



RAPHAEL DE OLIVEIRA GARCIA

MÉTODOS DE VOLUMES FINITOS CENTRADOS *UNSPLITTING*
UTILIZADOS NA OBTENÇÃO DE SOLUÇÕES EM
MAGNETOHIDRODINÂMICA RELATIVÍSTICA: APLICAÇÕES EM
DISCOS E JATOS

CAMPINAS

2014



UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA
E COMPUTAÇÃO CIENTÍFICA

RAPHAEL DE OLIVEIRA GARCIA

MÉTODOS DE VOLUMES FINITOS CENTRADOS *UNSPPLITTING*
UTILIZADOS NA OBTENÇÃO DE SOLUÇÕES EM
MAGNETOHIDRODINÂMICA RELATIVÍSTICA: APLICAÇÕES EM DISCOS
E JATOS

Tese apresentada ao Instituto de Matemática, Estatística e
Computação Científica da Universidade Estadual de Campi-
nas como parte dos requisitos exigidos para a obtenção do
título de Doutor em Matemática Aplicada.

Orientador: Prof. Dr. SAMUEL ROCHA DE OLIVEIRA

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA
TESE DEFENDIDA PELO ALUNO RAPHAEL DE OLIVEIRA
GARCIA, E ORIENTADA PELO PROF. DR. SAMUEL ROCHA
DE OLIVEIRA.

Assinatura do Orientador

A handwritten signature in black ink, reading "Samuel R. Oliveira.", is written over a horizontal line. The signature is in a cursive style.

CAMPINAS

2014

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Maria Fabiana Bezerra Muller - CRB 8/6162

G165m Garcia, Raphael de Oliveira, 1982-
Métodos de volumes finitos centrados unsplitting utilizados na obtenção de soluções em magnetohidrodinâmica relativística: aplicações em discos e jatos / Raphael de Oliveira Garcia. – Campinas, SP : [s.n.], 2014.

Orientador: Samuel Rocha de Oliveira.
Tese (doutorado) – Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica.

1. Método dos volumes finitos. 2. Astrofísica relativística. 3. Magnetoidrodinâmica. 4. Discos magnéticos. 5. Jatos (Matemática). I. Oliveira, Samuel Rocha de, 1962-. II. Universidade Estadual de Campinas. Instituto de Matemática, Estatística e Computação Científica. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Centered finite volume methods unsplitting used in the obtaining of solutions in relativistic magnetohydrodynamics: applications in disks and jets

Palavras-chave em inglês:

Finite volume method
Relativistic astrophysics
Magnetohydrodynamics
Magnetic disks
Jet bundles (Mathematics)

Área de concentração: Matemática Aplicada

Titulação: Doutor em Matemática Aplicada

Banca examinadora:

Samuel Rocha de Oliveira [Orientador]
Alberto Vazquez Saa
Eduardo Cardoso de Abreu
Anderson Caproni

Vera Jatenco Silva Pereira

Data de defesa: 21-02-2014

Programa de Pós-Graduação: Matemática Aplicada

Tese de Doutorado defendida em 21 de fevereiro de 2014 e aprovada

Pela Banca Examinadora composta pelos Profs. Drs.

Samuel R. Oliveira.

Prof(a). Dr(a). SAMUEL ROCHA DE OLIVEIRA

Alberto

Prof(a). Dr(a). ALBERTO VAZQUEZ SAA

Eduardo Cardoso de Abreu.

Prof(a). Dr(a). EDUARDO CARDOSO DE ABREU

Anderson Caproni

Prof(a). Dr(a). ANDERSON CAPRONI

Vera Jatenco Silva Pereira.

Prof(a). Dr(a). VERA JATENCO SILVA PEREIRA

Abstract

We have developed a new computer program in Fortran 90, in order to obtain numerical solutions of a system of partial differential equations of Relativistic Magnetohydrodynamics with predetermined gravitation (GRMHD), capable of simulate the formation of relativistic jets from the accretion disk of matter up to his ejection. Initially we carried out a study on numerical methods of Unidimensional Finite Volume, namely Lax-Friedrichs, Lax-Wendroff, Nessyahu-Tadmor method and Godunov methods dependent on Riemann problems, applied to equations Euler in order to verify their main features and make comparisons among those methods. It was then implemented the methods of Finite Volume Centered Lax-Friedrichs and Nessyahu-Tadmor, which are numerical schemes that have a formulation free and without dimensional separation Riemann problem solvers, even in two or more spatial dimensions, at this point, already applied in equations GRMHD. A Lax-Wendrof Runge-Kutta method of order 3, with the property of Total Value Descending (TVD) in time and size separation, was also applied to the same problem. Finally, with Nessyahu - Tadmor method was possible to obtain solutions stable numerical - without spurious oscillations or excessive dissipation - from the process of magnetized accretion disk, in rotation with respect to a central black hole (BH) Schwarzschild and immersed in a magnetosphere, to the ejection of matter in the form of jet over a distance of fourteen times the radius of the BH, a record in terms of astrophysical simulation.

Resumo

Neste trabalho foi desenvolvido um novo programa computacional em Fortran 90, com o objetivo de obter soluções numéricas de um sistema de equações diferenciais parciais de Magnetohidrodinâmica Relativística, com gravitação pré-determinada (GRMHD), capaz de simular a formação de jatos relativísticos desde a acreção de disco de matéria até sua ejeção. De início fez-se um estudo sobre métodos numéricos de Volumes Finitos Unidimensionais, a saber método Lax-Friedrichs, Lax-Wendroff, Nessyahu-Tadmor e métodos de Godunov dependentes de problemas de Riemann, aplicados nas equações de Euler com

o intuito de verificar as suas principais características e de efetuar comparações entre aqueles métodos. Em seguida implementou-se os métodos de Volumes Finitos Centrados Lax-Friedrichs e Nessyahu-Tadmor, que são esquemas numéricos que possuem uma formulação sem separação dimensional e livres de resolvedores de problemas de Riemann, mesmo em duas ou mais dimensões espaciais; neste ponto, já aplicados nas equações de GRMHD. Um método Lax-Wendroff com Runge-Kutta de ordem 3, com a propriedade de ser Valor Total Decrescente (TVD) no tempo e com separação dimensional, também foi aplicado no mesmo problema. Por fim, com o método Nessyahu-Tadmor foi possível obter soluções numéricas estáveis - sem oscilações espúrias nem dissipação excessiva - desde o processo de acreção do disco magnetizado, em rotação com relação a um buraco negro central (BH) de Schwarzschild e imerso a uma magnetosfera, até a ejeção de matéria em forma de jato ao longo de uma distância de quatorze vezes o raio do BH, um recorde em termos de simulação astrofísica.

Sumário

Introdução	1
1 Jatos Relativísticos - Aspectos do Fenômeno	5
1.1 Aspectos históricos	5
1.2 A formação de jatos	9
1.2.1 Jatos em AGN	13
1.2.2 Jatos em Microquasares	15
1.2.3 Jatos em Supernovas	17
1.2.4 Jatos em Explosões de Raios <i>gamma</i>	18
1.3 Simulações	19
1.3.1 Disco de Acreção	19
1.3.2 Jatos	21
1.3.3 Jatos e Discos	23
2 Modelagem de Jatos Relativísticos	25
2.1 As equações de GRMHD ideal	25
2.2 Métrica de Schwarzschild	29
2.3 Disco de Acreção	30
2.4 Equação de Estado	33
3 Métodos de Volumes Finitos	35
3.1 Formulação Geral de Leis de Conservação	35
3.1.1 Leis de Conservação unidimensionais	36
3.1.2 Leis de Conservação	43

3.2	Formulação Geral de Métodos de Volumes Finitos Unidimensionais	46
3.3	Convergência, Consistência e Estabilidade	51
3.3.1	Convergência	52
3.3.2	Consistência	54
3.3.3	Estabilidade	56
3.3.4	Teorema de Lax-Richtmyer	58
3.3.5	A análise de von Neumann	59
3.3.6	Estabilidade não linear	61
3.4	Dispersão e Dissipação	62
3.5	Métodos de Volumes Finitos Centrados	63
3.5.1	O método de Lax-Friedrichs	68
3.5.2	O Método de Nessyahu-Tadmor	70
3.5.3	O método de Lax-Wendroff	75
3.6	Métodos de Volumes Finitos do tipo Godunov	77
3.6.1	Método de Godunov de primeira ordem	79
3.7	Comparações	84
3.7.1	Recursos Computacionais	85
3.7.2	Exemplo 1 - problema de Sod	86
3.7.3	Exemplo 2 - problema de Sod	89
3.7.4	Exemplo 3 - problema de Woodward & Colella	95
3.8	Métodos Numéricos com termo fonte	112
3.9	Métodos Numéricos Bidimensionais	113
3.9.1	Métodos com separação dimensional	113
3.9.2	Métodos sem separação dimensional	115
3.9.3	O Método Nessyahu-Tadmor Bidimensional	115
4	Métodos Numéricos aplicados em problemas de GRMHD	121
4.1	Autovalores e Autovetores	121
4.2	Códigos em Magnetohidrodinâmica Relativística	124
4.2.1	HARM	124

4.2.2	cosmos++	125
4.2.3	whiskyMHD	126
4.2.4	ECHO	127
4.2.5	WHARM	127
4.2.6	CoCoA/CoCoNuT	128
4.2.7	GRHydro	128
4.2.8	Outros Códigos	129
4.2.9	Observações	129
4.3	Métodos Implementados	129
5	Implementações Computacionais	131
5.1	As equações Implementadas	131
5.2	Equações Complementares	133
5.3	O código	137
6	Simulações	143
6.1	Exemplo 1	143
6.2	Exemplo 2	147
6.3	Exemplo 3	157
7	Considerações Finais e Trabalhos Futuros	159
7.1	Considerações Finais	159
7.2	Trabalhos Futuros	162
	Referências Bibliográficas	165
	Apêndices	175
A	Código	175

“Sonho que se sonha só é só um sonho que se sonha só,
mas sonho que se sonha juntos é realidade.”

Prelúdio - Raul Seixas

Agradecimentos

Ao Samuel Rocha de Oliveira por todos os ensinamentos e orientações feitos nesses anos de convivência, pela sua dedicação e sabedoria, pelo seu carisma, pelo seu companheirismo e pelas inúmeras ajudas realizadas.

Aos meus pais - Edson e Ailda - que não mediram esforços para me incentivar, apoiar e me sustentar nos estudos, pela maravilhosa educação que vocês passaram não somente a mim, mas para todos os seus filhos.

Aos meus irmãos - Alexandre e Adriana - que sempre me apoiaram, por todos os anos de amizade e por tudo que vivemos juntos. Ao grande Enzo que inicia seus primeiros passos nesta vida.

À Graciele por todo amor, carinho, atenção, paciência, respeito, por todos esses maravilhosos anos que estamos vivendo juntos. A toda luta que enfrentamos juntos ao longo desses anos. E a todos os instantes que estamos juntos.

Aos meus familiares que sempre torceram por mim - minhas tias Nina, Fátima e Vilma, meus tios Foca, Zé, Nenem e Nunes, meus primos Edmilson, Edmundo, Gabriel, Felipe, Rodrigo e Leonardo, minhas primas Érika, Janaina, Jacyara e Carol, meus avôs Bento e Pedro e minhas avós Flôr e Aulida.

A todos os meus familiares São Paulinos que fazem permanecer a força tricolor em nossa família.

Aos professores, pesquisadores e funcionários do IMECC que colaboraram diretamente ou indiretamente para que esses anos de estudos resultassem nesta tese.

Lista de Figuras

1.1	Imagem obtida pelo telescópio Hubble do jato M87 - fonte: NASA Hubble Space Telescope Images [99].	5
1.2	Na primeira linha à esquerda encontram-se os elementos fundamentais envolvidos no processo de formação de jatos astrofísicos e à direita a esquematização da formação de um vento não colimado. Na segunda linha à esquerda têm-se a formação de jato em um dos polos do objeto compacto central e à direita o jato formado a partir do disco de acreção.	11
1.3	Imagens do telescópio Hubble da galáxia Cygnus A [99].	14
1.4	Representação gráfica da formação de jatos em um quasar cuja matéria acrescida advém da galáxia que hospeda tal sistema. Figura fora de escala.	15
1.5	Representação gráfica da formação de jatos em um microquasar cuja matéria acrescida advém de uma estrela companheira. Figura fora de escala.	16
1.6	À esquerda tem-se a representação gráfica da formação de jatos em uma supernova cuja matéria ejetada forma objetos chamados de Herbig-Haro e à direita tem-se o objeto Herbig-Haro 47 captado pelo telescópio Hubble. [99]	18
1.7	À esquerda tem-se uma representação do campo magnético achatando o envelope que envolve um toro e à direita tem-se a acreção desenvolvendo-se e delimitando duas regiões separadas por um cone: uma região de instabilidade que é evoluída no próprio disco e suas adjacências e uma outra região dentro do cone que expelle uma baixa quantidade de matéria não colimada pelos polos do BH - tal ejeção é chamada de vento . Simulações desta natureza encontram-se em [14, 24].	22

3.1	A ilustração mostra uma representação geral dos métodos de volumes finitos. Os pontos x_i 's - localizados nos pontos médio de cada volume finito Ω_i 's - constituem o domínio espacial computacional onde no tempo t_n a solução numérica é conhecida e no tempo t_{n+1} a solução é obtida através dos valores U_i^n 's e dos valores $F_{i+1/2}^n$'s definidos pela função fluxo numérico.	50
3.2	Representação gráfica de Métodos de Volumes Finitos Centrados. Cada polinômio p_i está definido em sua respectiva célula média e cada descontinuidade está localizada no ponto médio do intervalo I_i , isto é, no ponto x_i . Nota-se que os fluxos F_i são diretamente encontrados, pois a solução numérica é conhecida em todos os pontos x_i 's e esses fluxos estão representados por pontos \times .	66
3.3	Representação de métodos de volumes finitos centrados em que a condição CFL é satisfeita, para que as evoluções de descontinuidades dentro de I_i não atinjam os extremos. As retas que saem da descontinuidade em $x_{i+1/2}$ representam a evolução temporal das descontinuidades de t_n até t_{n+1} . No caso, a solução numérica no tempo t_{n+1} é obtida em uma malha deslocada pela equação (3.42) através dos polinômios p_i e p_{i+1} definidos no tempo t_n .	67
3.4	Representação gráfica do método de Lax-Friedrichs em que os polinômios p_i 's são constantes dentro da células médias Ω_i 's.	68
3.5	Representação gráfica do método de Nessyahu-Tadmor cujos polinômios p_i 's são retas dentro das células médias Ω_i que passam pelo ponto (x_i, \bar{w}_i^n) .	71
3.6	Gráfico de β em função de α cujos valores estão relacionados com a condição CFL e a função <i>minmod</i> .	76
3.7	Representação gráfica do método Godunov de primeira ordem cujos polinômios p_i 's são constantes dentro da células médias Ω_i 's. Em cada intervalo $I_i = (x_i, x_{i+1})$ tem-se um problema de Riemann local.	79

3.8	Representação de métodos de volumes finitos do tipo Godunov de primeira ordem, cuja função w é constante por partes e está representada pelos p_i 's. As evoluções das discontinuidades estão representadas nas interfaces $x_{i+1/2}$'s e para se obter os fluxos $F_{i+1/2}$'s são necessários resolver problemas de Riemann. O ponto representado por \times representa onde temos a solução numérica $\bar{w}_i^{n+1} = U_i^{n+1}$	83
3.9	Condições iniciais: à esquerda tem-se a densidade inicial e à direita a velocidade inicial.	87
3.10	Condições iniciais: à esquerda tem-se a pressão inicial e à direita a energia interna inicial.	87
3.11	Evolução da densidade: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov, com o solucionador de problemas de Riemann HLL.	88
3.12	Evolução da densidade: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	89
3.13	Evolução da densidade: à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	89
3.14	Evolução da velocidade: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.	90
3.15	Evolução da velocidade: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	90
3.16	Evolução da velocidade: à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	91
3.17	Evolução da pressão: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.	91

3.18	Evolução da pressão: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	92
3.19	Evolução da pressão: à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	92
3.20	Evolução da energia interna: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o resolucionador de problemas de Riemann HLL.	93
3.21	Evolução da energia interna: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	93
3.22	Evolução da energia interna: à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	94
3.23	Evolução da densidade: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo método de Godunov com o resolucionador de problemas de Riemann HLL.	95
3.24	Evolução da densidade: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	96
3.25	Evolução da densidade: à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	96
3.26	Evolução da velocidade: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o resolucionador de problemas de Riemann HLL.	97
3.27	Evolução da velocidade: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	97
3.28	Evolução da velocidade: à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	98

