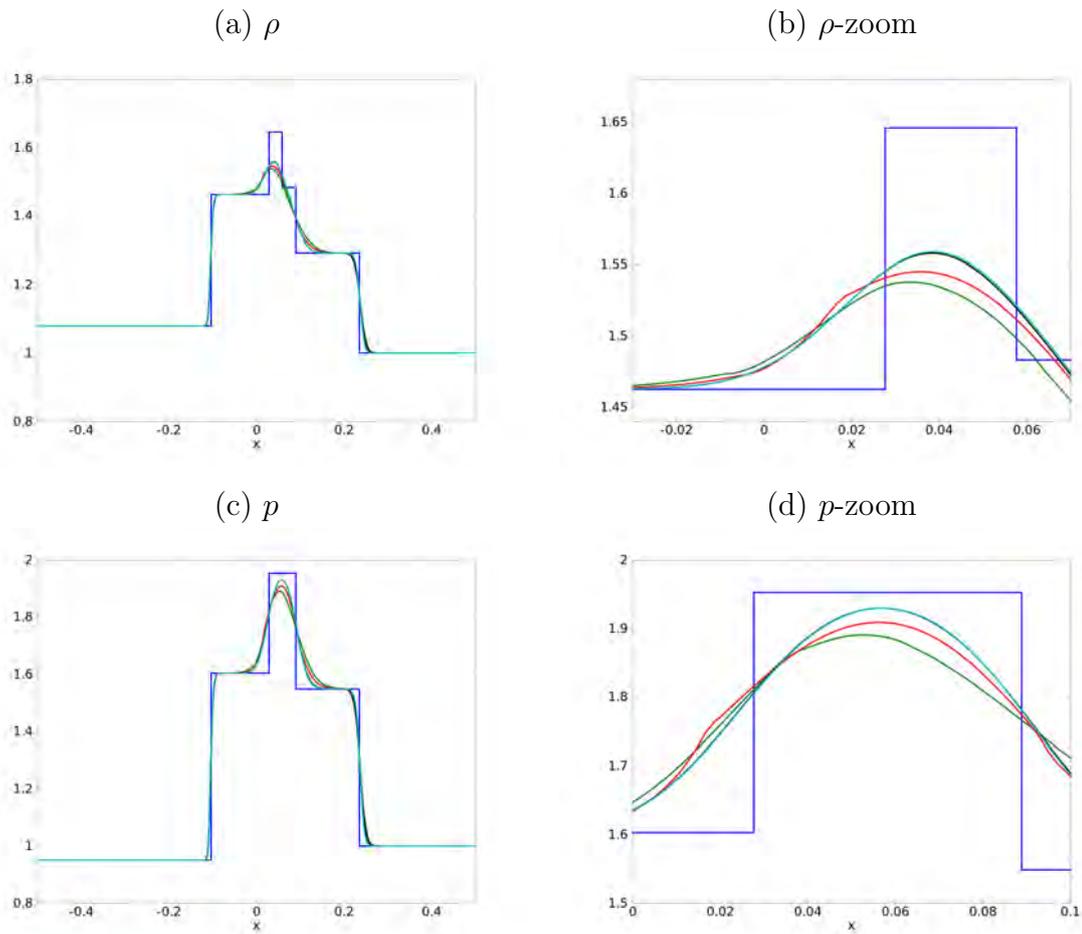


Figura 9.2 - Variáveis densidade  $\rho$  e pressão  $p$  simuladas utilizando o fluxo HLL,  $c_{CFL} = 0.3$  e uma malha adaptativa de  $512 \times 512$  células, no tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$ . São comparadas as soluções aproximadas com o valor  $\epsilon$  igual a 0.01 (verde), 0.005 (vermelho) e 0.001 (preto), a solução exata (azul escuro) e a solução não-adaptativa (azul claro).

Como esperado, na Figura 9.2 observa-se que a variável densidade é melhor representada para o menor valor de  $\epsilon$ . No caso do Riemann solver HLL, apenas é possível obter uma representação satisfatória da solução para o limiar de truncamento mais alto testado. Isso acontece porque o fluxo numérico HLL, como discutido no Capítulo 5, não é eficiente para aproximar a solução do modelo MHD, ou seja, não consegue aproximar de forma eficiente os tipos de discontinuidades presentes na solução. Quando essas discontinuidades não são bem aproximadas, o algoritmo adaptativo também não representa a solução numérica de forma eficiente, mesmo para um  $\epsilon$

bem pequeno. O mesmo acontece para as outras variáveis que compõem a solução. Em particular, a componente  $z$  do campo magnético e  $y$  da velocidade, como nas Figuras 9.3 e 9.4.

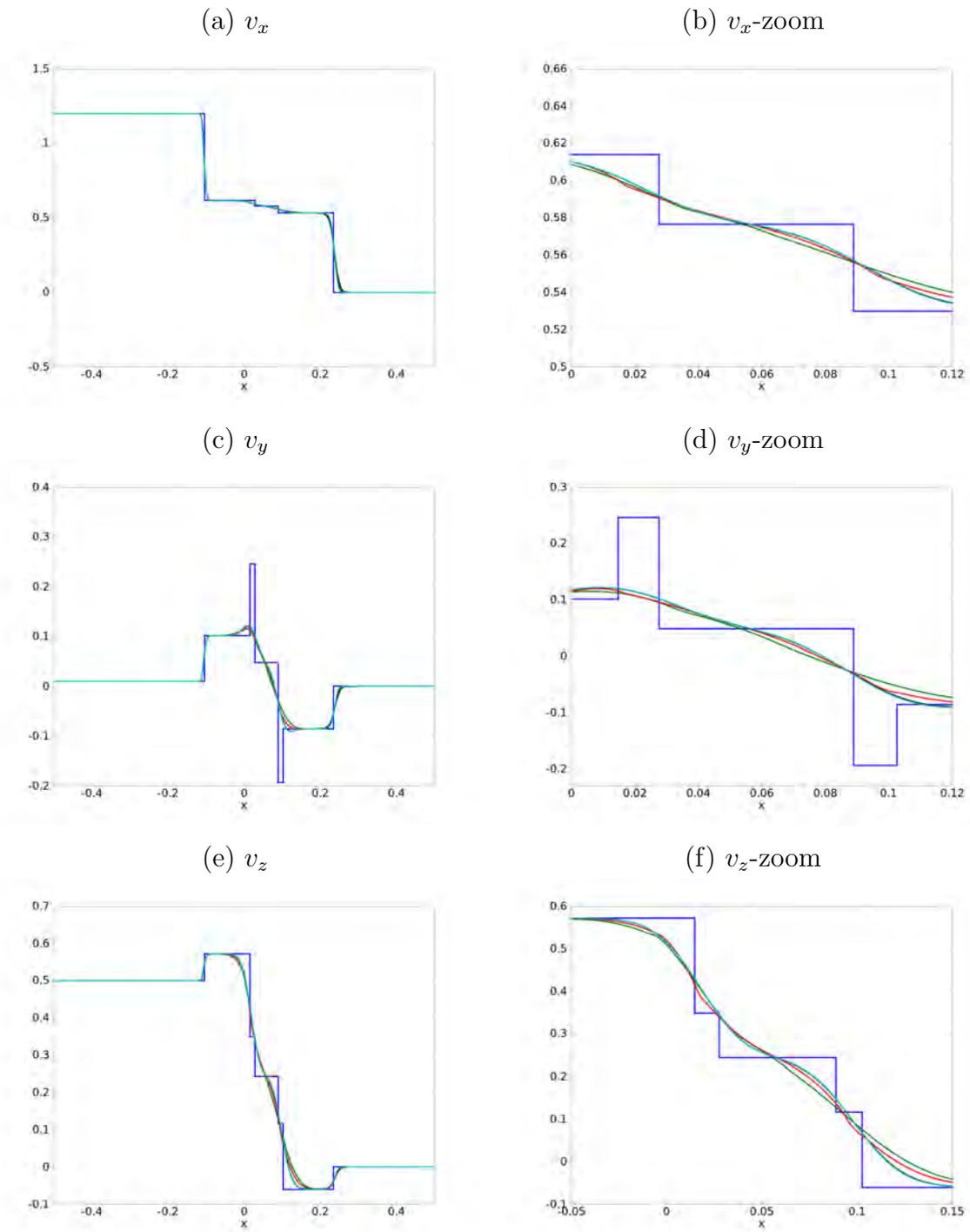


Figura 9.3 - Componentes do vetor velocidade simuladas utilizando o fluxo HLL,  $c_{CFL} = 0.3$  e uma malha adaptativa de  $512 \times 512$  células, no tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$ . São comparadas as soluções aproximadas com o valor  $\epsilon$  igual a 0.01 (verde), 0.005 (vermelho) e 0.001 (preto), a solução exata (azul escuro) e a solução não-adaptativa (azul claro).

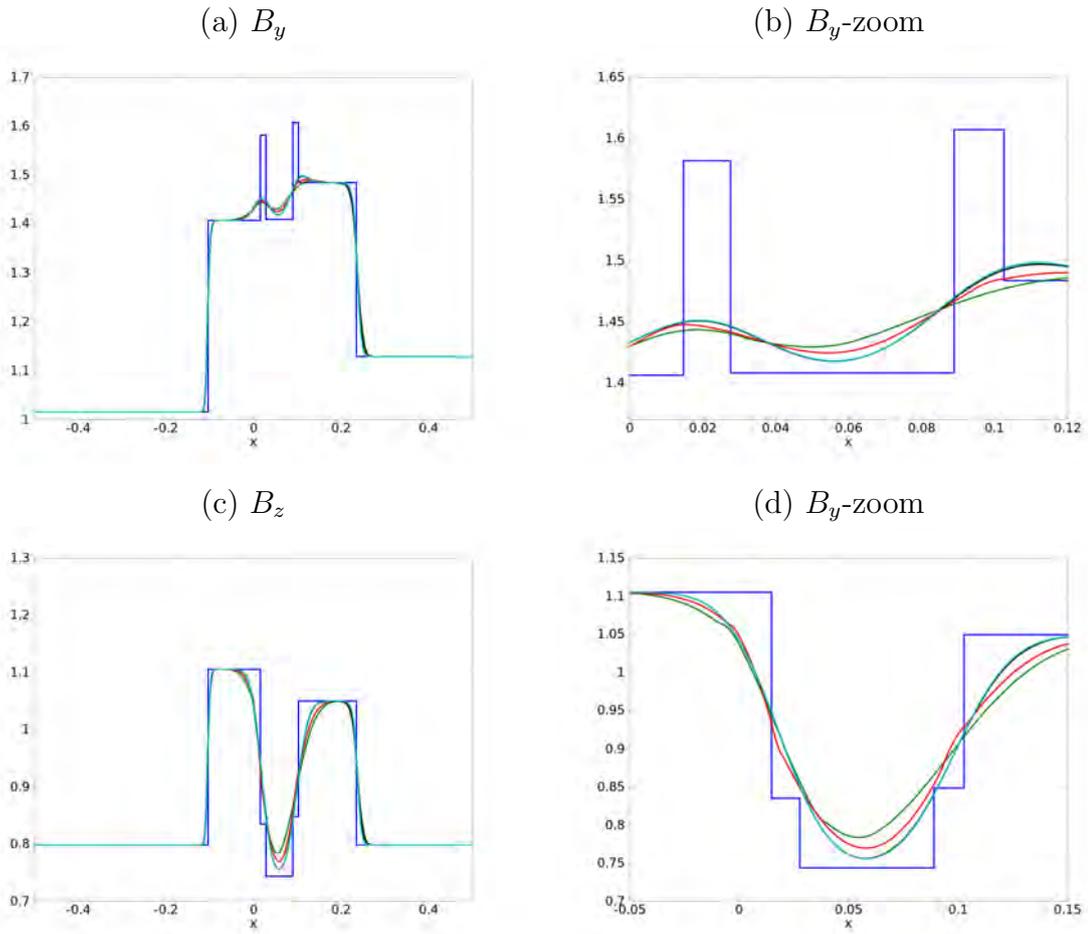


Figura 9.4 - Componentes do vetor campo magnético simuladas utilizando o fluxo HLL,  $c_{CFL} = 0.3$  e uma malha adaptativa de  $512 \times 512$  células, no tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$ . São comparadas as soluções aproximadas com o valor  $\epsilon$  igual a 0.01 (verde), 0.005 (vermelho) e 0.001 (preto), a solução exata (azul escuro) e a solução não-adaptativa (azul claro).

Como o Riemann solver HLLD mostrou-se mais eficiente na aproximação da solução do modelo EGLM-MHD, analisa-se, também, a representação multirresolução relacionada desse Riemann solver. Em particular, as variáveis densidade  $\rho$ , componente  $z$  do campo magnético  $B_z$  e  $y$  da velocidade  $v_y$ , como mostra-se nas Figuras 9.5, 9.6 e 9.7.

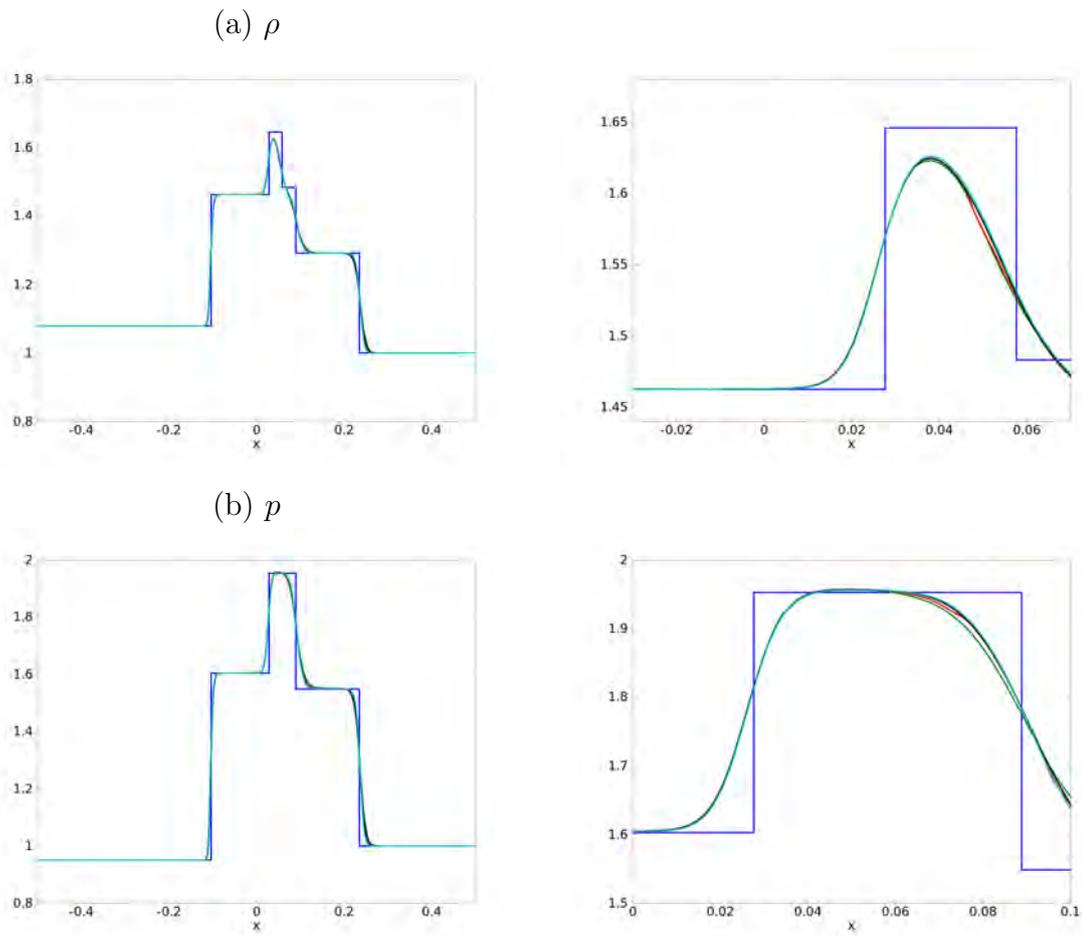


Figura 9.5 - Variáveis densidade (a) e pressão (b) simuladas utilizando o fluxo HLLD,  $c_{CFL} = 0.3$  e uma malha adaptativa de  $512 \times 512$  células, no tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$ . São comparadas as soluções aproximadas com o valor  $\epsilon$  igual a 0.01 (verde), 0.005 (vermelho) e 0.001 (preto), a solução exata (azul escuro) e a solução não-adaptativa (azul claro).

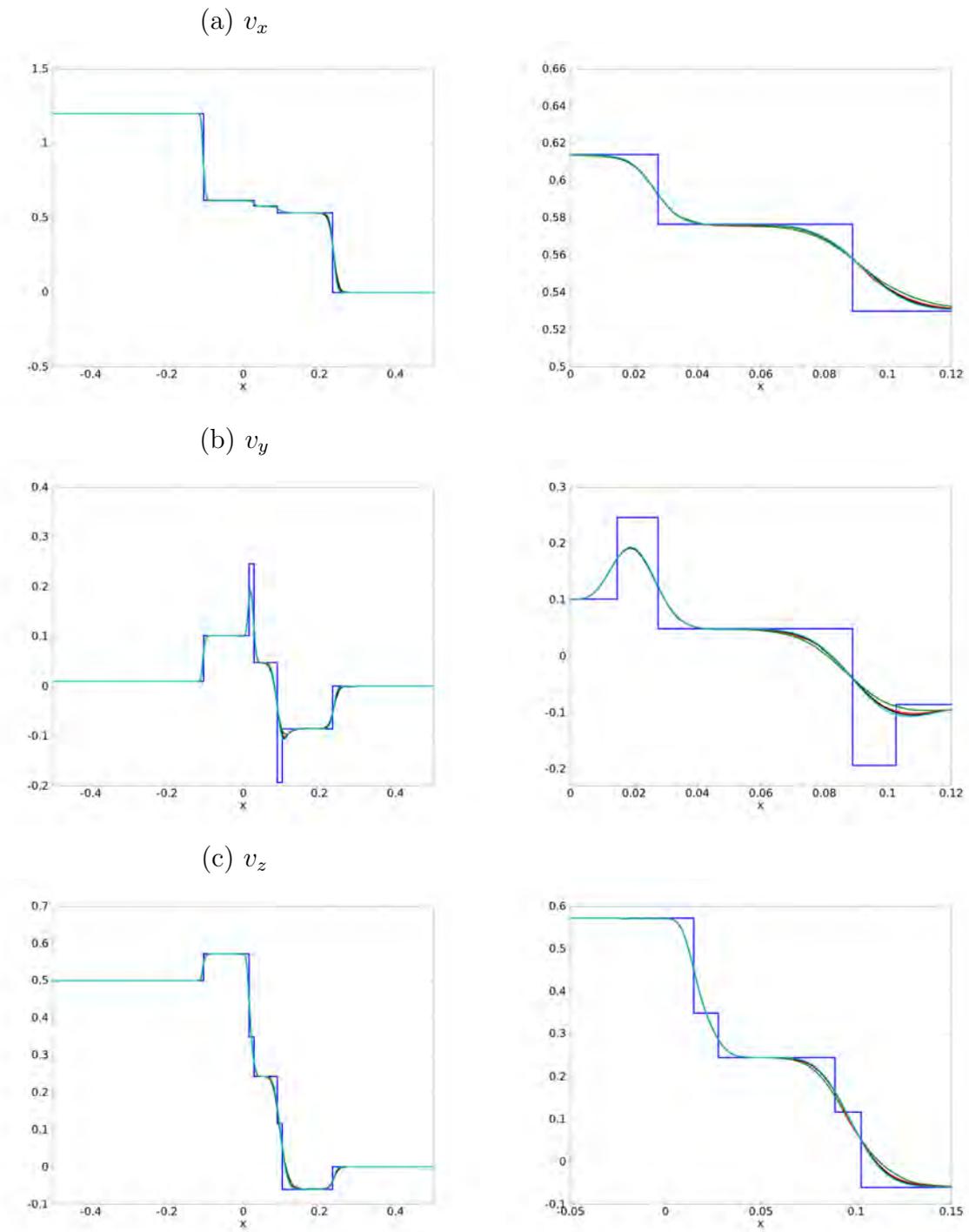


Figura 9.6 - Componentes do vetor velocidade simuladas utilizando o fluxo HLLD,  $c_{CFL} = 0.3$  e uma malha adaptativa de  $512 \times 512$  células, no tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$ . São comparadas as soluções aproximadas com o valor  $\epsilon$  igual a 0.01 (verde), 0.005 (vermelho) e 0.001 (preto), a solução exata (azul escuro) e a solução não-adaptativa (azul claro).

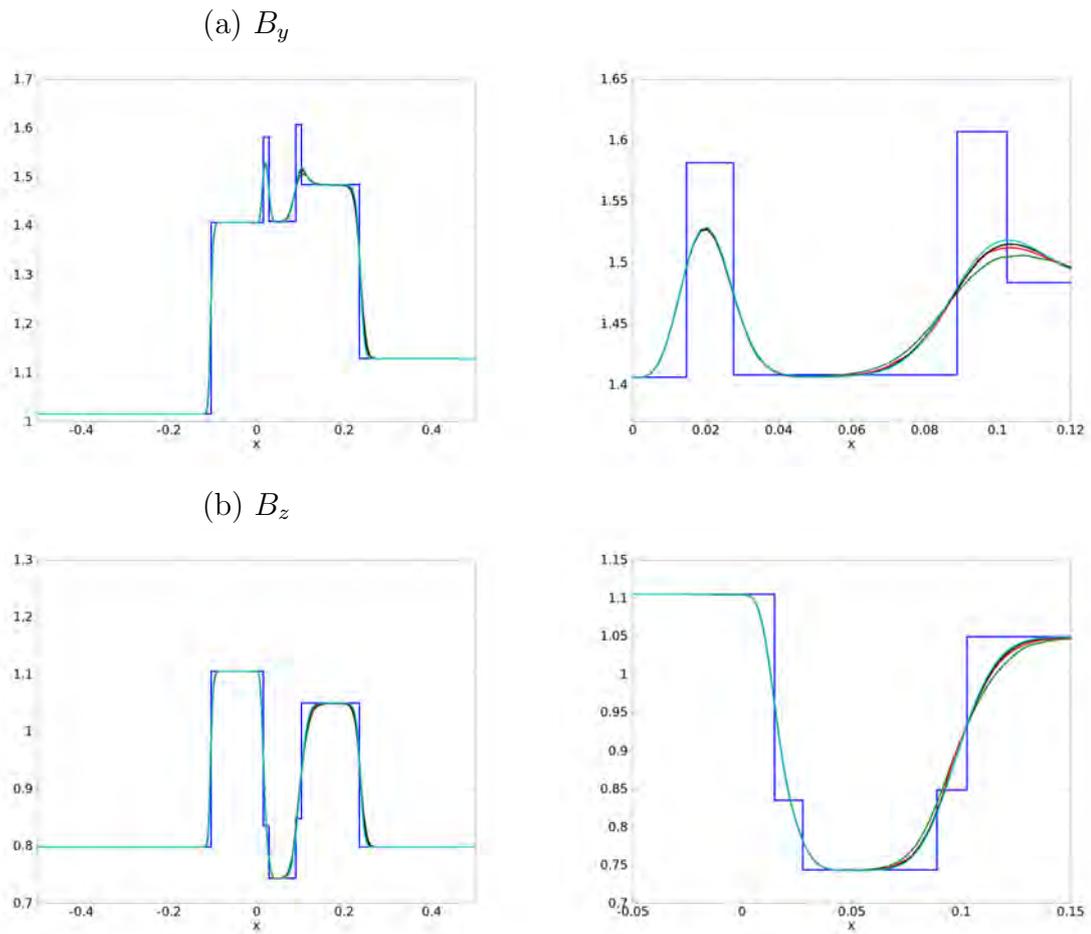


Figura 9.7 - Componentes do vetor campo magnético simuladas utilizando o fluxo HLLD,  $c_{CFL} = 0.3$  e uma malha adaptativa de  $512 \times 512$  células, no tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$ . São comparadas as soluções aproximadas com o valor  $\epsilon$  igual a 0.01 (verde), 0.005 (vermelho) e 0.001 (preto), a solução exata (azul escuro) e a solução não-adaptativa (azul claro).

Nas Figuras 9.8 e 9.9, tem-se as malhas adaptativas utilizadas para aproximar a solução do modelo EGLM-MHD com os fluxos HLL e HLLD, para cada  $\epsilon$  utilizado. Com a visualização das malhas adaptativas, a relação entre o refinamento local da malha e o valor de  $\epsilon$  escolhido fica mais evidente.

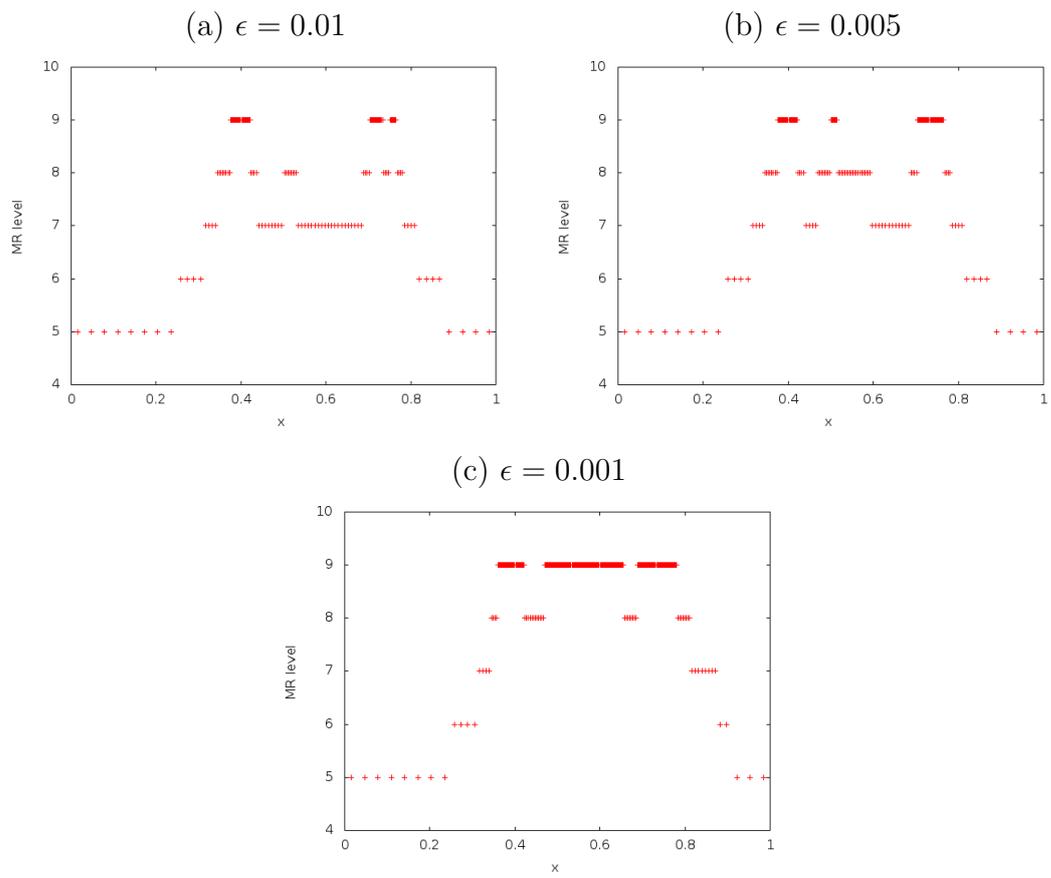


Figura 9.8 - Malhas adaptativas referentes ao Caso 1, para os diferentes valores de  $\epsilon$  utilizados, obtidas ao aproximar a solução com o fluxo HLL.

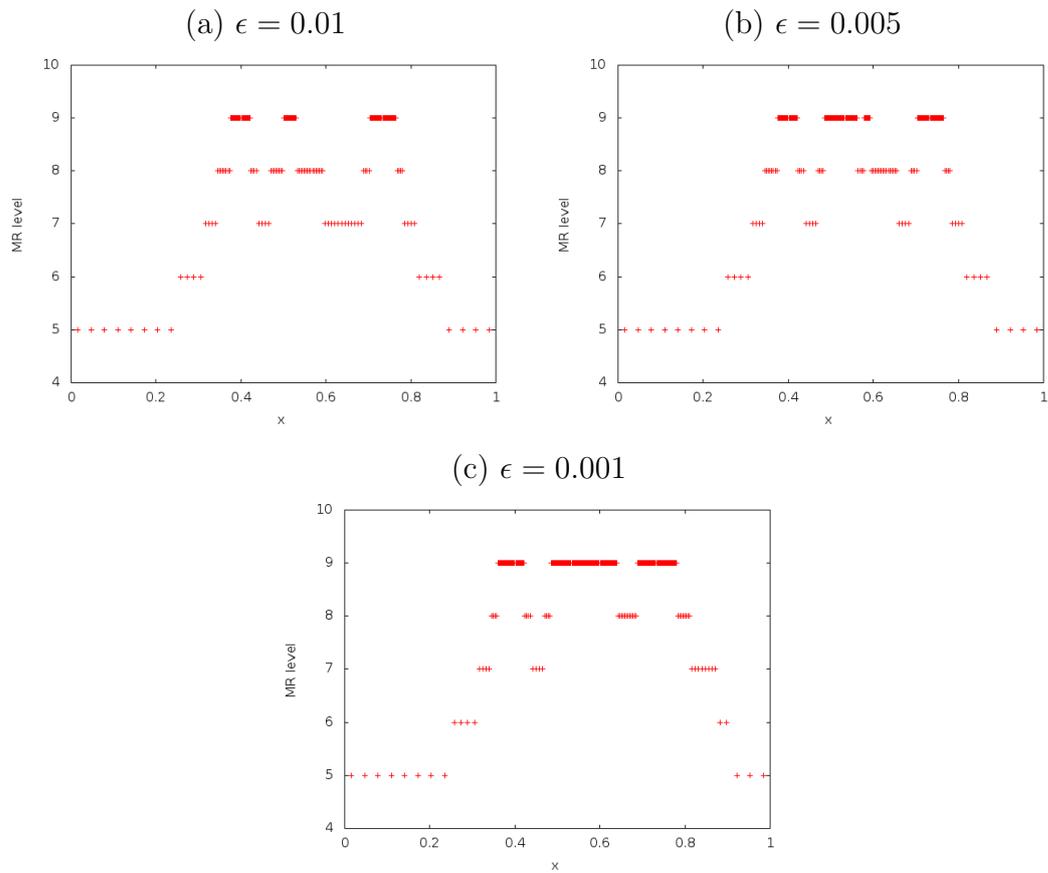


Figura 9.9 - Malhas adaptativas referentes ao Caso 1, para os diferentes valores de  $\epsilon$  utilizados, obtidas ao aproximar a solução com o fluxo HLLD.

Na Tabela 9.1, tem-se o tempo de CPU gasto para aproximar a solução do modelo EGLM-MHD de acordo com o valor de  $\epsilon$  escolhido e, também, o tempo gasto pelo código MHD-FV.

Tabela 9.1 - Tempos de CPU, compressão de CPU e memória para os diferentes valores de  $\epsilon$  utilizados e para o caso não-adaptativo do código MHD-FV. As malhas utilizadas têm  $512 \times 512$  células,  $c_{CFL} = 0.3$  e  $t = 0.1$ .

<b>Tempo de CPU (min)</b>				
Fluxo	MR			FV
	$\epsilon$			
	0.01	0.005	0.001	
HLL	11.63	13.63	17.65	65.08
HLLD	16.17	18.15	20.37	65.00
<b>Compressão de CPU (%)</b>				
Fluxo	MR			FV
	$\epsilon$			
	0.01	0.005	0.001	
HLL	17.87	20.94	27.12	100
HLLD	24.88	27.92	31.34	100
<b>Compressão de memória (%)</b>				
Fluxo	MR			FV
	$\epsilon$			
	0.01	0.005	0.001	
HLL	23.05	26.60	34.10	100
HLLD	27.13	30.11	33.31	100

A partir dos valores da Tabela 9.1, tem-se que o tempo de CPU referente à aproximação utilizando o algoritmo adaptativo é significativamente menor em relação ao tempo de CPU gasto pela aproximação sem o algoritmo adaptativo (MHD-FV). Para o menor valor de  $\epsilon$  utilizado, com o qual obtém-se uma representação eficiente, o tempo computacional utilizado chega a ser, em média, 50% menor. Já no caso dos valores  $\epsilon = 0.01$  e  $\epsilon = 0.005$ , há uma redução de, em média, 70% no tempo de CPU. Uma redução do custo computacional com tal magnitude é de extrema importância, principalmente, quando se trabalha com uma malha de simulação muito grande e esquemas numéricos mais caros computacionalmente.

Para completar a análise dos resultados, é importante avaliar o valor dos erros obtidos na simulação da solução do modelo EGLM-MHD. A comparação da solução

exata é feita, matematicamente, pelo erro na norma  $L^1$ . Na Tabela 9.2, tem-se o erro na norma  $L^1$  entre as soluções exata e aproximada, a partir dos valores obtidos na simulação da solução do modelo EGLM-MHD.

Tabela 9.2 - Caso 1: Erro na norma  $L_1$  calculado entre as soluções exata e aproximada, para os fluxos HLL e HLLD e diferentes valores de  $\epsilon$  e, também, para o caso não-adaptativo do código MHD-FV.

MHD-FV				
Fluxo	Variáveis			
	$\rho$	$p$	$v_x$	$v_y$
HLL	$10.3 \cdot 10^{-03}$	$17.2 \cdot 10^{-03}$	$6.70 \cdot 10^{-03}$	$7.08 \cdot 10^{-03}$
HLLD	$7.43 \cdot 10^{-03}$	$11.5 \cdot 10^{-03}$	$5.99 \cdot 10^{-03}$	$5.28 \cdot 10^{-03}$
	$v_z$	$B_x$	$B_y$	$B_z$
HLL	$7.49 \cdot 10^{-03}$	$4.16 \cdot 10^{-07}$	$9.91 \cdot 10^{-03}$	$10.5 \cdot 10^{-03}$
HLLD	$4.01 \cdot 10^{-03}$	$4.16 \cdot 10^{-07}$	$7.76 \cdot 10^{-03}$	$6.26 \cdot 10^{-03}$
$\epsilon = 0.001$				
Fluxo	Variáveis			
	$\rho$	$p$	$v_x$	$v_y$
HLL	$11.1 \cdot 10^{-03}$	$18.5 \cdot 10^{-03}$	$7.80 \cdot 10^{-03}$	$7.26 \cdot 10^{-03}$
HLLD	$8.23 \cdot 10^{-03}$	$12.9 \cdot 10^{-03}$	$7.12 \cdot 10^{-03}$	$5.55 \cdot 10^{-03}$
	$v_z$	$B_x$	$B_y$	$B_z$
HLL	$7.70 \cdot 10^{-03}$	0.00	$10.6 \cdot 10^{-03}$	$11.2 \cdot 10^{-03}$
HLLD	$4.28 \cdot 10^{-03}$	$5.86 \cdot 10^{-12}$	$8.59 \cdot 10^{-03}$	$6.94 \cdot 10^{-03}$
$\epsilon = 0.005$				
Fluxo	Variáveis			
	$\rho$	$p$	$v_x$	$v_y$
HLL	$11.8 \cdot 10^{-03}$	$20.6 \cdot 10^{-03}$	$8.01 \cdot 10^{-03}$	$7.53 \cdot 10^{-03}$
HLLD	$8.28 \cdot 10^{-03}$	$1.30 \cdot 10^{-02}$	$7.14 \cdot 10^{-03}$	$5.54 \cdot 10^{-03}$
	$v_z$	$B_x$	$B_y$	$B_z$
HLL	$8.89 \cdot 10^{-03}$	$6.64 \cdot 10^{-11}$	$10.9 \cdot 10^{-03}$	$12.8 \cdot 10^{-03}$
HLLD	$4.41 \cdot 10^{-03}$	$2.73 \cdot 10^{-11}$	$8.59 \cdot 10^{-03}$	$7.09 \cdot 10^{-03}$
$\epsilon = 0.01$				
Fluxo	Variáveis			
	$\rho$	$p$	$v_x$	$v_y$
HLL	$13.1 \cdot 10^{-03}$	$23.4 \cdot 10^{-03}$	$8.63 \cdot 10^{-03}$	$8.05 \cdot 10^{-03}$
HLLD	$8.75 \cdot 10^{-03}$	$14.0 \cdot 10^{-03}$	$7.61 \cdot 10^{-03}$	$5.78 \cdot 10^{-03}$
	$v_z$	$B_x$	$B_y$	$B_z$
HLL	$10.3 \cdot 10^{-03}$	$1.95 \cdot 10^{-10}$	$11.4 \cdot 10^{-03}$	$14.9 \cdot 10^{-03}$
HLLD	$4.82 \cdot 10^{-03}$	$2.15 \cdot 10^{-11}$	$8.90 \cdot 10^{-03}$	$7.68 \cdot 10^{-03}$

De um modo geral, comparando proporcionalmente os erros computados e o ganho em CPU, tem-se um ganho em todos os casos estudados. Os valores dos erros na norma  $L_1$  para a solução do modelo EGLM-MHD foram obtidos a partir dos valores da solução exata e da solução numérica com o algoritmo de multirresolução, para verificar o quanto a solução adaptativa se aproxima da solução exata para um determinado  $\epsilon$ .

A representação multirresolução para  $\epsilon = 0.005$  da solução aproximada pelo fluxo numérico HLLD é mais satisfatória. Comparada com a solução aproximada pelo código MHD-FV, a solução aproximada pelo CARMEN mostrou-se mais vantajosa em todos os aspectos que foram analisados nessa seção.

O próximo resultado de verificação deste trabalho é a abordagem numérica de multirresolução adaptativa para a condição inicial tipo Riemann 2D, como apresentado na seção seguinte.

## **9.2 Condição Inicial tipo Riemann 2D**

Nesta seção, é abordado o Caso 2 para o modelo EGLM-MHD, como apresentado no Capítulo 6. Utilizando a condição inicial tipo Riemann 2D da Tabela 6.3, uma malha de  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ , para  $t = 0.1$ , obtém-se a solução numérica do modelo EGLM-MHD.

### 9.2.1 Resultados: Fluxo HLL

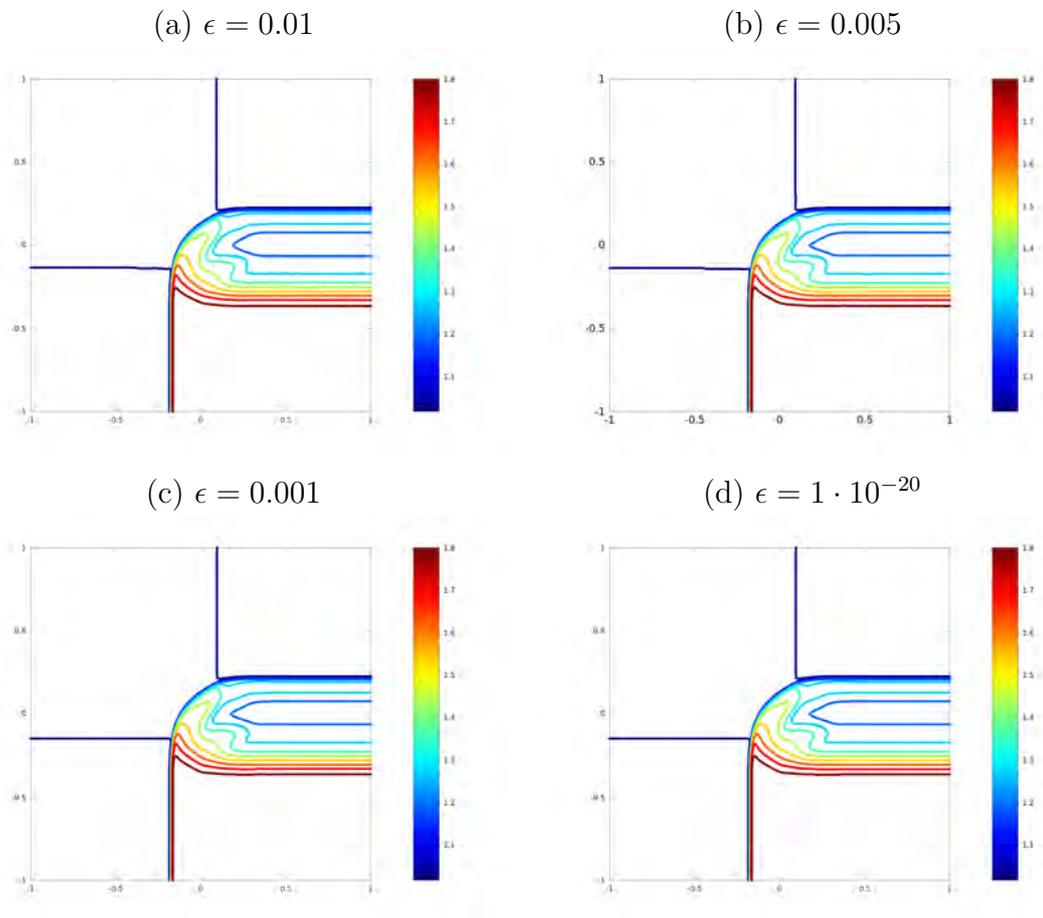


Figura 9.10 - Variável densidade aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

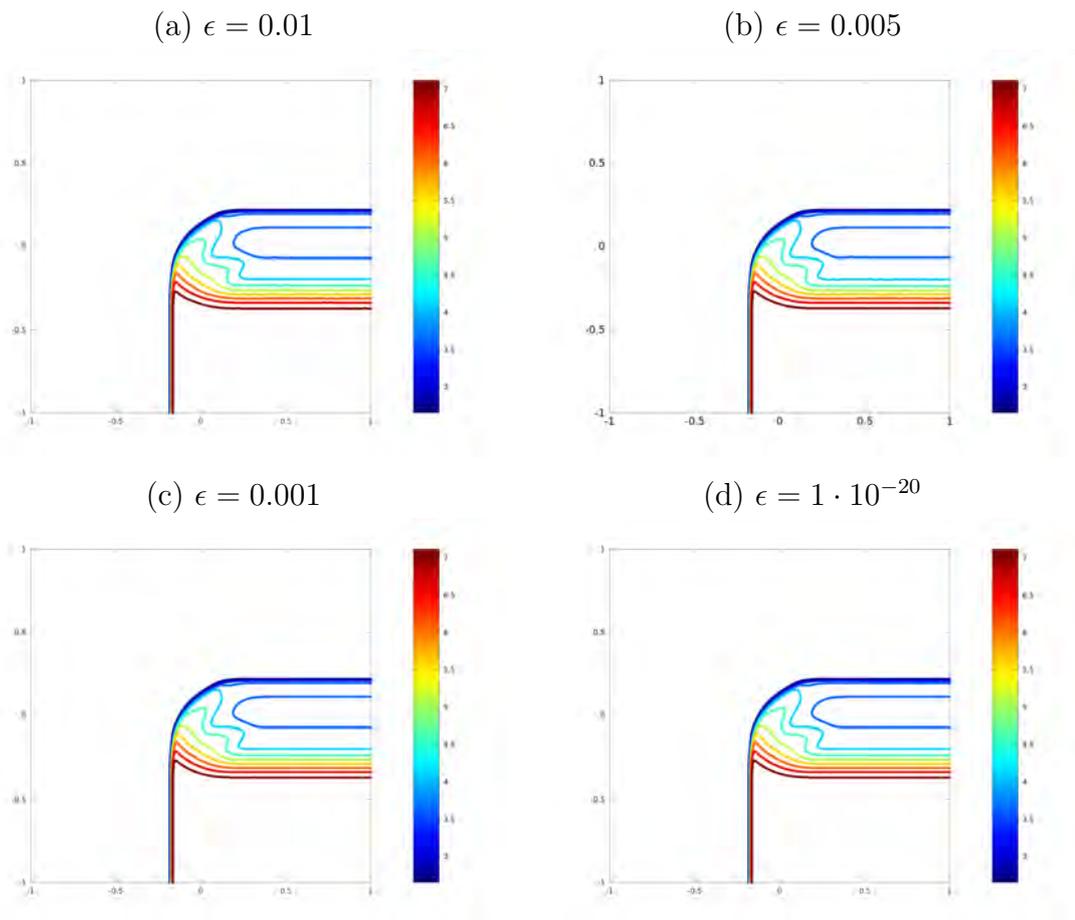


Figura 9.11 - Variável pressão aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

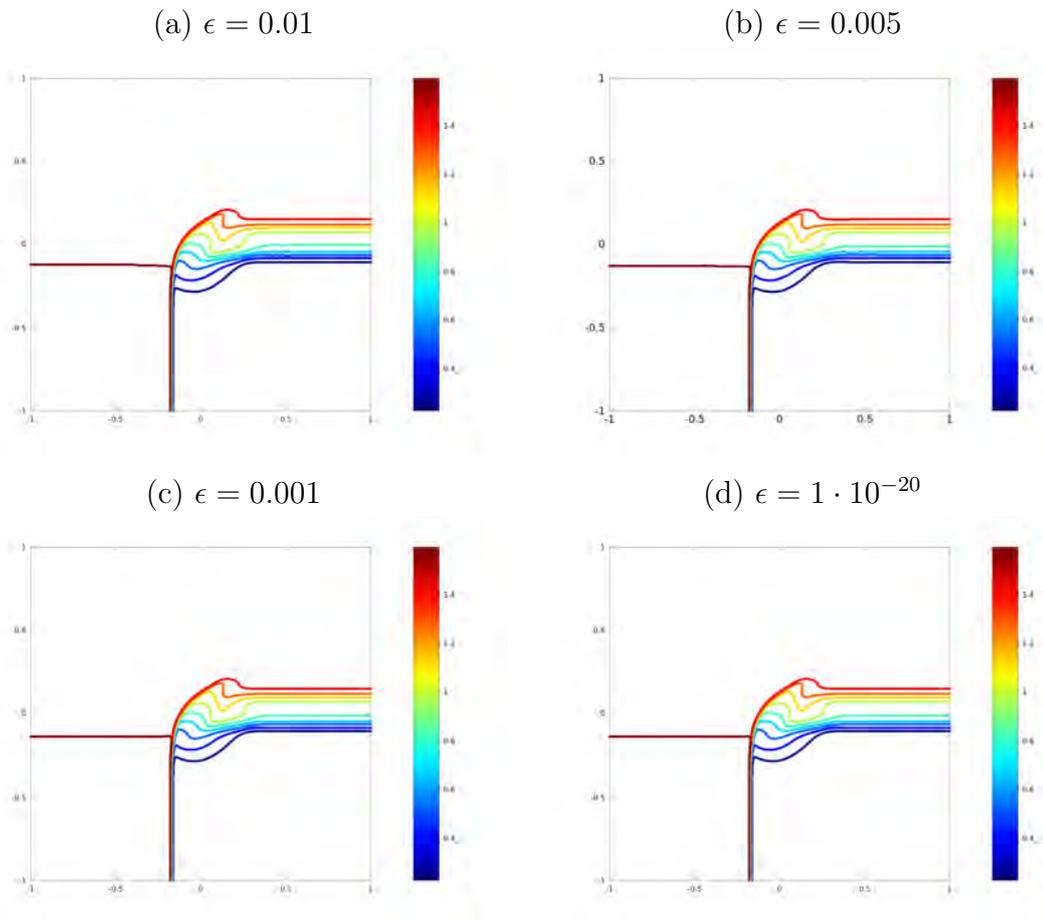


Figura 9.12 - Componente  $x$  da velocidade aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

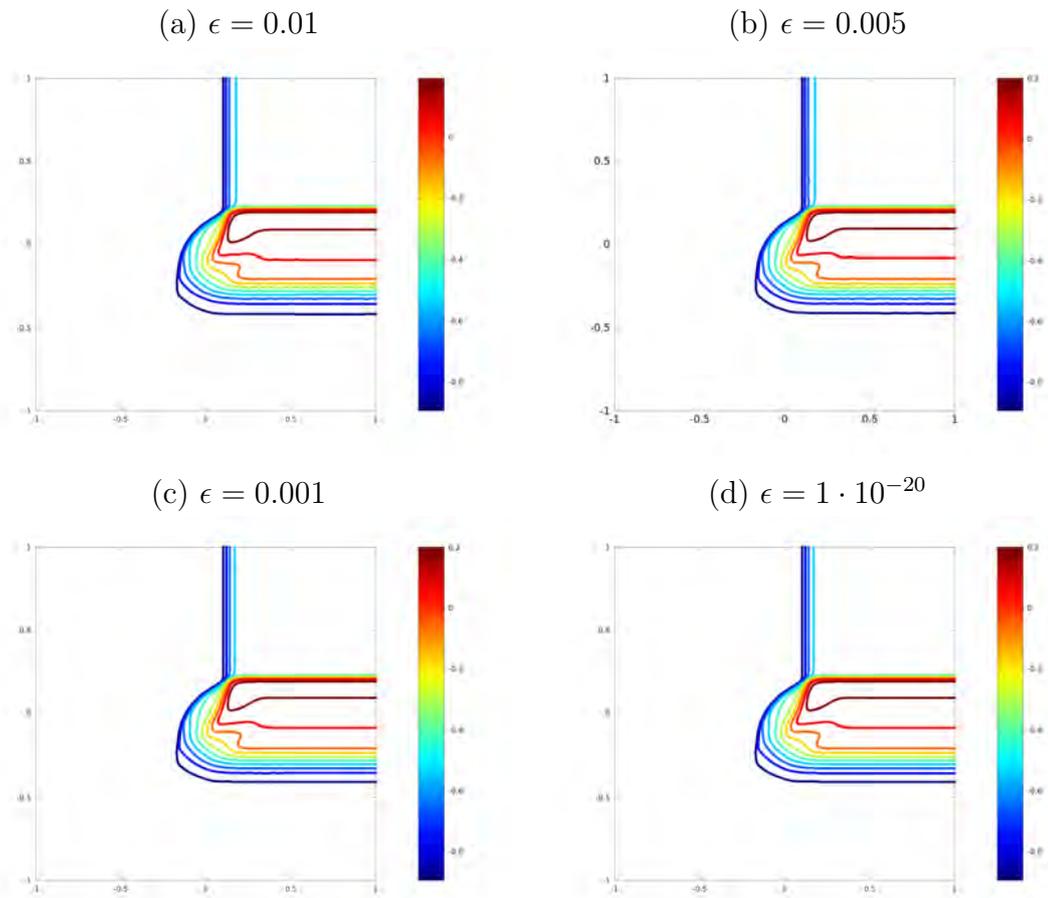


Figura 9.13 - Componente  $y$  da velocidade aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

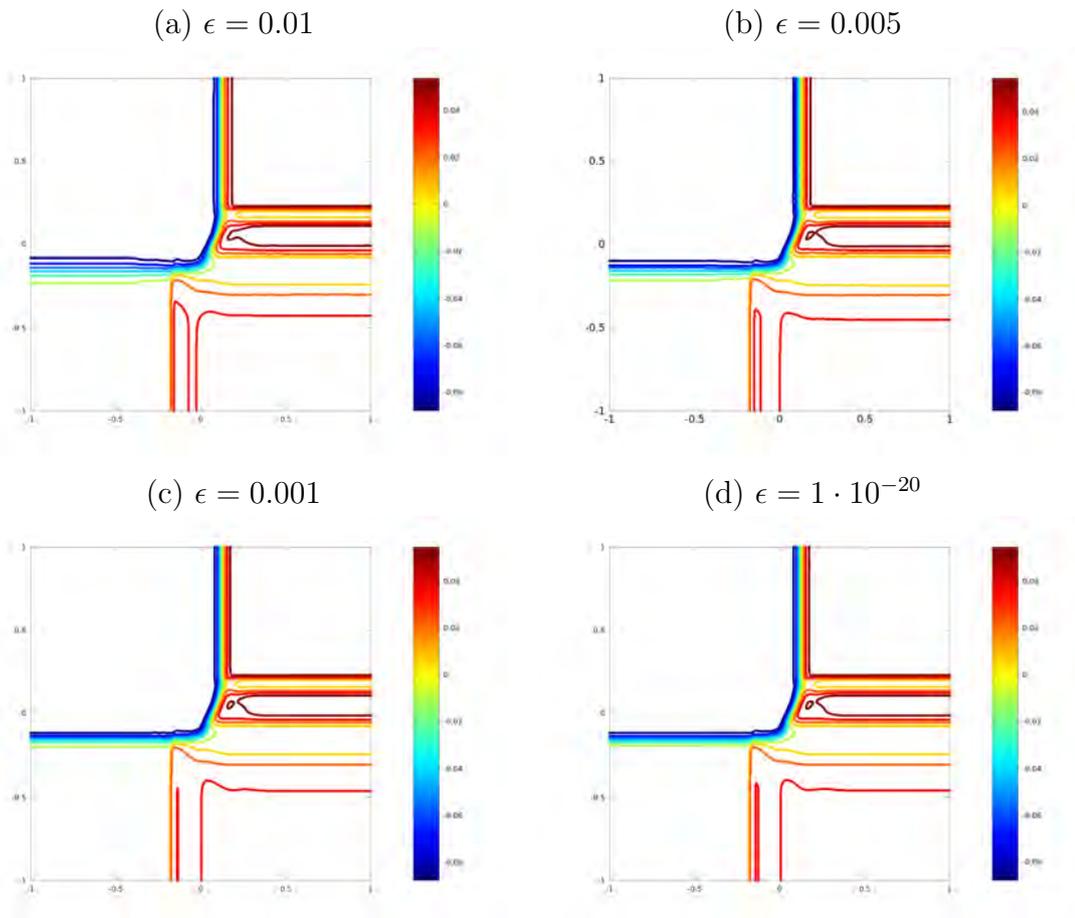


Figura 9.14 - Componente  $z$  da velocidade aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

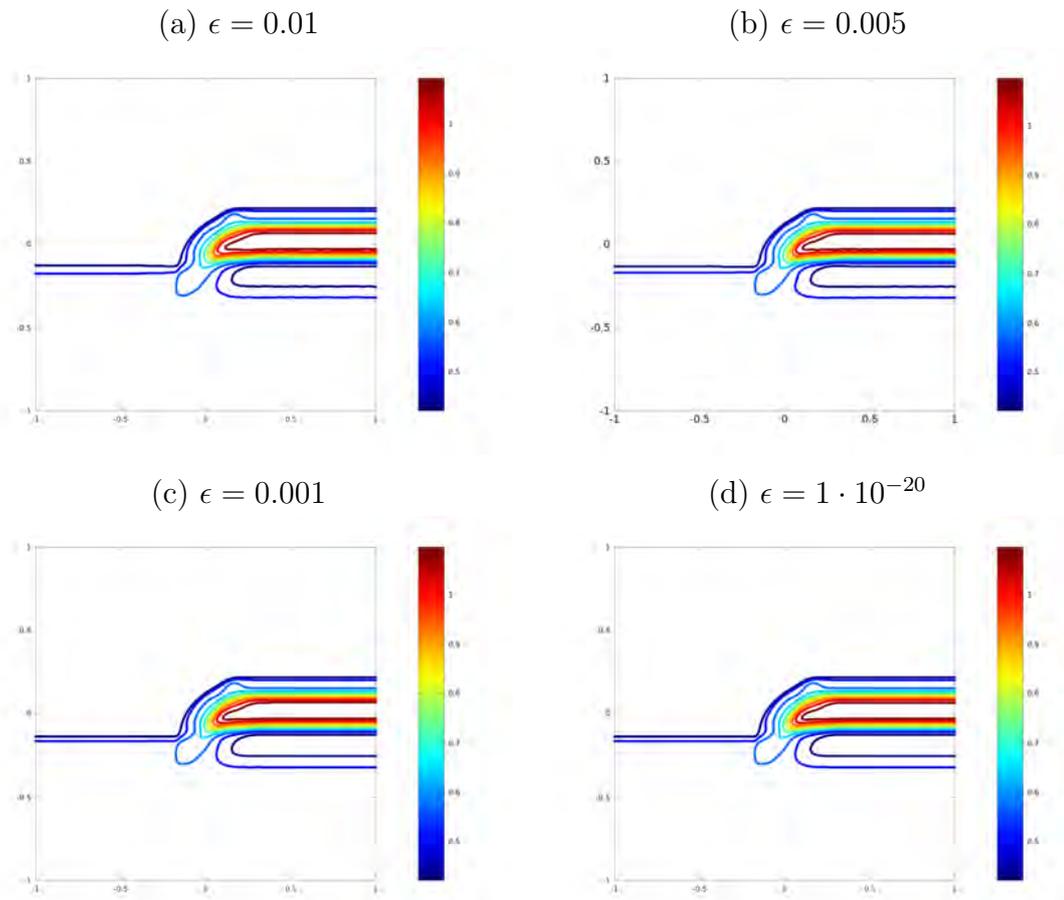


Figura 9.15 - Componente  $x$  do campo magnético aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

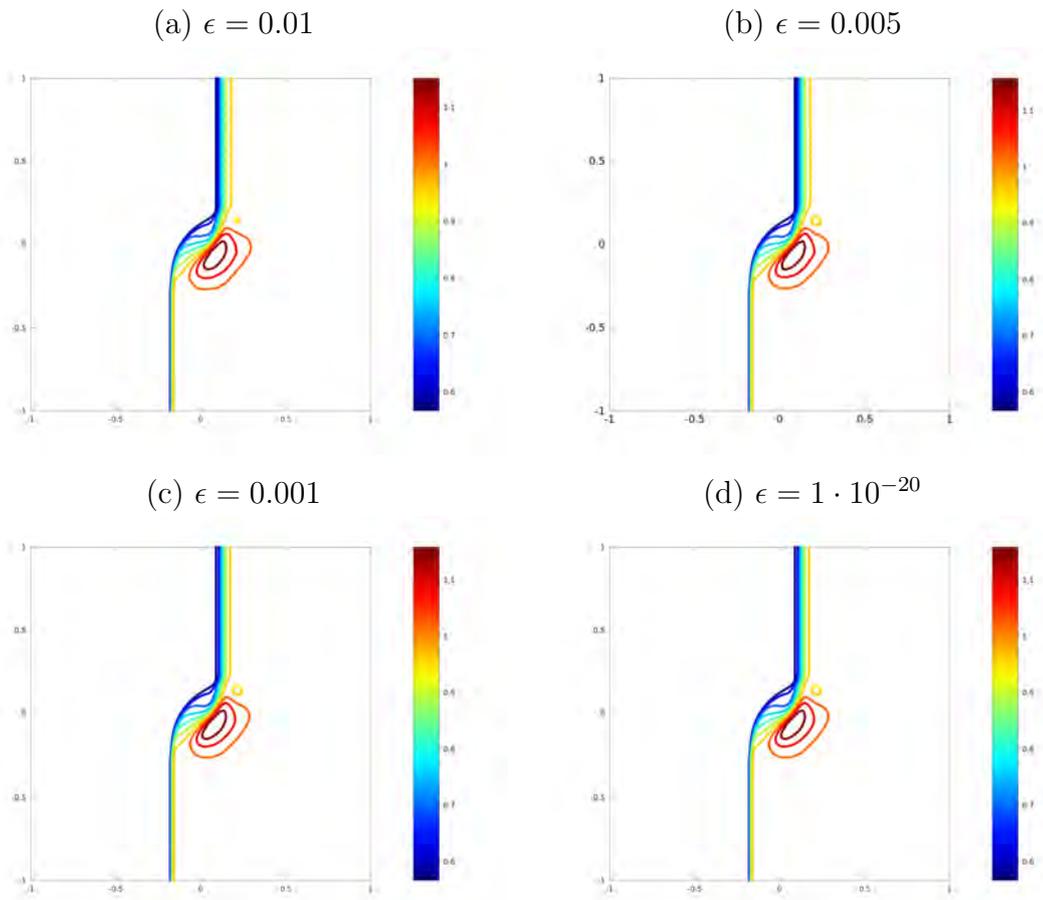


Figura 9.16 - Componente  $y$  do campo magnético aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