3.29	Evolução da pressão: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.	98
3.30	Evolução da pressão: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	99
3.31	Evolução da pressão: à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	99
3.32	Evolução da energia interna: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.	100
3.33	Evolução da energia interna: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	100
3.34	Evolução da energia interna: à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	101
3.35	Condições iniciais: à esquerda tem-se a densidade inicial e à direita a velocidade inicial.	102
3.36	Condições iniciais: à esquerda tem-se a pressão inicial e à direita a energia interna inicial.	102
3.37	Evolução da densidade: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.	105
3.38	Evolução da densidade: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	105
3.39	Evolução da densidade: à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	106

3.40	Evolução da velocidade: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.	106
3.41	Evolução da velocidade: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	107
3.42	Evolução da velocidade: à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	107
3.43	Evolução da pressão: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.	108
3.44	Evolução da pressão: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	108
3.45	Evolução da pressão: à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	109
3.46	Evolução da energia interna: à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.	109
3.47	Evolução da energia interna: à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.	110
3.48	Evolução da energia interna: à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.	110
3.49	Evolução da densidade: à esquerda tem-se a solução da densidade e à direita da velocidade, ambas obtida pelo método LWR com RK3-TVD. Consideramos a CFL de 0,01, que apresentou uma oscilação inicial maior do que a oscilação para $CFL = 0,02$	111

3.50	Evolução da densidade: À esquerda tem-se a solução da pressão e à direita da energia interna, ambas obtida pelo método LWR com RK3-TVD. Consideramos a CFL de 0,01, que apresentou uma oscilação inicial maior do que a oscilação para $CFL = 0,02$	111
5.1	Fluxograma do programa principal.	141
6.1	Densidade: à esquerda tem-se a evolução da densidade no tempo final de 11,8 e à direita no tempo final em 19,6.	145
6.2	Densidade: à esquerda tem-se a evolução da densidade no tempo final de 39,3 e à direita no tempo final em 58,9.	145
6.3	Intensidade do Campo Magnético: à esquerda tem-se a evolução da intensidade do campo magnético no tempo final de 11,8 e à direita no tempo final em 19,6.	146
6.4	Intensidade do Campo Magnético: à esquerda tem-se a evolução da intensidade do campo magnético no tempo final de 39,3 e à direita no tempo final em 58,9.	146
6.5	À esquerda encontra-se o gráfico da condição inicial da densidade e à direita da intensidade da velocidade.	148
6.6	À esquerda tem-se o gráfico da condição inicial da energia total e à direita da intensidade da pressão.	148
6.7	Condição inicial da intensidade do campo magnético.	148
6.8	Densidade: solução numérica no tempo de 11,8.	149
6.9	Densidade: solução numérica no tempo de 19,6.	149
6.10	Densidade: solução numérica no tempo de 39,3.	150
6.11	Densidade: solução numérica no tempo de 58,9.	150
6.12	Intensidade da Velocidade: solução numérica no tempo de 11,8.	151
6.13	Intensidade da Velocidade: solução numérica no tempo de 19,6.	151
6.14	Intensidade da Velocidade: solução numérica no tempo de 39,3.	152
6.15	Intensidade da Velocidade: solução numérica no tempo de 58,9.	152
6.16	Intensidade da Pressão: solução numérica no tempo de 11,8	153

6.17	Intensidade da Pressão: solução numérica no tempo de 19,6	153
6.18	Intensidade da Pressão: solução numérica no tempo de 39,2	154
6.19	Intensidade da Pressão: solução numérica no tempo de 58,9	154
6.20	Energia Total: solução numérica no tempo de 11,8.	155
6.21	Energia Total: solução numérica no tempo de 19,6.	155
6.22	Energia Total: solução numérica no tempo de 39,2.	156
6.23	Energia Total: solução numérica no tempo de 58,9.	156

Introdução

Nas últimas décadas, as aplicações de métodos numéricos e técnicas computacionais em simulações de sistemas astrofísicos vêm sendo cada vez mais intensificadas, graças aos avanços obtidos em pesquisas de análise numérica, computação científica, estudos analíticos, técnicas observacionais e experimentais. Contudo, novos cenários, fenômenos, propriedades e características são descobertos e detalhados colaborando para o entendimento de escoamento de fluidos em processos astrofísicos.

Cenários cujo objeto compacto central atrai a matéria ao redor e resultam em formação de jatos relativísticos ganham destaque por serem grandes emissores de energia, que influenciam muitos processos galácticos [18]. Tais fenômenos associados são modelados pelas equações de Magnetohidrodinâmica ¹ e Relatividade Geral (GRMHD) [87].

Diante da complexidade do sistema de equações diferenciais oriundas de GRMHD, a dificuldade de tratamento computacional imposto pelo caráter não linear das equações e por envolver fenômenos que atuam em diferentes ordens de grandezas, os processos de acreção e ejeção de plasma pelo objeto compacto central foram inicialmente tratados separadamente [9, 8, 23, 79].

Simulações que consideram o processo de acreção, a formação do jato e sua ejeção em uma mesma evolução temporal vêm sendo recentemente pesquisados por Koide [40], Nishikawa [69], McKinney [61], Shibata [38], entre outros. Entretanto, as implementações computacionais desses fenômenos por meio de métodos numéricos ainda representam desafios por requerer muita atenção em questões que envolvem ordem de precisão, discontinuidades na solução do sistema de equações, estabilidade dos métodos numéricos, tempo

¹Neste texto, optou-se pela grafia desta palavra como geralmente aparece nos textos de Física, mesmo sabendo que a grafia oficial é Magnetoidrodinâmica

de processamento e memória utilizada.

Um outro ponto de dificuldade é o momento de transição entre o disco de acreção e o início da formação de jatos colimados, pois nesta situação as grandezas envolvidas possuem variações bruscas características do fenômeno e um tratamento numericamente inadequado pode acrescentar oscilações espúricas ou dissipação numérica excessiva que prejudicam o andamento da simulação. Tais dificuldades podem ser percebidas através de dois fatos: desde o trabalho pioneiro de Koide, Shibata e Kudoh de 1999 [38] até os dias atuais, tais pesquisadores ainda utilizam o método Lax-Wendroff com propriedade de ser Valor Total Decrescente (TVD) do tempo, acréscimo de Viscosidade Artificial, com separação dimensional (*Splitting dimensional*) e limitadores de inclinações para se evitar oscilações; e o outro fato é que os atuais códigos em GRMHD disponíveis não possuem a modelagem de problemas de formação de jatos relativísticos desde a acreção de disco até a ejeção de matéria.

Com o levantamento realizado nesta tese sobre os principais códigos utilizados para as equações de GRMHD, percebeu-se uma ênfase nos métodos de Godunov dependentes de solucionadores de problemas de Riemann analíticos ou aproximados, com separação dimensional e limitadores de inclinações, mesmo sabendo que o sistema de equações de GRMHD possuem degenerescência de autovalores associados à matriz Jacobiana do sistema [24]. Tal fato limita a aplicabilidade de métodos desse tipo. Uma outra questão restritiva associada a esses métodos é a utilização de separação dimensional, que não são recomendados em problemas cujos autovalores assumem grandes diferenças nas ordens de magnitudes. Essas diferenças acontecem exatamente na transição que define a formação de jatos.

Neste contexto, o objetivo principal deste trabalho foi desenvolver um novo código de GRMHD aplicado na formação de jatos relativísticos, capaz de descrevê-lo desde a acreção de matéria em forma de disco até sua ejeção colimada, caracterizando um jato através da aplicação de métodos de Volumes Finitos Centrados sem separação dimensional, com o intuito de melhor representar a evolução temporal desse sistema astrofísico, com uma extensão maior do que a encontrada na literatura[40, 38].

Os objetivos específicos traçados a fim de se alcançar o objetivo principal foram:

- Investigar as propriedades e características de métodos de volumes finitos dependentes ou não de problemas de Riemann;
- Comparar as soluções de métodos de volumes finitos entre si e entre a solução de referência, em problemas que envolvem evolução temporal de descontinuidades;
- Modelar o problema de formação de jatos relativísticos através das equações de Magnetohidrodinâmica Relativística (GRMHD), tomando como base a formulação de Shibata [40];
- Programar um novo código em Fortran 90 para obter soluções numéricas do sistema de equações diferenciais que compõem a Magnetohidrodinâmica Relativística (GRMHD);
- Implementar os métodos de Volumes Finitos Centrados - Lax-Friedrichs e Nessyahu-Tadmor - sem separação dimensional no problema de formação de jatos relativísticos.

A seguir, apresenta-se a forma como a tese foi conduzida para se chegar aos objetivos listados.

No capítulo 1 foi feito um levantamento do desenvolvimento dos fenômenos associados à astrofísica moderna com o foco em jatos de discos relativísticos e apresentou-se o atual cenário de pesquisa na área.

As equações diferenciais parciais (EDPs), juntamente com as equações que compõem a modelagem do cenário astrofísico que possibilitam a formação de jatos relativísticos estão expostas no capítulo 2.

O capítulo 3 foi destinado às questões relacionadas aos métodos de Volumes Finitos aplicados em Leis de Balanço e Leis de Conservação. Os métodos de Lax-Friedrichs, Lax-Wendroff's, Godunov com resolvidor de problemas Riemann e o método de Nessyahu-Tadmor são detalhados e aplicados às equações de Euler unidimensionais com o intuito de explorar e comparar os métodos numéricos desenvolvidos. Neste momento, desenvolveu-se códigos em Octave para as equações de Euler unidimensionais, que descrevem o escoamento de fluidos ideais em situações que são encontradas em sistemas mais complexos como no caso das equações de GRMHD.

Um levantamento sobre os principais códigos utilizados em GRMHD foi feito no capítulo 4, juntamente com apontamentos que destacam as diferenças entre o código desenvolvido em Fortran 90 nesta tese e os demais códigos.

No capítulo 5 expôs-se as equações implementadas e as estruturas de Fortran 90 utilizadas na programação, tais como alocação dinâmica de memória, estruturas de blocos Modular, entre outros.

As simulações de formação de jatos relativísticos estão no capítulo 6. De início implementou-se o método Lax-Friedrichs bidimensional sem separação dimensional, com o intuito de testar e avaliar tendências na evolução das simulações. Depois, utilizou-se o método Lax-Wendroff com Runge-Kutta TVD de três estágios com *Splitting* dimensional e sem limitador de inclinações e verificou-se que a propriedade TVD não foi suficiente para estabilizar o método.

Partiu-se então para a implementação e aplicação do método Nessyahu-Tadmor bidimensional, sem separação de dimensões espaciais e método de Euler com quatro estágios para o acréscimo do termo fonte. Com este método, foi possível descrever a evolução temporal do sistema astrofísico inicialmente composto de disco de acreção em rotação em torno de um buraco negro (BH) central de Schwarzschild, imersos em uma magnetosfera em queda com relação ao BH. Mostramos que tal sistema passou pela transição de disco de acreção e ejeção de matéria em forma de jato com estabilidade numérica, cuja extensão atingiu 14 vezes o raio do BH. Além disso, o método Nessyahu-Tadmor foi capaz de revelar subestruturas formadas tanto no jato quanto do disco.

Por fim no capítulo 7 traz um retrospecto dos estudos realizados nos capítulos precedentes e as considerações finais realizadas. São apresentados ainda, sugestões de trabalhos futuros que possam contribuir para a continuidade ou a melhoria do que foi proposto.

Capítulo 1

Jatos Relativísticos - Aspectos do Fenômeno

1.1 Aspectos históricos

A Astronomia é uma ciência que surgiu das observações de corpos celestes durante centenas de anos. O acúmulo de diversas observações possibilitou o desenvolvimento de teorias que auxiliam na compreensão de fenômenos naturais. Em 1917, Haber Curtis notou uma estrutura que descreveu como *um curioso raio em linha reta conectado a um núcleo*. Tal objeto atualmente é conhecido como o **jato M87** e é uma das ejeções mais estudadas associada a um Núcleo Ativo de Galáxia - **AGN**. Ver Figura 1.1.

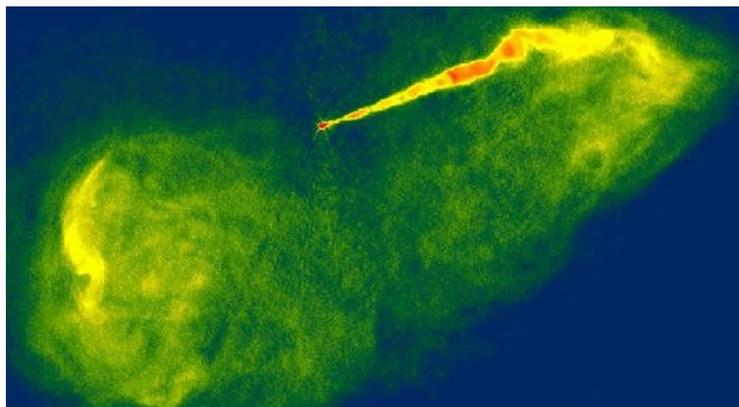


Figura 1.1: Imagem obtida pelo telescópio Hubble do jato M87 - fonte: NASA Hubble Space Telescope Images [99].

De acordo com uma concepção moderna, objetos massivos centrais estão presentes na maioria das Galáxias e permanecem ativos se, e somente se, uma quantidade suficiente de matéria cair neles. Além do objeto compacto central, Núcleos Ativos de Galáxias necessitam de fluxo de matéria em queda na direção do centro de massa do sistema. Este mecanismo é chamado de **acrecção de matéria** e está associado ao campo gravitacional gerado pelo objeto central.

No início da década de 1950, com o desenvolvimento da Radioastronomia iniciaram-se estudos sobre as emissões de radiação térmica e não térmica e, também sobre a luminosidade de objetos estelares. No entanto, observações nas faixas de raio-X e raio gamma possibilitaram inúmeras descobertas de fontes não térmicas do Universo, com destaque para as emissões difusas de rádio em torno de galáxias ativas.

Após essas descobertas, rapidamente tais emissões foram associadas a ejeções de plasmas colimados que escoam para fora do núcleo galáctico [8]. A matéria expelida por esses jatos tem uma energia muito elevada e em muitos casos, a propagação da ejeção estende-se por grandes distâncias - em comparação ao raio do objeto central - com velocidades relativísticas até antes de atingir um perceptível freamento devido à interação com o meio intergaláctico [8, 45, 50].

No final da década de 1960, evidências da existência de objetos em que processos eletrodinâmicos são decisivos na liberação de energia impulsionaram os primeiros estudos de modelos eletromagnéticos para fontes compactas [30, 37].

Em 1976, Blandford e Znajek apresentam um mecanismo que estrutura o cenário básico de extração de energia de um Buraco Negro (BH) em rotação. Tal mecanismo é conhecido como processo de *Blandford-Znajek*, que expõem sobre a necessidade de um disco de acreção com um campo magnético suficientemente forte em torno do eixo de rotação do BH [10]. A ideia chave que está na base teórica de tal processo é o fato de considerar o objeto compacto - imerso em um campo magnético externo - como um indutor unipolar.

Inicialmente no desenvolvimento da teoria analítica, discos de acreção e jatos foram tratados separadamente. Tais teorias podem ser encontradas em [14] para disco de acreção e para jatos na referência [9].

Um tratamento levando em consideração as equações de Maxwell e as leis de conservação relativísticas de massa, momento e energia - chamando de *Magnetohidrodinâmica Relativística - GRMHD* - foi proposto por Thorne, Price e MacDonald (1986) [87].

Avanços em recursos computacionais, em técnicas de computação e desenvolvimento de Métodos Numéricos proporcionaram implementar simulações que abrangem cenários astrofísicos que envolvem a formação de jatos desde o processo de acreção do disco até a propagação do plasma ejetado.

Koide, Shibata e Kudoh (1999) [38] utilizaram as equações de GRMHD ideal, sem termo resistivo, para mostrar em simulações o processo de formação do jato a partir do disco de acreção em volta de um buraco negro central, imerso em uma magnetosfera em queda. Em uma de suas configurações implementadas, o jato foi formado na superfície do disco mostrando a possibilidade da existência tanto de jatos que se formam nos polos do BH quanto jatos ejetados por discos.

O desenvolvimento de teoria analítica e de simulações que envolvem o disco de acreção e condições iniciais que resultam em um jato inicialmente formado, buscam explorar as propriedades e características evidenciadas pela propagação da ejeção. Nesses estudos tanto as técnicas analíticas via soluções auto-similares das equações de Grad-Shafranov [12] quanto as simulações via GRMHD [61] mostram que a existência de um campo magnético forte é essencial para a colimação do jato.

Avanços na área observacional vêm adicionando e evidenciando características do disco de acreção, formação e ejeção de jatos. Atualmente, as modelagens matemáticas de jatos que envolvem a dinâmica de plasmas altamente energéticos, através de teorias analíticas e simulações de GRMHD, destacam-se por estarem em consonância com as observações [18, 58].

Erupções em forma de jatos representam uma das aparências visíveis da atividade de objetos astrofísicos. Certamente, este é um mecanismo universal de liberação de energia [8] que influencia em diversos processos astrofísicos [18].

Contudo, apesar dos cenários favoráveis referentes aos avanços nos estudos das teorias analíticas, das simulações computacionais, das observações e de experiências laboratoriais de plasmas, ainda o assunto apresenta grandes dificuldades e desafios por conter muitas

questões em aberto.

Após quase cem anos da descoberta de Heber Curtis, a principal característica de muitos jatos estáveis - a propagação da ejeção do plasma colimado por longas distâncias em comparação ao raio do objeto compacto central - permanece inexplicada. É fato que o campo magnético forte oriundo do disco de acreção tem suas linhas alteradas ao ponto de colimar o jato, de direcionar a propagação do plasma e preservá-lo durante a evolução temporal, porém pesquisas sobre estabilidades de jatos (ambos relativísticos e não relativísticos) possuem um grande número de modos de instabilidade [61].

Ao longo das últimas décadas, pesquisadores apresentaram critérios de instabilidade e técnicas analíticas com o intuito de estabilizar a propagação de jatos, porém até agora nenhuma análise de estabilidade foi suficiente para explicar a extensão de diversos jatos observáveis [8].

Um outro ponto intrigante são as subestruturas formadas durante a propagação de jatos. Estudos apontam que as formas possíveis de jatos e de suas subestruturas são influenciadas pelas propriedades do disco de acreção, do campo magnético e do meio em que o sistema está imerso (magnetosfera) [61]. Entretanto, os ajustes de valores que entrarão nas condições iniciais para as análises analíticas e para as simulações computacionais ainda são desafiadores por permanecer difícil a comparação da teoria com observações (Worral, 2009) [95] e com experimentos laboratoriais de plasmas (Ciardi, 2009) [15].

Diante deste cenário, simulações em GRMHD seguem-se em destaque no estudo de modelos de jatos mais realistas, porém deve-se ressaltar a importância da experiência acumulada com os estudos analíticos e observações para ajustar os parâmetros envolvidos nas condições iniciais [8].

A complexidade do sistema de equações diferenciais oriundas de GRMHD, a dificuldade de tratamento computacional imposto pelo caráter não linear das equações, que envolvem fenômenos que atuam em diferentes ordens de grandezas, fizeram com que os processos de acreção e de ejeção de plasma pelo objeto compacto central fossem inicialmente tratados separadamente, [9], [8], [23], [79], entre outros.

Simulações que consideram o processo de acreção, a formação do jato e sua ejeção em uma mesma evolução temporal vêm sendo recentemente pesquisadas por Koide [40],

Nishikawa [69], McKinney [61], Shibata [38], entre outros. Entretanto, a implementação computacional desses fenômenos por meio de métodos numéricos ainda representa desafios por requerer muita atenção em questões que envolvem ordem de precisão, descontinuidades na solução do sistema de equações, estabilidade dos métodos numéricos, tempo de processamento e memória utilizada.

Na seção seguinte apresentam-se os elementos fundamentais que resultam em formação de jatos astrofísicos, também expõem-se suas propriedades e características.

1.2 A formação de jatos

Nas últimas décadas, diversos sistemas astrofísicos vêm sendo descobertos com a propriedade de gerar escoamento colimado de plasma, com velocidades que podem aproximar da velocidade da luz.

Um **jato** é uma linha de corrente fortemente colimado de fluido, gás ou plasma. Tipicamente, possui energia cinética, energia interna, momento linear e, dependendo do sistema envolvido, também poderá possuir momento angular. Embora existam diferentes métodos propostos para produção de jatos colimados, as ideias principais estão embasadas em magnetohidrodinâmica (para jatos não relativísticos) e em magnetohidrodinâmica relativística (GRMHD) - para jatos relativísticos [8].

As equações de GRMHD são utilizadas para descrever a formação e propagação de jatos colimados em aceleração até atingir velocidades relativísticas, enquanto ocorre colapso gravitacional de um disco em rotação imerso em uma magnetosfera. A matéria em acreção escoar em direção a um objeto compacto central que, por sua vez, impõe uma curvatura para a estrutura espaço-tempo. Em seguida, descreve-se os elementos fundamentais que estão envolvidos na formação de jatos astrofísicos estudados.

Um objeto compacto central com rotação atrai matéria imersa em um meio de baixa densidade e com um campo magnético forte - chamado de magnetosfera. A componente poloidal do campo magnético enfileira a matéria imersa deixando-a em forma de disco - chamado de disco de acreção.

Com uma variedade de velocidades de rotação no disco de acreção, a matéria é escoada

em direção ao objeto compacto levando consigo as linhas de campo magnético. A matéria depositada na vizinhança do objeto compacto é espremida pelas linhas de campo e pela matéria que continua em colapso. As diferentes velocidades de rotação do disco produzem um campo magnético helicoidal sobre o eixo de rotação que, além de apertar a matéria na vizinhança do objeto compacto, direcionará a matéria conforme as linhas do campo magnético.

Dependendo da importância relativa do campo magnético, da densidade do disco de acreção em comparação com a densidade do meio, do ângulo de abertura do disco e da velocidade de rotação do disco, uma variedade de resultados são possíveis, tais como: um amplo vento não colimado; dois escoamentos não relativísticos saindo em cada um dos polos do objeto compacto; um jato relativístico altamente colimado ou até mesmo um jato formado no próprio disco de acreção. Ver Figura 1.2.

Sendo assim, o campo magnético gerado pelo disco de acreção é a ligação natural entre o objeto compacto central e o jato, interligando também a rotação do disco com a rotação do objeto central. Como o escoamento de saída do plasma é direcionado pelas linhas do campo magnético, o fluxo de liberação de energia pelo jato também ocorre ao longo das linhas de campo. A propagação do jato define uma corrente elétrica longitudinal gerando assim um campo magnético induzido toroidal capaz de exercer uma pressão magnética que contribui para a colimação do jato.

No entanto, deve-se notar que neste sistema astrofísico, a corrente total que flui a partir do objeto central deveria desaparecer, pois a força de Ampère descolimaria o jato uma vez que as correntes anti-paralelas repelem-se. Portanto, um meio externo se faz necessário para colimar e ajudar a manter a colimação do jato formado.

Além disso, uma questão intrigante e não trivial é “ se é possível considerar um BH imerso em um campo magnético externo como um indutor unipolar ”. Blandford e Znajek [10] desenvolveram as bases teóricas para o entendimento de tal questão e atualmente pode-se considerar que o disco magnetizado em rotação serve como uma bateria que determina a energia liberada a partir do objeto central de alta condutividade [8].

Embora tenhamos o cenário favorável descrito acima para a formação, colimação e ejeção do jato, deve-se destacar que sua manutenção por longas distâncias não é nada

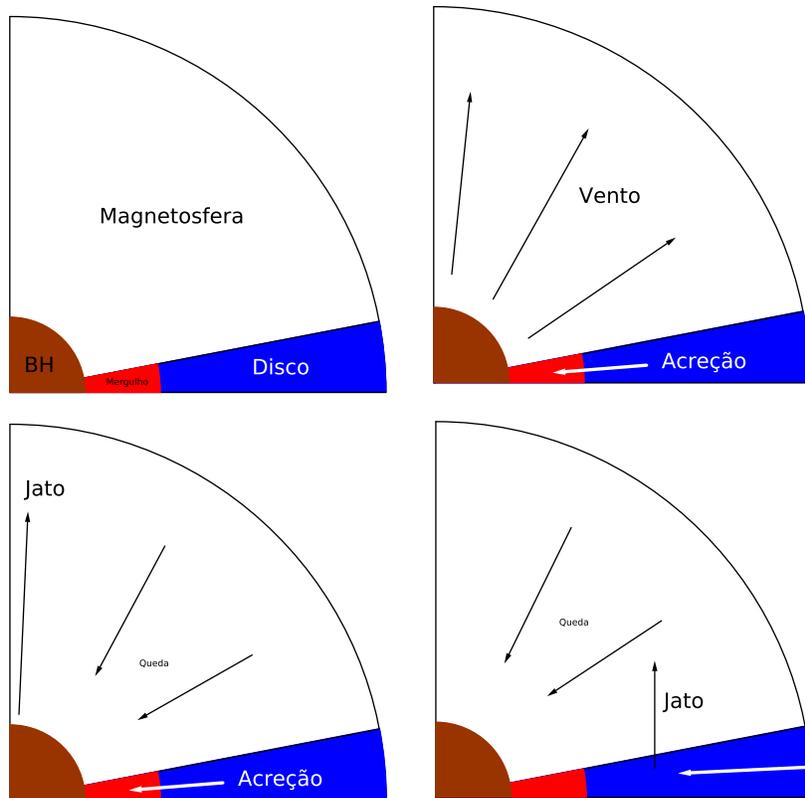


Figura 1.2: Na primeira linha à esquerda encontram-se os elementos fundamentais envolvidos no processo de formação de jatos astrofísicos e à direita a esquematização da formação de um vento não colimado. Na segunda linha à esquerda têm-se a formação de jato em um dos polos do objeto compacto central e à direita o jato formado a partir do disco de acreção.

trivial pois nas últimas décadas pesquisas apontam para um grande número de modos de instabilidade que podem resultar em uma total ruptura da colimação.

Exemplos de modos de instabilidade em jatos incluem: modo de reflexão ressonante, instabilidade de Kelvin-Helmholtz, modo de corrente conduzida e modo de torção [8]. Diante disso, Kruskal e Shafranov observam uma relação conhecida como *critério de instabilidade de Kruskal-Shafranov - KS*.

Critério de instabilidade - KS: *Jatos magnetizados formados por sistemas que envolvem disco de acreção magnetizado, objeto compacto central em rotação e magnetosfera, serão instáveis se,*

$$\left| \frac{B_\phi}{B_\theta} \right| > \frac{2\pi R}{r} ,$$

em que B_ϕ e B_θ são respectivamente as componentes toroidal e poloidal do campo magnético, $R = r \sin\theta$ é o raio do toro e r é a extensão poloidal (direção radial).

O critério KS implica que jatos serão instáveis antes de atingir $\gamma \gtrsim 3$ em regiões depois de $r \cong 100.M$, em que r é aproximadamente cinquenta vezes maior que o tamanho do raio do objeto central no sistema ($G = c = 1$) [61]. Alguns jatos observáveis satisfazem tal critério, porém um contra-exemplo é o jato M87.

Instabilidades não foram observadas apenas em jatos, estudos sobre acreção de discos também apontam para instabilidades no processo de acreção de matéria magnetizada via simulações de MHD. Turbulências geradas por instabilidades magnetorotacional (MRI) são capazes de desestruturar o disco e espalhar matéria não colimada para fora do disco, como mostrado em simulações [26]. Por outro lado, acredita-se que a instabilidade magnetorotacional seja o principal mecanismo que direciona a acreção de discos astrofísicos [5].

Jatos relativísticos são obtidos em sistemas com as seguintes características: o disco de acreção possui velocidades relativísticas e o objeto compacto central é um buraco negro ou uma estrela de nêutrons. Estes são encontrados em núcleos ativos de galáxias.

1.2.1 Jatos em AGN

As principais propriedades do objeto compacto central em um AGN, que são aceitas pela maioria dos astrofísicos atuais, podem ser resumidas conforme expõem-se a seguir.

No centro da galáxia existe um buraco negro supermassivo no qual ao seu redor ocorre acreção de matéria [79, 23]. Somente com este cenário é possível explicar a eficiência muito elevada da liberação de energia e a compacidade do objeto central, o que fornece para o universo uma grande fonte de energia. A atividade do núcleo galáctico é relacionado tanto com a energia de rotação do BH via

$$E_{rot} = \frac{J_r \Omega^2}{2} \cong 10^{62} \left(\frac{M}{10^9 M_\odot} \right) \left(\frac{\Omega_H r_S}{c} \right)^2 \text{ erg} , \quad (1.1)$$

quanto a energia de matéria acrecida. Na equação (1.1),

$$r_S = \frac{2GM}{c^2}$$

é o raio de Schwarzschild do BH, J_r é o momento de inércia, Ω_H é a velocidade angular do BH, M é a massa do BH, c a velocidade da luz, G a constante universal da gravitação, $M_\odot \cong 2.10^{33}g$ a massa do Sol e para que o buraco negro seja classificado como supermassivo sua massa é aproximadamente de 10^6 a 10^9 vezes a massa do Sol.

A existência de objetos supermassivos é justificada pelo fato de que a luminosidade de Eddington é dada por,

$$L_{Edd} \cong 10^{47} \left(\frac{M}{10^9 M_\odot} \right) \text{ erg/s} , \quad (1.2)$$

isto é, a força gravitacional que age na matéria acrecida é equilibrada pela força de pressão da radiação determinada pela luminosidade [97]. Os AGN's possuem uma luminosidade característica próxima de L_{Edd} e a duração da fase ativa é estimada usando-se as equações (1.1) e (1.2), isto é,

$$\tau = \frac{E_{rot}}{L_{Edd}} ,$$

que é da ordem de 10^7 anos.

A acreção de matéria pelo BH possui uma forma de disco e está de acordo com observações expostas por Lynden-Bell [57]. Assim, a direção preferencial - o eixo de rotação - surge naturalmente no espaço no qual a formação de jatos é possível. Como um buraco negro não possui campo magnético próprio - *Teorema sem cabelos* - *No Hair Theorem*, a geração de campos magnéticos fortes se devem ao disco de acreção [23].

Os núcleos ativos de galáxias produzem os maiores (em extensão) e os mais energéticos jatos encontrados no Universo, embora eles não forneçam os mais rápidos e mais potentes (potência instantânea). As velocidades medidas em jatos de galáxias de rádio assumem fatores de Lorentz que variam de $\gamma \cong 1,005$ até $\gamma \cong 50$ [55], por exemplo, na galáxia M87, seu jato formado nas proximidades do núcleo possui na maior parte do escoamento velocidades de $\gamma \cong 6$ [36] e estende-se por longas distâncias, quando comparado ao raio do objeto compacto central [62].

Em muitos casos, a matéria expelida pelo jato continua movendo-se com velocidades relativísticas por grandes distâncias, em comparação ao raio do BH, até atingir um perceptível travamento devido a interação com o meio intergaláctico. No caso da galáxia Cygnus A, as extensões dos dois jatos superam 10 ordens de magnitude com relação ao tamanho do BH central, conforme Figura 1.3.

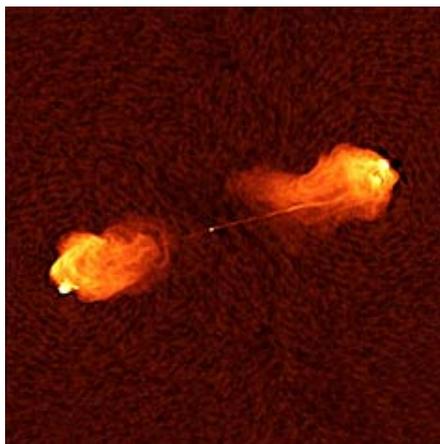


Figura 1.3: Imagens do telescópio Hubble da galáxia Cygnus A [99].

No caso da galáxia M87, a parte mais luminosa concentra-se ao redor do núcleo e, em partes, ao longo do jato, enquanto que a parte menos luminosa aparece como uma galáxia aproximadamente elíptica. Na galáxia Cygnus A, o BH central e as subestruturas

formadas nas pontas dos jatos possuem maior luminosidade. Apesar dos jatos das duas galáxias serem colimados, as espessuras angulares dos jatos da Cygnus A são bem mais finos que a espessura do jato da M87. Uma outra diferença entre os jatos dessas galáxias é que a galáxia M87 possui um jato enquanto que a Cygnus A possui dois jatos. Ver Figuras 1.1 e 1.3.

1.2.2 Jatos em Microquasares

Embora os quasares supermassivos residentes no centro de galáxias sejam alimentados por estrelas, gases ou plasmas coletados de sua vizinhança, existem versões menos massivas desses sistemas que podem ser alimentados por uma única faixa de atmosfera de uma estrela companheira. Uma parcela desses sistemas binários chamados de microquasares produzem jatos, conforme Figuras 1.4 e 1.5.

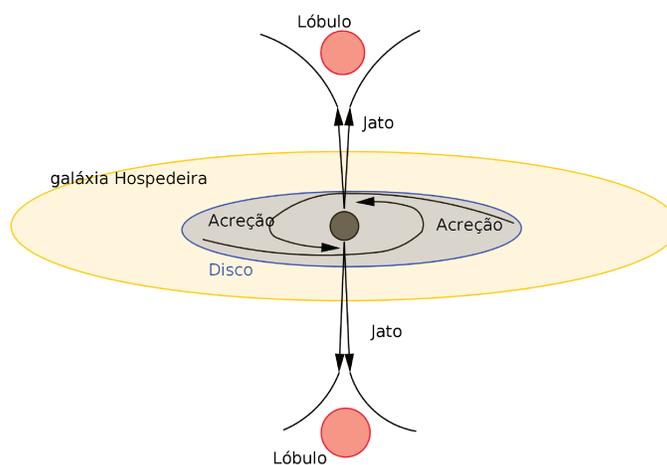


Figura 1.4: Representação gráfica da formação de jatos em um quasar cuja matéria acrecida advém da galáxia que hospeda tal sistema. Figura fora de escala.