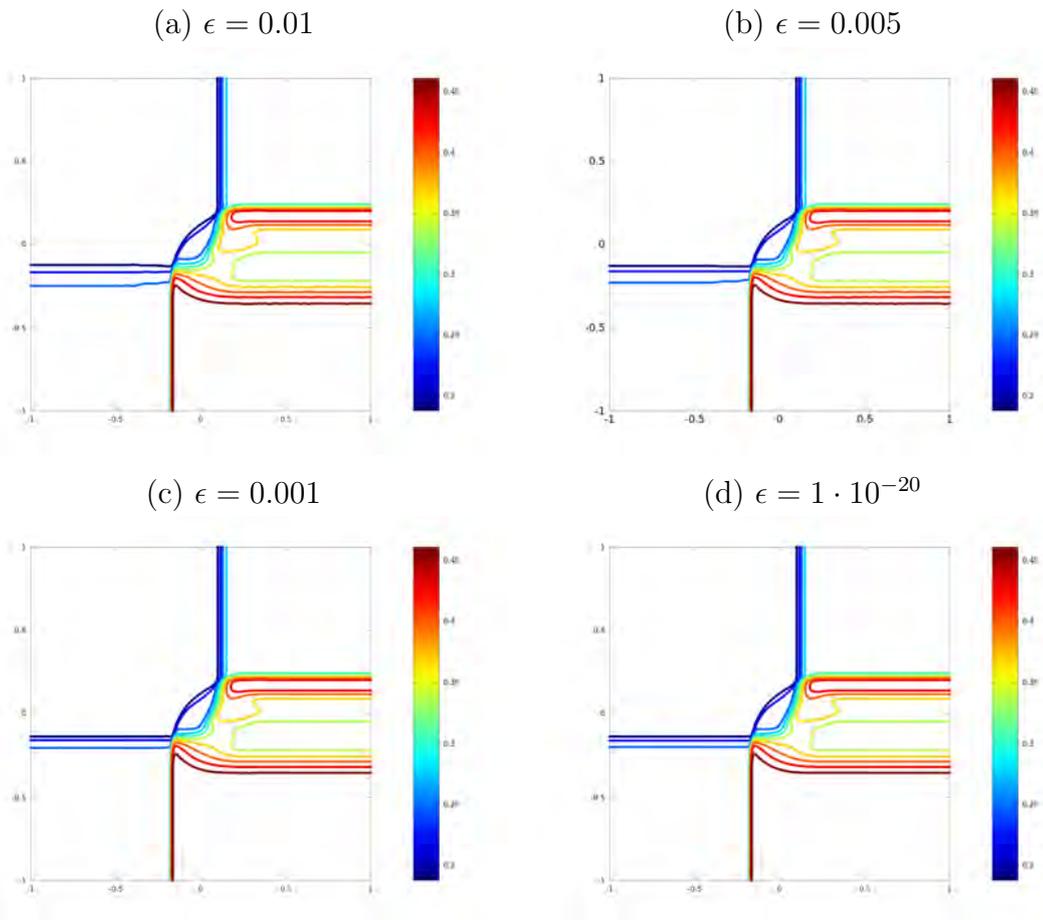


Figura 9.17 - Componente  $z$  do campo magnético aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

## 9.2.2 Resultados: Fluxo HLLD

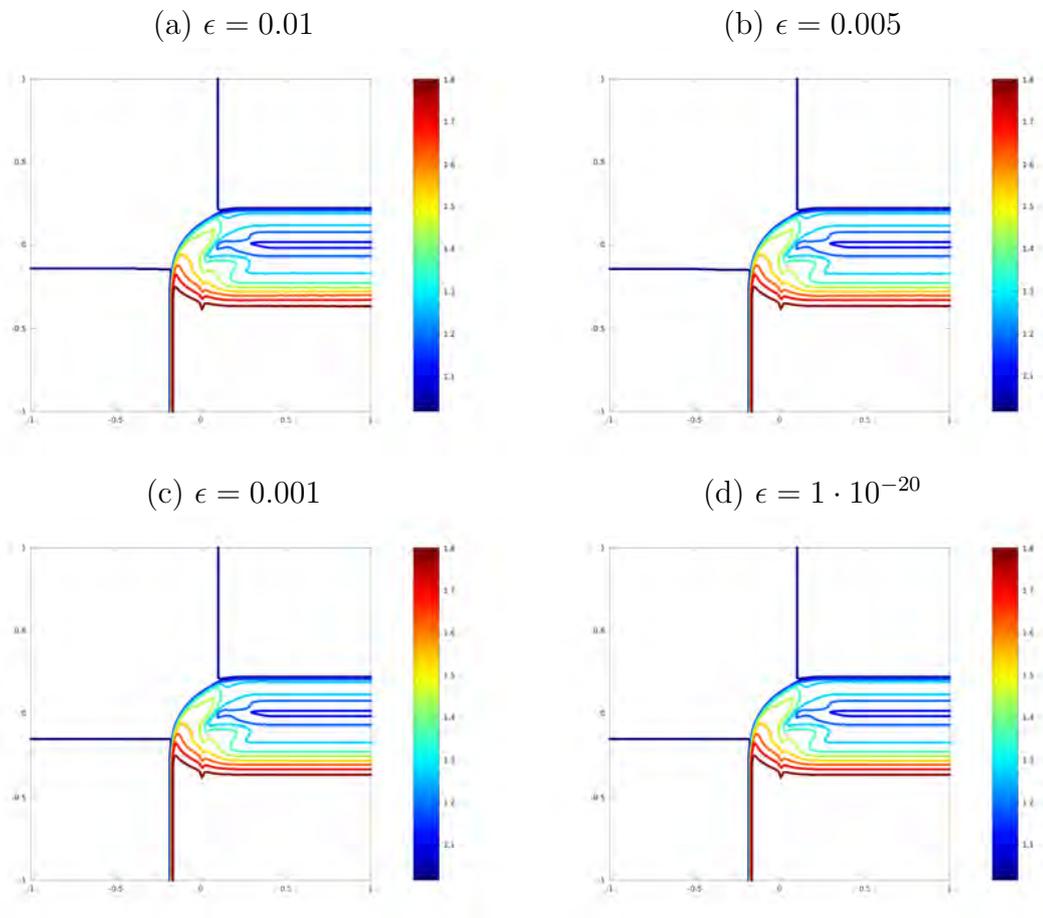


Figura 9.18 - Variável densidade aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

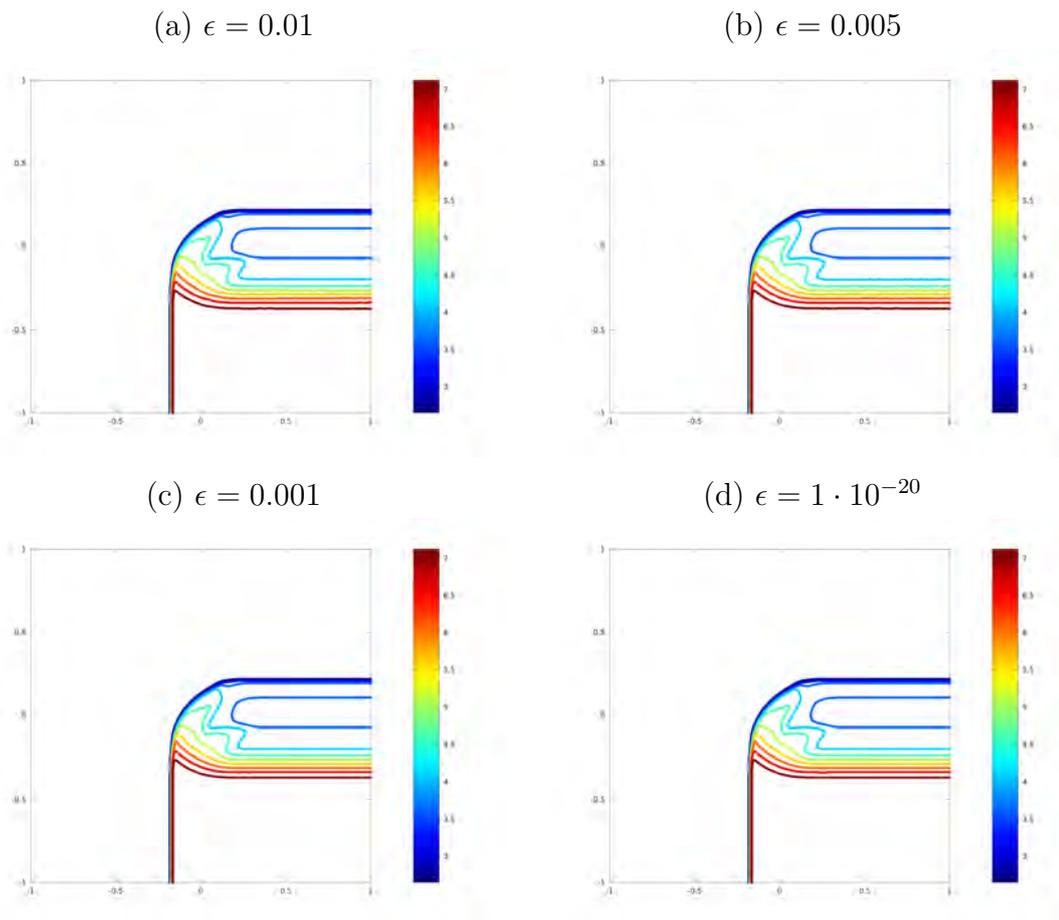


Figura 9.19 - Variável pressão aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

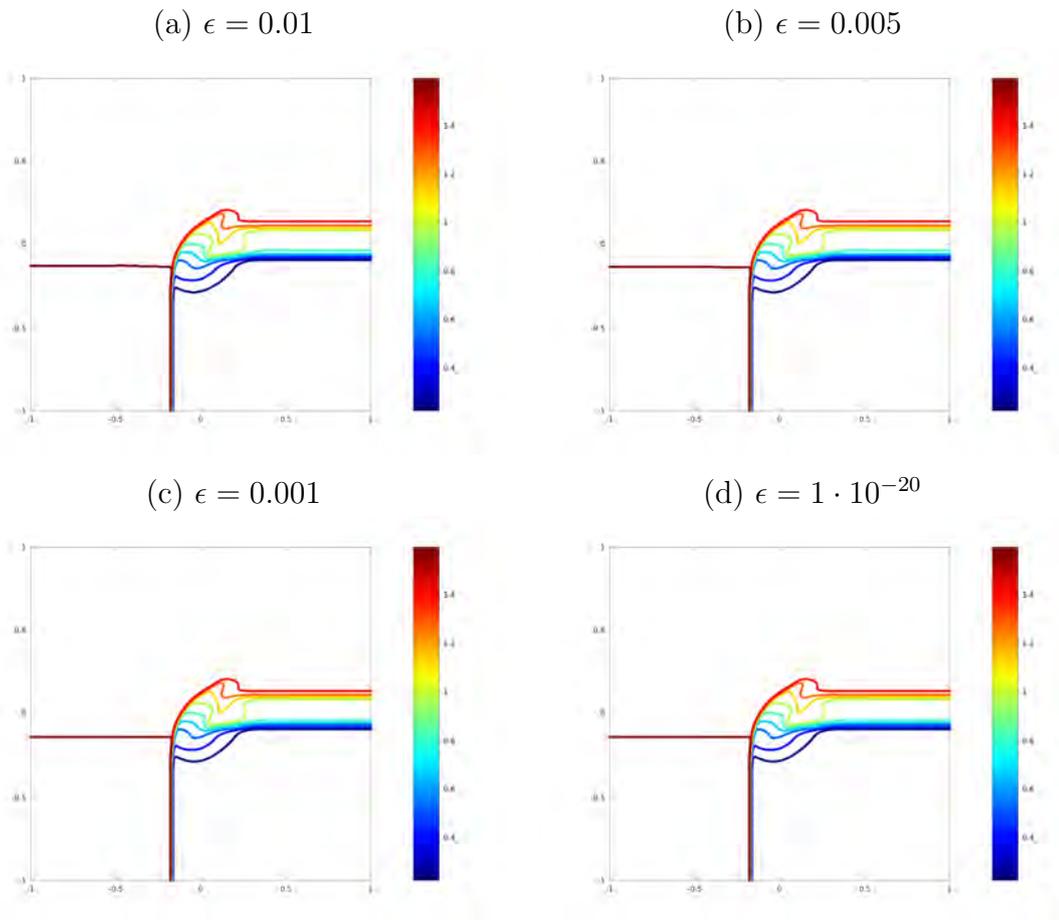


Figura 9.20 - Componente  $x$  da velocidade aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

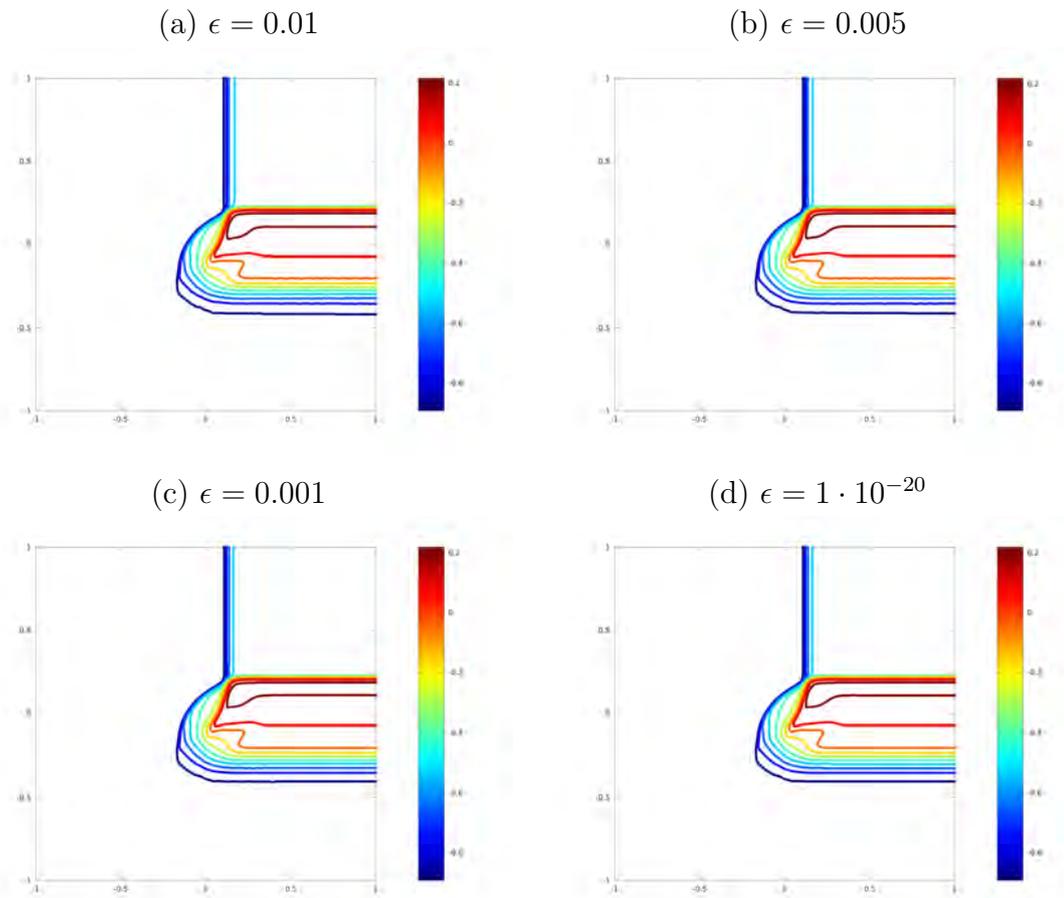


Figura 9.21 - Componente  $y$  da velocidade aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

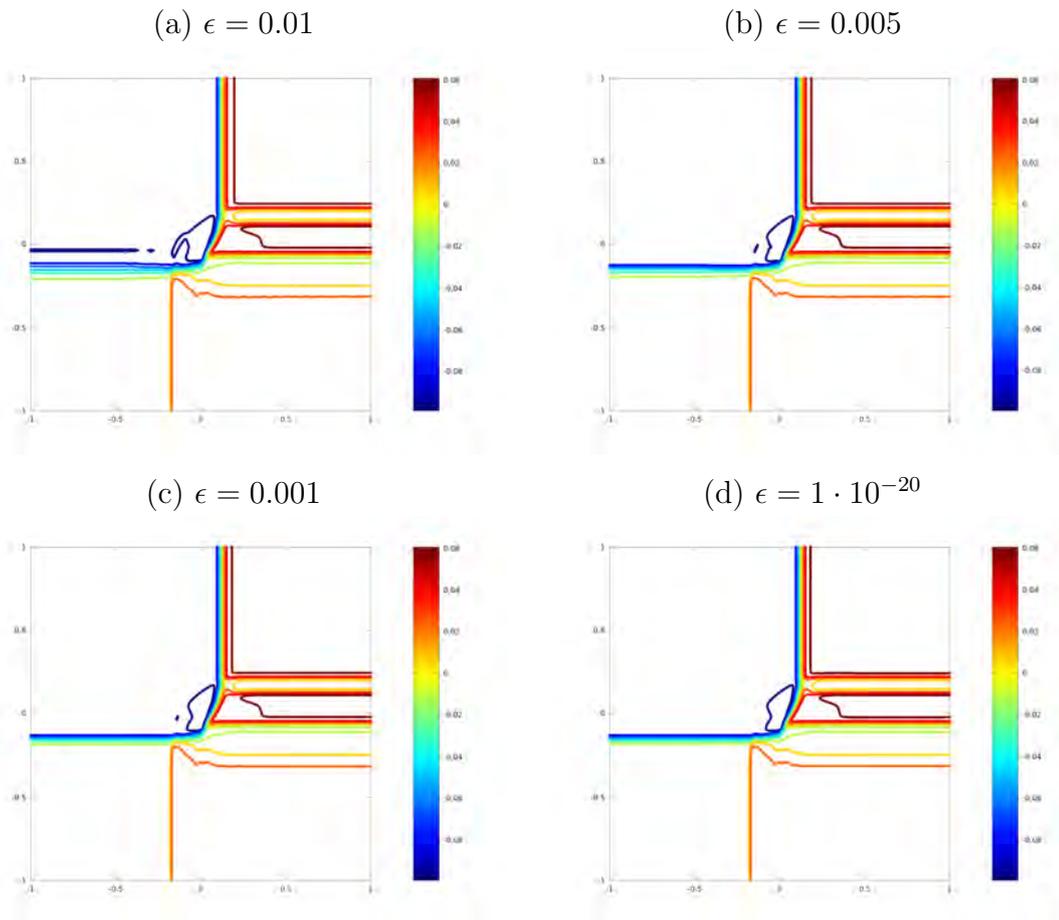


Figura 9.22 - Componente  $z$  da velocidade aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

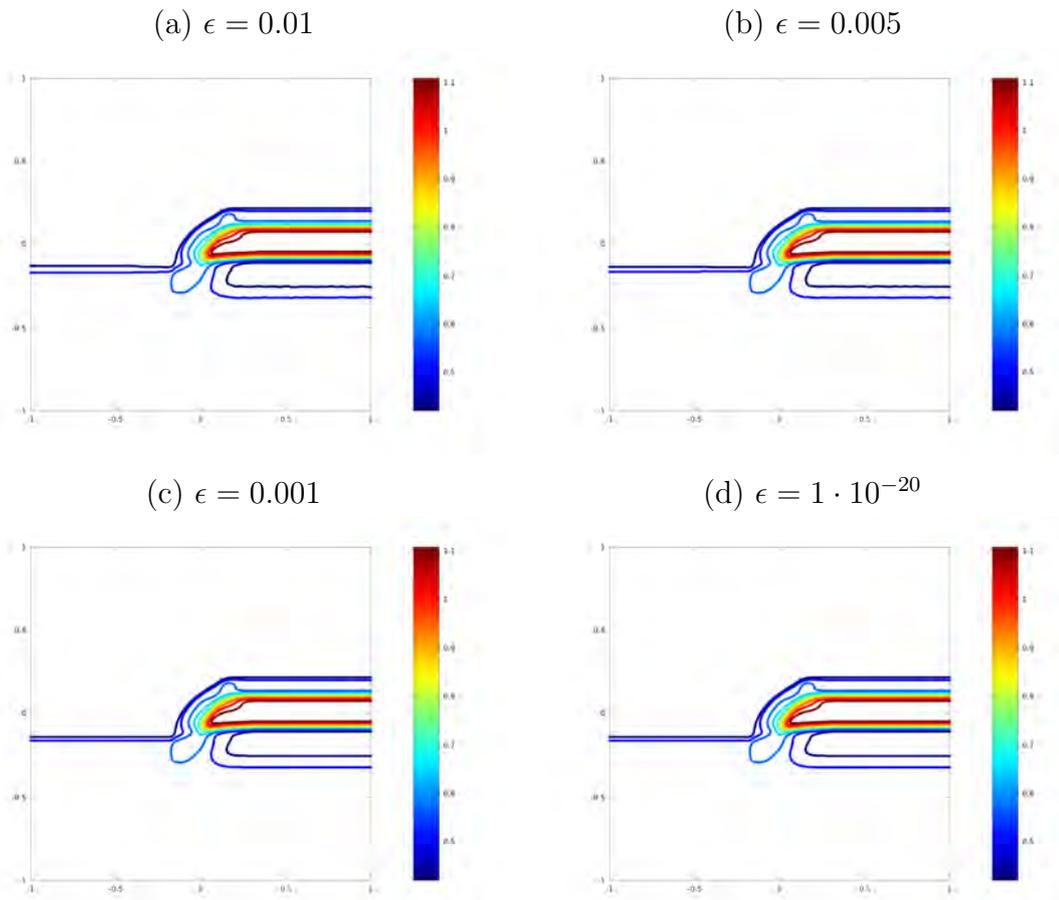


Figura 9.23 - Componente  $x$  do campo magnético aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

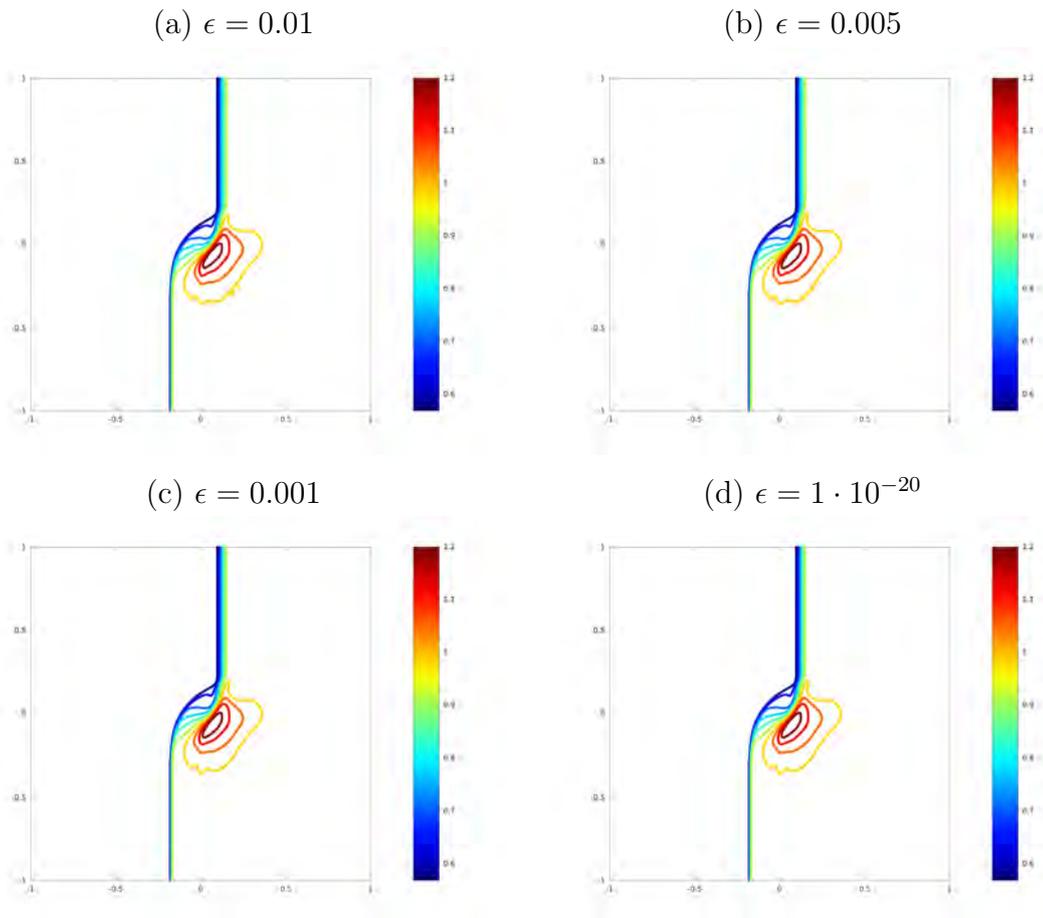


Figura 9.24 - Componente  $y$  do campo magnético aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

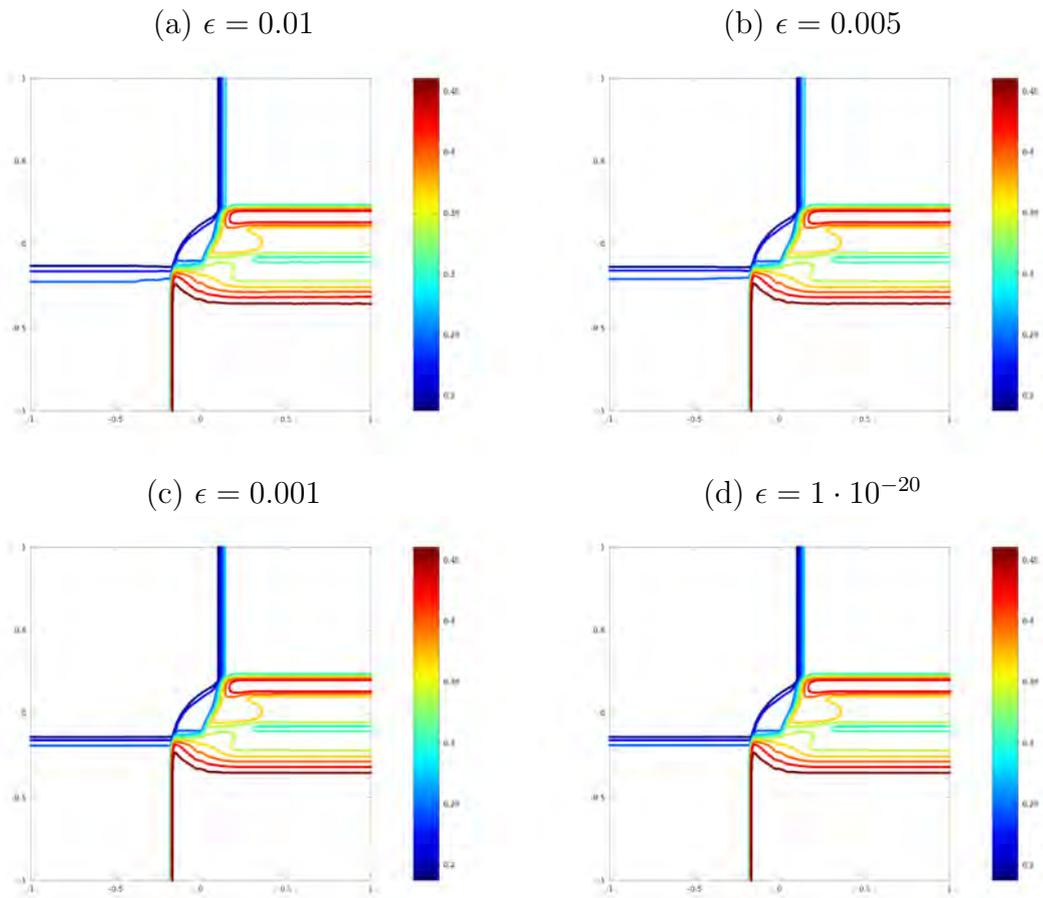


Figura 9.25 - Componente  $z$  do campo magnético aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $512 \times 512$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

A seguir, as malhas adaptativas referentes à simulação do Caso 2, para os valores de  $\epsilon$  iguais a 0.01, 0.005, 0.001 e  $1 \cdot 10^{-20}$ , obtidas ao utilizar os fluxos HLL e HLLD.