Existem dois tipos de microquasares diferenciados pelos objetos centrais: em um deles o objeto compacto é um BH cuja massa é aproximadamente $10M_{\odot}$ e o outro é caracterizado por residir em uma estrela de nêutrons de massa aproximadamente igual a $1,4M_{\odot}$. O segundo caso produz um jato moderadamente relativístico pois em média as velocidades assumem valores próximos de $0,26c$ ($\gamma \cong 1,036$), enquanto que no primeiro caso tem-se $0,92c$ ($\gamma \cong 2,5$).

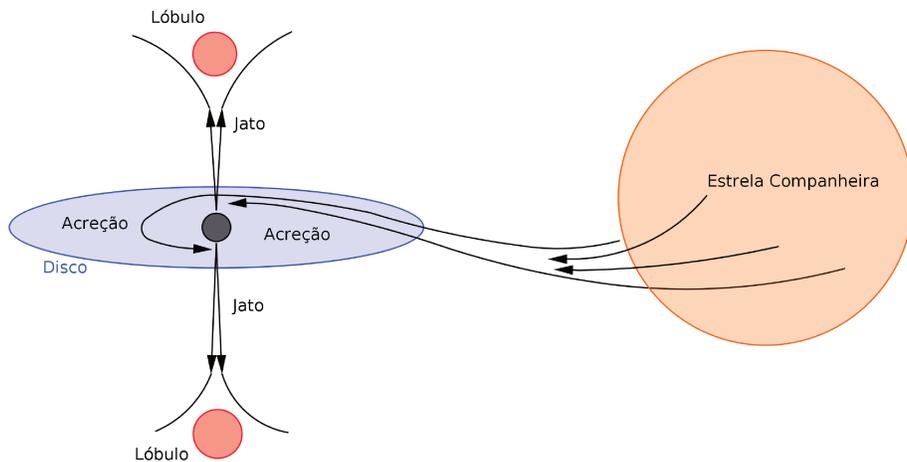


Figura 1.5: Representação gráfica da formação de jatos em um microquasar cuja matéria acrescida advém de uma estrela companheira. Figura fora de escala.

Um exemplo de jatos em microquasares são os jatos gêmeos da *fonte SS 433* que possuem uma velocidade média de $0,26c$ ($\gamma \cong 1,036$). Até agora não foi observado um microquasar cuja velocidade se aproxima do fator de Lorentz de $\gamma \cong 10$. Uma possível causa para isto se deve ao fato de que os microquasares evoluem por várias fases de sua vida em poucos anos ou décadas [62].

Nesses sistemas a taxa de fluxo de matéria emergida é maior do que a quantidade de matéria que o objeto compacto possa engolir. Uma consequência que pode surgir é a formação de jatos devido ao aumento do momento angular da matéria em acreção e o aumento de pressão pelo acúmulo de matéria.

Observações de microquasares mostram que jatos estão relacionados com a espessura do disco de acreção, sendo que não são observados jatos em sistemas com discos relativamente finos. A razão pela qual isso ocorre ainda não é clara: talvez um disco fino seja insuficiente para colimar o fluxo emergente ou o campo magnético gerado pelo disco fino não é suficientemente forte [8].

Historicamente, o primeiro objeto revelado desta classe foi o *SS 433* possuindo uma taxa de liberação de energia total de 10^{38} erg/s e o gás ejetado com velocidades moderadas. Tais velocidades podem ser explicadas pela pressão de radiação altamente aquecida pelas regiões mais internas do disco. Com relação a jatos relativísticos, a primeira fonte foi

observada em 1994 [64].

Um fato importante de se destacar em jatos formados por microquasares é que, na maioria dos casos, bolhas são formadas a grandes distâncias do objeto central. Acredita-se que tal subestrutura chamada de *Lóbulo* seja formada pelos diversos ciclos de rotação do objeto compacto central. Ver Figura 1.5.

Estudos observacionais de microquasares são de fundamental importância. Sabendo que a escala de tempo para eventos que ocorrem perto da acreção de objetos compactos é diretamente proporcional a massa do objeto, os microquasares possibilitam evoluções de fenômenos na escala de anos ou décadas ao passo que em quasares os fenômenos estão associados a escala de milhares de anos [62].

1.2.3 Jatos em Supernovas

Existem evidências de que certos tipos de supernovas, que não estejam envolvidas por um gigante envelope vermelho, explodem de forma esférica.

Estudos teóricos e observacionais apontam para a possibilidade de que tal assimetria seria causada pela ejeção de jatos gêmeos que parte do núcleo central de uma supernova, enquanto colapsa para formar uma estrela de nêutrons.

A acreção de gás na estrela de nêutrons central pode criar um disco que impulsiona jatos altamente colimados, impulsionando a matéria para fora da estrela em velocidades relativísticas entre $0,25c$ e $0,6c$. Mesmo que apenas uma pequena fração de seu núcleo seja ejetado - aproximadamente 1%, a energia cinética assume valores suficientes para alterar substancialmente a estrutura de expansão da supernova e, em alguns casos, pode desempenhar um papel crucial nas próximas fases da evolução de supernovas. De fato, como observado, a potência explosiva é da ordem de 10^{52} ergs em poucos segundos [39] e isto pode limitar a quantidade de matéria que será acrescida pelo objeto central.

Observações da colisão entre o jato colimado e a matéria que compõe a supernova mostrou que uma das possíveis subestruturas formadas é o objeto Herbig-Haro [32], como ilustrado na Figura 1.6.

Uma das grandes dificuldades atuais para os modelos matemáticos propostos, juntamente com as simulações computacionais oriundas de códigos em GRMHD, é conseguir

unir fenômenos de acreção e ejeção de jatos com a formação de subestruturas a longas distâncias do objeto central.

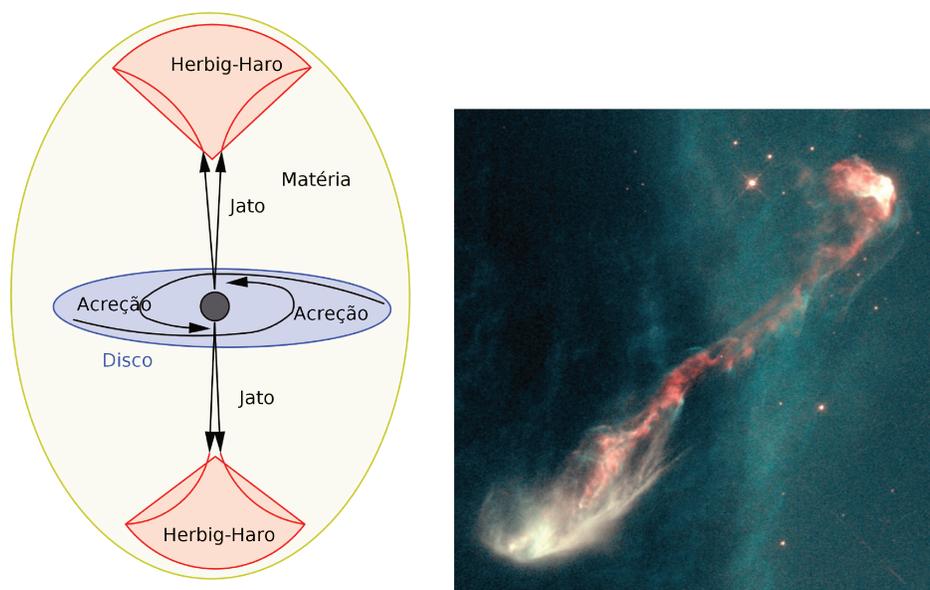


Figura 1.6: À esquerda tem-se a representação gráfica da formação de jatos em uma supernova cuja matéria ejetada forma objetos chamados de Herbig-Haro e à direita tem-se o objeto Herbig-Haro 47 captado pelo telescópio Hubble. [99]

1.2.4 Jatos em Explosões de Raios *gamma*

Explosões de raios *gamma* - GRB's - fornecem atualmente o exemplo mais extremo de escoamento relativístico de jatos. Esses jatos podem assumir velocidades associadas a fatores de Lorentz de aproximadamente $\gamma \cong 100$ ou até mesmo $\gamma \cong 300$, com uma liberação de energia na ordem de 10^{53} ergs por segundo [8].

Os altos valores de energia e de fator de Lorentz são consistentes com a formação de um BH estelar de massa aproximadamente igual a $10M_{\odot}$, com 1% de matéria saindo pelo jato extremamente relativístico. Este é um caso extremo e raro de supernovas que evoluem para o estágio de formação de BH ao invés de parar no estágio de estrela de nêutrons. Uma outra situação que pode desencadear um evento similar está associada a fusão de duas estrelas de nêutrons [62].

Nestes casos, o BH central possui uma rotação bem elevada e o processo de formação de jatos é bem semelhante ao caso de AGN's, com a diferença de que o campo magnético assume intensidades elevadíssimas, da ordem de 10^{15} Gauss. Somente com essa intensidade de campo magnético explica-se a quantidade de energia liberada pelo BH e acredita-se que tamanha intensidade é obtida após o colapso do núcleo da supernova ou após a fusão de estrelas de nêutrons [48].

1.3 Simulações

Nas últimas duas décadas, o desenvolvimento da computação, da tecnologia de hardwares e de técnicas computacionais possibilitou um grande avanço em simulações numéricas de cenários astrofísicos que ajudam no entendimento da dinâmica de fenômenos observáveis. No entanto, as pesquisas atuais ainda encontram cenários desafiadores que iniciam-se na modelagem matemática dos fenômenos físicos em questão, passam pela escolha adequada dos métodos numéricos, chegando na interpretação dos resultados obtidos e na comparação com observações e resultados encontrados na literatura.

Devido a uma grande quantidade de fenômenos que os sistemas astrofísicos envolvem, tais como: dinâmica de fluidos relativísticos magnetizados, descrita por equações diferenciais parciais acopladas dependentes do tempo - equações de mecânica de fluidos e equações de Maxwell relativísticas, campo gravitacional descrito pelas equações de Einstein, dinâmica para campos gravitacionais, processos adiabáticos - termodinâmica, viscosidade, resistividade, equação de estado - EOS, física de partículas, radiação, entre outros; os processos de acreção de matéria e os de ejeção de jatos foram tratados separadamente.

1.3.1 Disco de Acreção

Modelos teóricos de acreção basicamente descrevem propriedades estacionárias de disco de acreção com várias possibilidades de aberturas e propriedades físicas embutidas. Tais modelos analíticos investigaram vários processos físicos em discos de acreção destacando-se as grandezas físicas envolvidas para a descrição fenomenológica, entretanto os resultados

obtidos nesses estudos aplicados em representações da dinâmica ficam contraditórios [14].

Um exemplo contraditório é o caso em que estudos analíticos apontavam para a não existência de descontinuidades em modelos de disco enquanto que estudos numéricos em descrições dependentes do tempo mostram o surgimento de descontinuidades e, até mesmo, modos de instabilidades [5].

Apesar do avanço de técnicas observacionais indicarem que os modelos dinâmicos simulados via magnetohidrodinâmica relativística estão mais coerentes com os dados coletados, os modelos analíticos servem para testar as simulações, principalmente no que diz respeito a resultados assintóticos.

Dois modelos em destaque por possuírem resultados analíticos, numéricos e observacionais em consonância são os modelos de acreção inviscosa de Wilson e o modelo de acreção de discos viscosos de Chakrabarti [14].

No primeiro modelo, Wilson investigou numericamente o acréscimo de matéria inviscida em um buraco negro em rotação, utilizando as equações da hidrodinâmica dependentes do tempo [91]. Com um código em coordenadas cilíndricas, para estudar a formação de ondas de choque e de emissão de *raios-X* nas proximidades do BH, Wilson foi capaz de acompanhar a formação de espessuras de discos durante as simulações.

O segundo modelo mostrou que a formação de ondas de choque em discos de acreção depende fortemente da viscosidade e que duas regiões ficam bem distintas: uma anterior a onda de choque e a outra posterior. Após a dissipação da onda de choque, o disco torna-se do tipo Kepleriano como previsto por estudos analíticos.

A formulação de Wilson e suas variações foram extensivamente utilizadas em simulações que envolvem a acreção de discos espessos. O surgimento de modos de instabilidades fez com que os pesquisadores entedessem mais sobre o assunto, porém logo percebeu-se a necessidade de incorporar campos magnéticos e auto-gravitação [24, 25].

Em 1977, Wilson propôs um sistema numérico para GRMHD para estudar a evolução de plasma com simetria axial perto de um buraco negro de Kerr, porém apenas nos últimos 15 anos as simulações que envolvem GRMHD foram possíveis de serem implementadas, na tentativa de obter esclarecimentos sobre a dinâmica de discos magnetizados em regiões de campos gravitacionais intensos [5, 16, 26, 33, 65].

Neste contexto foram identificados Instabilidades Magnetorotacionais - MRI - em simulações que utilizam tanto a métrica de Schwarzschild quanto a de Kerr, independente do uso de um código com simetria axial ou de um tridimensional no espaço [5, 33].

As simulações realizadas com GRMHD ideal aplicado em discos espessos de acreção (inicialmente na forma de um Toro magnetizado) possibilitaram descrever o seguinte processo básico de acreção: primeiro, a componente toroidal do campo magnético cresce rapidamente impulsionada pelo cisalhamento; em seguida a instabilidade magnetorotacional desenvolve-se até atingir uma saturação e por fim a acreção entra em um regime quase estacionário em que o disco é mantido pela turbulência que redistribui a densidade e o momento angular do disco, no próprio disco, ver Figura 1.7. O crescimento exponencial da MRI acontece dentro do primeiro período orbital do Toro inicial, logo após o crescimento do campo magnético que foi suficiente para distorcer o Toro e iniciar o processo de acreção de matéria no BH [25].

As ejeções de matéria pelos polos do BH podem ser intensificadas pela rotação do BH, pela intensidade do campo magnético inicial, pelas linhas de campo magnético, pela quantidade de matéria do disco, pela espessura do disco e pela diferença entre as densidades do disco e da magnetosfera que permeiam o sistema. Sendo assim, com condições iniciais adequadas para a formação de jatos, a evolução temporal do disco descrito no sistema acima - representada pela Figura 1.7 - pode alimentar os jatos que propagam-se na região interior do cone e que são colimados pelo campo magnético forte.

1.3.2 Jatos

Estudos de simulações que impõe condições iniciais favoráveis para a propagação de jatos vêm sendo realizados na tentativa de se obter propriedades e características que contribuem para a colimação, extensão, velocidades e formação de subestruturas relacionadas com o jato.

Pudritz (2000) [72] realizou simulações de jatos alimentados por discos magnetizados de acreção, na direção de um objeto compacto e comparou a grande eficiência de liberação de energia desse sistema com dados observacionais.

O desenvolvimento das pesquisas caminharam para explorar as variações dos parâ-

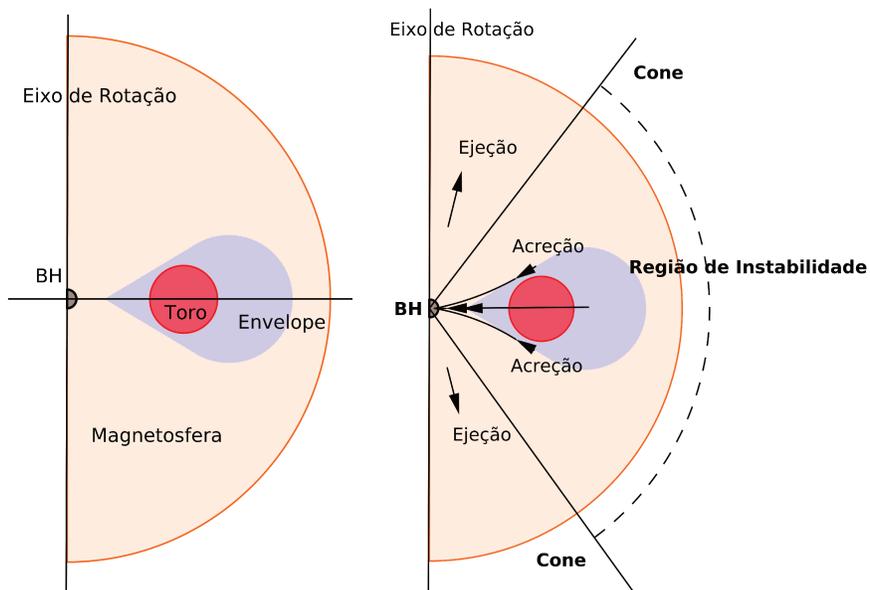


Figura 1.7: À esquerda tem-se uma representação do campo magnético achatando o envelope que envolve um toro e à direita tem-se a acreção desenvolvendo-se e delimitando duas regiões separadas por um cone: uma região de instabilidade que é evoluída no próprio disco e suas adjacências e uma outra região dentro do cone que expela uma baixa quantidade de matéria não colimada pelos polos do BH - tal ejeção é chamada de **vento**. Simulações desta natureza encontram-se em [14, 24].

metros envolvidos destacando-se pela diferenciação entre discos finos e discos espessos. Com a modelagem inicial que desenvolveu para jatos dominados pelo vetor de Poyting, alimentados inicialmente por discos Toroidais e colimados por campos magnéticos fortes - com componentes poloidal e toroidal não nulos, McKinney (2006) [60] conseguiu simular jatos relativísticos de longa extensão e, também, a formação de subestruturas chamadas de *Lóbulo*.

Uma análise mais detalhada de subestruturas de jatos foi realizada por Mizuta *et al* (2004) [59], porém as equações utilizadas não envolvem o campo magnético. Na mesma linha, estruturas internas de jatos relativísticos em regiões de intensidade gravitacional e sem campo magnético foram simuladas em [85].

As estabilidades de jatos alimentados por discos finos em rotações relativísticas estão sendo recentemente investigadas em simulações de GRMHD por Komissarov *et al* (2009) [44] e McKinney e Blandford (2009) [61] que são aplicadas tanto em AGN's quanto em explosões de *raios gamma*. Tais estudos simularam a propagação de jatos gêmeos e também de jatos unipolares. Entretanto, como alerta Worrall, a fenomenologia associada a jatos relativísticos está longe de ser totalmente compreendida e atualmente a pesquisa na área necessita cada vez mais do desenvolvimento de teorias analíticas, juntamente com desenvolvimento de simulações mais realistas e globais (que envolvam mais fenômenos associados ao problema), dados observacionais e experimentos de plasmas em laboratórios.

1.3.3 Jatos e Discos

Na linha de se obter simulações mais realistas, Koide, Shibata e Kudoh (1999) [38] foram os pioneiros em simulações envolvem a evolução de sistemas astrofísicos desde o início da acreção de discos geometricamente finos magnetizados até a ejeção de plasma colimado através de jatos.

A formulação de Koide, Shibata e Kudoh descrita em [38] possibilitou iniciar a investigação do processo de formação de jatos partindo-se inicialmente de um disco fino magnetizado em acreção em torno de um buraco negro em rotação, ambos envolvidos por uma magnetosfera. Tal formulação destaca-se pelo fato de que o jato não é imposto pelas condições iniciais do sistema astrofísico.

Mesmo com um sistema mais completo que os anteriores, as simulações deste tipo ainda foram pouco exploradas. As simulações das equações de GRMHD ideal - sem termo resistivo, sem condução de calor e com condutividade ideal - com termo fonte que acrescenta a gravitação pré-determinada pelo buraco negro, são descritas semelhantemente às Leis de Balanço [38], porém o sistema de equações diferenciais parciais dependente do tempo não necessariamente forma um sistema estritamente hiperbólico. Isso dificulta a aplicabilidade de Métodos Numéricos que dependem do conjunto de autovalores para saber as velocidades de propagação de ondas, isto é, Métodos do tipo Godunov que dependem de *Resolvedores de Problemas de Riemann* - chamados de *Riemann Solvers* [24].

Desse modo recomenda-se o uso de esquemas que não dependam de Resolvedores de Problemas de Riemann, isto é, métodos numéricos livres de *Riemann Solvers*, sendo assim, neste trabalho implementamos e desenvolvemos um novo código para GRMHD aplicável em simulações de jatos - desde o processo de acreção até a ejeção - utilizando métodos livres de *Riemann Solvers*.

Diferentemente do método Lax-Wendroff com propriedade de ser Valor Total Decrescente (TVD), com Viscosidade Artificial e *Splitting* dimensional utilizados pelo Koide, Shibata e Kudoh (1999) [38], o código desenvolvido nesta tese aplica os métodos Lax-Friedrichs e Nessyahu-Tadmor, ambos sem separação dimensional.

A modelagem de sistemas mais complexos segue em discussão, com o intuito de resolver questões que envolvem o acréscimo de um termo resistivo nas equações de GRMHD reescritas pela formulação de Shibata [40].

Capítulo 2

Modelagem de Jatos Relativísticos

Neste capítulo estão expostas as equações que modelam fenômenos relacionados com escoamento de fluidos eletricamente carregados (plasmas), em regiões magnetizadas na presença de campos gravitacionais.

Na modelagem, desconsiderou-se a viscosidade do fluido e que o fluido é um condutor perfeito. Estas aproximações são exatamente as hipóteses das equações de Magneto-hidrodinâmica Relativística ideal com gravitação (GRMHD ideal) e são estas equações consideradas nesta tese para descrever a formação de jatos relativísticos desde a acreção de matéria (plasma) em forma de disco até sua ejeção.

2.1 As equações de GRMHD ideal

As equações que fornecem base para a modelagem matemática de jatos relativísticos advêm da Magnetohidrodinâmica Relativística (GRMHD), que consiste de leis de balanço¹ relativística de massa, momento e energia, acopladas com as equações de Maxwell [87],

¹Leis de Balanço são Leis de conservação hiperbólicas com termos fonte *stiff*.

$$\nabla_{\mu}(\rho U^{\nu}) = 0 , \quad (2.1)$$

$$\nabla_{\mu} T_g^{\mu\nu} = 0 , \quad (2.2)$$

$$\partial_{\mu} F_{\nu\lambda} + \partial_{\nu} F_{\lambda\mu} + \partial_{\lambda} F_{\mu\nu} = 0 , \quad (2.3)$$

$$\nabla_{\mu} F^{\mu\nu} = 0 , \quad (2.4)$$

em que U^{ν} estão associados às componentes do quadrivetor velocidade. Os escalares ρ , p e ϵ são a densidade própria, pressão própria e a densidade de energia total própria

$$\epsilon = \rho c^2 + \frac{p}{\Gamma - 1} ,$$

respectivamente. A constante Γ está relacionada ao calor específico, ∇_{ν} é a derivada covariante. O tensor momento-energia é definido por

$$T_g^{\mu\nu} = p g^{\nu\mu} + (\epsilon + p) U^{\nu} U^{\mu} + F_{\sigma}^{\mu} F^{\nu\sigma} - \frac{1}{4} g^{\mu\nu} F^{\lambda\kappa} F_{\lambda\kappa} ,$$

em que $g^{\mu\nu}$ e $F^{\nu\mu}$ são as componentes da métrica do espaço-tempo e do tensor força do campo magnético, respectivamente. No caso da métrica satisfazer $g_{\mu\nu} = 0$ para $\mu \neq \nu$, tem-se $h_0 = \sqrt{-g_{00}}$, $h_1 = \sqrt{g_{11}}$, $h_2 = \sqrt{g_{22}}$ e $h_3 = \sqrt{g_{33}}$.

As equações da modelagem são reescritas para possuir uma formulação semelhante a uma lei de conservação, porém com um termo fonte adicional conforme a seguinte expressão:

$$\mathbf{U} = (D \quad P_1 \quad P_2 \quad P_3 \quad \epsilon \quad B_1 \quad B_2 \quad B_3)^T ;$$

em que D representa a densidade; P_1 , P_2 e P_3 são as componentes do momento; ϵ energia interna; B_1 , B_2 e B_3 são componentes do campo magnético; x^1 , x^2 e x^3 são coordenadas espaciais e $h_0 f$ o termo fonte devido ao campo gravitacional. O interesse em formular tais equações no formato de leis de conservação advém do uso de métodos de volumes finitos para a obtenção de soluções numéricas. Tal formulação foi inicialmente desenvolvida por Koide, Shibata e Kudoh [38] na forma conservativa e outras formulações encontram-se em

[24].

Observa-se que as componentes dos vetores \mathbf{v} de velocidade, \mathbf{B} de campo magnético e \mathbf{E} de campo elétrico, em coordenadas fiduciais são, respectivamente,

$$v_i = \frac{c}{\gamma} h_i U^i, \quad (2.5)$$

$$B_i = \epsilon_{ijk} \frac{h_i}{J} F^{jk}, \quad (2.6)$$

$$E_i = \frac{1}{h_0 h_i} F^{0i}, \quad (2.7)$$

em que os índices $i, j, k = 1, 2, 3$, γ é o fator de Lorentz e $J = h_1 h_2 h_3$ é o Jacobiano.

As quantidades conservativas em coordenadas fiduciais são dadas por,

$$D = \gamma \rho, \quad (2.8)$$

$$\epsilon = (e + p)\gamma^2 - p - Dc^2 + \frac{1}{2} \left(B^2 + \frac{E^2}{c^2} \right), \quad (2.9)$$

$$\mathbf{P} = \frac{1}{c^2} [(e + p)\gamma^2 \mathbf{v} + \mathbf{E} \times \mathbf{B}], \quad (2.10)$$

e o tensor momento-energia,

$$T = p\mathbf{I} + \frac{1}{c^2} (e + p)\gamma^2 \mathbf{v}\mathbf{v} - \mathbf{B}\mathbf{B} - \frac{1}{c^2} \mathbf{E}\mathbf{E} + \frac{1}{2} \left(B^2 + \frac{E^2}{c^2} \right) \mathbf{I}, \quad (2.11)$$

com as componentes sendo,

$$T^{ij} = h_i h_j T_g^{ij}, \quad i, j = 1, 2, 3.$$

Os vetores campo magnético e campo elétrico são normalizados pela permeabilidade magnética μ , isto é,

$$\mathbf{B} = \frac{\mathbf{B}_*}{\sqrt{\mu}} \quad \text{e} \quad \mathbf{E} = \frac{\mathbf{E}_*}{\sqrt{\mu}},$$

em que as quantidades com subíndice asterisco estão em unidades padrão do sistema internacional (SI). Assim, as equações (2.1)-(2.4) são reescritas na seguinte forma

$$\frac{\partial \mathbf{u}}{\partial t} = -q \circ \left[\frac{\partial}{\partial x^1} (h_0 b_1 \circ w_1) + \frac{\partial}{\partial x^2} (h_0 b_2 \circ w_2) + \frac{\partial}{\partial x^3} (h_0 b_3 \circ w_3) \right] + h_0 f, \quad (2.12)$$

em que a operação \circ é definida por

$$a \circ b = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} \circ \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} = \begin{pmatrix} a_1 b_1 \\ \vdots \\ a_n b_n \end{pmatrix}.$$

As variáveis apresentadas pela equação (2.12) são definidas pelas seguintes expressões matriciais

$$\mathbf{u} = \begin{pmatrix} D \\ P_1 \\ P_2 \\ P_3 \\ \epsilon \\ B_1 \\ B_2 \\ B_3 \end{pmatrix}, \quad q = \begin{pmatrix} 1/J \\ 1/J \\ 1/J \\ 1/J \\ 1/J \\ h_1/J \\ h_2/J \\ h_3/J \end{pmatrix}, \quad (b_1 | b_2 | b_3) = \begin{pmatrix} h_2 h_3 & h_3 h_1 & h_1 h_2 \\ h_2 h_3 & h_3 h_1 & h_1 h_2 \\ h_2 h_3 & h_3 h_1 & h_1 h_2 \\ h_2 h_3 & h_3 h_1 & h_1 h_2 \\ h_2 h_3 & h_3 h_1 & h_1 h_2 \\ 0 & h_3 & h_2 \\ h_3 & 0 & h_1 \\ h_2 & h_1 & 0 \end{pmatrix} \text{ e}$$

$$(w_1 | w_2 | w_3) = \begin{pmatrix} Dv_1 & Dv_2 & Dv_3 \\ T^{11} & T^{12} & T^{13} \\ T^{21} & T^{22} & T^{23} \\ T^{31} & T^{32} & T^{33} \\ c^2(P_1 - Dv_1) & c^2(P_2 - Dv_2) & c^2(P_3 - Dv_3) \\ 0 & E_3 & -E_2 \\ -E_3 & 0 & E_1 \\ E_2 & -E_1 & 0 \end{pmatrix}.$$

O termo fonte da equação (2.12) é dado por

$$f = \begin{pmatrix} 0 \\ (\epsilon + Dc^2)G_{01} + G_{12}T^{21} + G_{13}T^{31} - G_{21}T^{22} - G_{31}T^{33} \\ (\epsilon + Dc^2)G_{02} + G_{23}T^{32} + G_{21}T^{12} - G_{32}T^{33} - G_{12}T^{11} \\ (\epsilon + Dc^2)G_{03} + G_{31}T^{13} + G_{32}T^{23} - G_{13}T^{11} - G_{23}T^{22} \\ c^2(G_{01}P_1 + G_{02}P_2 + G_{03}P_3) \\ 0 \\ 0 \\ 0 \end{pmatrix} .$$

Nas equações de GRMHD, a gravitação é adicionada através do termo fonte e as componentes da métrica - h_μ , $\mu = 0, 1, 2, 3$ - estão representadas pelas componentes $G_{\mu\nu}$ definidas por

$$G_{\mu\nu} = -\frac{1}{h_\mu h_\nu} \left(\frac{\partial h_\mu}{\partial x^\nu} \right), \quad (2.13)$$

onde $\mu, \nu = 0, 1, 2, 3$.

O fluido é considerado livre de forças eletromagnéticas e esta restrição é conhecida como condição *frozen-in* dada pela seguinte relação,

$$\mathbf{E} = -\mathbf{v} \times \mathbf{B} . \quad (2.14)$$

Nota-se que as derivadas espaciais e temporal estão separadas, o que define o formalismo (3 + 1) das equações de GRMHD. Como a intenção é utilizar tais equações na modelagem de formação de jatos, inicialmente o modelo deve partir de um disco de acreção com rotação e em queda com relação a um objeto compacto central, que no caso foi considerado um Buraco Negro de Schwarzschild representado pela *métrica* de Schwarzschild.

2.2 Métrica de Schwarzschild

O problema de jatos relativísticos modelado é o de escoamento de fluido no espaço-tempo de Schwarzschild cuja métrica é estática. Isso significa que o buraco negro central é

responsável por toda gravitação envolvida no problema. Logo, no modelo o fluido não possui auto-gravitação e então, a gravitação é totalmente pré-determinada pela seguinte métrica de Schwarzschild

$$ds^2 = -\alpha^2 dt^2 + \frac{1}{\alpha^2} dr^2 + r^2 d\theta^2 + r^2 \sin^2 \theta d\phi^2 \quad (2.15)$$

donde

$$\alpha = \sqrt{1 - \frac{r_s}{r}}, \quad \text{com } c = G = 1, \quad M = \frac{1}{2} r_s \text{ (massa BH)},$$

$$h_0 = \alpha, \quad h_1 = \frac{1}{\alpha}, \quad h_2 = r, \quad h_3 = r \sin \theta,$$

em que α é a função lapso e r_s é o raio de Schwarzschild usado como unidade de referência. As constantes c e G são respectivamente a velocidade da luz e a constante universal da gravitação.

As coordenadas (t, r, θ, ϕ) são respectivamente o tempo, a coordenada radial, a coordenada polar e a azimutal.

Dentro desta geometria tem-se inicialmente um disco de acreção em rotação que não está em queda na direção do buraco negro, isto é, sem velocidade radial (direção r) descrito a seguir.

2.3 Disco de Acreção

Para que se tenha formação de jatos relativísticos é fundamental que se tenha um disco de acreção cuja característica de sua espessura seja geometricamente fino. Discos espessos são capazes de gerar modos de instabilidade que podem eliminar qualquer possibilidade de ejeção de matéria em forma de jato (ver Figura 1.7 - seção 1.3.1).

A velocidade de rotação do disco está na direção azimutal e tal componente da velocidade coincide com a velocidade Kepleriana [9], isto é,

$$v_\phi = v_K = \frac{c}{\sqrt{2\left(\frac{r}{r_s} - 1\right)}}. \quad (2.16)$$

O fluido do *disco de acreção* está inicialmente localizado em

$$|\cot \theta| \leq \delta \quad \text{e} \quad r \geq r_D = d_{in} r_s \quad (2.17)$$

em que $d_{in} > 1$ é a posição da borda interna do disco cuja velocidade de rotação é

$$v_K = \frac{c}{\sqrt{2(d_{in} - 1)}}.$$

Na referência [38], um valor estudado para a abertura do disco é $\delta = 0,125$, isto é, se θ é o ângulo com relação ao equador - abertura do disco - pertencente ao intervalo $]0, \pi/2[$ radianos, então $\delta = 0,125$ implica que o disco ocupa um setor de um oitavo do setor correspondente ao intervalo $]0, \pi/2[$.