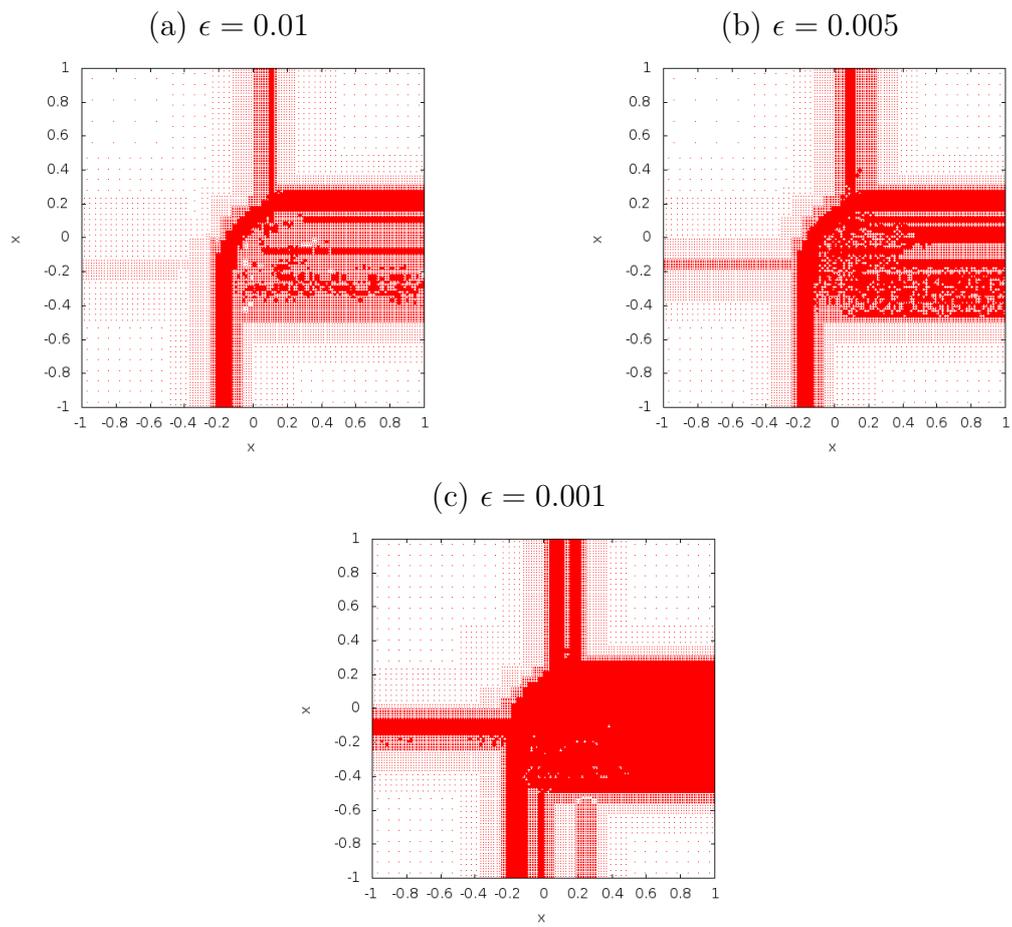


Figura 9.26 - Malhas adaptativas referentes ao Caso 2, para os diferentes valores de  $\epsilon$  utilizados, obtidas ao aproximar a solução com o fluxo HLL.

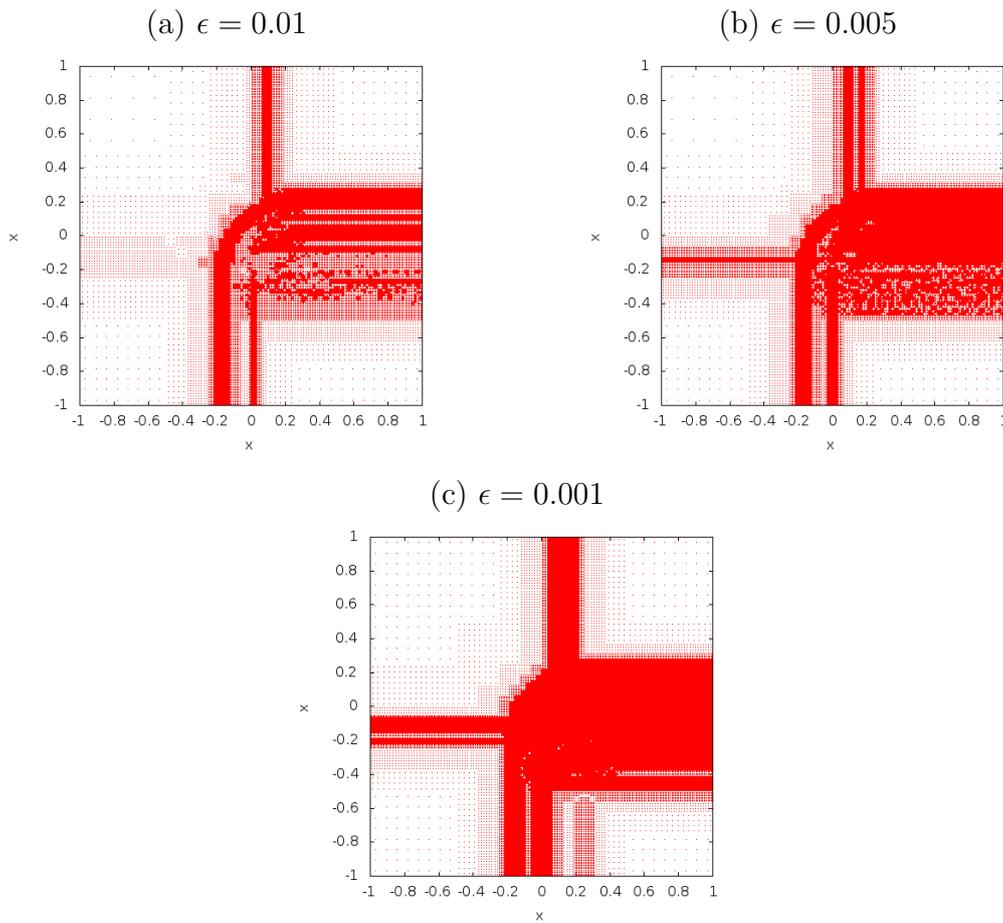


Figura 9.27 - Malhas adaptativas referentes ao Caso 2, para os diferentes valores de  $\epsilon$  utilizados, obtidas ao aproximar a solução com o fluxo HLLD.

Nas Figuras 9.26 e 9.27, principalmente para  $\epsilon = 0.01$  e  $\epsilon = 0.005$ , é possível observar onde o algoritmo adaptativo localiza as estruturas presentes na solução do modelo EGLM-MHD. Fazendo um comparativo entre os fluxos HLL e HLLD, para esses valores de  $\epsilon$  citados, observa-se, também, que existe uma localização de estruturas na malha para o fluxo HLLD superior à localização dessas estruturas para o fluxo HLL, como esperado. Fazendo, assim, com que o fluxo HLLD necessite de um valor de  $\epsilon$  menor para obter a representação dessa solução.

Na Tabela 9.3, faz-se um comparativo entre os tempos de CPU, compressão de CPU e compressão de memória de cada caso adaptativo, juntamente com o caso não-adaptativo (MHD-FV).

Tabela 9.3 - Tempos de CPU, compressão de CPU e memória para os diferentes valores de  $\epsilon$  utilizados e para o caso não-adaptativo do código MHD-FV. As malhas utilizadas têm  $512 \times 512$  células,  $c_{CFL} = 0.3$  e  $t = 0.1$ .

<b>Tempo de CPU (min)</b>					
Fluxo	MR				FV
	$\epsilon$				
	0.01	0.005	0.001	$1 \cdot 10^{-20}$	
HLL	9.27	11.03	13.52	33.78	44.82
HLLD	11.53	13.65	15.78	40.68	43.87
<b>Compressão de CPU (%)</b>					
Fluxo	MR				FV
	$\epsilon$				
	0.01	0.005	0.001	$1 \cdot 10^{-20}$	
HLL	20.68	24.61	30.16	75.37	100
HLLD	26.28	31.11	35.97	92.73	100
<b>Compressão de memória (%)</b>					
Fluxo	MR				FV
	$\epsilon$				
	0.01	0.005	0.001	$1 \cdot 10^{-20}$	
HLL	24.22	28.07	34.50	84.78	100
HLLD	25.89	29.42	34.34	85.02	100

Para o Caso 2, não existe uma solução exata para fins de comparação. Por isso, calcula-se uma solução de referência dois níveis mais refinada que a utilizada nesse caso, ou seja, obtém-se uma solução numérica para uma malha com  $2048 \times 2048$  células (ver Apêndice F). A partir dessa solução de referência, obtém-se a solução dois níveis menos refinada,  $512 \times 512$ , e faz-se a comparação com o teste simulado nesse trabalho (Caso 2). Na Tabela 9.4, calcula-se o erro  $L^1$  entre a solução de referência e a solução teste para o Caso 2.

Tabela 9.4 - Erro na norma  $L_1$  calculada entre as soluções exata e aproximada, para os fluxos HLL e HLLD e diferentes valores de  $\epsilon$ . A solução de referência é calculada com o fluxo HLLD.

$\epsilon = 0.001$				
Fluxo	Variáveis - $10^{-03}$			
	$\rho$	$p$	$v_x$	$v_y$
HLL	$20.0 \cdot 10^{-03}$	$102.3 \cdot 10^{-03}$	$33.5 \cdot 10^{-03}$	$27.1 \cdot 10^{-03}$
HLLD	$16.4 \cdot 10^{-03}$	$80.2 \cdot 10^{-03}$	$16.7 \cdot 10^{-03}$	$20.8 \cdot 10^{-03}$
	$v_z$	$B_x$	$B_y$	$B_z$
HLL	$7.32 \cdot 10^{-03}$	$29.2 \cdot 10^{-03}$	$10.5 \cdot 10^{-03}$	$10.3 \cdot 10^{-03}$
HLLD	$3.63 \cdot 10^{-03}$	$12.7 \cdot 10^{-03}$	$6.54 \cdot 10^{-03}$	$6.43 \cdot 10^{-03}$
$\epsilon = 0.005$				
Fluxo	Variáveis			
	$\rho$	$p$	$v_x$	$v_y$
HLL	$20.9 \cdot 10^{-03}$	$106.0 \cdot 10^{-03}$	$34.3 \cdot 10^{-03}$	$28.6 \cdot 10^{-03}$
HLLD	$17.3 \cdot 10^{-03}$	$85.6 \cdot 10^{-03}$	$1.83 \cdot 10^{-03}$	$22.9 \cdot 10^{-03}$
	$v_z$	$B_x$	$B_y$	$B_z$
HLL	$7.94 \cdot 10^{-03}$	$30.0 \cdot 10^{-03}$	$11.1 \cdot 10^{-03}$	$11.0 \cdot 10^{-03}$
HLLD	$4.54 \cdot 10^{-03}$	$14.4 \cdot 10^{-03}$	$7.55 \cdot 10^{-03}$	$7.42 \cdot 10^{-03}$
$\epsilon = 0.01$				
Fluxo	Variáveis			
	$\rho$	$p$	$v_x$	$v_y$
HLL	$21.4 \cdot 10^{-03}$	$109.4 \cdot 10^{-03}$	$34.8 \cdot 10^{-03}$	$29.5 \cdot 10^{-03}$
HLLD	$18.1 \cdot 10^{-03}$	$90.1 \cdot 10^{-03}$	$20.4 \cdot 10^{-03}$	$24.3 \cdot 10^{-03}$
	$v_z$	$B_x$	$B_y$	$B_z$
HLL	$8.25 \cdot 10^{-03}$	$30.6 \cdot 10^{-03}$	$11.2 \cdot 10^{-03}$	$11.5 \cdot 10^{-03}$
HLLD	$5.38 \cdot 10^{-03}$	$16.7 \cdot 10^{-03}$	$7.99 \cdot 10^{-03}$	$8.40 \cdot 10^{-03}$

Observa-se, na Tabela 9.4, que à medida que  $\epsilon$  aumenta, o erro  $L^1$  da solução aproximada com o fluxo HLL se aproxima do erro do HLLD. Para  $\epsilon = 0.01$ , na maioria das variáveis, o erro na norma  $L^1$  para o caso HLL é menor que o erro na norma  $L^1$  para o caso HLLD.

Nos estudos realizados, a divergência de  $\mathbf{B}$  se manteve estabilizada. No caso da

condição inicial tipo Riemann 1D, ela foi zero utilizando o código MHD-FV, e próxima ao zero da máquina para no código carmen ( $\mathcal{O}(\infty^{-\infty\nabla})$ ). No caso da condição inicial tipo Riemann 2, em ambos os códigos o valor da divergência de  $\mathbf{B}$  estabilizou-se na ordem de  $10^{-5}$ . Esses valores da divergência elevados devem-se ao modelo implementado ser somente 2D. Para melhorar esses resultados de divergência no caso 2D, é necessário a implementação de um modelo 3D, usualmente chamado *2,5D*.

## 10 CONCLUSÕES E TRABALHOS FUTUROS

O estudo feito para esse trabalho envolveu uma abordagem de multirresolução adaptativa, a qual não havia ainda sido estudada no contexto do modelo EGLM-MHD, utilizando uma discretização em volumes finitos. Foram testadas duas diferentes condições iniciais do tipo Riemann e a solução desse modelo foi aproximada com dois diferentes tipos de fluxos numéricos, o HLL e o HLLD.

Para o caso analisado, o algoritmo de multirresolução adaptativa mostrou-se eficiente em termos de representação da solução e custo computacional para ambos os casos testados nesse trabalho. O refinamento da malha em uma simulação numérica utilizando um algoritmo de multirresolução, principalmente para o fluxo HLLD, é uma alternativa atraente e competitiva no contexto da magnetohidrodinâmica. Como pode-se verificar, a malha adaptativa, para o fluxo HLLD, localiza de maneira mais eficiente as estruturas presentes na solução do modelo EGLM-MHD, ao ser comparado com o fluxo HLL. A simulação da solução pode, ainda, vir a se tornar mais acurada, e o algoritmo adaptativo pode vir a localizar as estruturas presentes na solução de maneira ainda mais eficiente, desde que sejam utilizados fluxos numéricos que aproximam a solução do modelo ainda melhor do que os utilizados nesse trabalho, o que é um outro desafio nesse contexto.

Este trabalho foi um ponto de partida para a utilização do algoritmo multirresolução adaptativa em um modelo MHD, discretizado pelo método dos volumes finitos. Por ser um código inicial neste contexto MHD adaptativo, ainda há como otimizar esse código e é necessário continuar sua verificação em outros casos, a fim de obter soluções aproximadas em um tempo de CPU ainda mais eficiente do que os obtidos. Com todos os resultados e evidências apresentadas, conclui-se, então, que o algoritmo multirresolução adaptativa pode vir a ser uma alternativa competitiva, se comparada a outros métodos adaptativos mais conhecidos.

### Trabalhos Futuros

A partir deste trabalho, existem diversas possibilidades de trabalhos futuros. Algumas delas são apresentadas a seguir.

- Desenvolver código para calcular o modelo EGLM-MHD tridimensional com a abordagem adaptativa;

- Estudar do comportamento da divergência do campo magnético  $\mathbf{B}$  dentro da abordagem de multirresolução adaptativa;
- Implementar diferentes fluxos numéricos, a fim de melhorar a solução aproximada e a adaptatividade da malha;
- Implementar e verificar a solução do modelo EGLM-MHD com invariância de Galileu (DEDNER et al., 2002) para o caso adaptativo;
- Simular a solução utilizando outros tipos de condições iniciais, como as apresentadas em Wesenberg(2003);
- Implementar código resistivo.

## REFERÊNCIAS BIBLIOGRÁFICAS

- ABGRALL, R.; HARTEN, A. Multiresolution representation in unstructured meshes. **SIAM J. Numer. Anal.**, v. 35, n. 6, p. 2128–2146, 1998. 62
- BIHARI, B. L.; HARTEN, A. Multiresolution schemes for the numerical solution of 2-D conservation laws I. **SIAM J. Sci. Comput.**, v. 18, n. 2, p. 315–354, 1997. 73
- BITTENCOURT, J. **Fundamentals of plasma physics**. 3. ed.. ed. New York: Springer-Verlag New York, Inc., 2004. 6
- BRACKBILL, J. U.; BARNES, D. C. The effect of nonzero [backward difference] · b on the numerical solution of the magnetohydrodynamic equations. **Journal of Computational Physics**, v. 35, n. 3, p. 426 – 430, 1980. ISSN 0021-9991.  
Disponível em: <<http://www.sciencedirect.com/science/article/B6WHY-4DD1TWX-VK/2/a9f5c6809996b57224bbb6ec585950b6>>. 6
- CASTILHO, J. E.; DOMINGUES, M. O.; PAGAMISSE, A.; JR., O. M. **Introdução ao mundo das wavelets**. [S.l.]: SBMAC, 2012. 61
- COHEN, A. Wavelet methods in numerical analysis. In: CIARLET, P. G.; LIONS, J. L. (Ed.). **Handbook of numerical analysis**. Amsterdam: Elsevier, 2000. VII. 62, 69
- COHEN, A.; DAUBECHIES, I.; FEAUVEAU, J. C. Biorthogonal bases of compactly supported wavelets. **Comm. Pure Appl. Math.**, v. 45, p. 485–560, 1992. 68
- COURANT, R.; FRIEDRICHS, K.; LEWY, H. On the partial difference equations of mathematical physics. **IBM Journal of Research and Development**, v. 11, n. 2, p. 215 –234, march 1967. ISSN 0018-8646. 25
- DAI, W.; WOODWARD, P. R. A simple Riemann solver and high-order Godunov schemes for hyperbolic systems of conservation laws. **Journal of Computational Physics**, v. 121, p. 51–65, 1995. 16, 40, 55
- DEDNER, A.; KEMM, F.; KRÖNER, D.; MUNZ, C.-D.; T.SCHNITZER;  
WESENBERG, M. Hyperbolic divergence cleaning for the MHD equations. **Journal of Computational Physics**, v. 175, p. 645–673, 2002. xv, 2, 7, 8, 10, 12, 13, 15, 43, 44, 54, 55, 59, 116, 123, 131, 135

DEITERDING, R.; DOMINGUES, M. O.; GOMES, S. M.; ROUSSEL, O.; SCHNEIDER, K. Adaptive multiresolution or adaptive mesh refinement: a case study for 2D Euler equations. **ESAIM Proceedings**, v. 29, p. 28–42, 2009. 2, 73

DESLAURIERS, G.; DUBUC, S. Symmetric iterative interpolation processes. **Constructive Approximation**, v. 5, p. 49–68, 1989. 162

DOMINGUES, M. O.; FERREIRA, P. J. S. G.; GOMES, S. M.; GOMIDE, A.; PEREIRA, J. R.; PINHO, P. Grid structure impact in sparse point representation of derivatives. **Journal of Computational and Applied Mathematics**, v. 234, p. 2377–2389, 2010. 63, 77

DOMINGUES, M. O.; GOMES, S. M.; ROUSSEL, O.; SCHNEIDER, K. Adaptive multiresolution methods. **ESAIM Proceedings**, v. 34, p. 1–96, December 2011. 61, 64, 65, 72, 161, 164, 165, 168, 171

DONOHO, D. L. **Interpolating wavelet transforms**. San Francisco: [s.n.], October 1992. 167

DONOHO David L. Smooth wavelet decompositions with blocky coefficient kernels. In: SCHUMAKER, L. L.; WEBB, G. (Ed.). **Recent Advances in Wavelet Analysis**. San Francisco: Academic Press, 1993. v. 3, p. 364. 68

HARTEN, A. Multiresolution representation of data: a general framework. **SIAM**, v. 33, p. 1205–1256, 1996. 62, 170

HARTEN, A.; LAX, P. D.; LEER, B. V. On upstream differencing and Godunov-type scheme for hyperbolic conservation laws. **Society for Industrial and Applied Mathematics**, v. 25, p. 35–61, 1983. 1, 27, 29, 32

HOLMSTRÖM, M. **Wavelet based methods for time dependent PDEs**. Tese (Doutorado) — Uppsala University, 1997. 169

LEDVINA, S. A.; MA, Y. J.; KALLIO, E. Modeling and simulating flowing plasmas and related phenomena. **Space Science Reviews**, v. 139, p. 143–189, 2008. 1, 5

LEVEQUE, R. J. **Numerical methods for conservation laws**. Basel; Boston; Berlin: Birkhäuser Verlag, 1990. 214 p. 5, 11, 16

- \_\_\_\_\_. **Finite volume methods for hyperbolic problems**. Cambridge: Cambridge University Press, 2002. 558 p. [21](#), [24](#)
- LI, S. An HLLC Riemann solver for magneto-hydrodynamics. **Journal of Computational Physics**, v. 203, p. 344–357, 2005. [16](#), [44](#), [45](#), [131](#), [135](#)
- MALLAT, S. **A wavelet tour of signal processing**. Burlington: Academic Press, 1999. [62](#)
- MIYOSHI, T.; KUSANO, K. A multi-state HLL approximate riemann solver for ideal magnetohydrodynamics. **Journal of Computational Physics**, v. 208, p. 315–344, 2005. [xv](#), [1](#), [27](#), [29](#), [32](#), [35](#), [39](#), [40](#), [45](#), [46](#)
- PIERRO, B. D. Méthode d’annulation de la divergence pour les EDP hyperboliques application aux équations de la magnéto-hydrodynamique. 2009. [43](#), [131](#)
- POWELL, K. G. **An approximate Riemann solver for magnetohydrodynamics (that works in more than one dimension)**. Hampton: [s.n.], 1994. [123](#)
- ROUSELL, O.; SCHNEIDER, K.; TSIGULIN, A.; BOCKHORN, H. A conservative fully adaptive multiresolution algorithm for parabolic PDEs. **Journal of Computational Physics**, v. 188, p. 493–523, 2003. [129](#)
- ROUSSEL, O. **Developpement d’un algorithme multiresolution adaptatif tridimensionnel pour la resolution des equations aux derivees partielles paraboliques. application aux instabilites thermodiffusives de flamme**. Tese (Doutorado) — Universite de la Mediterranee, 2003. [79](#), [134](#)
- TORO, E. F. **Riemann solvers and numerical methods for fluid dynamics: a practical introduction**. 2. ed.. ed. New York; Barcelona; Hong Kong; London; Milan; Paris; Singapore; Tokyo: Springer-Verlag Berlin Heidelberg, 1999. [30](#), [31](#), [33](#)
- TORRILHON, M. **Exact solver and uniqueness conditions for Riemann problems of ideal magnetohydrodynamics**. Zürich: [s.n.], April 2002. [17](#)
- \_\_\_\_\_. Uniqueness conditions for Riemann problems of ideal magnetohydrodynamics. **Journal of Plasma Physics**, v. 96, p. 253–276, 2003. [16](#), [45](#)

TÓTH, G. The  $\nabla \cdot B$  constraint in shock-capturing magnetohydrodynamics codes. **Journal of Computational Physics**, v. 161, p. 605–652, 2000. 7

WESENBERG, M. **Efficient finite-volume schemes for magnetohydrodynamics simulations in solar physics**. Tese (Doutorado) — Albert-Ludwigs-Universität Freiburg im Breisgau, 2003. 16, 29, 40, 41, 116

## APÊNDICE A - TERMOS DO SISTEMA EGLM-MHD

No Capítulo 3, pág. 121, apresenta-se um sistema hiperbólico de EDPs pela Equação A.1

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = \mathcal{S}(\mathbf{U}), \quad (\text{A.1})$$

em que  $\mathbf{U}$  é o vetor de variáveis em sua forma conservativa,  $\mathbf{F}(\mathbf{U}) = (F(\mathbf{U})_x, F(\mathbf{U})_y, F(\mathbf{U})_z)$  é o fluxo físico e  $\mathcal{S}(\mathbf{U})$  é um vetor de termos de fonte. Neste apêndice, os termos  $\mathbf{U}$ ,  $\mathbf{F}(\mathbf{U})$  e  $\mathcal{S}(\mathbf{U})$  são expandidos de acordo com o modelo EGLM-MHD, de forma que sua compreensão seja facilitada.

Expandindo os termos de  $\mathbf{U}$ ,  $\mathcal{S}(\mathbf{U})$  e  $\mathbf{F}(\mathbf{U})$ , obtém-se

$$\mathbf{U} = \begin{pmatrix} \rho \\ E \\ \psi \\ \rho v_x \\ \rho v_y \\ \rho v_z \\ B_x \\ B_y \\ B_z \end{pmatrix}, \quad \mathcal{S}(\mathbf{U}) = \begin{pmatrix} 0 \\ -\left( B_x \frac{\partial \psi}{\partial x} + B_y \frac{\partial \psi}{\partial y} + B_z \frac{\partial \psi}{\partial z} \right) \\ -\frac{c_h^2}{c_p^2} \psi \\ -B_x \left( \frac{\partial \mathbf{B}}{\partial x} + \frac{\partial \mathbf{B}}{\partial y} + \frac{\partial \mathbf{B}}{\partial z} \right) \\ -B_y \left( \frac{\partial \mathbf{B}}{\partial x} + \frac{\partial \mathbf{B}}{\partial y} + \frac{\partial \mathbf{B}}{\partial z} \right) \\ -B_z \left( \frac{\partial \mathbf{B}}{\partial x} + \frac{\partial \mathbf{B}}{\partial y} + \frac{\partial \mathbf{B}}{\partial z} \right) \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad (\text{A.2})$$

$$\mathbf{F}(\mathbf{U}) = \begin{pmatrix}
(E + p + \frac{\rho v_x}{2} \frac{|\mathbf{B}|^2}{c_h^2} v_x - B_x(\mathbf{u} \cdot \mathbf{B})) & (E + p + \frac{\rho v_y}{2} \frac{|\mathbf{B}|^2}{c_h^2} v_y - B_y(\mathbf{u} \cdot \mathbf{B})) & (E + p + \frac{\rho v_z}{2} \frac{|\mathbf{B}|^2}{c_h^2} v_z - B_z(\mathbf{u} \cdot \mathbf{B})) \\
\rho v_x^2 + p + \frac{|\mathbf{B}|^2}{2} - B_x^2 & \rho v_y v_x - B_y B_x & \rho v_z v_x - B_z B_x \\
\rho v_x v_y - B_x B_y & \rho v_y^2 + p + \frac{|\mathbf{B}|^2}{2} - B_y^2 & \rho v_z v_y - B_z B_y \\
\rho v_x v_z - B_x B_z & \rho v_y v_z - B_y B_z & \rho v_z^2 + p + \frac{|\mathbf{B}|^2}{2} - B_z^2 \\
v_x B_y - B_x v_y & \psi & v_z B_x - B_z v_x \\
v_x B_z - B_x v_z & v_y B_z - B_y v_z & v_z B_y - B_z v_y \\
& & \psi
\end{pmatrix}, \tag{A.3}$$

em que  $\mathbf{u} \cdot \mathbf{B} = (v_x B_x + v_y B_y + v_z B_z)$  e  $|\mathbf{B}|^2 = \mathbf{B} \cdot \mathbf{B} = (B_x^2 + B_y^2 + B_z^2)$ .

## APÊNDICE B - EGLM-MHD: AUTOVALORES E AUTOVETORES

Os autovalores e autovetores do modelo EGLM-MHD, como apresentados no Capítulo 3, podem ser calculados utilizando o programa de livre distribuição de matemática simbólica *GNU/Maxima*<sup>1</sup>. Neste trabalho, utilizou-se uma interface gráfica do *GNU/Maxima* chamada *wxMaxima*<sup>2</sup> para fazer esse cálculo. Neste trabalho, os valores dos autovalores obtidos com o *wxMaxima* foram verificados e condizentes com as literaturas de Dedner et al (2002) e Powell 1994. No caso dos autovetores, essa verificação não foi possível devido à normalização feita nos autovalores presentes em Powell (1994).

### B.1 Cálculo simbólico

A seguir, são descritas as etapas desenvolvidas para obtenção dos autovalores e autovetores utilizando o *wxMaxima*. Os valores (%i·) e (%o·) denotam os valores de entrada e saída (input e output, em inglês), respectivamente, e o símbolo (·) é o número que indica a ordem de cálculo de cada entrada e a relaciona com sua respectiva saída.

Como entrada inicial, deve-se escrever a matriz com a qual se deseja obter os autovalores e autovetores. Nesse exemplo, utiliza-se a matriz  $A_x$  como na Equação 3.5, apresentada na pág. 13. Posteriormente, também utiliza-se a matrix  $B_x$  como na Equação 3.11, apresentada na pág. 15, para o cálculo dos autovalores da matrix  $A_x - B_x$ , a qual é relacionada ao modelo EGLM-MHD.

```
(%i1) Ax : matrix([vx,0,0,rho,0,0,0,0,0],
  [0,vx,(1-gama)*Bx,gama*p,0,0,(gama-1)*(vx*Bx+vy*By+vz*Bz),0,0],
  [0,0,0,0,0,0,ch*ch,0,0],
  [0,1/rho,0,vx,0,0,-Bx/rho,By/rho,Bz/rho],
  [0,0,0,0,vx,0,-By/rho,-Bx/rho,0],
  [0,0,0,0,0,vx,-Bz/rho,0,-Bx/rho],
  [0,0,1,0,0,0,0,0,0],
  [0,0,0,By,-Bx,0,-vy,vx,0],
  [0,0,0,Bz,0,-Bx,-vz,0,vx]);
```

---

<sup>1</sup><http://maxima.sourceforge.net/>

<sup>2</sup><http://wxmaxima.sourceforge.net/>

$$\begin{pmatrix}
vx & 0 & 0 & \rho & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & vx & Bx(1-gama) & gamap & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{\rho} & 0 & vx & 0 & 0 & 0 & 0 & \frac{Bz}{\rho} & 0 \\
0 & 0 & 0 & 0 & 0 & vx & 0 & -\frac{Bx}{\rho} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & vx & 0 & 0 & -\frac{Bx}{\rho} \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & By & -Bx & 0 & -vy & vx & 0 & 0 \\
0 & 0 & 0 & Bz & 0 & -Bx & -vz & 0 & 0 & vx
\end{pmatrix}$$

(%o1)

```

(%i2) Bx : matrix([0,0,0,0,0,0,0,0,0,0],
[0,0,(gama-1)*Bx,0,0,0,0,0,0,0],
[0,0,0,0,0,0,0,0,0],
[0,0,0,0,0,0,-Bx/rho,0,0],
[0,0,0,0,0,0,-By/rho,0,0],
[0,0,0,0,0,0,-Bz/rho,0,0],
[0,0,0,0,0,0,0,0,0],
[0,0,0,0,0,0,0,0,0],
[0,0,0,0,0,0,0,0,0]);

```

$$\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & Bx(gama-1) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{Bx}{\rho} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{By}{\rho} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{Bz}{\rho} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$

(%o2)

(%i3) eigenvalues(Ax) ;

$$\begin{aligned}
 & \sqrt{\frac{2 \sqrt{\text{gamma}^2 p^2 + 2 Bz^2 + 2 Bx^2} - 2 \sqrt{\text{gamma}^2 p^2 + (2 Bz^2 + 2 By^2 - 2 Bx^2) \text{gamma} p + Bz^4 + (2 By^2 + 2 Bx^2) Bz^2 + By^4 + 2 Bx^2 By^2 + Bx^4}}{\rho}} \\
 & \sqrt{\frac{2 \sqrt{\text{gamma}^2 p^2 + 2 Bz^2 + 2 Bx^2} - 2 \sqrt{\text{gamma}^2 p^2 + (2 Bz^2 + 2 By^2 - 2 Bx^2) \text{gamma} p + Bz^4 + (2 By^2 + 2 Bx^2) Bz^2 + By^4 + 2 Bx^2 By^2 + Bx^4}}{\rho}} \\
 & \sqrt{\frac{2 \sqrt{\text{gamma}^2 p^2 + (2 Bz^2 + 2 By^2 - 2 Bx^2) \text{gamma} p + Bz^4 + (2 Bx^2) Bz^2 + By^4 + 2 Bx^2 By^2 + Bx^4}}{\rho}} + \frac{2 \text{gamma} p + 2 Bz^2 + 2 Bx^2}{\rho} \\
 & \sqrt{\frac{2 \sqrt{\text{gamma}^2 p^2 + (2 Bz^2 + 2 By^2 - 2 Bx^2) \text{gamma} p + Bz^4 + (2 Bx^2) Bz^2 + By^4 + 2 Bx^2 By^2 + Bx^4}}{\rho}} + \frac{2 \text{gamma} p + 2 Bz^2 + 2 Bx^2}{\rho} \\
 & \frac{Bx \sqrt{\rho} - \rho vx + Bx \sqrt{\rho}}{\rho}, -ch, ch, vx, [1, 1, 1, 1, 1, 1, 1]
 \end{aligned}$$

(%i4) eigenvalues(Ax-Bx) ;

$$\begin{aligned}
 & \sqrt{\frac{2 \sqrt{\text{gamma}^2 p^2 + 2 Bz^2 + 2 Bx^2} - 2 \sqrt{\text{gamma}^2 p^2 + (2 Bz^2 + 2 By^2 - 2 Bx^2) \text{gamma} p + Bz^4 + (2 By^2 + 2 Bx^2) Bz^2 + By^4 + 2 Bx^2 By^2 + Bx^4}}{\rho}} \\
 & \sqrt{\frac{2 \sqrt{\text{gamma}^2 p^2 + 2 Bz^2 + 2 Bx^2} - 2 \sqrt{\text{gamma}^2 p^2 + (2 Bz^2 + 2 By^2 - 2 Bx^2) \text{gamma} p + Bz^4 + (2 By^2 + 2 Bx^2) Bz^2 + By^4 + 2 Bx^2 By^2 + Bx^4}}{\rho}} \\
 & \sqrt{\frac{2 \sqrt{\text{gamma}^2 p^2 + (2 Bz^2 + 2 By^2 - 2 Bx^2) \text{gamma} p + Bz^4 + (2 Bx^2) Bz^2 + By^4 + 2 Bx^2 By^2 + Bx^4}}{\rho}} + \frac{2 \text{gamma} p + 2 Bz^2 + 2 Bx^2}{\rho} \\
 & \sqrt{\frac{2 \sqrt{\text{gamma}^2 p^2 + (2 Bz^2 + 2 By^2 - 2 Bx^2) \text{gamma} p + Bz^4 + (2 Bx^2) Bz^2 + By^4 + 2 Bx^2 By^2 + Bx^4}}{\rho}} + \frac{2 \text{gamma} p + 2 Bz^2 + 2 Bx^2}{\rho} \\
 & \frac{Bx \sqrt{\rho} - \rho vx + Bx \sqrt{\rho}}{\rho}, -ch, ch, vx, [1, 1, 1, 1, 1, 1, 1]
 \end{aligned}$$

Para calcular os autovetores à direita, ou seja, calcular os vetores  $\mathbf{r}_i$ ,  $i \in \{1, \dots, 9\}$ , do sistema  $A_x \mathbf{r}_i = \lambda_i \mathbf{r}_i$ , utiliza-se o comando `eigenvectors(Ax)`. Já para o caso dos autovetores à esquerda, ou seja, os vetores  $\mathbf{l}_i$  do sistema  $\mathbf{l}_i A_x = \mathbf{l}_i \lambda$ , basta calcular os autovetores da matriz  $A_x^T$ , que é a matriz transposta de  $A_x$ .

```
--> eigenvectors(Ax);
```

```
--> Axt : transpose(Ax);
```

```
--> eigenvectors(Axt);
```

Caso seja necessário, pode-se utilizar os seguintes comandos para fatorar e expandir o resultado obtido a fim de simplificá-lo.

```
--> factor(results);
```

```
--> expand(results);
```

Esse mesmo processo pode ser feito para calcular os autovalores e autovetores das matrizes  $A_y$  e  $A_y - B_y$ . Basta fazer as modificações necessárias ao entrar com a matriz no programa.

## APÊNDICE C - CONDIÇÃO DE CONTORNO

Nesse apêndice são apresentadas as condições de contorno e evolução temporal utilizadas para o cálculo do modelo EGLM-MHD nos códigos MHD-FV e CARMEN.

### C.1 Condição de Contorno

A condição de contorno utilizada é conhecida como *condição de Neumann*. A mesma é utilizada em ambos os códigos MHD-FV e CARMEN. Em uma matriz de dados, essa condição consiste em espelhar o valor das linhas e colunas nas extremidades da matrix para as células virtuais utilizadas no código. Para ilustrar a condição de contorno, considera-se uma matriz  $T$ ,  $6 \times 6$ , qualquer

$$T = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} & a_{36} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} & a_{46} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} & a_{56} \\ a_{61} & a_{62} & a_{63} & a_{64} & a_{65} & a_{66} \end{pmatrix}. \quad (\text{C.1})$$

A borda da matriz  $T$ , marcadas com a cor vermelha, são espelhadas nas extremidades da matriz maior, ou seja, após aplicar a condição de contorno de *Neumann*, obtém-se a matriz

$$T_{cc} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} \\ a_{11} & a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} & a_{16} \\ a_{21} & a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} & a_{26} \\ a_{31} & a_{31} & a_{32} & a_{33} & a_{34} & a_{35} & a_{36} & a_{36} \\ a_{41} & a_{41} & a_{42} & a_{43} & a_{44} & a_{45} & a_{46} & a_{46} \\ a_{51} & a_{51} & a_{52} & a_{53} & a_{54} & a_{55} & a_{56} & a_{56} \\ a_{61} & a_{61} & a_{62} & a_{63} & a_{64} & a_{65} & a_{66} & a_{66} \\ a_{61} & a_{62} & a_{63} & a_{64} & a_{65} & a_{66} \end{pmatrix}. \quad (\text{C.2})$$

A matriz  $T_{cc}$  é utilizada para obter a solução do modelo EGLM-MHD e tem dimensões  $8 \times 8$ , pois são acrescentadas as células virtuais, para os cálculos nas extremidades. No entanto, essas células servem apenas como um auxílio especificamente para esse cálculo, não sofrendo a evolução temporal. A aplicação da condição de contorno

é essencial para uma simulação numérica, principalmente quando se deseja calcular a solução nas extremidades da malha.

## APÊNDICE D - CARMEN: EVOLUÇÃO TEMPORAL

Como já foi dito ao longo deste trabalho, para discretização do modelo EGLM-MHD é utilizada a formulação clássica do método dos volumes finitos em sua forma conservativa, como apresentado no Capítulo 4. Considerando uma lei de conservação para  $(x, t) \in \Omega \times [0, +\infty)$ ,  $\Omega \in \mathbb{R}^d$ , na forma

$$\frac{\partial \mathbf{U}}{\partial t} = \mathcal{D}(\mathbf{U}, \nabla \mathbf{U}), \quad (\text{D.1})$$

em que  $\Omega$  é o domínio, com condição inicial  $u(x, 0) = u_0(x)$  e condição de contorno apropriada. O operador  $\mathcal{D}(\mathbf{U}, \nabla \mathbf{U}) = -\nabla \cdot \mathbf{F}(\mathbf{U}, \nabla \mathbf{U}) + \mathbf{S}(\mathbf{U})$ , formado pela divergência do fluxo e pelos termos fonte. A função fluxo contém termos advectivos e difusivos da forma  $\mathbf{F}(\mathbf{U}, \nabla \mathbf{U}) = \mathbf{f}(\mathbf{U}) - \nu \nabla \mathbf{U}$ , com a constante de difusão  $\nu \geq 0$ .

Considera-se uma partição do domínio computacional  $\Omega$  como  $(\Omega_k)_{k \in \Gamma}$ ,  $\Gamma = \{1, \dots, k_{max}\}$ . Integrando a Equação D.1 em  $\Omega_k$ , tem-se

$$\int_{\Omega_k} \frac{\partial \mathbf{U}(x, t)}{\partial t} dx = \int_{\Omega_k} \mathcal{D}(\mathbf{U}(x, t), \nabla \mathbf{U}(x, t)) dx, \quad (\text{D.2})$$

ou seja,

$$\frac{\partial \bar{u}_k(t)}{\partial t} = \bar{\mathcal{D}}_k(t), \quad (\text{D.3})$$

em que  $\bar{u}_k(t)$  é a média celular da célula  $\Omega_k$ . No caso da evolução temporal utilizada no código CARMEN, tem-se o esquema *Runge-Kutta 2* (RK2), o qual é explícito e de segunda ordem (ROUSELL et al., 2003). Denotando por  $\Delta t$  o passo de tempo e  $\bar{u}_k^n = \bar{u}_k(t_n)$ , em que  $t_n = n\Delta t$ , o esquema RK2 utilizado tem a forma

$$\bar{u}_k^* = \bar{u}_k^n + \Delta t \bar{\mathcal{D}}_k^n, \quad (\text{D.4a})$$

$$\bar{u}_k^{n+1} = \frac{1}{2} [\bar{u}_k^n + \bar{u}_k^* + \bar{\mathcal{D}}_k^*]. \quad (\text{D.4b})$$

Denotando por  $\bar{\mathbf{U}}^n$  o vetor  $(\bar{u}_k^n)_{k \in \Gamma}$ , o operador discreto de evolução temporal  $\bar{\mathbf{E}}(\Delta t)$  é definido por

$$\bar{\mathbf{U}}^{n+1} = \bar{\mathbf{E}}(\Delta t) \cdot \bar{\mathbf{U}}^n, \quad (\text{D.5})$$

em que

$$\bar{\mathbf{E}}(\Delta t) = \mathbf{I} + \frac{\Delta t}{2} [\bar{\mathcal{D}} + \bar{\mathcal{D}}(\mathbf{I} + \Delta t \bar{\mathcal{D}})], \quad (\text{D.6})$$

e  $\mathbf{I}$  denota o operador identidade.

Considera-se um tempo fixo  $t_n$ , mas o índice  $n$  é omitido nos próximos passos. para o caso 2D,  $\Omega_{k,m}$  é um retângulo com volume de tamanho  $|\Omega_{k,m}| = \Delta x_k \Delta y_m$ . A Equação D.7 pode ser escrita como

$$\frac{\partial \bar{u}_{k,m}(t)}{\partial t} = \bar{\mathcal{D}}_{k,m}(t), \quad (\text{D.7})$$

O operador  $\bar{\mathcal{D}}_k$  é escrito como

$$\bar{\mathcal{D}}_{k,m} = -\frac{1}{\Delta x_k} (\bar{\mathbf{F}}_{k+1/2,m} - \bar{\mathbf{F}}_{k-1/2,m}) - \frac{1}{\Delta y_m} (\bar{\mathbf{F}}_{k,m+1/2} - \bar{\mathbf{F}}_{k,m-1/2}) + \bar{S}_{k,m} \quad (\text{D.8})$$

Para a parte advectiva da função fluxo, utiliza-se o esquema de Roe com uma interpolação ENO de segunda ordem, enquanto que, para a parte difusiva, utiliza-se um esquema centrado com precisão de segunda ordem. Em ... mostra-se que o esquema global resultante, o qual é não-linear, é de segunda ordem no espaço. O fluxo numérico é aplicado nas duas direções  $x$  e  $y$ .

$$\hat{\mathbf{F}}_{k+1/2,m} = f^R \left( \bar{u}_{k+1/2,m}^+, \bar{u}_{k+1/2,m}^- \right) - \nu \frac{\bar{u}_{k+1,m} - \bar{u}_{k,m}}{\Delta x_{k+1/2}}, \quad (\text{D.9a})$$

$$\hat{\mathbf{F}}_{k,m+1/2} = f^R \left( \bar{u}_{k,m+1/2}^+, \bar{u}_{k,m+1/2}^- \right) - \nu \frac{\bar{u}_{k,m+1} - \bar{u}_{k,m}}{\Delta y_{m+1/2}}, \quad (\text{D.9b})$$

em que  $\Delta x_{k+1/2} = \frac{1}{2}(\Delta x_k) + \Delta x_{k+1}$  e  $\Delta y_{m+1/2} = \frac{1}{2}(\Delta y_m) + \Delta y_{m+1}$ . O termo  $f^R$  denota, para a parte advectiva, a solução aproximada de Roa para o problema de Riemann dados os valores de  $\mathbf{U}$  à esquerda ( $-$ ) e à direita ( $+$ ).

## APÊNDICE E - IMPLEMENTAÇÕES

Neste capítulo são apresentados detalhes das implementações do modelo EGLM-MHD desenvolvidas ao longo deste trabalho, para ambos os códigos MHD-FV e CARMEN.

### E.1 Código MHD-FV

A implementação do código MHD-FV é inspirado na implementação do código MHD desenvolvido por Pierro (2009) em Fortran 90, o qual é baseado em Dedner et al (2002).

O código MHD-FV foi implementado na linguagem  $C^{++}$ . Com esse código é possível simular a solução do modelo EGLM-MHD para quatro diferentes condições iniciais

- 1 Condição inicial tipo Riemann 1D, como em Dedner et al (2002);
- 2 Condição inicial tipo Riemann 1D, como em Miyoshi e Kusano (2005), direção  $x$ ;
- 3 Condição inicial tipo Riemann 2D, como em Dedner et al (2002);
- 4 Condição inicial tipo Riemann 1D, como em Miyoshi e Kusano (2005), direção  $y$ .

e dois diferentes fluxos numéricos: o HLL e o HLLD.

Para compilar o código MHD-FV, acessa-se o diretório do código pelo terminal e utiliza-se o comando `g++ MHD_FV.cpp -o MHD-FV`. Para rodá-lo, no mesmo diretório utiliza-se o comando `./MHD-FV`.

O código MHD-FV possui a seguinte rotina para a simulação da solução do modelo EGLM-MHD

- 1 Escolher condição inicial, fluxo numérico e parâmetros da simulação;
- 2 Criar a malha das variáveis (depende da condição inicial);
- 3 Aplicar condição de contorno `applyBCAbsorptionNeumann`;
- 4 Calcular fluxo físico `computeFlux`;
- 5 Calcular fluxo numérico `RiemannSolver`;

- 6 Corrigir fluxo numérico de acordo com a correção de divergência livre `fluxCorrectionX, fluxCorrectionY`;
- 7 Evoluir as variáveis para no tempo  $t_{n+1}$ , utilizando a função de evolução temporal `conservationStep`;
- 8 Corrigir as variáveis no tempo  $t_{n+1}$  `computeCorrection`.

A condição inicial entra no código a partir dos arquivos `data_1D.dat`, `data_1Db.dat` e `data_2Dprim.dat`, de acordo com a condição inicial desejada. Os parâmetros da simulação são escolhidos dentro do próprio código, como no trecho a seguir.

```
#define IMAX 202
#define JMAX 202
#define IM (IMAX-2)
#define JM (JMAX-2)
#define VAR 9 //(density, pression, psi, vx, vy, vz, Bx, By, Bz);
#define TVAR 18
#define T4VAR 36
#define KTEMPMAX 20000
#define IMAX2 23104
#define IC_case 3
#define epsilon 1e-8
```

```
double longX = (double)1.0, longY = (double)0.5, CFL = (double)0.3,
    gama = (double)5.0/(double)3.0, gravity = (double)9.8;
```

A função `computeFlux` calcula o fluxo físico referente ao modelo EGLM-MHD, nas direções  $x$  e  $y$ . O fluxo numérico é calculado pela função `RiemannSolver`. O fluxo a ser utilizado depende do parâmetro a ser definido no início do código, como pode ser visto no trecho a seguir. O Caso 1 refere-se ao fluxo HLL e o Caso 2, ao fluxo HLLD.

```
switch(SchemeNb)
{
    case 1:
        for(int kvar=0;kvar<VAR; kvar++)
        {
            if(slopeLeft>0)F[0][kvar] = F[0][kvar];
            if(slopeRight>=0 && slopeLeft<=0)F[0][kvar] =
                ( slopeRight*F[0][kvar]-slopeLeft*F[1][kvar]
                + slopeLeft*slopeRight*(vec[1][kvar]-vec[0][kvar]))
```

```

                                //(slopeRight-slopeLeft);
    if(slopeRight<0)F[0][kvar] = F[1][kvar];
}
break;

case 2:

for(int kvar=0;kvar<VAR; kvar++){
//Flux Function - Equation 66
//F_L
if(slopeLeft>0)F[0][kvar] = F[0][kvar]
//F-star left // FL=FLstar
else if(slopeLeftStar>=0 && slopeLeft<=0){F[0][kvar] = F[0][kvar]
+ slopeLeft*(Ustar[0][kvar] - vec[0][kvar]);}
//F-star-star left
else if(slopeM>=0 && slopeLeftStar<=0){F[0][kvar] = F[0][kvar]
+ slopeLeftStar*Ustarstar[0][kvar]
- (slopeLeftStar - slopeLeft)*Ustar[0][kvar]
- slopeLeft*vec[0][kvar];}

//F-star-star right
else if(slopeRightStar>=0 && slopeM<=0){F[0][kvar] = F[1][kvar]
+ slopeRightStar*Ustarstar[1][kvar]
- (slopeRightStar - slopeRight)*Ustar[1][kvar]
- slopeRight*vec[1][kvar];}

//F-star right
else if(slopeRight>=0 && slopeRightStar<=0){F[0][kvar] = F[1][kvar]
+ slopeRight*(Ustar[1][kvar] - vec[1][kvar]);}

//F_R
else if(slopeRight<0)F[0][kvar] = F[1][kvar];
}
break;
}

```

Para calcular os estados intermediários do fluxo, existem algumas condições a serem satisfeitas, as quais dependem de um certo valor chamado *epsilon*, o qual deve ser definido no início do código, como apresentado em um trecho nesta seção. Um exemplo das condicionais existentes no cálculo dos pode ser vista no trecho a seguir, retirado da função `stateUstar`.

```

if((half*mf*mf/min((half*B1*B1),(half*Br*Br))) < (epsilon*epsilon))
{
for(int k=0;k<VAR;k++){
Ustarstar[0][k] = Ustar[0][k];

```

```

Ustarstar[1][k] = Ustar[1][k];
}
}else{
//density - Equation 49
rholss = rhols;
rhorss = rhors;

```

A solução aproximada obtida após a simulação feita com o código MHD-FV é escrita nos arquivos `primitiveIT.out` e `PrimDiv_varN.out`, em que IT e N denotam o número da iteração e o número referente à variável, respectivamente.

## E.2 Código CARMEN

O código CARMEN foi implementado por Roussel (2003) em  $C^{++}$ , possibilitando que a simulação de seis modelos diferentes fossem realizadas utilizando um algoritmo de multirresolução adaptativa ou uma malha sem adaptatividade.

Como parte deste trabalho, tem-se a implementação do modelo EGLM-MHD no código CARMEN, com objetivo de simular a solução do modelo EGLM-MHD, utilizando o algoritmo adaptativo já implementado. O modelo EGLM-MHD passou a ser a Equação 7 no código CARMEN. Para tal, o código CARMEN foi modificado de acordo com os seguintes passos

- 1 Escolheu-se o modelo de Navier-Stokes como base para a implementação;
- 2 Adicionou-se as variáveis do modelo;
- 3 Adicionou-se arquivo com a condição inicial;
- 4 Adicionou-se os parâmetros referentes ao modelo e o cálculo do passo de tempo;
- 5 Adicionou-se as variáveis ao arquivo de saída;
- 6 Implementou-se função fluxo físico;
- 7 Implementou-se funções para calcular o fluxo numérico;
  - HLL:
    - i. Implementou-se função para calcular o fluxo HLL.
  - HLLD:

- i. Implementou-se função para calcular o fluxo HLLD;
- ii. Implementou-se função para calcular estados intermediários  $\mathbf{U}^*$  e  $\mathbf{U}^{**}$ .

8 Implementou-se função para correção do fluxo numérico;

9 Implementou-se função para correção parabólica-hiperbólica nas variáveis no tempo  $t_{n+1}$ .

Para o código CARMEN, foram implementadas as condições iniciais:

1 Condição inicial tipo Riemann 2D, como em Dedner et al (2002);

2 Condição inicial tipo Riemann 1D, como em Miyoshi e Kusano (2005), direção  $x$ ;

3 Condição inicial tipo Riemann 1D, como em Miyoshi e Kusano (2005), direção  $y$ .

e dois diferentes fluxos numéricos: o HLL e o HLLD.

Para compilar o código CARMEN, acessa-se o diretório do código pelo terminal e utiliza-se o comando `make`. Para rodá-lo, no mesmo diretório utiliza-se o comando `./carmen`. Nesse código, a solução inicial é implementada no arquivo `carmen.ini` e, os parâmetros, no arquivo `carmen.par`. No arquivo `carmen.par` são definidos parâmetros como o fluxo numérico a ser utilizado, o modelo a ser simulado, o tempo físico, a condição inicial, a constante CFL, o domínio, o nível multirresolução utilizado, o parâmetro de truncamento  $\epsilon$  e a constante  $\gamma$ . A seguir, tem-se uma parte do arquivo `carmen.par`.

```
// Carmen parameter file
// Generated by Carmen Editor

// 1) Time integration -----
StepNb = 2;
SchemeNb = 1;
PhysicalTime = 0.1000000e-00;
CFL = 3.00000e-01;
ConstantTimeStep = false;

// 2) Solved equations -----
```

```

EquationType = 7;

// 3) Geometry -----

Dimension = 2;

XMin[1] = 0.000000e+00;
XMax[1] = 1.000000e+00;
XMin[2] = 0.000000e+00;
XMax[2] = 1.000000e+00;

// 4) Multiresolution -----

Multiresolution = true;
ScaleNb = 7;
Tolerance = 8.000000e-3;

// 5) Physics -----

Gamma = 1.666666666666666e+00;

```

Os arquivos de saída são criados a partir dos arquivos `PrintGrid.h`, `PrintGrid.cpp` e `PrintIntegral.cpp`. As funções que calculam os fluxos físicos, fluxos numéricos e correções de fluxo fazem parte da biblioteca `carmen.h`. As funções que calculam os fluxos numéricos, `SchemeHLL.cpp` e `SchemeHLLD.cpp`, retornam o valor `Result`, referente ao valor obtido com os fluxos e as correções de fluxo, as quais são calculadas pela função `fluxCorrection`. A seguir, tem-se trechos de ambos os códigos de fluxo numérico.

### E.2.1 Fluxo HLL

```

// --- HLL Riemann Solver ---

for(int i=1;i<=QuantityNb;i++)
{
    //F_L
    if(SL>0.)
        Result.setValue(i, FL.value(i));
    else if(SR>=0.)
        Result.setValue(i, (SR*FL.value(i) - SL*FR.value(i)
            + SR*SL*(RightAverage.value(i) - LeftAverage.value(i)))/(SR-SL));
    else if(SR<0.)

```

```

        Result.setValue(i, FR.value(i));
    }

    fluxCorrection(Result, LeftAverage, RightAverage, AxisNo);

    return Result;

```

## E.2.2 Fluxo HLLD

```

// --- HLLD Riemann Solver ---

for(int i=1;i<=QuantityNb;i++)
{
    //Flux Function - Equation 66
    //F_L
    if(SL>0.)
        Result.setValue(i, FL.value(i));
    //F-star left // FL=FLstar
    else if(SLS>=0.)
        Result.setValue(i, FL.value(i) + SL*(U.value(i,1) - LeftAverage.value(i)));
    //F-star-star left
    else if(SM>=0.)
        Result.setValue(i, FL.value(i) + SLS*U.value(i,3)
            - (SLS - SL)*U.value(i,1) - SL*LeftAverage.value(i));
    //F-star-star right
    else if(SRS>=0.)
        Result.setValue(i, FR.value(i) + SRS*U.value(i,4)
            - (SRS - SR)*U.value(i,2) - SR*RightAverage.value(i));
    //F-star right
    else if(SR>=0.)
        Result.setValue(i, FR.value(i) + SR*(U.value(i,2) - RightAverage.value(i)));
    //F_R
    else if(SR<0.)
        Result.setValue(i, FR.value(i));
}

    fluxCorrection(Result, LeftAverage, RightAverage, AxisNo);

    return Result;

```

Utilizando os valores obtidos com o fluxo numérico, calcula-se a divergência do fluxo, valor que é utilizado para fazer a evolução temporal, como apresentada no Apêndice C. Após a evolução temporal, utiliza-se as variáveis no tempo  $t_{n+1}$  e faz-se

a correção de divergência livre nessas variáveis e atualiza os valores na árvore. A variável `StepNb` é definida como 2, pois utiliza-se um Runge-Kutta 2.

### E.2.3 Evolução temporal

```
for (StepNo = 1; StepNo <= StepNb; StepNo++)
{
// --- Compute divergence ---
Root->computeDivergence();

// --- Runge-Kutta step ---
Root->RungeKutta();

// --- Divergence-free correction ---
    if (EquationType == 7) Root->computeCorrection();

// --- Refresh tree structure ---
RefreshTree(Root);
}
```

Depois do processo de evolução temporal, é realizado o processo de multirresolução adaptativa, de acordo com o  $\epsilon$  escolhido para a simulação. Para isso, utiliza-se a função `Remesh(Mesh)`. Após o processo de adaptação da malha, cria-se os arquivos de saída e o arquivo `carmen.prf`, com as informações da simulação, como é possível observar no seguinte trecho do arquivo `main.cpp`.

```
// --- Time evolution procedure ---
if (Multiresolution)
TimeEvolution(Mesh);
else
TimeEvolution(FMesh);

// --- Remesh ---
if (Multiresolution) Remesh(Mesh);

// --- Check CPU Time ---
CPUTime.check();

// --- Write information every (Refresh) iteration ---
if ((IterationNo-1)%Refresh == 0)
{
// - Write integral values -
if (rank==0) PrintIntegral("Integral.dat");
```

```
if (Cluster == 0)
ShowTime(CPUTime); // Show time on screen
//else
if (rank==0) Performance("carmen.prf"); // Refresh file "carmen.prf"
}
```

As saídas referentes à simulação são escritas nos arquivos `Average.dat` e `Mesh.dat`, referentes às médias celulares das variáveis calculadas e à malha adaptativa, respectivamente.



## APÊNDICE F - SOLUÇÃO DE REFERÊNCIA PARA O CASO 2

Neste Apêndice são apresentadas as soluções de referência obtidas com os códigos MHD-FV e CARMEN. A solução de referência é calculada para uma malha de  $2048 \times 2048$ , utilizando-se dos mesmos parâmetros utilizados para obter as soluções teste, como apresentadas nos Capítulos 6 e 9.

As soluções de referência calculadas com o códigos MHD-FV e CARMEN têm os parâmetros  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $t = 0.1$ ,  $c_h/c_p^2 = 1$  e condição inicial tipo Riemann 2D, como apresentada na Tabela 6.3.