A matéria que envolve o sistema composto do disco de acreção e do buraco negro é menos densa do que o fluido que compõe o disco. Esta matéria é conhecida como **magnetosfera** e está em queda em direção ao buraco negro contendo apenas a componente radial da velocidade diferente de zero. Assim, as condições iniciais referentes a densidade e a velocidade da matéria ao redor do buraco negro são dadas por

Densidade:

$$\rho = \rho_{mag} + \rho_{disco} \quad (2.18)$$

onde ρ_{mag} é a densidade do fluido na região da magnetosfera e ρ_{disco} é a densidade do fluido na região do disco. Para a delimitação do disco tem-se

$$\rho_{disco} = \begin{cases} \kappa_\rho \rho_{mag}, & \text{se } r > r_D \text{ e } |\cot \theta| < \delta \\ 0, & \text{se } r \leq r_D \text{ ou } |\cot \theta| \geq \delta \end{cases}, \quad (2.19)$$

em que κ_ρ é a razão entre a densidade do disco e a densidade da magnetosfera.

Velocidade:

$$(v_r, v_\theta, v_\phi) = \begin{cases} (0, 0, v_K), & \text{se } r > r_D \text{ e } |\cot \theta| < \delta \\ (-v_{mag}, 0, 0), & \text{se } r \leq r_D \text{ ou } |\cot \theta| \geq \delta \end{cases}, \quad (2.20)$$

donde v_K é a velocidade Kepleriana de rotação do disco, v_{mag} é a velocidade de queda do fluido localizado na magnetosfera, r_D é a borda interna do disco e δ está relacionado com o ângulo de abertura do disco.

Inicialmente o disco possui campo magnético gerado pelo disco e para isto utiliza-se a solução de Wald para adicionar tal propriedade física [89], isto é, as componentes do campo magnético são dadas por

Campo Magnético:

$$\begin{cases} B_r = B_0 \cos \theta \\ B_\theta = -\alpha B_0 \sin \theta \\ B_\phi = 0 \end{cases}, \quad (2.21)$$

em que $B_0 = \kappa_B \sqrt{\rho_D c^2}$ e ρ_D é a densidade na borda interna do disco.

Uma maneira de determinar v_{mag} é através da seguinte equação [38],

$$\alpha = \frac{H(\gamma^{-2} + \Gamma - 2)}{(\Gamma - 1)\gamma}, \quad (2.22)$$

onde H é a constante relacionada com a entalpia do fluido, Γ é a constante relacionada com o calor específico, α é a função lapso e γ é o fator de Lorentz. A equação (2.22) garante que existe um ponto sônico entre os valores de γ que satisfazem a igualdade. Tal ponto separa regiões em que o fluido possui velocidades transônicas e subsônicas e esta propriedade torna a acreção de disco mais realista [9].

Na região da magnetosfera tem-se apenas a componente radial da velocidade $v_r = -v_{mag}$. Sendo assim, resolvendo a equação (2.22) pelo método de Newton-Raphson encontramos os valores de γ para cada α e então determina-se v_{mag} através da expressão do fator de Lorentz, neste caso, $\gamma = 1/\sqrt{1 - \beta^2}$ em que $\beta = v_{mag}/c$.

Com a hipótese de que o gás (plasma) é politrópico, isto é, $p = \rho^\Gamma$ e que a razão entre a densidade e pressão é dada por [9, 38],

$$a = \frac{p}{\rho} = \frac{\Gamma - 1}{\Gamma} \left(\frac{H}{\alpha\gamma} - 1 \right) c^2 , \quad (2.23)$$

resulta que as expressões da densidade ρ_q e da pressão são respectivamente,

$$\rho_{mag} = a^{1/(\Gamma-1)} \quad e \quad p = a^{1+1/(\Gamma-1)} . \quad (2.24)$$

Essas informações são suficientes para compor o cenário inicial do disco de acreção magnetizado em rotação, envolvido por uma magnetosfera, ambos influenciados pelo buraco negro de Schwarzschild. No entanto, para determinar as expressões encontradas, as equações diferenciais de GRMHD que modelam a evolução temporal do fenômeno, necessitam de uma expressão que relaciona densidade, pressão, campo magnético e energia, isto é, será necessária uma *Equação de Estado*.

2.4 Equação de Estado

O sistema de equações algébricas, que relacionam as grandezas das equações de GRMHD utilizadas em problemas de formação de jatos relativísticos, está representado pelas seguintes equações

$$\begin{aligned} x(x+1) [\Gamma a x^2 + (2\Gamma a - b)x + \Gamma a - b + d \frac{\Gamma}{2} y^2]^2 = \\ = (\Gamma x^2 + 2\Gamma x + 1)^2 [\tau^2(x+1)^2 + 2\sigma y + 2\sigma x y + \beta^2 y^2] , \end{aligned} \quad (2.25)$$

$$\begin{aligned} [\Gamma(a - \beta^2)x^2 + (2\Gamma a - 2\Gamma\beta^2 - b)x + \Gamma a - b + d - \beta^2 + \frac{\Gamma}{2}y] y \\ = \sigma(x+1)(\Gamma x^2 + 2\Gamma x + 1) , \end{aligned}$$

em que

$$x = \gamma - 1 , \quad y = \gamma(\mathbf{v} \cdot \mathbf{B}) , \quad a = D + \epsilon , \quad b = (\Gamma - 1)D , \quad d = (1 - 0,5\Gamma)B^2 ,$$

$$\tau = P , \quad \beta = B \quad e \quad \sigma = \mathbf{B} \cdot \mathbf{P} .$$

No sistema de equações (2.25) as incógnitas são x e y e o modo escolhido para resolvê-lo foi através do Método de Newton-Raphson [71, 75]. Como a cada passo temporal das equações de GRMHD é necessário resolver tal sistema, a aproximação inicial exigida pelo método de Newton-Raphson é escolhida como sendo as soluções x e y encontradas no passo temporal anterior. Detalhes sobre a equação de estado encontram-se em [63] e seus estudos iniciais em [77].

Capítulo 3

Métodos de Volumes Finitos

Este capítulo trata de métodos numéricos que foram utilizados na resolução de sistemas de equações diferenciais parciais hiperbólicas dependentes do tempo, que auxiliaram na escolha de métodos capazes de obter soluções aproximadas das equações de GRMHD, aplicadas no estudo de formação de jatos relativísticos.

De início apresenta-se a formulação geral dos *métodos de volumes finitos*. Depois, os métodos Lax-Friedrichs, Lax-Wendroff, Nessyahu-Tadmor e Godunov são expostos e comparados em exemplos de escoamento de fluidos descrito pelas equações de Euler. Por fim, o método Nessyahu-Tadmor bidimensional, sem *splitting* dimensional é apresentado.

3.1 Formulação Geral de Leis de Conservação

Leis de conservação são observadas em vários ramos das ciências e são úteis para obter equações fundamentais de fenômenos que possuem grandezas ou quantidades que são balanceadas ao longo de um processo, dentro de uma região definida. Tais leis são representadas por equações diferenciais ou por equações integrais que são consideradas como equações governantes ou equações de movimento do processo em questão. Estas equações descrevem (ditam) como os processos se desenvolvem a medida que há uma evolução temporal.

A seguir encontram-se as formulações integral e diferencial de Leis de Conservação em espaços unidimensionais e espaços de dimensão maior. O texto escrito nesta seção tem

como referências [22, 56, 78, 83].

3.1.1 Leis de Conservação unidimensionais

Seja $u = u(x, t)$ uma quantidade que depende de uma variável espacial $x \in \mathbb{R}$ e do tempo $t \in \mathbb{R}, t > 0$, sendo assim, por definição u varia somente na direção x de tal modo que sua quantidade é igualmente distribuída ao longo de uma seção transversal de área fixa A . Se $I = [a, b]$ é um intervalo onde u está definido, então a quantidade total de u dentro do intervalo I no tempo t é dada por

$$u_{total} = \int_a^b u(x, t) A dx .$$

Assumindo que existe apenas movimento na direção x , o fluxo $\phi(x, t)$ de u na posição x e no tempo t é a quantidade de u que escoar através de A por unidade de Área, por unidade de tempo. Assim, em um tempo t a taxa líquida de u no intervalo I é a taxa com que u atravessa para dentro de I em $x = a$ menos a taxa com que u escoar para fora de I em $x = b$, isto é,

$$taxa líquida = A\phi(a, t) - A\phi(b, t) .$$

Dentro do intervalo I a grandeza u pode ser acrescida ou perdida por uma fonte que é representada por uma *função fonte* ou também chamado de *termo fonte*. Tal termo é uma função local que atua em cada ponto $x \in (a, b)$ podendo depender de x, t, u e das derivadas de u e será representado por $s = s(x, t, u)$, sendo assim, f é a taxa com que u é acrescentada ou diminuída na posição x e no tempo t por unidade de volume. Se s é positivo tem-se uma fonte e se s é negativo tem-se um sumidouro. Contudo, pode-se calcular a taxa com que a quantidade total de u é acrescentada (ou retirada) de I ,

$$taxa que u é produzida = \int_a^b s(x, t, u) A dx .$$

A *Lei de Balanço* fundamental formulada para a quantidade u em qualquer intervalo I é dado por: a taxa de variação de u_{total} com relação ao tempo é igual a taxa líquida

de u escoada em I mais a taxa com que u é produzida no interior de I . Logo, a Lei de Balanço em notação integral é representado por,

$$\frac{d}{dt} \int_a^b u(x, t) = \phi(a, t) - \phi(b, t) + \int_a^b s(x, t, u) dx . \quad (3.1)$$

A equação (3.1) define a formulação integral de uma lei de balanço unidimensional. Se as funções u , ϕ e s são continuamente diferenciáveis em \mathbb{R}^2 então pode-se definir uma formulação diferencial para a lei de balanço representada pela equação (3.1).

Sejam u , ϕ e s continuamente diferenciáveis em \mathbb{R}^2 . Recorrendo ao a) teorema fundamental do cálculo e b) ao teorema de Leibniz,

a)

$$\int_a^b \phi_x(x, t) dx = \phi(b, t) - \phi(a, t) ;$$

b)

$$\frac{d}{dt} \int_a^b u(x, t) dx = \int_a^b u_t(x, t) dx ,$$

a lei de balanço pode ser reescrita por

$$\int_a^b [u_t(x, t) + \phi_x(x, t) - s(x, t, u)] dx = 0 ,$$

em que $(\)_w \equiv \frac{\partial}{\partial w} (\)$. Como o integrando é uma função contínua de x e o resultado de sua integral definida sobre o intervalo $I = [a, b]$ é igual a zero, então

$$u_t(x, t) + \phi_x(x, t) = s(x, t, u) . \quad (3.2)$$

A equação (3.2) é a formulação diferencial da *Lei de Balanço*, isto é, uma equação diferencial parcial que representa a lei de balanço. Se o intervalo $I = [a, b]$ for um intervalo qualquer dos *Reais*, então a lei de balanço tanto na sua forma integral - equação (3.1) - quanto na sua forma diferencial - equação (3.2) - vale para todo $x \in \mathbb{R}$ e para todo $t > 0$.

As leis de balanço também podem ser obtidas dentro de um intervalo tão pequeno quanto se queira, isto é, um intervalo $I = [x, x + \Delta x]$ com $\Delta x \rightarrow 0$. Assumindo suavidade das funções u , ϕ e s temos a lei de balanço no intervalo I dado por

$$\frac{\partial}{\partial t}(u(\xi, t)A\Delta x) = A\phi(x, t) - A\phi(x + \Delta x, t) + s(\eta, t, u(\eta, t))A\Delta x ,$$

em que ξ e η são pontos de $(x, x + \Delta x)$, garantido pelo teorema do valor médio. Agora dividindo a expressão por $A\Delta x$ obtêm-se

$$\frac{\partial}{\partial t}u(\xi, t) = \frac{\phi(x, t) - \phi(x + \Delta x, t)}{\Delta x} + s(\eta, t, u(\eta, t)) ,$$

e para $\Delta x \rightarrow 0$, tem-se

$$\frac{\partial}{\partial t}u(x, t) + \frac{\partial}{\partial x}\phi = s(x, t, u) \quad \text{ou} \quad u_t + \phi_x = s .$$

Este procedimento obtém-se uma formulação forte para as Leis de Conservação, válidas para $t \in (0, T^*)$.

Se nas leis de balanço definidas pelas equações (3.1) e (3.2) as funções ϕ e s forem definidas por $\phi = f(u)$ e $s = 0$ (sem termo fonte dentro do intervalo I), então tais leis definem as formulações diferencial e integral para as *Leis de Conservação*.

Definição 3.1.1 *Seja $u = u(x, t)$ uma quantidade dependente de uma variável espacial $x \in I \subset \mathbb{R}$ e do tempo $t \in \mathbb{R}, t > 0$. Se a quantidade u satisfaz a equação*

$$\frac{d}{dt} \int_a^b u(x, t) = f(u(a, t)) - f(u(b, t)) , \quad (3.3)$$

*então a quantidade u é conservada. A equação (3.3) define uma **formulação integral para Leis de Conservação** para quantidades que se conserva no sentido da expressão (3.3).*

Na definição , a função u é integrável no sentido de Lebesgue.

Definição 3.1.2 *Seja $u = u(x, t)$ uma quantidade dependente de uma variável espacial $x \in I \subset \mathbb{R}$ e do tempo $t \in \mathbb{R}, t > 0$. Se u e $f = f(u)$ são funções contínuas e deriváveis em I , então a partir da definição 3.1.1 pode-se obter uma **formulação diferencial para***

Leis de Conservação, dada pela seguinte expressão:

$$\frac{\partial}{\partial t}u(x,t) + \frac{\partial}{\partial x}f(u(x,t)) = 0 \quad \text{ou} \quad u_t + f(u)_x = 0. \quad (3.4)$$

Se quantidade u satisfaz a equação (3.4), então u é conservada no sentido da expressão (3.4).

Assim, um conjunto de equações diferenciais parciais, em que cada EDP satisfaz a definição 3.1.2, define um sistema de Leis de conservação.

Definição 3.1.3 (Leis de Conservação) *Um sistema de m equações diferenciais parciais dependentes do tempo são chamadas de Leis de Conservação unidimensional se puder ser reescrito da seguinte forma,*

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = 0,$$

onde

$$\mathbf{U} = \begin{pmatrix} u_1 \\ \vdots \\ u_m \end{pmatrix}, \quad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} f_1 \\ \vdots \\ f_m \end{pmatrix},$$

com

$$\begin{aligned} U = \mathbf{U}(x,t) &: \Omega \times [0, T] \rightarrow \mathbb{U}_m \\ \mathbf{F} = \mathbf{F}(\mathbf{U}) &: \mathbb{U}_m \rightarrow \mathbb{R}^m \end{aligned},$$

em que os símbolos $\Omega \subset \mathbb{R}$ representa o domínio espacial de interesse e $\mathbb{U}_m \subset \mathbb{R}^m$ é o conjunto de estados físicos admissíveis (espaço de fase).

Na definição 3.1.3, \mathbf{U} é chamado de *vetor de variáveis conservadas*, $\mathbf{F} = \mathbf{F}(\mathbf{U})$ é chamado de vetor de fluxo e f_i para $i = 1, \dots, m$ são chamadas de funções fluxo.

Uma particularidade associada aos sistemas de Leis de Conservação está relacionada ao conjunto de autovalores e autovetores, que definem sua hiperbolicidade. Alguns métodos de volumes finitos foram criados de modo a utilizar tal conjunto, para obter mais informações sobre o problema modelado e, conseqüentemente, tornarem-se métodos mais

eficazes em algum sentido. Para um sistema de Leis de Conservação ser hiperbólico deve-se satisfazer a seguinte condição:

Definição 3.1.4 (Matriz Jacobiana de fluxo) *A matriz jacobiana do vetor fluxo \mathbf{F} da definição 3.1.3 é representada por*

$$\mathcal{A}(\mathbf{U}) = \frac{\partial \mathbf{F}}{\partial \mathbf{U}} = \partial_{\mathbf{U}} \mathbf{F} = \begin{pmatrix} \frac{\partial f_1}{\partial u_1} & \dots & \frac{\partial f_1}{\partial u_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial u_1} & \dots & \frac{\partial f_m}{\partial u_m} \end{pmatrix}.$$

Definição 3.1.5 (Sistema Hiperbólico) *Um sistema de m equações diferenciais na forma de leis de conservação é dito hiperbólico no ponto $(x, t) \in \Omega \times [0, T]$, se a matriz jacobiana \mathcal{A} da definição 3.1.4 tiver m autovalores Reais, $\lambda_1, \dots, \lambda_m$ e, também se o conjunto de autovetores correspondentes a direita, $\mathbf{K}^{(1)}, \dots, \mathbf{K}^{(m)}$, formar um conjunto linearmente independente.*

Se os autovalores da definição 3.1.5 forem todos distintos, então o sistema hiperbólico é chamado de *Sistema estritamente hiperbólico*. Nota-se que hiperbolicidade estrita implica em hiperbolicidade, pois autovalores reais e distintos garantem a existência de um conjunto linearmente independente de autovetores associados aos autovalores do sistema hiperbólico [22, 83].