## F.1 Solução de Referência - MHD-FV

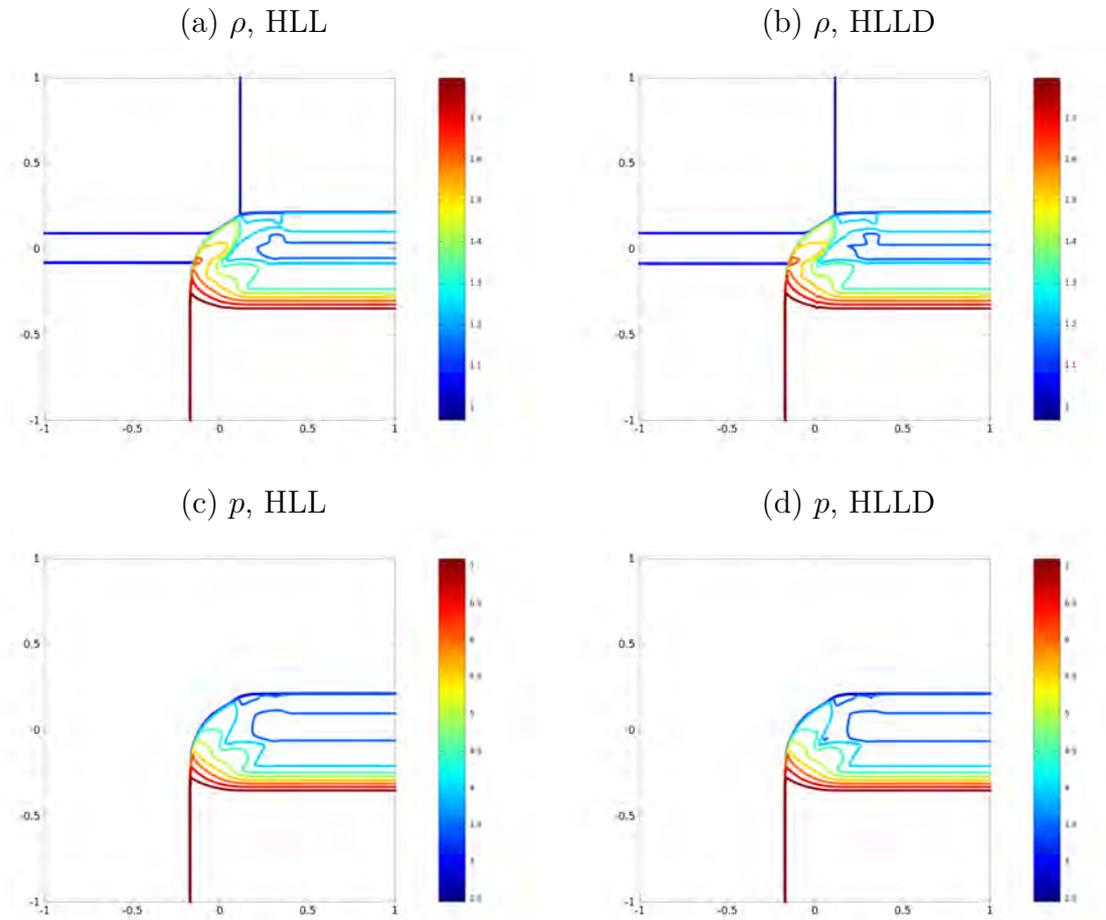


Figura F.1 - Variáveis densidade e pressão para os fluxos HLL (a,c) e HLLD (b,d), respectivamente, em uma malha com  $2048 \times 2048$  células, para tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $c_{CFL} = 0.3$ .

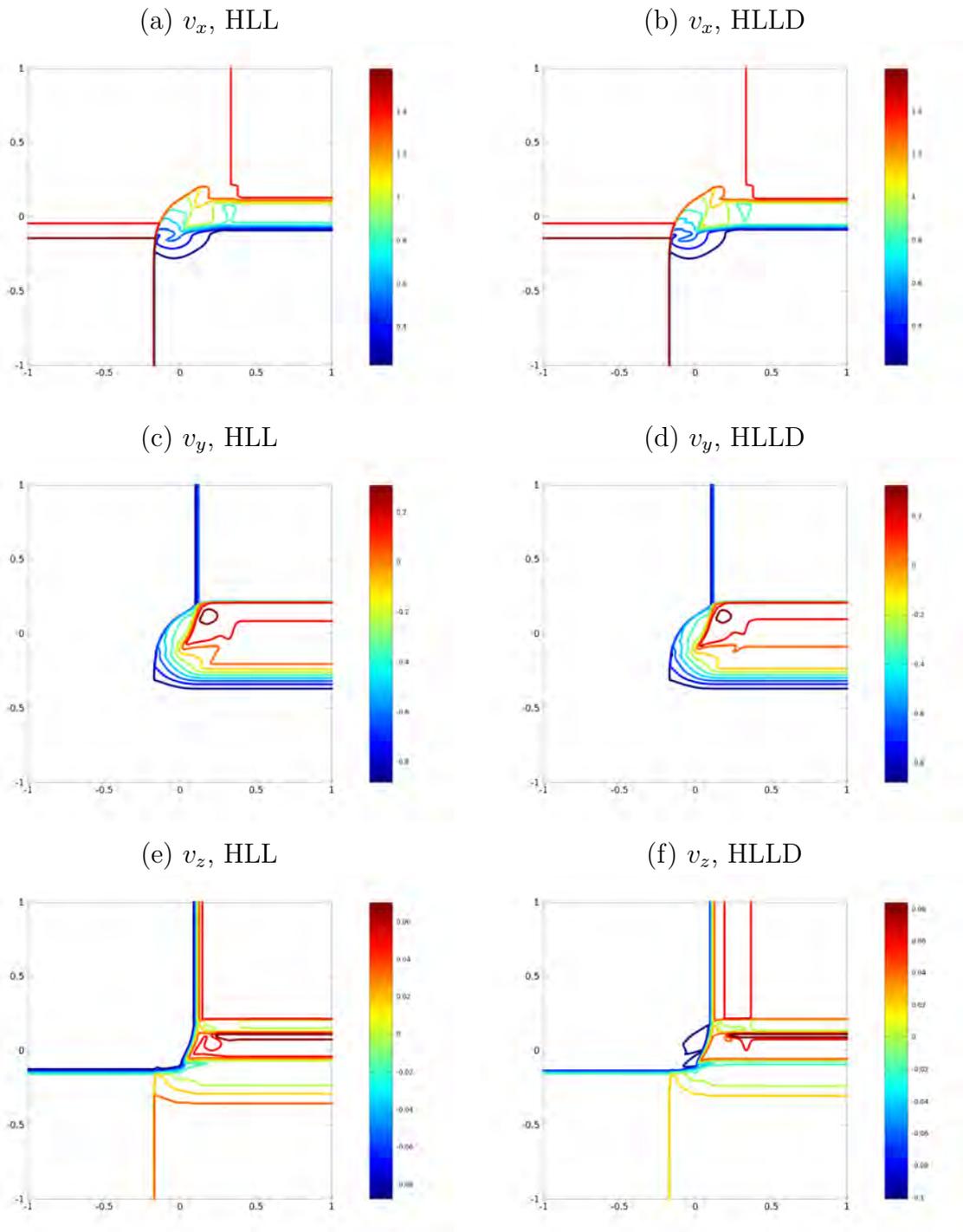


Figura F.2 - Componentes  $x$ ,  $y$ ,  $z$  da velocidade para os fluxos HLL (a,c,e) e HLLD (b,d,f), respectivamente, em uma malha com  $2048 \times 2048$  células, para tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $c_{CFL} = 0.3$ .

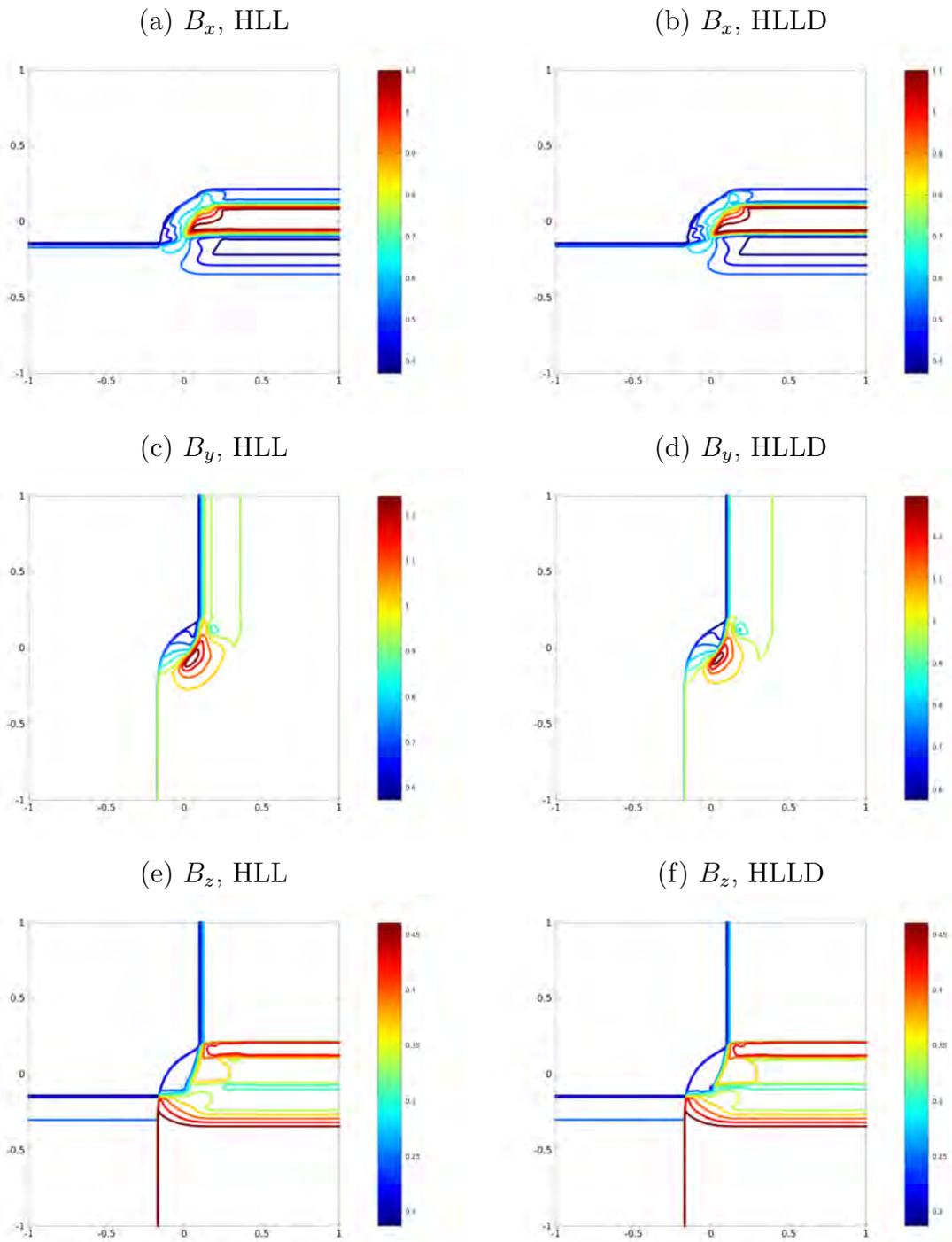


Figura F.3 - Componentes  $x$ ,  $y$ ,  $z$  do campo magnético para os fluxos HLL (a,c,e) e HLLD (b,d,f), respectivamente, em uma malha com  $2048 \times 2048$  células, para tempo  $t = 0.1$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $CCFL = 0.3$ .

## F.2 Solução de Referência - CARMEN

### F.2.1 Resultados: Fluxo HLL

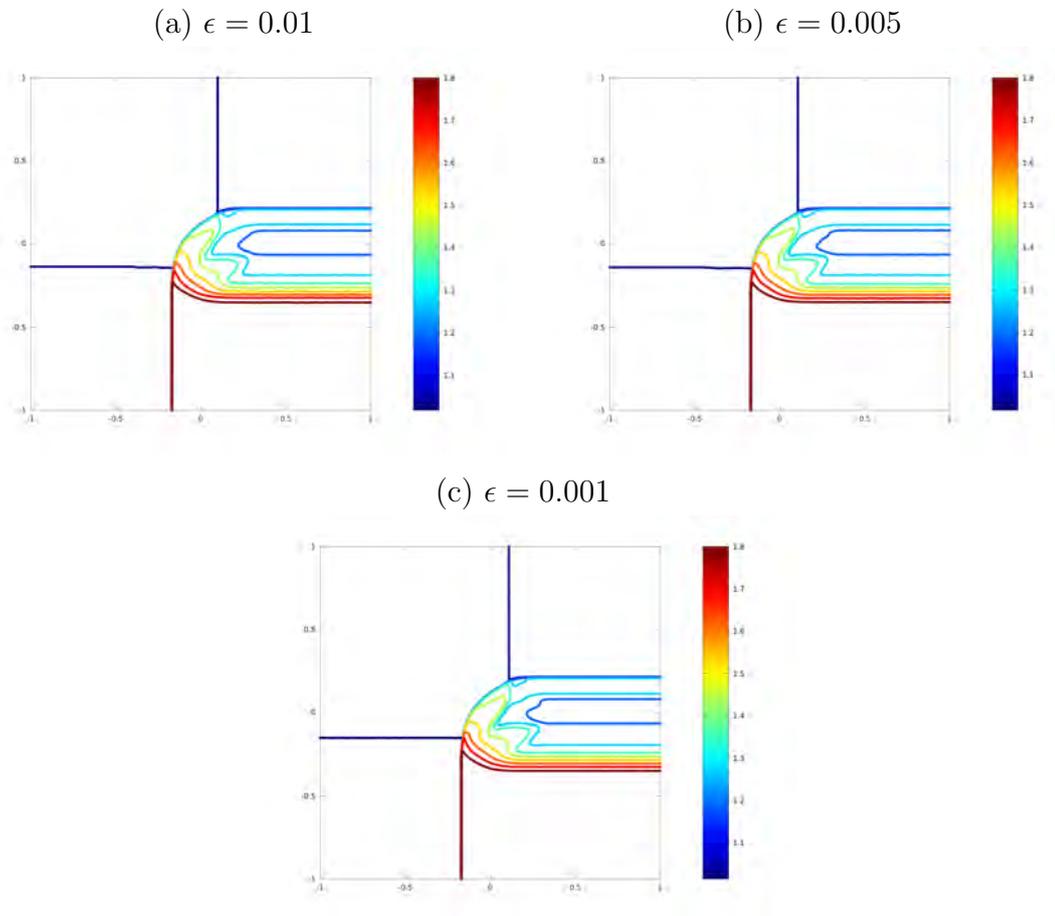


Figura F.4 - Variável densidade aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

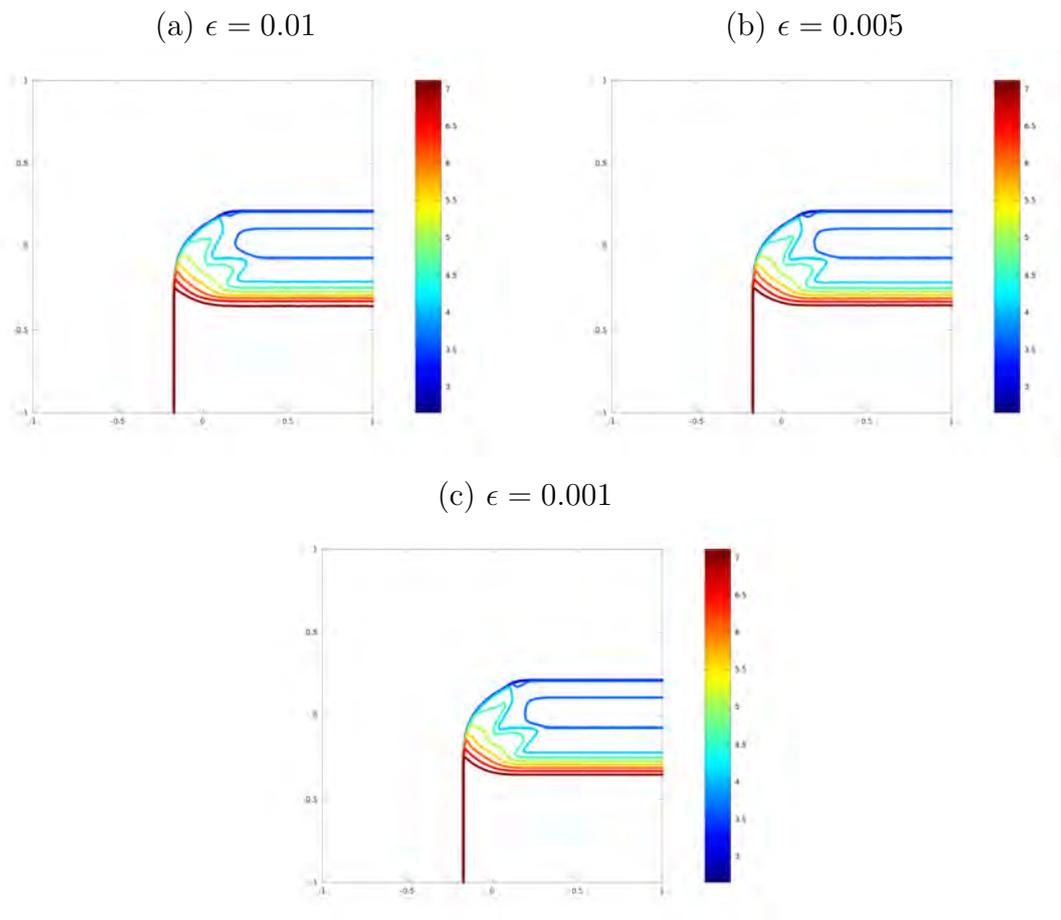


Figura F.5 - Variável pressão aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

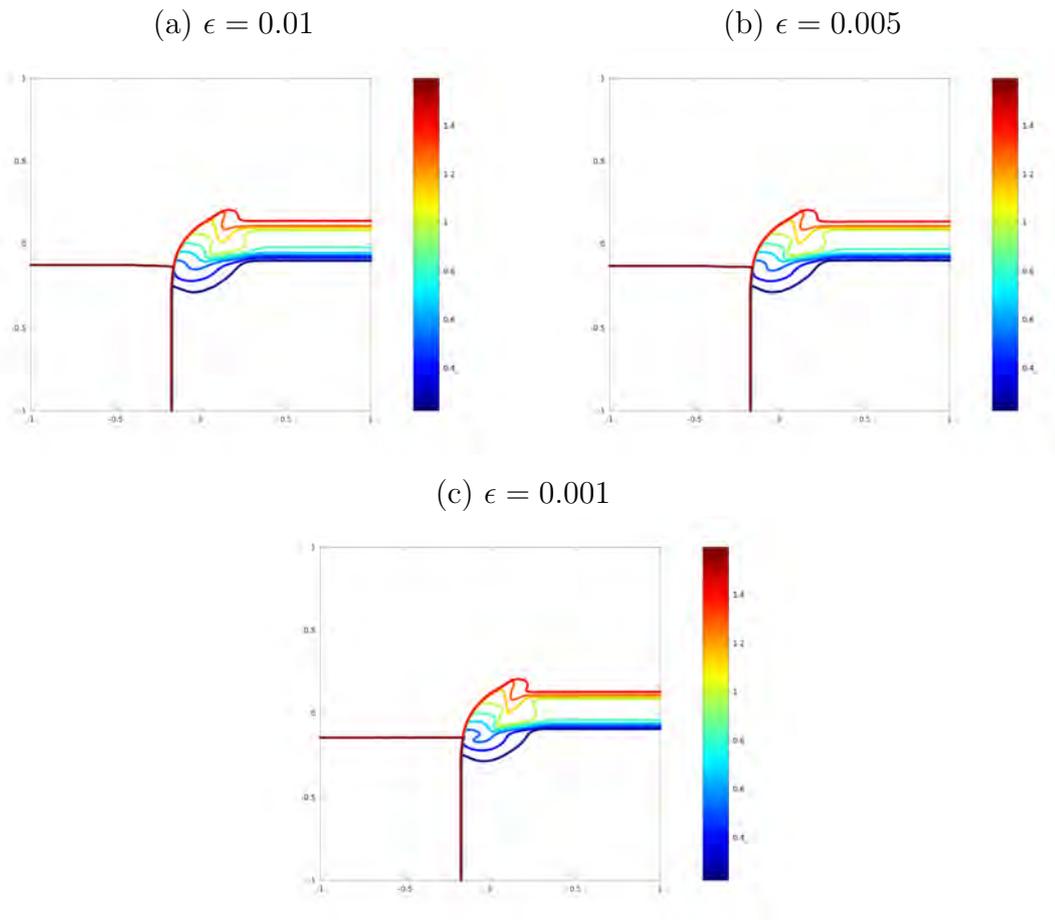


Figura F.6 - Componente  $x$  da velocidade aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

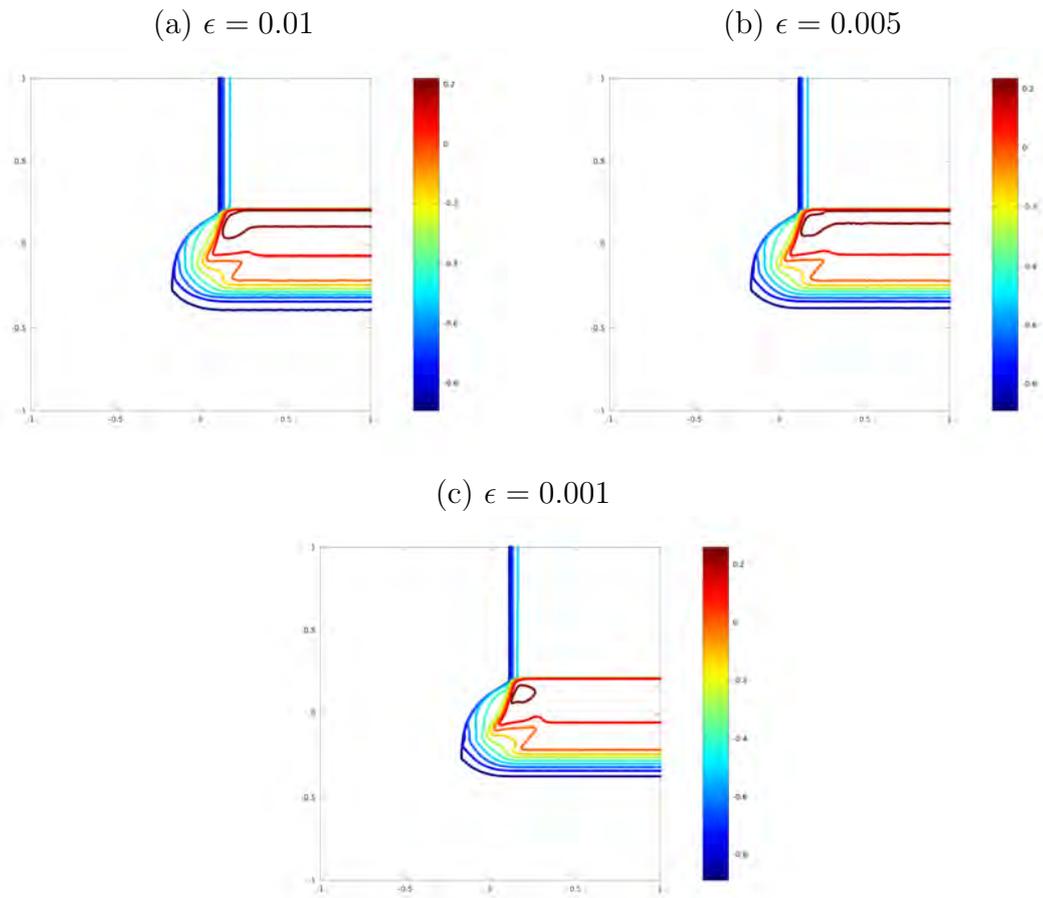


Figura F.7 - Componente  $y$  da velocidade aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

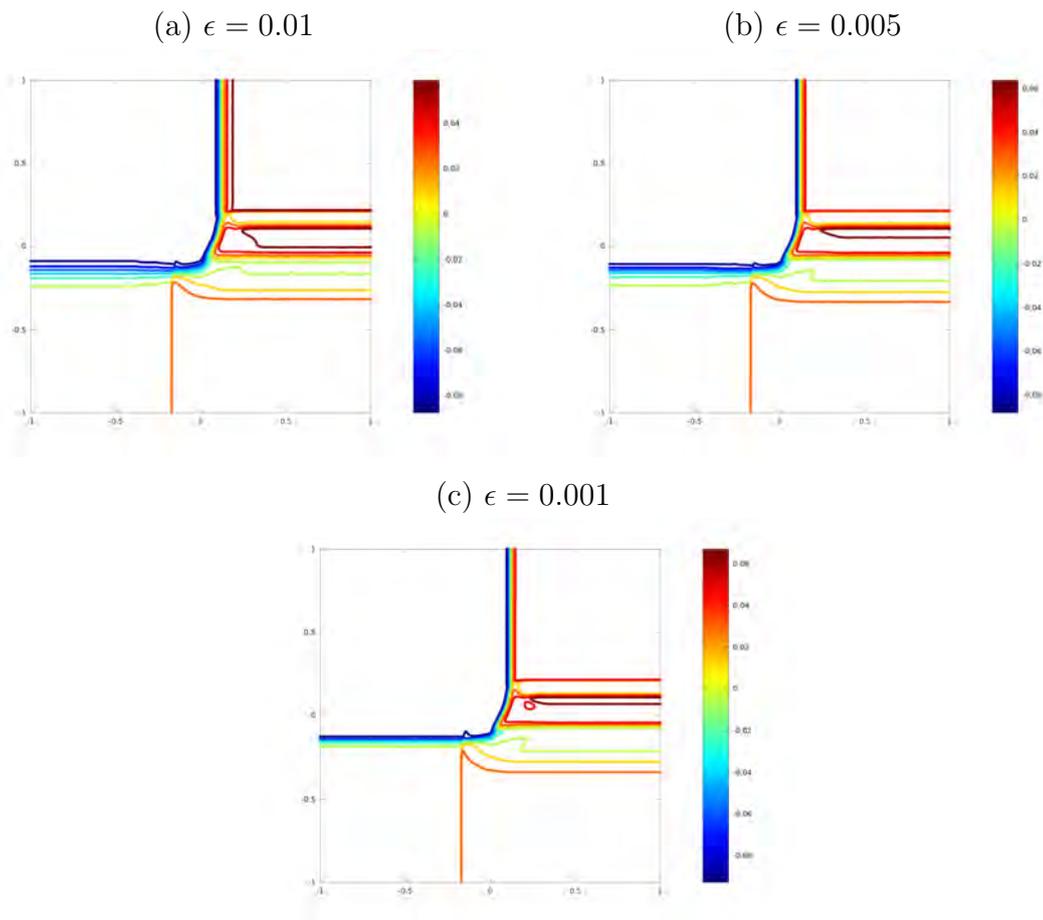


Figura F.8 - Componente  $z$  da velocidade aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

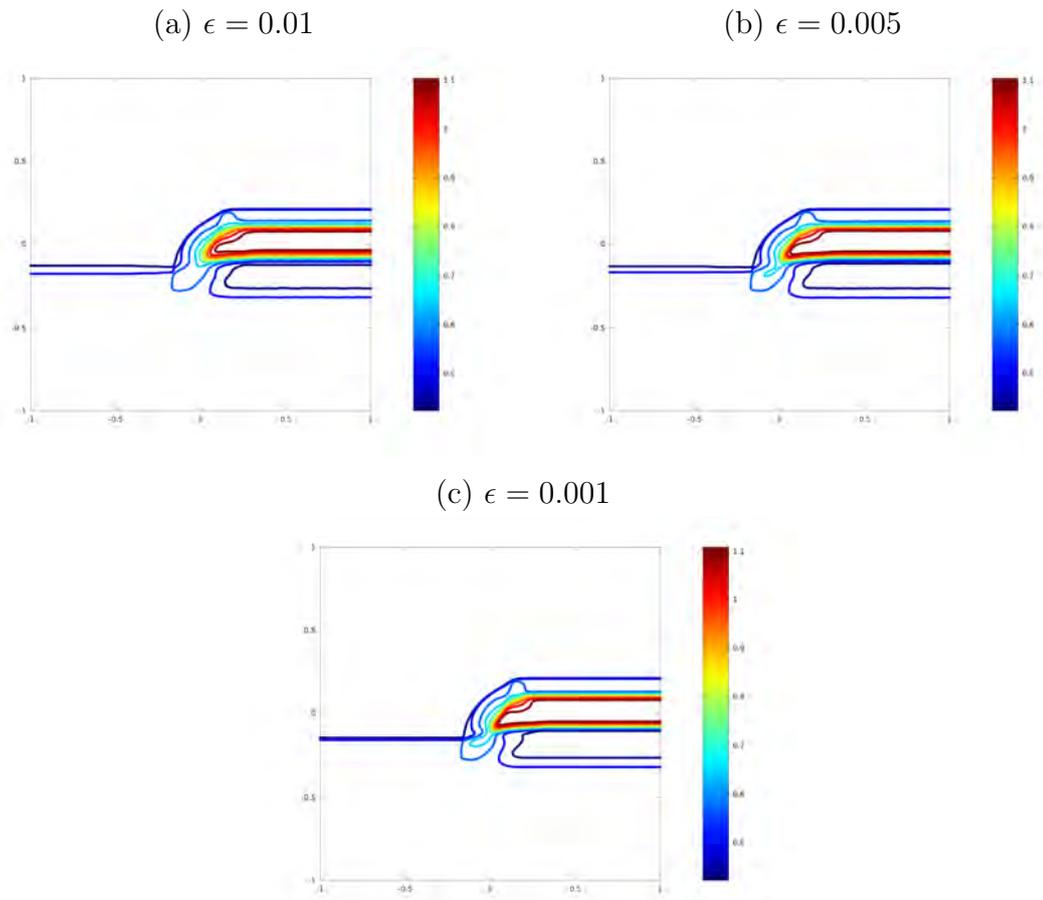


Figura F.9 - Componente  $x$  do campo magnético aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

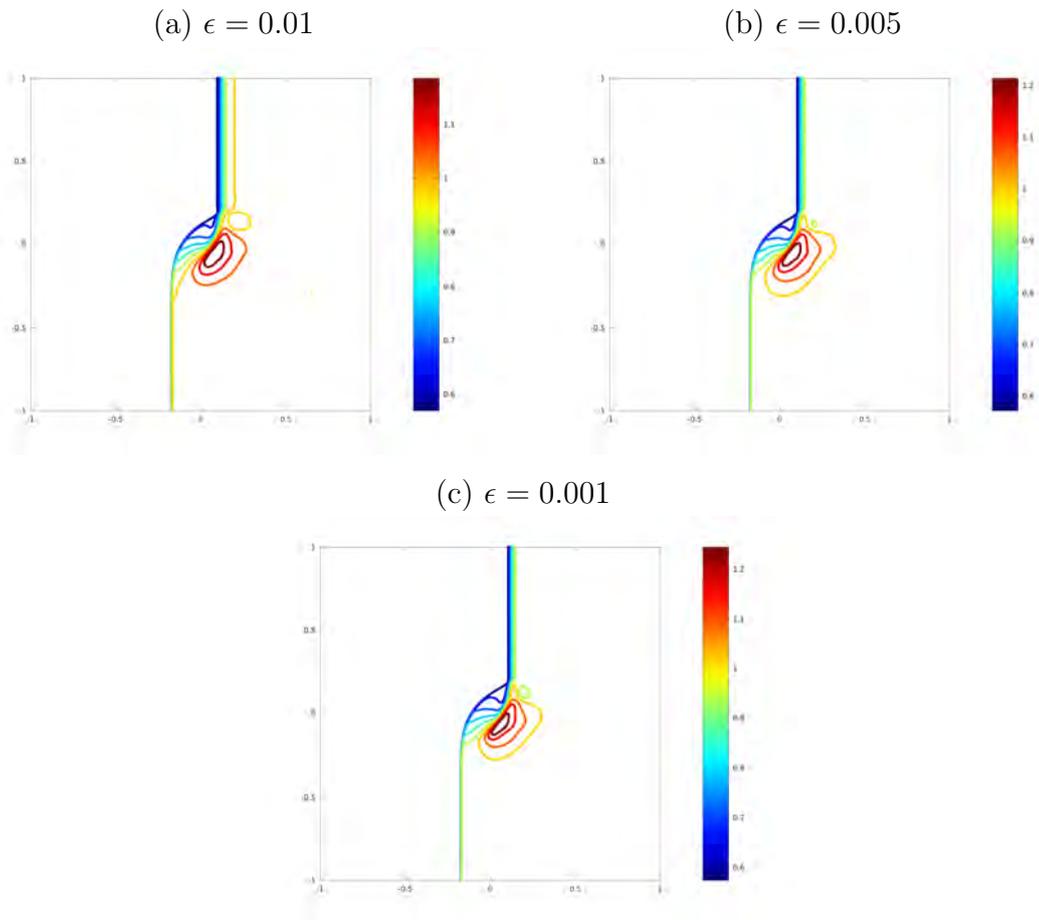


Figura F.10 - Componente  $y$  do campo magnético aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

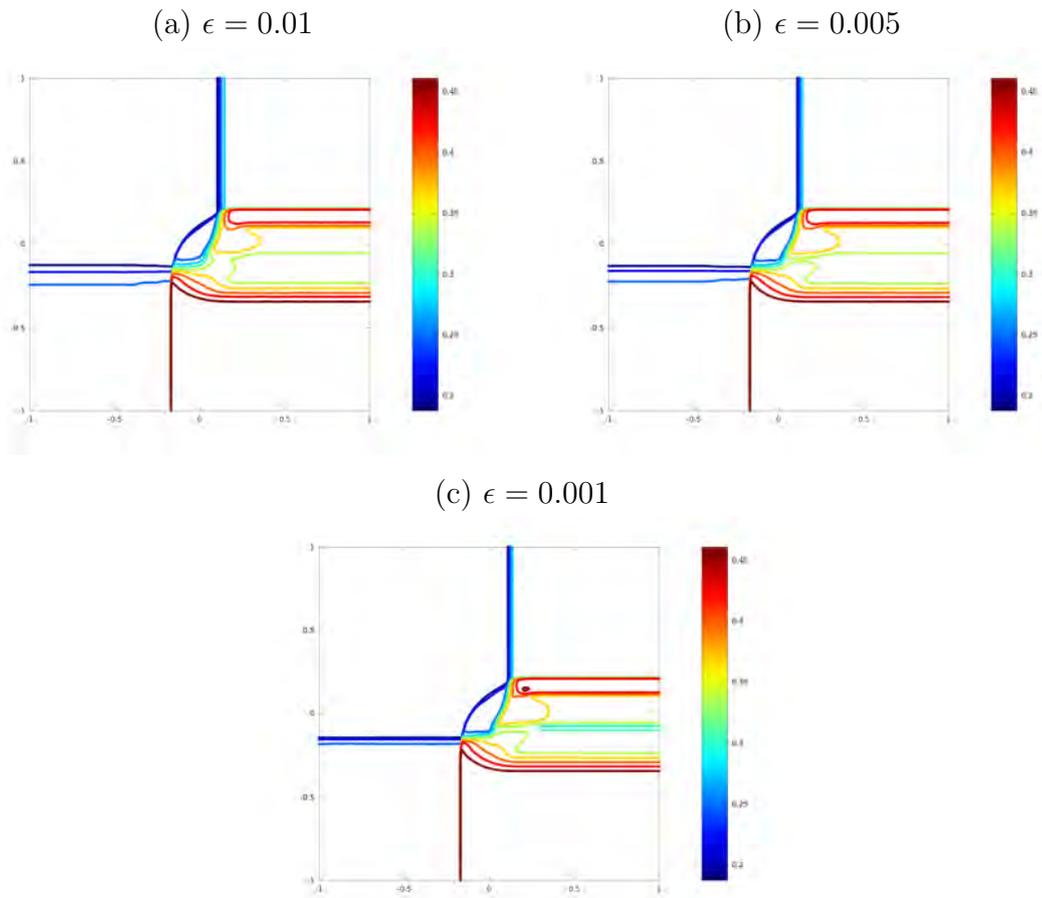


Figura F.11 - Componente  $z$  do campo magnético aproximada com o fluxo HLL para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

## F.2.2 Resultados: Fluxo HLLD

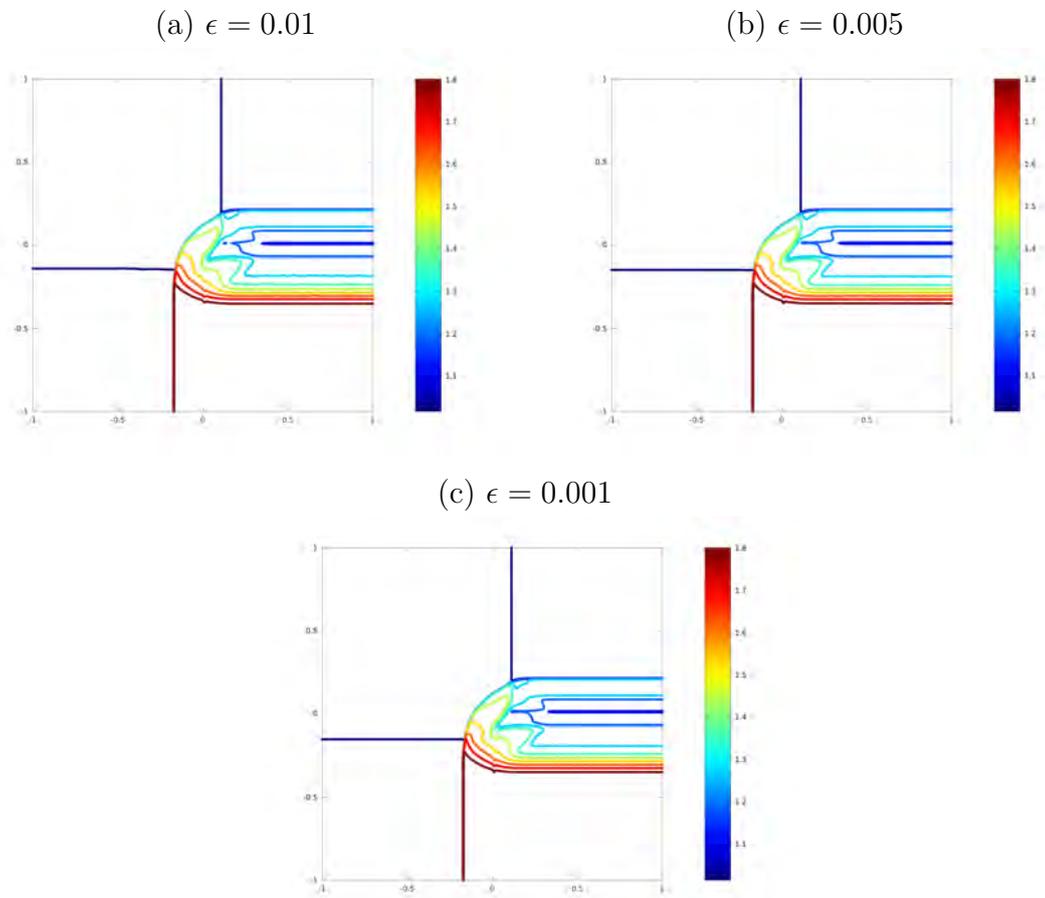


Figura F.12 - Variável densidade aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

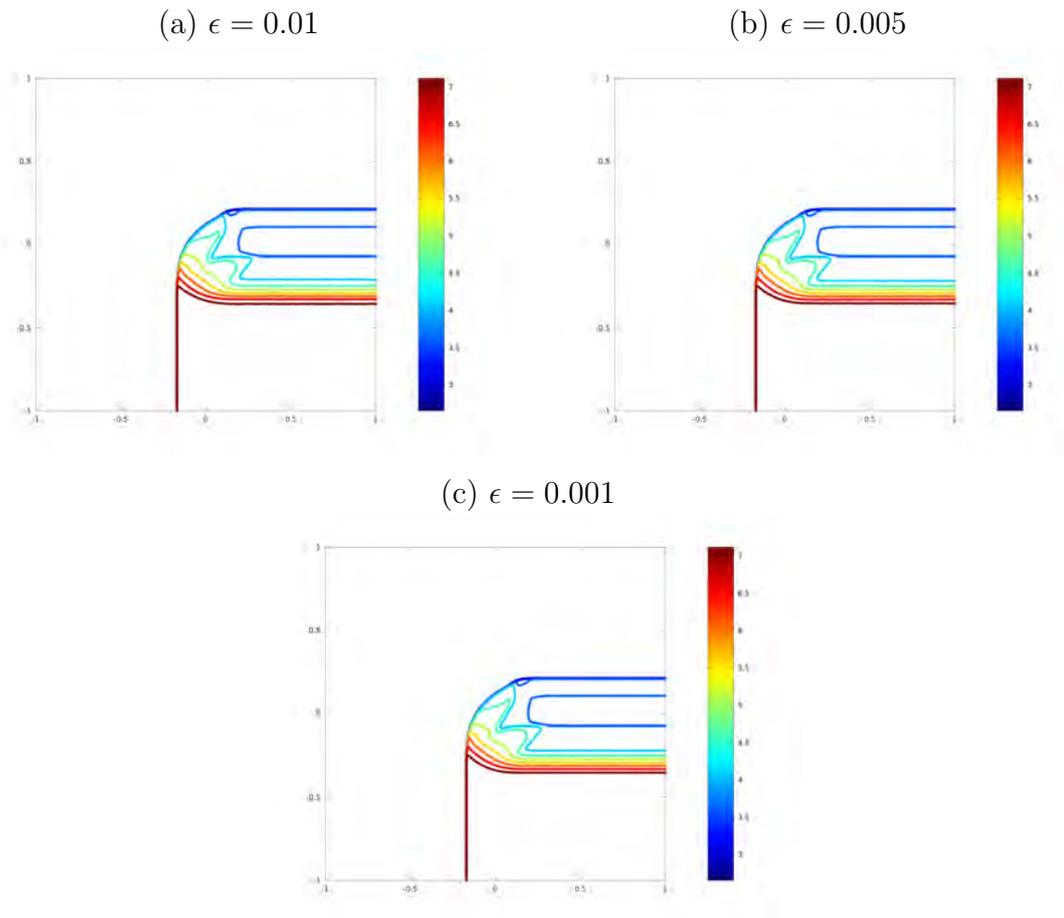


Figura F.13 - Variável pressão aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

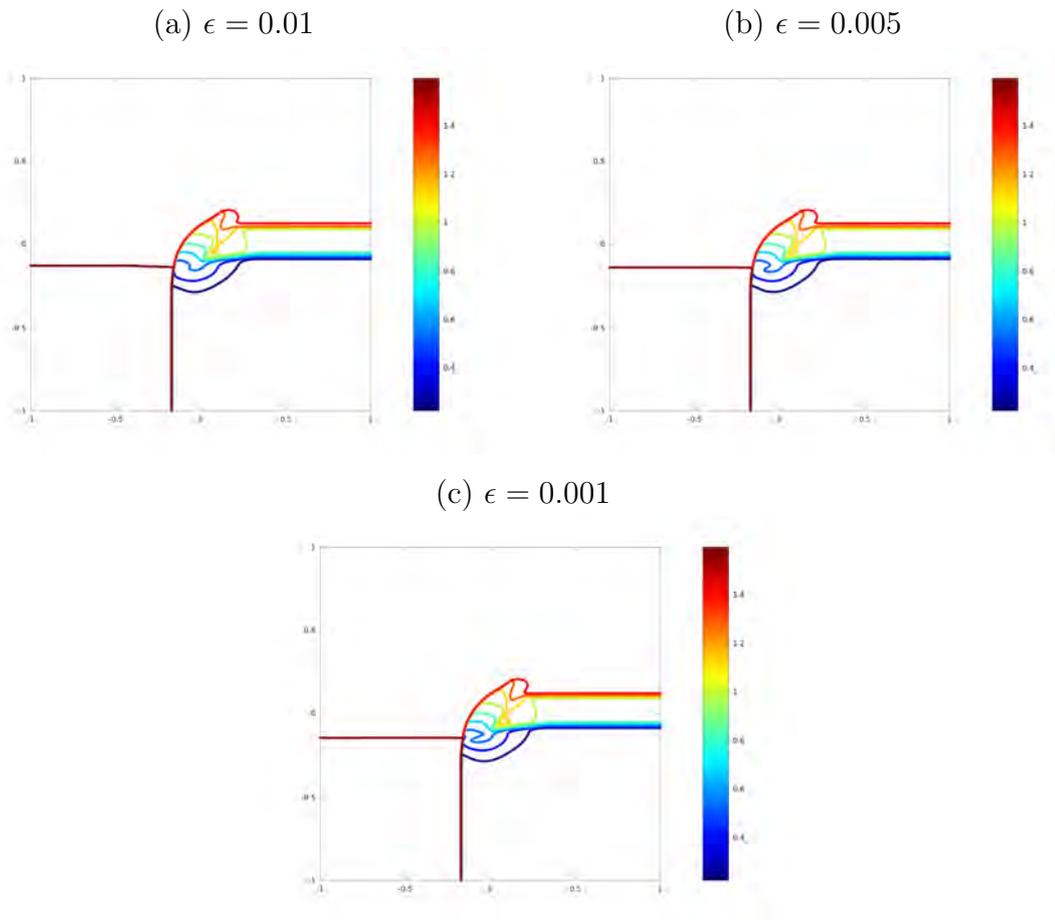


Figura F.14 - Componente  $x$  da velocidade aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

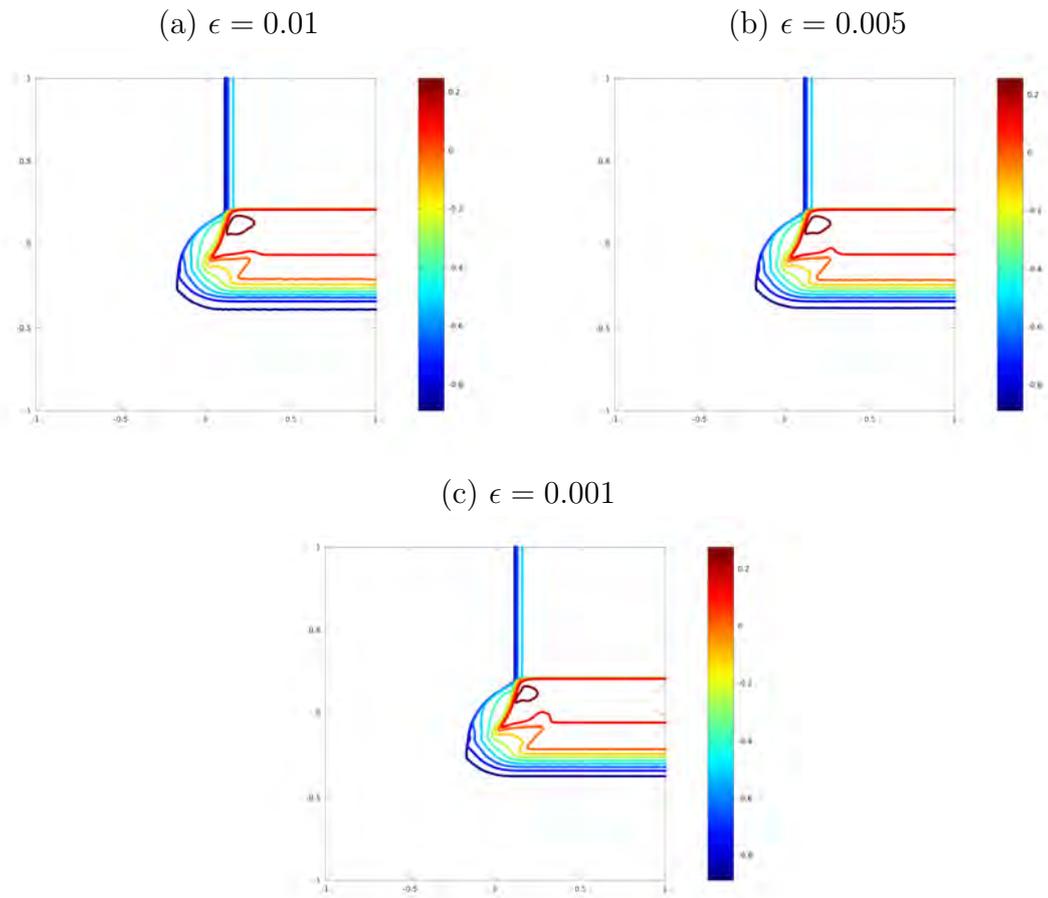


Figura F.15 - Componente  $y$  da velocidade aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

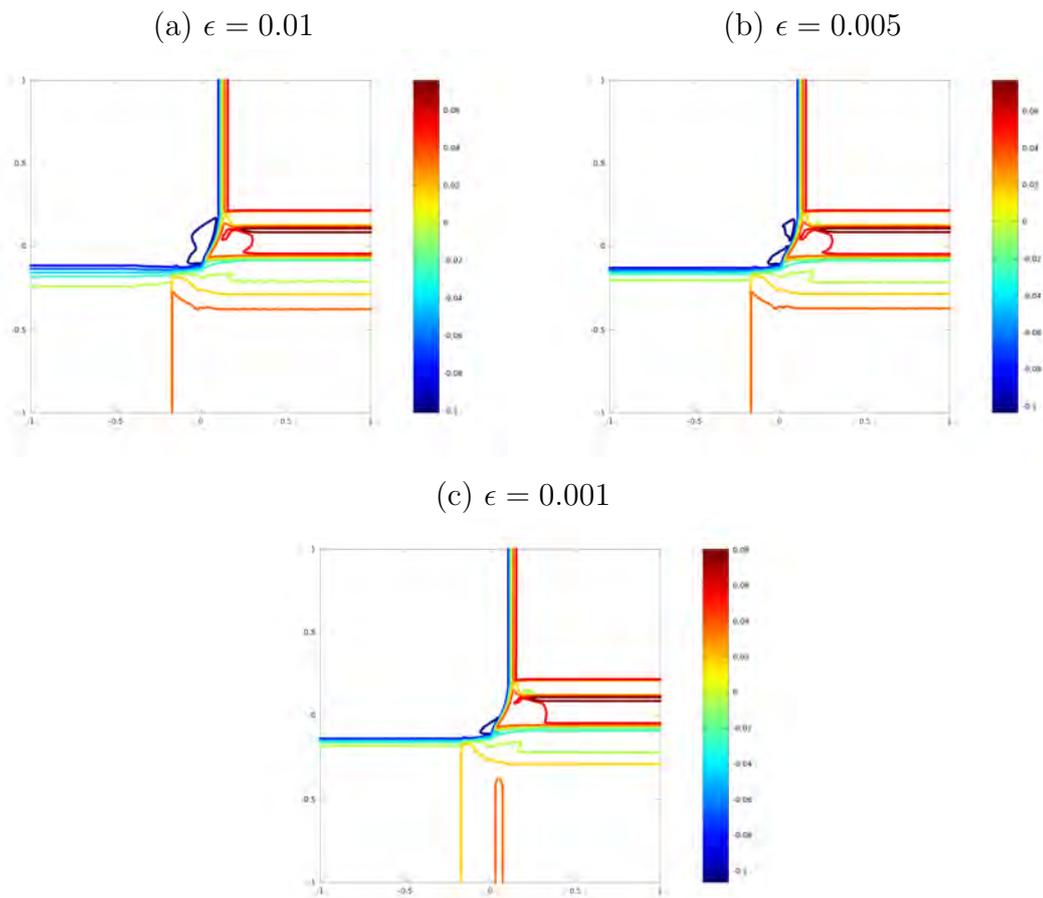


Figura F.16 - Componente  $z$  da velocidade aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

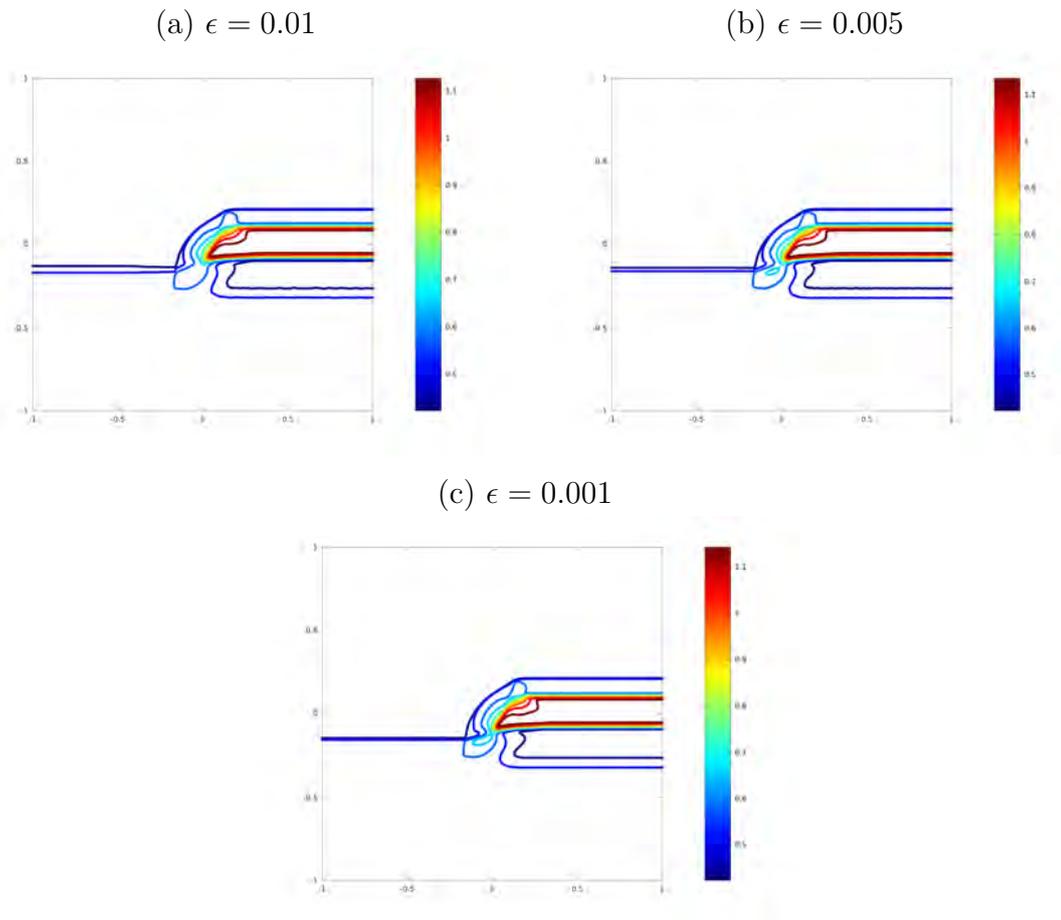


Figura F.17 - Componente  $x$  do campo magnético aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

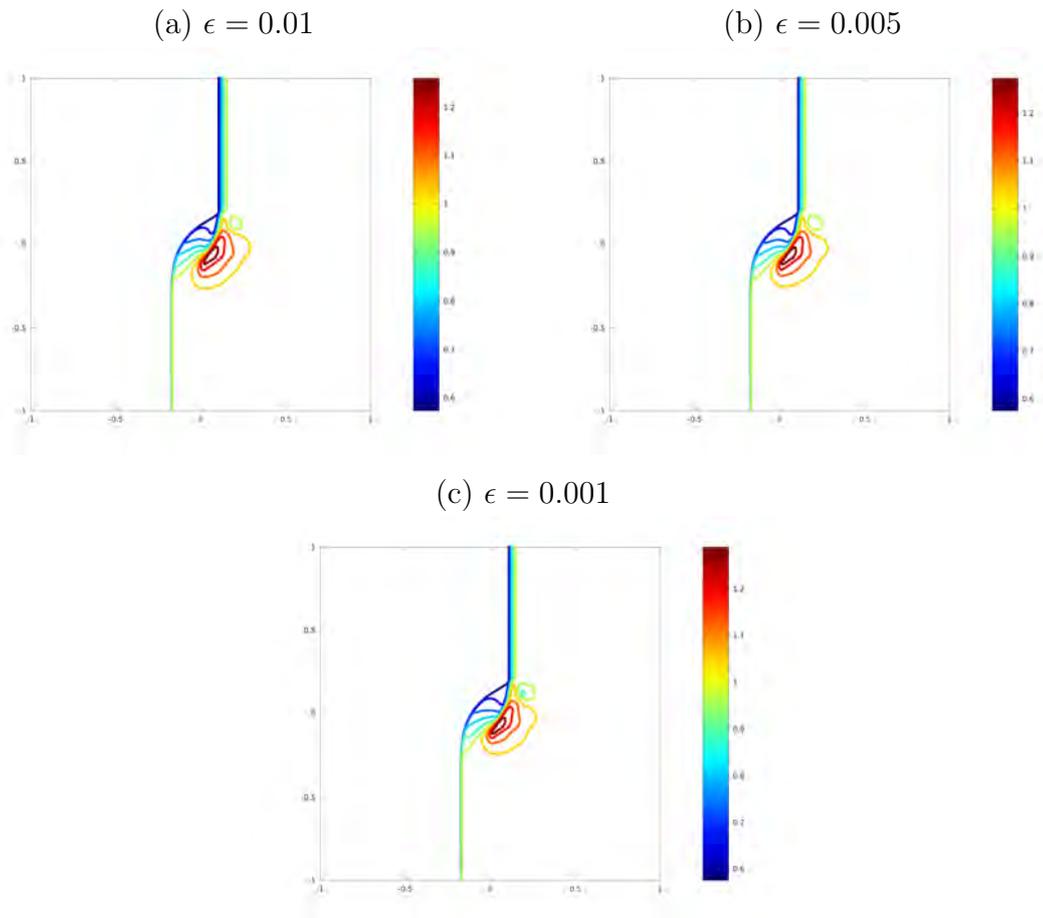


Figura F.18 - Componente  $y$  do campo magnético aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

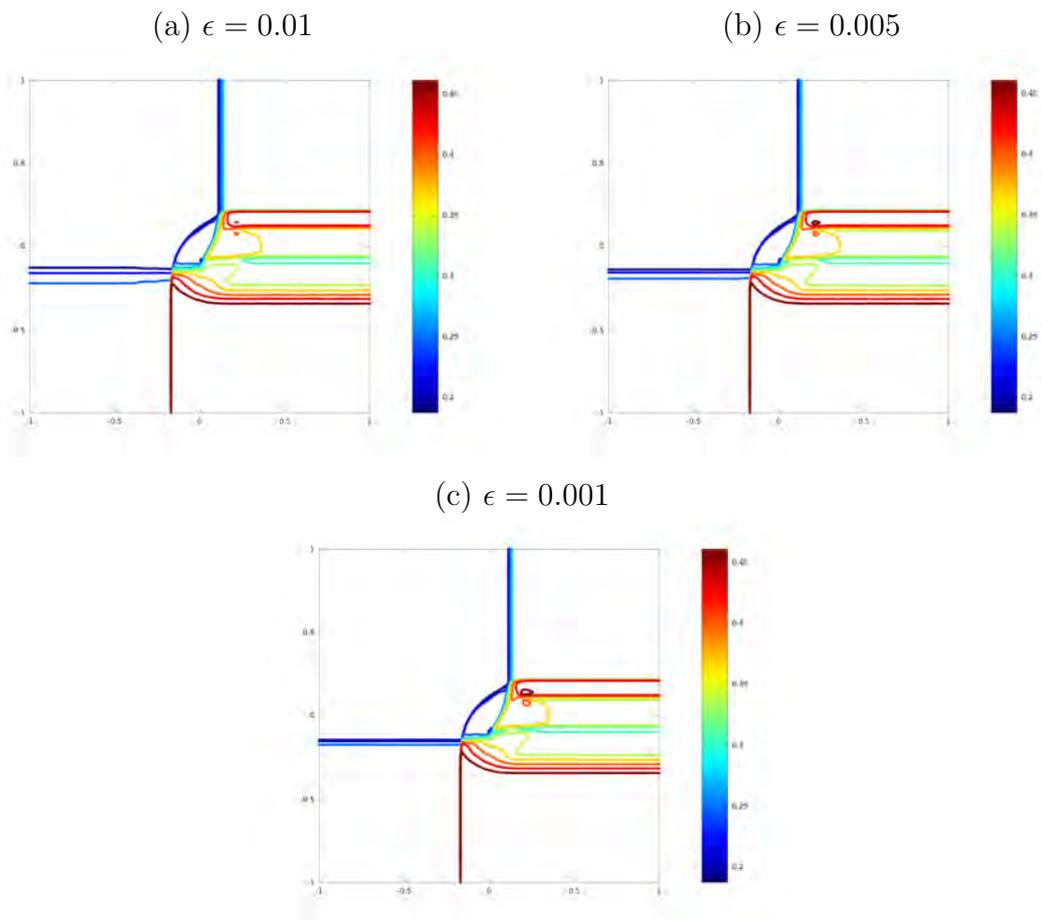


Figura F.19 - Componente  $z$  do campo magnético aproximada com o fluxo HLLD para o Caso 2. Utiliza-se uma malha adaptativa com  $2048 \times 2048$  células no intervalo  $[-1, 1] \times [-1, 1]$ ,  $c_{CFL} = 0.3$ ,  $\gamma = 5/3$ ,  $c_h/c_p^2 = 1$  e  $t = 0.1$ .