Uma classe de métodos numéricos que utilizam as informações de autovalores e autovetores em seu desenvolvimento são os métodos de Godunov. Tais métodos necessitam da resolução de problemas de Riemann que faz uso dos autovalores obtidos da matriz Jacobiana, para definir as velocidades de propagação de ondas do tipo: ondas de choque, de rarefação e de contato. Métodos que não necessitam de resolver problemas de Riemann muitas vezes são chamados de *Métodos livres de solucionadores de problemas de Riemann*.

Um sistema hiperbólico conforme as definições 3.1.3 e 3.1.5 possui infinitas soluções se nenhuma condição for acrescentada na função \mathbf{U} , sendo assim, com o intuito de investigar existência e unicidade de soluções de problemas hiperbólicos, adicionam-se condições

adequadas ao sistema para torná-lo **bem posto**. Discussões sobre existência e unicidade de soluções de Equações Diferenciais Parciais e, também, em particular para Leis de Conservação encontram-se nas referências [22, 35, 56, 83].

Definição 3.1.6 (Problema de Cauchy) *O Problema de Valor Inicial para a definição 3.1.3, representado por*

$$\begin{cases} \partial_t \mathbf{U} + \sum_{i=1}^n \partial_{x_i} \mathbf{F}_i(\mathbf{U}) = 0, & \forall \mathbf{x} \in \mathbb{R}^n, t \in (0, T), \\ \mathbf{U}(\mathbf{x}, 0) = \mathbf{U}_0(\mathbf{x}), & \forall \mathbf{x} \in \mathbb{R}^n \end{cases} \quad (3.5)$$

em que $\mathbf{U}_0(\mathbf{x}) \in C^0(\mathbb{R}^n, \mathbf{U}_m)$, $\mathbf{F}_i \in C^1(\mathbf{U}_m, \mathbb{R}^m)$ e $\mathbf{x} \in \mathbb{R}^n$ é chamado de **problema de Cauchy**. Se $\mathbf{U} \in C^1(\mathbb{R}^n \times [0, T], \mathbf{U}_m)$ satisfaz o problema de Cauchy, então \mathbf{U} é uma **solução clássica** do problema de Cauchy.

Sistemas hiperbólicos não lineares têm a característica de desenvolver descontinuidades ao longo da evolução temporal, mesmo que a condição inicial do problema de Cauchy seja suficientemente suave. Dessa maneira, espera-se a existência de soluções clássicas para o problema de Cauchy apenas em um tempo finito. Portanto, no sentido de se obter soluções globais (para qualquer tempo $t > 0$), necessita-se da generalização do conceito de solução para o problema de Cauchy.

A seguir, exemplifica-se tal situação em um problema de Cauchy em uma equação hiperbólica unidimensional.

Exemplo 3.1.1 (Equação de Burger sem viscosidade) *Considere o seguinte problema de Cauchy,*

$$\begin{cases} \partial_t u + u \partial_x u = 0 \\ u(x, 0) = u_0(x) \end{cases}$$

em que

$$u_0(x) = \begin{cases} 1 & \text{se } x < 0 \\ 1 - x & \text{se } 0 \leq x \leq 1 \\ 0 & \text{se } x > 1 \end{cases} .$$

A solução clássica para o exemplo 3.1.1 definida no intervalo $0 \leq t < 1$ é dada por

$$u(x, t) = \begin{cases} 1 & \text{se } x < t < 1 \\ \frac{1-x}{1-t} & \text{se } t \leq x < 1 \\ 0 & \text{se } t < 1 \leq x \end{cases} .$$

Note que no exemplo 3.1.1, a condição inicial é contínua e a solução é contínua apenas para $t \in [0, 1)$. Para $t = 1$ a solução é descontínua em $x = 1$, o que exemplifica que condições iniciais contínuas podem gerar soluções descontínuas. Para detalhes do exemplo 3.1.1 ver [35].

A partir do tempo $t \geq 1$ o exemplo 3.1.1 não possui solução clássica, tornando-se necessário generalizar o conceito de solução. Tal generalização é chamada de *solução fraca* [52].

Uma outra característica associada aos sistemas hiperbólicos é que condições iniciais descontínuas podem evoluir para soluções contínuas, porém neste caso, a condição inicial não é de classe C^1 e portanto não satisfaz as hipóteses da definição do problema de Cauchy. Entretanto, sistemas com condições iniciais descontínuas enquadram-se em uma outra classe de problemas.

Definição 3.1.7 (Problema de Riemann) *Um Problema de Valor Inicial para a definição 3.1.3, representado por*

$$\begin{cases} \partial_t \mathbf{U} + \sum_{i=1}^n \partial_{x_i} \mathbf{F}_i(\mathbf{U}) = 0, & \forall \mathbf{x} \in \mathbb{R}^n, t > 0 \\ \mathbf{U}(\mathbf{x}, 0) = \mathbf{U}_0(\mathbf{x}), & \forall \mathbf{x} \in \mathbb{R}^n \end{cases} \quad (3.6)$$

em que

$$\mathbf{U}_0(\mathbf{x}) = \begin{cases} \mathbf{U}_L(\mathbf{x}), & \text{se } x < 0 \\ \mathbf{U}_R(\mathbf{x}), & \text{se } x > 0 \end{cases}$$

é chamado de **problema de Riemann**.

Com a definição de Problema de Riemann é possível exemplificar um problema no qual a condição inicial é descontínua e, conforme acontece a evolução temporal, a solução passa a ser contínua.

Exemplo 3.1.2 (Equação de Burger sem viscosidade) *Considere o seguinte problema de Riemann,*

$$\begin{cases} \partial_t u + u \partial_x u = 0 \\ u(x, 0) = u_0(x) \end{cases}$$

em que

$$u_0(x) = \begin{cases} 0 & \text{se } x < 0 \\ 1 & \text{se } x > 0 \end{cases} .$$

No exemplo 3.1.2, a solução definida para $t > 0$ é dada por

$$u(x, t) = \begin{cases} 0 & \text{se } x < 0, t \geq 0 \\ \frac{x}{t} & \text{se } 0 \leq x \leq t, t \neq 0 \\ 1 & \text{se } x > t \geq 0 \end{cases} .$$

A solução do exemplo 3.1.2 é contínua para $t > 0$, mesmo que se tenha uma condição inicial descontínua. Portanto, ao se fazer uma evolução temporal da solução, percebe-se que a descontinuidade inicial não foi propagada e que apenas tem-se descontinuidades nas derivadas. Este comportamento é característico de fenômenos puramente não lineares, mais detalhes e exemplos de tal comportamento encontram-se em [22], [56] e [35].

3.1.2 Leis de Conservação

Em um Espaço Euclidiano n dimensional - \mathbb{R}^n - pode-se formular leis de conservação baseadas nas expressões unidimensionais.

Sejam $x = (x_1, \dots, x_n) \in \mathbb{R}^n$ e $u = u(x, t)$ uma função escalar definida em um domínio $D \subset \mathbb{R}^n$. Neste domínio seja V uma região arbitrária com uma fronteira suficientemente suave denotada por ∂V . A quantidade total de u em V é dada pela seguinte integral volumétrica,

$$u_{total} = \int_V u(x, t) dV ,$$

onde dV é um elemento de volume de V . Se $s = s(x, t, u)$ denota o termo fonte contido

em V , então a taxa com que u é produzida ou consumida em V é dada por

$$\text{taxa de produção de } u = \int_V s(x, t, u) dV .$$

Se $\mathbf{n}(x)$ é um vetor unitário na direção normal de um elemento de área da superfície ∂V , então a taxa líquida com que o vetor fluxo $\phi(x, t)$ atravessa a fronteira ∂V é fornecida pela seguinte integral de superfície

$$\text{taxa líquida} = \int_{\partial V} \phi(x, t) \cdot \mathbf{n}(x) dS .$$

Logo, a *formulação integral da lei de balanço* para a quantidade u na região V é dada por

$$\frac{d}{dt} \int_V u dV = - \int_{\partial V} \phi \cdot \mathbf{n} dS + \int_V s dV . \quad (3.7)$$

Se u e ϕ são suficientemente suaves, pode-se utilizar o *Teorema do Divergente*,

$$\int_V \text{div } \phi dV = \int_{\partial V} \phi \cdot \mathbf{n} dS ,$$

em que *div* é o operador divergente, para reescrever a equação (3.7)

$$\int_V u_t dV = - \int_V u dV + \int_V s dV$$

e obter a *formulação diferencial da lei de balanço*,

$$u_t + \text{div } \phi = s(x, t, u) \quad \forall x \in D, \quad t > 0 . \quad (3.8)$$

Se nas leis de balanço definidas pelas equações (3.7) e (3.8) as funções ϕ e s forem definidas por $\phi = (f_1(u), \dots, f_n(u))$ e $s = 0$ (sem termo fonte dentro de V), então tais leis definem as formulações diferencial e integral para as *Leis de Conservação*.

Definição 3.1.8 *Seja $u = u(x, t)$ uma quantidade dependente de uma variável espacial*

$x \in D \subset \mathbb{R}^n$ e do tempo $t \in \mathbb{R}, t > 0$. Se a quantidade u satisfaz a equação

$$\frac{d}{dt} \int_a^b u(x, t) = - \int_{\partial V} \phi(x, t) \cdot \mathbf{n}(x) dS, \quad (3.9)$$

então a quantidade u é conservada. A equação (3.9) define uma **formulação integral para Leis de Conservação** para quantidade que se conserva no sentido da expressão (3.9).

Definição 3.1.9 Seja $u = u(x, t)$ uma quantidade dependente de $x \in D \subset \mathbb{R}^n$ e do tempo $t \in \mathbb{R}, t > 0$. Se u e as $f_i = f_i(u)$, com $i = 1, \dots, n$, são funções contínuas e deriváveis em D , então a partir da definição 3.1.8 pode-se obter uma **formulação diferencial para Leis de Conservação** dada pela seguinte expressão:

$$\frac{\partial}{\partial t} u(x, t) + \operatorname{div} (f(u(x, t))) = 0 \quad \text{ou} \quad u_t + \operatorname{div}(f(u)) = 0. \quad (3.10)$$

Se quantidade u satisfaz a equação (3.10), então u é conservada no sentido da expressão (3.10).

Um conjunto de equações diferenciais parciais dependentes do tempo pode constituir um sistema de Leis de Conservação se satisfizer a seguinte definição.

Definição 3.1.10 (Leis de Conservação) Um sistema de m equações diferenciais parciais dependentes do tempo são chamadas de Leis de Conservação se puder ser reescrito da seguinte forma

$$\partial_t \mathbf{U} + \operatorname{div} (\mathbf{F}) = 0, \quad (3.11)$$

donde

$$\mathbf{U} = \begin{pmatrix} u_1 \\ \vdots \\ u_m \end{pmatrix}, \quad \mathbf{F} = [\mathbf{F}_1(\mathbf{U}) | \dots | \mathbf{F}_n(\mathbf{U})], \quad \mathbf{F}_i(\mathbf{U}) = \begin{pmatrix} f_{(i)1} \\ \vdots \\ f_{(i)m} \end{pmatrix},$$

$\Omega_n \subset \mathbb{R}^n$ representa o domínio espacial de interesse e $\mathbf{U}_m \in \mathbb{R}^m$ é o conjunto de estados físicos admissíveis (espaço de fase) com

$$u_j = u_j(\mathbf{x}, t) \in \mathbf{U}_m, \quad \mathbf{x} = \mathbf{x}(x_1, \dots, x_n) \in \mathbb{R}^n, \quad t \in [0, T] \subset \mathbb{R} \quad \text{e} \quad \mathbf{F}_i \in \mathbb{R}^m$$

para $i = 1, \dots, n$.

Na definição 3.1.10, $\mathbf{U} : \Omega_n \times [0, T] \rightarrow \mathbb{U}_m$ é chamado de **vetor de variáveis conservadas** e $\mathbf{F}_i(\mathbf{U}) : \mathbb{U}_m \rightarrow \mathbb{R}^m$ são chamados de **vetores de fluxos** ou funções fluxos. Em particular, destacam-se as leis de conservação de sistemas hiperbólicos dependentes do tempo.

Definição 3.1.11 (Sistema Hiperbólico) *Um sistema de leis de conservação é dito hiperbólico em \mathbb{G}_m se o **Jacobiano** de cada função fluxo da definição 3.1.10,*

$$\mathcal{A}_e(\mathbf{U}) := \sum_{i=1}^n D\mathbf{F}_i(\mathbf{U})e_i$$

é diagonalizável em \mathbb{R}^m com seus autovalores Reais, para todo $U \in \mathbb{G}_m$ e para todo $e \in \mathbb{S}^{n-1} := \{(e_1, \dots, e_n) \in \mathbb{R}^n \mid \sum_{i=1}^n e_i^2 = 1\}$. Cada matriz $D\mathbf{F}_i$ é o jacobiano correspondente do fluxo \mathbf{F}_i .

3.2 Formulação Geral de Métodos de Volumes Finitos Unidimensionais

Esta seção trata da formulação geral de métodos de volumes finitos aplicados em leis de conservação dependentes do tempo e unidimensionais no espaço.

As ideias básicas para a obtenção de tais métodos são obtidos pela subdivisão do domínio espacial em intervalos chamados de *volumes finitos* ou *células da malha*.

Sendo assim, considere uma partição regular do domínio espacial

$$\Omega_x : x_1 < \dots < x_i < \dots < x_{N+1} \text{ ,}$$

com N subintervalos em que

$$\Delta x = \frac{x_{i+1} - x_i}{N} \text{ ,}$$

para qualquer valor de i existente. A i -ésima *célula média* ou o i -ésimo *volume finito* é definido por, $\Omega_i = (x_{i-1/2}, x_{i+1/2})$ onde $x_{i+1/2} = \frac{x_{i+1} + x_i}{2}$, nota-se que cada célula média

tem o tamanho do espaçamento Δx , pois $x_{i+1/2} - x_{i-1/2} = \Delta x$.

Sejam $u = u(x, t)$ uma função que representa uma grandeza física definida do domínio espacial Ω e Ω_x uma partição regular de Ω . Se u é conservada em Ω , então pela definição 3.3 a lei de conservação é satisfeita em cada célula média ou volume finito Ω_i , isto é,

$$\frac{d}{dt} \int_{\Omega_i} u(x, t) dx = f(u(x_{i-1/2}, t)) - f(u(x_{i+1/2}, t)). \quad (3.12)$$

Ao definir uma partição regular para a variável independente t em que $\Delta t = t_{n+1} - t_n$, um algoritmo explícito de evolução temporal de um único passo é obtido integrando (3.12) no tempo entre os valores t_n e t_{n+1} da seguinte maneira,

$$\int_{\Omega_i} u(x, t_{n+1}) dx - \int_{\Omega_i} u(x, t_n) dx = \int_{t_n}^{t_{n+1}} f(u(x_{i-1/2}, t)) dt - \int_{t_n}^{t_{n+1}} f(u(x_{i+1/2}, t)) dt ,$$

rearranjando os termos e dividindo a expressão por Δx , tem-se

$$\begin{aligned} \frac{1}{\Delta x} \int_{\Omega_i} u(x, t_{n+1}) dx &= \frac{1}{\Delta x} \int_{\Omega_i} u(x, t_n) dx + \\ &- \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(u(x_{i+1/2}, t)) dt - \int_{t_n}^{t_{n+1}} f(u(x_{i-1/2}, t)) dt \right]. \end{aligned} \quad (3.13)$$

A equação (3.13) fornece um procedimento para encontrar os valores de u no tempo t_{n+1} , nos pontos x_i . Nota-se que tal equação integral é exata (não possui aproximações) e é válida para todo subintervalo de Ω .

Com o intuito de se obter uma aproximação de diferenças finitas para a formulação diferencial da Lei de Conservação - definição 3.1.1 - as integrais

$$\frac{1}{\Delta x} \int_{\Omega_i} u(x, t_{n+1}) dx \quad \text{e} \quad \frac{1}{\Delta x} \int_{\Omega_i} u(x, t_n) dx ,$$

são vistas como o valor médio da grandeza u respectivamente nos tempos t_n e t_{n+1} , sobre o ponto x_i da célula Ω_i .

As últimas integrais do lado direito da equação (3.13) também podem ser vistas como valor médio do fluxo $f(u)$ nos pontos $x_{i+1/2}$ e $x_{i-1/2}$, porém em geral não se pode calcular tais integrais porque os valores de u nos extremos das células médias não são conhecidos.

Basicamente existem duas maneiras de aproximar $f(u(x_{i_{1/2}}), t)$ para resolver as integrais entre os intervalos t_n e t_{n+1} . Uma delas é aproximar a integral por um método de integração numérica e depois utilizar a série de Taylor para aproximar os valores da função nos extremos da célula média e obter um valor do fluxo médio no ponto médio do volume finito. Isto define os Métodos de Volumes Finitos que podem ser vistos como métodos de diferenças finitas cujos valores médios estão centrados nos x'_i s e que são chamados de *Métodos de Volumes Finitos do tipo Centrado*. Na seção 3.5 são apresentados e detalhados métodos com essas características.