## APÊNDICE G - MR PARA VALORES PONTUAIS

Considere a hierarquia de um intervalo unitário de malhas diádicas uniformes

$$X_j = \{x_{j,k} = k2^{-j}, k = 0, \dots, 2^j\}$$

com espaçamento  $2^{-j}$ . Como mostrado na Figura G.1, isso significa que para ir de  $X_j$  para uma malha mais refinada  $X_{j+1}$ , deve-se adicionar a  $X_j$  novos pontos médios  $(2k+1)2^{-j-1}$  entre os pontos antigos  $k2^{-j}$  e  $(k+1)2^{-j}$ , reduzindo pela metade o tamanho do passo.

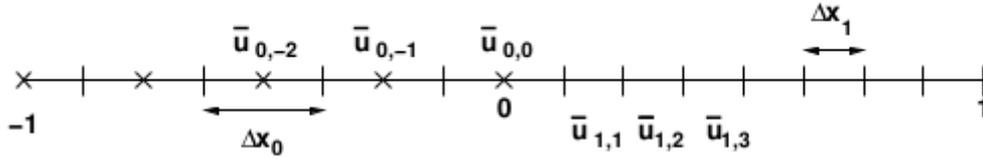


Figura G.1 - Malha diádica em um intervalo. Fonte: Domingues et al (2011).

Nessa geometria, em cada nível  $j$ , o operador de discretização  $\mathcal{D}_j : f \rightarrow \mathbf{f}_j$  associa uma função contínua  $f(x)$ , em um intervalo  $[0, 1]$ , a um vetor  $\mathbf{f}_j$ , o qual é formado pelos valores amostrados  $f_{j,k} = f(x_{j,k})$  dos pontos em  $X_j$ .

Para fazer o intercâmbio de informações entre dois níveis consecutivos  $j$  e  $j+1$ , os operadores de projeção e predição são necessários:

$$P_{j+1 \rightarrow j} : \mathbf{f}_{j+1} \rightarrow \mathbf{f}_j$$

$$P_{j \rightarrow j+1} : \mathbf{f}_j \rightarrow \tilde{\mathbf{f}}_{j+1}$$

O operador de projeção parte do nível  $j+1$  para o nível  $j$  fazendo apenas decimação, *i.e.*,

$$f_{j,k} = f_{j+1, 2k}.$$

O operador de predição é mais sutil, pois, utilizando o conhecimento dos valores amostrados em  $X_j$ , obtém-se os valores estimados  $\tilde{f}_{j+1,k} \approx f_{j+1,k}$  para os locais com

índice ímpar em  $X_{j+1}$ .

### G.0.3 Interpolando Predições

As principais propriedades requeridas para o operador interpolador de predição são

- 1 Nos pontos antigos, os dados são preservados:  $\tilde{f}_{j+1,2k} = f_{j+1,2k}$ .
- 2 O cálculo de  $\tilde{f}_{j+1,2k+1}$  requer apenas as amostras  $f_{j,m}$  em uma vizinhança fechada  $x_{j+1,2k+1}$ . (**Localização**)
- 3 A predição é exata para polinômios de um grau prescrito. (**Reprodução de polinômios**)

Um interesse particular surge, considerando as predições obtidas por interpolações feitas com polinômios de Lagrange. O exemplo mais simples é dado por uma interpolação linear

$$\tilde{f}_{j+1,2k+1} = \frac{f_{j,k} + f_{j,k+1}}{2}. \quad (\text{G.1})$$

Geralmente, para esquemas de ordem  $M$ , a idéia é definir  $\tilde{f}_{j+1,2k+1}$  pela estimativa em  $x_{j+1,2k+1}$  do polinômio  $p(x)$  de grau  $M-1$ , o qual interpola  $f_{j,m}$  nos  $M$  pontos  $x_{j,m} \in X_j$  que estão o mais perto possível de  $x_{j+1,2k+1}$ . Está claro que tal procedimento reproduz polinômios de grau  $M-1$ . O esquema é ilustrado na Figura G.2, para  $M=2$ .

Distante da fronteira do intervalo, e até mesmo para  $M=2L$ , tem-se o esquema clássico de subdivisão iterativa (DESLAURIERS; DUBUC, 1989), em que o estêncil interpolador é centrado em torno de  $x_{j+1,2k+1}$ , ou seja,  $x_{j,m}$ ,  $-L \leq m \leq L$ . Para exemplificar, considera-se o caso de interpolação cúbica ( $M=4$ ). No meio do intervalo, tem-se a fórmula

$$\tilde{f}_{j+1,2k+1} = \frac{9}{16}(f_{j,k} + f_{j,k+1}) - \frac{1}{16}(f_{j,k-1} + f_{j,k+2}), \quad 0 < k < 2^j.$$

Próximo à fronteira da esquerda, em  $x_{j+1,1}$ , escolhe-se o estêncil interpolador  $x_{j,m}$ ,  $m=0, 1, 2$  e  $3$ , resultando em

$$\tilde{f}_{j+1,1} = \frac{5}{16}f_{j,0} + \frac{15}{16}f_{j,1} - \frac{5}{16}f_{j,2} + \frac{1}{16}f_{j,3}, \quad (\text{G.2})$$

e, na fronteira à direita, em  $x_{j+1,2^{j+1}-1}$ , escolhe-se o estêncil interpolador  $x_{j,2^j-m}$ ,  $m = 0, 1, 2$  e  $3$  e inverte-se a fórmula para obter

$$\tilde{f}_{j+1,2^{j+1}-1} = \frac{1}{16}f_{j,2^j-3} - \frac{5}{16}f_{j,2^j-2} + \frac{15}{16}f_{j,2^j-1} + \frac{5}{16}f_{j,2^j}. \quad (\text{G.3})$$

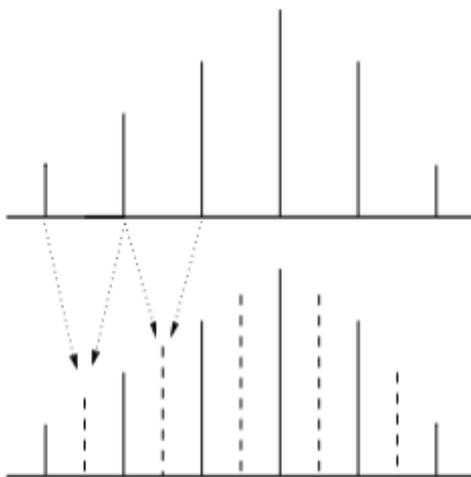


Figura G.2 - Predição utilizando a interpolação linear. Essa operação insere valores aproximados em/entre os valores antigos.

### G.1 Transformada MR

A análise multirresolução para valores pontuais é construída utilizando interpolação de predições. A diferença de informação entre um nível  $j$  e o nível acima  $j + 1$  ocorre nos pontos novos  $x_{j+1,2k+1} \in X_{j+1} \setminus X_j$ . Portanto, para medir esses detalhes, os coeficientes  $d_{j,k}$  são definidos como erros de predição em tais localidades, como indicado na Figura G.3.

Se esse procedimento for feito  $J$  vezes, do nível  $J - 1$  ao nível  $0$ , obtém-se os detalhes  $\mathbf{d}_j$  e o sinal  $\mathbf{f}_0$  no nível menos refinado da hierarquia, como mostra a Figura G.4. Na Figura G.5, a estrutura do processo reverso é mostrada, a qual se inicia por  $\mathbf{f}_0$  e  $\mathbf{d}_j$ ,  $j = 0, \dots, J - 1$ , e recupera  $\mathbf{f}_J$ .

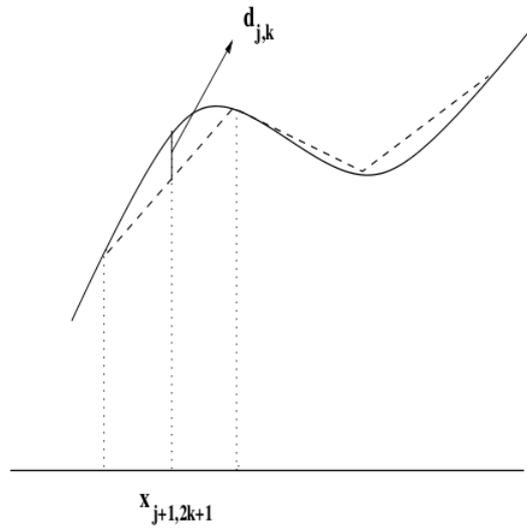


Figura G.3 - Coeficiente wavelet como sendo o erro de predição no caso linear. Fonte: Domingues et al (2011).

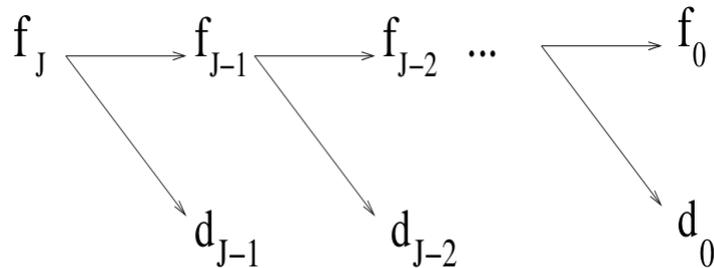


Figura G.4 - Estrutura do algoritmo de análise. Fonte: Domingues et al (2011).

### G.1.1 Análise - $\mathbf{f}_J \rightarrow \mathbf{f}_J^{MR} = (\mathbf{f}_0, \mathbf{d}_0, \dots, \mathbf{d}_{J-1})$

Para  $j = J - 1, \dots, 0$ , dado  $\mathbf{f}_{j+1}$ :

- Faça a projeção:  $P_{j+1 \rightarrow j} : \mathbf{f}_{j+1} \rightarrow \mathbf{f}_j$
- Faça a predição:  $P_{j \rightarrow j+1} : \mathbf{f}_j \rightarrow \tilde{\mathbf{f}}_{j+1}$
- Calcule erros de predição:  $d_{j,k} = f_{j+1,2k+1} - \tilde{f}_{\ell_{j+1}, 2k+1}$

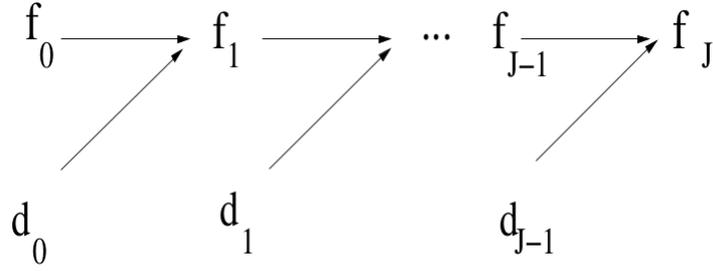


Figura G.5 - Estrutura do algoritmo de síntese. Fonte: Domingues et al (2011).

### G.1.2 Síntese - $\mathbf{f}_j^{MR} = (\mathbf{f}_0, \mathbf{d}_0, \dots, \mathbf{d}_{J-1}) \rightarrow \mathbf{f}_J$

Para  $j = 0, \dots, J - 1$ , dado  $\mathbf{f}_j, \mathbf{d}_j$ :

a) Faça a predição:  $P_{j \rightarrow j+1} : \mathbf{f}_j \rightarrow \tilde{\mathbf{f}}_{j+1}$

b) Calcule  $\mathbf{f}_{j+1}$

$$f_{j+1, 2k} = f_{j, k}$$

$$f_{j+1, 2k+1} = d_{j, k} + \tilde{f}_{j+1, 2k+1}$$

## G.2 Indicador wavelet de regularidade local

Como erros de interpolação, os coeficientes wavelet contém informações acerca da regularidade local de uma função analisada  $f(x)$ . Por exemplo, denotando por  $I_{j,k}$  o menor intervalo contendo o estêncil de interpolação usado para calcular  $\tilde{f}_{j+1, 2k+1}$ , a teoria de aproximação por interpolação polinomial declara que, para  $1 \leq s \leq M - 1$ , existe uma constante  $K = K(s, M)$  tal que a estimativa

$$|d_{j,k}| \leq K 2^{-(s+1)j} \max_{\xi \in I_{j,k}} |f^{(s+1)}(\xi)|$$

vale para funções  $f \in C^s(I_{j,k})$ , tendo derivada limitada de ordem  $s + 1$ .

Esse resultado pode ser usado para estimar o grau de regularidade de uma função em uma certa localidade, analisando a taxa de decaimento de magnitude dos coeficiente wavelet associados aos pontos em tal região.

### G.3 Contexto funcional

Para discretizações por valores pontuais, pode-se definir as decomposições de multirresolução do ponto de vista funcional, como descrito na Equação 7.1. Os valores pontuais da malha podem ser interpretados como

$$f_{j,k} = f(x_{j,k}) = \langle f, \Phi_{j,k}^* \rangle,$$

em que  $\Phi_{j,k}^*$  é a distribuição delta em  $x_{j,k}$ . Pela definição de wavelet dual  $\Psi_{j,k}^*$  tal que

$$d_{j,k} = f_{j+1,2k+1} - \tilde{f}_{j+1,2k+1} = \langle f, \Psi_{j,k}^* \rangle$$

o que implica em

$$\Psi_{j,k}^* = \Phi_{j,2k+1}^* - \sum_m h_m^k \Phi_{j,m}^*,$$

em que  $h_m^k$  são os coeficientes da predição interpoladora, tal que

$$\tilde{f}_{j+1,2k+1} = \sum_m h_m^k f_{j,m}.$$

Por exemplo, nas localidades centrais ao intervalo, tem-se  $h_m^k = \{1/2, 1/2\}$  para a interpolação linear e  $h_m^k = \{-1/16, 9/16, 9/16, -1/16\}$  para a interpolação cúbica.

A definição das funções básicas principais  $\Phi_{j,k}$  e  $\Psi_{j,k}$  baseia-se na convergência do operador de predição. Uma predição interpoladora é dita convergente se, para qualquer sequência inicial  $s_j = (s_{j,k})$ , existe uma função contínua  $s_j(x)$ , tal que  $s_j(x_{\ell,k}) = s_{\ell,k}$ , para  $\ell \geq j$ , em que  $s_\ell$  é definida pela subdivisão iterativa  $s_\ell = P_{\ell-1 \rightarrow \ell} s_{\ell-1}$ . Essa propriedade vale para predições feitas por interpolação utilizando polinômios de Lagrange de grau  $M - 1$ . Permite-se que a função básica  $\Phi_{j,k}(x)$  seja definida por um esquema de subdivisão iterativa de uma sequência de deltas de Dirac  $s_{j,m} = \delta(k - m)$ , o que significa que  $\Phi_{j,k}(x)$  satisfaz a propriedade de interpolação, *i.e.*,

$$\Phi_{j,k}(x_{j,m}) = \delta(k - m).$$

A relação de escala

$$\Phi_{j,k}(x) = \sum_m c_k(m) \Phi_{j+1,m}(x),$$

em que  $c_k(m) = \Phi_{j,k}(x_{j+1,m})$ , é uma consequência do fato de que as funções, em ambos os lados, coincidem em  $X_{j+1}$ . Observa-se que os coeficientes de escala são

$c_k(2m) = \delta(m - k)$  e  $c_k(2m + 1) = h_k^m$ . Além disso, as seguintes propriedades são válidas, como em Donoho (1992),

- A regularidade aumenta com grau de interpolação  $M - 1$ .
- Distante das fronteiras,  $\Phi_{j,k}(x) = \phi(2^j x - k)$ , em que  $\phi = \varphi_M$  é a função fundamental Dubuc-Deslauriers.
- $\Phi_{j,k}(x)$  tem um suporte compacto, com tamanho do suporte  $|\text{supp}(\Phi_{j,k})| = O(2^{-j})$  aumentando com  $M$ .

Definindo os subespaços funcionais

$$V_j = \text{span}\{\Phi_{j,k}(x), k = 0, \dots, 2^j\} \subset C[0, 1],$$

por construção, tem-se  $V_j \subset V_{j+1}$ . Para funções  $f \in C[0, 1]$ , considera-se a sequência de aproximações

$$f_j(x) = \sum_k f_{j,k} \Phi_{j,k}(x),$$

a qual corresponde aos operadores de interpolação de  $f$  em  $V_j$ . Pode-se provar que

$$\begin{aligned} \|f_j\|_\infty &\leq C \|f\|_\infty, \\ \|f - f_j\|_\infty &\rightarrow 0, \text{ as } j \rightarrow \infty. \end{aligned}$$

Considera-se as wavelets  $\Psi_{j,k}(x) = \Phi_{j+1,2k+1}(x)$  como as funções escala associadas aos pontos  $x_{j+1,2k+1} \in X_{j+1} \setminus X_j$  e ao espaço

$$W_j = \text{span}\{\Psi_{j,k}(x), k = 0, \dots, 2^j - 1\},$$

o qual é formado pelas funções em  $V_{j+1}$  que desaparecem em  $X_j$ . Portanto, isso resulta que  $V_{j+1}$  é a soma direta dos espaços  $V_j$  e  $W_j$ . E desde que as diferenças entre duas reconstruções consecutivas  $f_{j+1}(x) - f_j(x)$  tendem a zero em  $X_j$  e coincidem com  $d_{j,k} = f_{j+1,2k+1} - \tilde{f}_{j+1,2k+1}$  em  $X_{j+1} \setminus X_j$ , tem-se a representação bi-nível da soma direta por

$$\begin{aligned} V_{j+1} &= V_j + W_j, \\ \sum_k f_{j+1,k} \Phi_{j+1,k} &= \sum_k f_{j,k} \Phi_{j,k} + \sum_k d_{j,k} \Psi_{j,k}. \end{aligned}$$

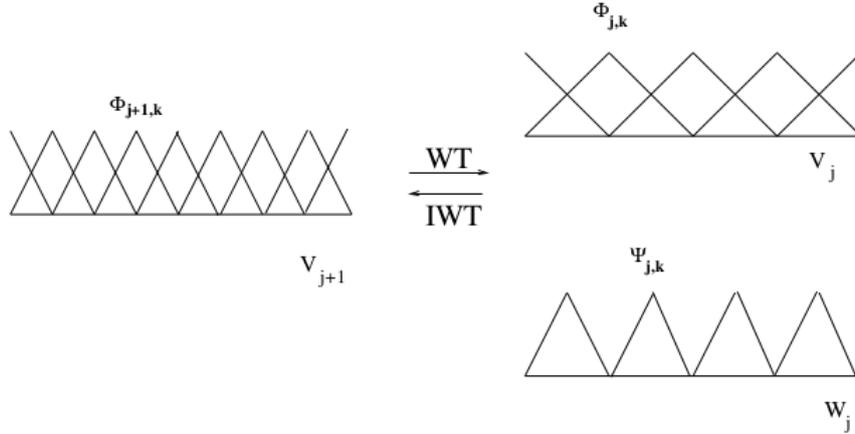


Figura G.6 - Decomposição bi-nível para análise multirresolução interpoladora para  $j = 4$ , associada à predição linear. Fonte: Domingues et al (2011).

A Figura G.6 ilustra a decomposição bi-nível para análise multirresolução interpoladora para  $j = 4$ , associada à predição linear.

Aplicando essa decomposição iterativamente, as transformadas multirresolução direta (análise) e inversa (síntese) são associadas à representação multirresolução

$$V_J = V_0 + \sum_{j=0}^{J-1} W_j, \quad (\text{G.4})$$

$$\sum_k f_{J,k} \Phi_{J,k}(x) = \sum_k f_{0,k} \Phi_{0,k}(x) + \sum_{j=0}^{J-1} \sum_k d_{j,k} \Psi_{j,k}(x). \quad (\text{G.5})$$

#### G.4 Compressão de dados

Dados os valores pontuais  $\mathbf{f}_J$  no nível mais refinado  $J$ , depois da etapa de análise da transformada MR, eles podem ser representados como

$$\mathbf{f}_J^{MR} = (\mathbf{f}_0, \mathbf{d}_0, \dots, \mathbf{d}_{J-1}).$$

Dados o limiar de truncamento  $\epsilon_j$ , a compressão de dados é obtida pelo operador de truncamento

$$\mathcal{T}_{j,k}^\epsilon = \begin{cases} 0 & \text{se } |d_{j,k}| \leq \epsilon_j, \\ d_{j,k} & \text{caso contrário.} \end{cases}$$

As duas principais propriedades do operador predição, chamadas localização e reprodução polinomial, têm uma consequência imediata no tamanho dos coeficientes wavelet nas regiões suaves. Por exemplo, para funções como as mostradas na Figura G.7, é esperado que a representação multirresolução comprimida

$$\mathbf{f}_J^{MR,\epsilon} = (\mathbf{f}_0, \mathbf{d}_0^\epsilon, \dots, \mathbf{d}_{J-1}^\epsilon)$$

tenha um número reduzido de termos que não desaparecem.

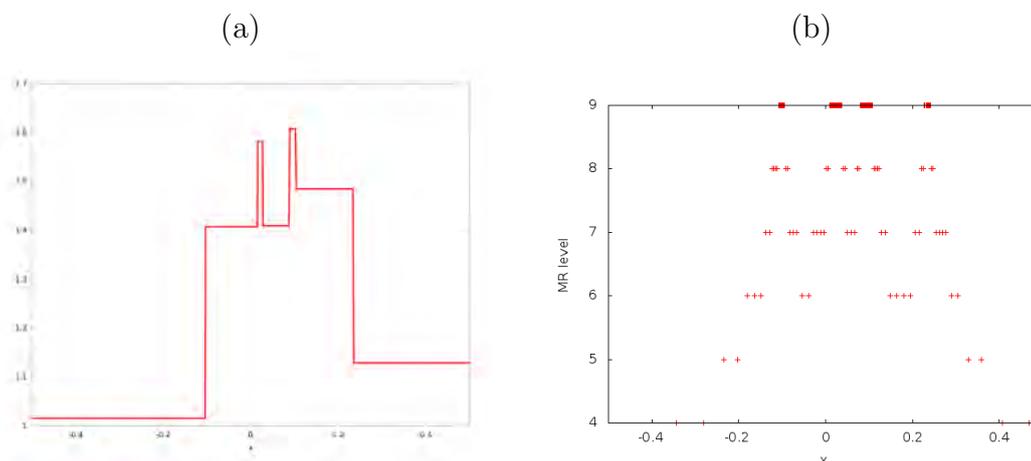


Figura G.7 - A função  $f(x)$  (a) e a posição dos coeficientes wavelet significantes  $|d_{j,k}| \geq 1 \cdot 10^{-3}$  (b).

Para o método de Representação de Pontos Esparços (SPR), no contexto dos métodos de diferenças finitas, a idéia principal é utilizar os coeficientes wavelet em uma representação multirresolução de valores pontuais da solução para construir malhas adaptativas: utilizar células menos refinadas nas regiões onde os coeficientes wavelet são pequenos (e a solução é suave) e refinar onde esses coeficientes são significantes. Essa técnica é introduzida por Holmström (1997).

## G.5 Estabilidade

Existe um outro aspecto-chave na compressão wavelet: depois da aplicação da transformada MR inversa no dado comprimido  $\mathbf{f}_j^{MR,\epsilon}$ , como fazer a comparação dos valores reconstruídos  $\mathbf{f}_j^\epsilon$  com os valores originais? Nota-se que uma perturbação nos coeficientes wavelet no nível  $j$  é transmitida para níveis mais altos após sucessivas predições. Tais perturbações poderiam ser amplificadas incontrolavelmente por esse processo?

No contexto da convergência de esquemas de subdivisão, o resultado de estabilidade é mantido (ver Harten (1996), Seção 3.C)

$$\|\mathbf{f}_J - \mathbf{f}_J^\epsilon\|_\infty \leq C \sum_{j=0}^{J-1} \epsilon_j.$$

### G.5.1 Análise MR para valores pontuais em dimensões superiores

Em dimensões superiores, os esquemas de multirresolução para malhas cartesianas podem ser obtidos ao fazer os produtos tensoriais dos esquemas unidimensionais. Por exemplo, no domínio quadrado  $[0, 1] \times [0, 1]$ , iniciando-se a partir de  $X_0$ , que é o nível menos refinado, a hierarquia de malhas uniformes  $X_j$ ,  $j \geq 1$  é

$$X_j = \{x_{j,(k,m)} = (k2^{-j}, m2^{-j}), 0 \leq k \leq 2^j, 0 \leq m \leq 2^j\}.$$

Como indicado na Figura G.8, o refinamento diádico  $X^{j+1}$ , é obtido pela inclusão de novos pontos médios  $(2k\Delta_x^{\ell+1}, (2m+1)\Delta_y^{\ell+1})$ ,  $((2k+1)\Delta_x^{j+1}, 2m\Delta_y^{j+1})$ , e  $((2k+1)\Delta_x^{j+1}, (2m+1)\Delta_y^{j+1})$ . Para a discretização

$$f_{j,(k,m)} = f(x_{j,(k,m)})$$

tem-se o operador de projeção

$$f_{j,(k,m)} = f_{j+1,(2k,2m)}.$$

A aplicação do produto tensorial da interpolação de predição corresponde à interpolação 1D na direção  $x$  para calcular  $\tilde{f}_{j+1,(2k+1,2m)}$  e na direção  $y$  para calcular  $\tilde{f}_{j+1,(2k,2m+1)}$ . Finalmente,  $\tilde{f}_{j+1,(2k+1,2m+1)}$  é obtida pela interpolação 1D na direção

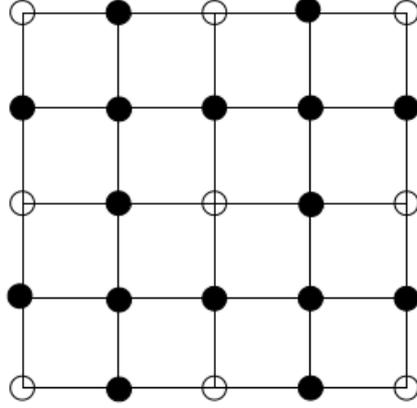


Figura G.8 - Malhas diádicas uniformes: o símbolo  $\circ$  representa os pontos em  $X_j$ , enquanto  $\bullet$  representa os pontos em  $X_{j+1} \setminus X_j$ .  
Fonte: Domingues et al (2011).

$x$  dos, já calculados, valores interpolados  $\tilde{f}_{j+1,(2s,2m+1)}$ . Então, os coeficientes wavelet são definidos como erros de predição (interpolação)

$$d_{j,(k,m)}^{(1)} = f_{j+1,(2k,2m+1)} - \tilde{f}_{j+1,(2k,2m+1)}$$

$$d_{j,(k,m)}^{(2)} = f_{j+1,(2k+1,2m)} - \tilde{f}_{j+1,(2k+1,2m)}$$

$$d_{j,(k,m)}^{(3)} = f_{j+1,(2k+1,2m+1)} - \tilde{f}_{j+1,(2k+1,2m+1)}$$

As transformadas MR em 2D herdam da abordagem 1D as propriedades de cancelamento de polinômios por coeficientes wavelet, localização e estabilidade, e os coeficientes wavelet podem ser usados como indicadores de regularidade. Esse princípio é utilizado na construção da SPR das funções 2D: apenas os valores pontuais correspondentes aos coeficientes wavelet significantes são mantidos na malha adaptativa.

## **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). Constam 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 sequência 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.