O outro caso é construir uma função polinomial definida em todo $x \in \Omega_i$ passando no ponto $(x_i, u(x_i, t))$. Ao extrapolar tal função polinomial nos pontos $x_{i+1/2}$ e $x_{i-1/2}$ obtêm-se os valores aproximados de $u(x_{i+1/2}, t)$ e $u(x_{i-1/2}, t)$. Entretanto, nota-se que em cada interface definida pelos extremos das células médias há dois valores para u , logo para encontrar uma aproximação para $f(u)$ é necessário resolver um Problema de Riemann em cada interface $x_{i+1/2}$. Tais características definem o que chamamos de *Métodos de Volumes Finitos do tipo Godunov* e no caso da função polinomial constante em cada volume de controle Ω_i , tem-se puramente um *Método de Godunov*. Na seção 3.6 são mostrados métodos com essas propriedades.

Independente das aproximações escolhidas, em ambas as classes de métodos de volumes finitos têm-se

$$U_i^n \cong \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} u(x, t_n) dx = \frac{1}{\Delta x} \int_{\Omega_i} u(x, t_n) dx \quad (3.14)$$

e

$$F_{i+1/2}^n \cong \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} f(u(x_{i+1/2}, t)) dt . \quad (3.15)$$

Os valores U_i^n e $F_{i+1/2}^n$ são respectivamente o valor médio aproximado da quantidade u sobre o volume finito Ω_i no tempo t_n e o valor médio do fluxo que atravessa $x_{i-1/2}$ entre os tempos t_n e t_{n+1} .

No contexto das aproximações (3.14) e (3.15) pode-se reescrever a expressão (3.12) e obter uma aproximação de diferenças para a Lei de Conservação na forma diferencial, ou seja,

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x} [F_{i+1/2}^n - F_{i-1/2}^n] . \quad (3.16)$$

em que $F_{i+1/2}^n = \mathcal{F}(U^n)$ é uma função que depende de valores de U obtidos no tempo t_n . Se de fato é possível aproximar tal fluxo médio através dos valores de U^n , então a equação (3.16) representa um **método discreto de volumes finitos** [53]. Dessa forma, pode-se definir uma expressão para os métodos de volumes finitos conservativos.

Definição 3.2.1 *Se \mathcal{F} é uma função de $(p + q + 1)$ argumentos de tal modo que*

$$F_{i+1/2}^n = \mathcal{F}(U_{i-p}^n, U_{i-p+1}^n, \dots, U_{i+q}^n) ,$$

*então a forma de um **método de volumes finitos conservativo** é dada por*

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x} [\mathcal{F}(U_{i-p}^n, U_{i-p+1}^n, \dots, U_{i+q}^n) - \mathcal{F}(U_{i-p-1}^n, U_{i-p}^n, \dots, U_{i+q-1}^n)] , \quad (3.17)$$

em que \mathcal{F} é chamada de função fluxo numérico, $\Delta x = x_{i+1} - x_i$ é o espaçamento de uma partição regular do domínio espacial e $\Delta t = t_{n+1} - t_n$ é o espaçamento entre dois tempos consecutivos. Ver Figura 3.1.

Nota-se que a equação (3.17) da definição 3.2.1 é a forma discreta da lei de conservação descrita pela definição 3.1.2. Na Figura 3.1, em cada célula Ω_i tem-se uma lei de conservação discreta representada pela equação (3.17) e o que diferencia os métodos de volumes finitos é a forma com que \mathcal{F} é obtida ou aproximada.

O método representado pela equação (3.17) define uma classe de métodos de volumes finitos explícitos de um passo, isto é, que os valores de U_i^{n+1} são obtidos explicitamente dos valores de U_i^n . Dessa maneira, tais métodos podem ser representados da seguinte forma

$$U^{n+1} = QU^n ,$$

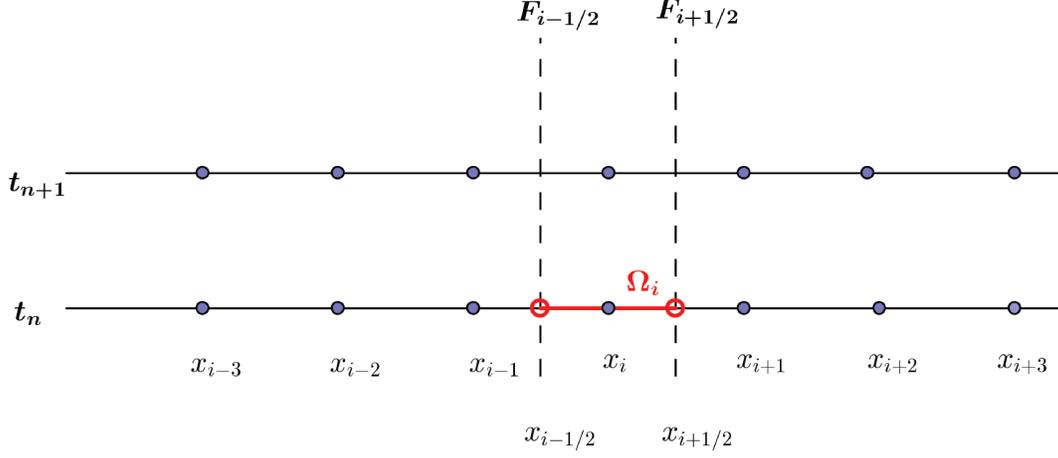


Figura 3.1: A ilustração mostra uma representação geral dos métodos de volumes finitos. Os pontos x_i 's - localizados nos pontos médio de cada volume finito Ω_i 's - constituem o domínio espacial computacional onde no tempo t_n a solução numérica é conhecida e no tempo t_{n+1} a solução é obtida através dos valores U_i^n 's e dos valores $F_{i+1/2}^n$'s definidos pela função fluxo numérico.

onde Q é um operador atuando no espaço das soluções numéricas da equação (3.17).

No caso do problema em estudo envolver termo fonte, isto é, ser representado por uma lei de balanço conforme a equação (3.1), a definição 3.2.1 é reescrita como segue

Definição 3.2.2 *Se \mathcal{F} é uma função de $(p + q + 1)$ argumentos de tal modo que*

$$F_{i+1/2}^n = \mathcal{F}(U_{i-p}^n, U_{i-p+1}^n, \dots, U_{i+q}^n) ,$$

então a forma de um método de volumes finitos para leis de balanço é dada por

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x} [F_{i+1/2} - F_{i-1/2}] + \Delta t S(x_i, t_n, U_i^n) , \quad (3.18)$$

em que \mathcal{F} é chamada de função fluxo numérico, $\Delta x = x_{i+1} - x_i$ é o espaçamento de uma partição regular do domínio espacial, $\Delta t = t_{n+1} - t_n$ é o espaçamento entre dois tempos consecutivos e $S = S(x, t, U)$ é o valor médio do termo fonte $s = s(x, t, u)$.

Em geral, os métodos de volumes finitos explicitos podem ser representados pela seguinte expressão,

$$U^{n+1} = QU^n + \Delta t G^n \quad , \quad (3.19)$$

em que G^n é o termo fonte no tempo t_n .

As equações (3.17) e (3.18) podem ser obtidas tanto através do conceito de volumes finitos quanto da aproximação de diferenças finitas que advém de séries de Taylor, sendo assim, as análises e resultados de métodos numéricos de volumes finitos e de métodos de diferenças finitas são equivalentes.

A ideia é encontrar soluções a partir da equação (3.17) e U aproximadamente igual a u em algum sentido e assim, U constitui uma solução aproximada para a equação (3.4) que representa a lei de conservação na forma diferencial.

A seguir apresentam-se definições e resultados que fornecem condições para avaliar as propriedades e características de soluções aproximadas obtidas pelos métodos de volumes finitos. Tais resultados estão relacionados com os conceitos de Convergência, Consistência e Estabilidade.

3.3 Convergência, Consistência e Estabilidade

Nesta seção conceitos e resultados referentes aos métodos numéricos e soluções numéricas são explorados com o intuito de investigar as propriedades e características de tais soluções, quando utilizadas como aproximações de soluções analíticas de Leis de Conservação ou de Leis de Balanço.

O principal objetivo é fazer algumas considerações teóricas de métodos numéricos para verificar sob quais condições e em que sentido a solução numérica converge para a solução analítica da equação diferencial. Tal convergência é analisada através dos conceitos de consistência e estabilidade, que é embasada pelo *Teorema de Lax-Richtmyer* ou simplesmente Teorema de Lax, cuja aplicabilidade depende da linearidade do método numérico.

O desenvolvimento desta seção segue a notação: a equação diferencial parcial (EDP) é representada pela equação $\mathcal{L}u = F$ e um esquema de diferenças (métodos de diferenças finitas ou métodos de volumes finitos) associado a EDP é $L_i^n U_i^n = G_i^n$, em que L é um operador diferencial que envolve termos espaciais e termo temporal, L_i^n é a discretização

do operador \mathcal{L} , u é a solução analítica da EDP, U_i^n é uma solução aproximada obtida pelo método numérico, F é o termo fonte da EDP e G_i^n é o termo fonte discretizado. O seguinte texto basea-se nas referências [34, 52, 53, 82, 86, 88].

3.3.1 Convergência

Ao estudar métodos numéricos para resolver equações diferenciais é natural questionar se a solução obtida por tal método é adequada para representar a solução analítica. Esses questionamentos remetem-se ao conceito de convergência de métodos numéricos, isto é, se as soluções produzidas pelo método numérico estão suficientemente próximas da solução analítica.

De início, os conceitos de convergência de soluções aproximadas obtidas de métodos numéricos são definidos para problemas de Cauchy. Assim, para as definições a seguir considera-se uma equação diferencial parcial dependente do tempo e unidimensional no espaço, com uma condição inicial,

$$\begin{cases} \mathcal{L}u = F, & \forall x \in \mathbb{R}, t \in (0, T) \subset \mathbb{R} \\ u(x, 0) = f(x), & \forall x \in \mathbb{R} \end{cases}.$$

Definição 3.3.1 (Convergência Pontual) *Um esquema numérico é pontualmente convergente se para todo (x, t) com $(i\Delta x, (n+1)\Delta t) \rightarrow (x, t)$ verifica-se que*

$$U_i^{n+1} \rightarrow u(x, t), \text{ quando } \Delta t \rightarrow 0 \text{ e } \Delta x \rightarrow 0,$$

em que $\{U^{n+1}\}$ é um conjunto de soluções aproximadas de u - que satisfaz $\mathcal{L}u = F$ - obtida pontualmente pelo esquema numérico $L_i^n U_i^n = G_i^n$.

Nota-se que a definição 3.3.1 não depende do fato de que todas as soluções U^{n+1} sejam soluções aproximadas de u ; tal definição só garante que no limite tem-se U^{n+1} suficientemente próxima de u . Também, podemos ter situações em que U_i^{n+1} esteja suficientemente próximo de u para alguns valores de i e para outros não. Contudo, uma outra definição de convergência é escrita para fornecer a ideia de convergência uniforme.

Definição 3.3.2 (Convergência Uniforme) *Um esquema numérico $L_i^n U_i^n = G_i^n$ que produz soluções aproximadas para $\mathcal{L}u = F$ é uniformemente convergente no tempo t se para todo t com $(n+1)\Delta t \rightarrow t$, tem-se*

$$\|\mathbf{U}^{n+1} - \mathbf{u}^{n+1}\| \rightarrow 0 \text{ , quando } \Delta t \rightarrow 0 \text{ e } \Delta x \rightarrow 0 \text{ ,}$$

na norma do sup do espaço normado $(l^\infty, \|\cdot\|)$, em que l^∞ é o espaço das seqüências limitadas, $\mathbf{U}^n = \{U_k^n\}$, $\mathbf{u}^n = \{u(i\Delta x, n\Delta t)\} \in l^\infty$.

Com a definição de convergência, pode-se questionar sobre a velocidade de convergência dos métodos numéricos e para isso temos a seguinte definição.

Definição 3.3.3 (Ordem de Convergência) *Um esquema numérico $L_i^n U_i^n = G_i^n$ que produz soluções aproximadas para $\mathcal{L}u = F$ é um esquema convergente de ordem (p, q) , se para qualquer t , com $(n+1)\Delta t \rightarrow t$, tem-se*

$$\|\mathbf{U}^{n+1} - \mathbf{u}^{n+1}\| = \mathcal{O}(\Delta x^p) + \mathcal{O}(\Delta t^q) \text{ , quando } \Delta t \rightarrow 0 \text{ e } \Delta x \rightarrow 0 \text{ .}$$

Assim, dizemos que o esquema é de ordem p no espaço e de ordem q no tempo.

As definições 3.3.1, 3.3.2 e 3.3.3 descritas até agora referem-se ao conceito de convergência para Problemas de Valor Inicial (PVI), nos quais a solução está definida em toda reta Real. Entretanto em diversos problemas e, também, em domínios computacionais têm-se restrições que limitam o tamanho do domínio. Tal situação exige que se acrescentem condições na fronteira do domínio transformando o problema em um problema de valor inicial com condições de contorno (PVIC), sendo necessária uma definição para problemas desse tipo. Considere o seguinte PVIC,

$$\begin{cases} \mathcal{L}u = F , & x \in \Omega \subset \mathbb{R}, t \in (0, T) \subset \mathbb{R} \\ u(x, 0) = f(x) , & x \in \Omega \subset \mathbb{R} \\ u|_{\partial\Omega} = h \end{cases}$$

em que $u|_{\partial\Omega}$ é uma restrição à função u na fronteira de Ω conhecida. Tais condições podem ser do tipo: condições de Dirichlet, de Neumann ou de Robin.

Seja $\Omega_{\Delta x_j}$ uma partição regular descrita pelo espaçamento Δx_j do domínio Ω . Para cada partição regular aplicada para discretizar o domínio de solução tem-se uma solução aproximada obtida por um esquema numérica de $\mathcal{L}u = F$ dado.

Assim, seja X_j um espaço normado de dimensão finita contendo as soluções associadas ao incremento Δx_j cuja norma seja representada por $\|\cdot\|_j$. Se $\{\Delta x_j\}_1^\infty$ é uma sequência qualquer de partições tal que $\Delta x_j \Rightarrow 0$ quando $j \rightarrow \infty$, então define-se convergência de PVIC da seguinte maneira.

Definição 3.3.4 (Convergência em PVIC) *Um esquema numérico é dito convergente no tempo t se para qualquer sequência de partições $\{\Delta x_j\}$ com $(n+1)\Delta t \Rightarrow t$, tem-se*

$$\|U^n - u^n\|_j \rightarrow 0, \text{ quando } j \rightarrow \infty \text{ e } \Delta t \rightarrow 0,$$

em que U e u são respectivamente a solução numérica e a solução analítica do PVIC, no domínio discretizado.

Em geral, provar a convergência de um método numérico diretamente da definição é trabalhoso, entretanto existem dois conceitos relacionados que podem auxiliar nesta tarefa: a consistência e a estabilidade.

3.3.2 Consistência

Definição 3.3.5 (Consistência Pontual) *Um esquema de diferenças $L_i^n U_i^n = G_i^n$ é pontualmente consistente com a equação diferencial $\mathcal{L}u = F$ em um ponto (x, t) qualquer, se para qualquer função suficientemente $\phi = \phi(x, t)$ tem-se*

$$(\mathcal{L}\phi - F)_i^n - [L_i^n \phi(i\Delta x, n\Delta t) - G_i^n] \rightarrow 0,$$

quando $\Delta x, \Delta t \rightarrow 0$ e $(i\Delta x, (n+1)\Delta t) \rightarrow (x, t)$.

Se o esquema numérico da forma $L_i^n U_i^n = G_i^n$ for de um único passo, isto é, para determinar a solução no tempo $t^{n+1} = (n+1)\Delta t$ é necessário apenas conhecer a solução

do tempo t^n , então pode-se reescrevê-lo da seguinte maneira

$$U^{n+1} = QU^n + \Delta t G^n, \quad (3.20)$$

onde Q é um operador que atua no espaço das soluções aproximadas de um método específico.

Ao substituir a solução analítica no método numérico representado pela equação (3.20) a igualdade não é satisfeita, pois o método numérico é uma aproximação da EDP. Logo, tem-se um erro intrínseco ao método numérico. Ao adicionar tal erro em (3.20), a equação é satisfeita quando aplica-se a variável u e assim podemos definir um conceito de consistência mais exigente.

Definição 3.3.6 (Consistência Uniforme) *Um esquema numérico da forma $U^{n+1} = QU^n + \Delta t G^n$ é uniformemente consistente com a equação diferencial associada na norma $\|\cdot\|$, se a solução u da equação diferencial satisfaz,*

$$u^{n+1} = Qu^n + \Delta t G^n + \Delta t \tau^n \quad (3.21)$$

e

$$\|\tau^n\| \rightarrow 0,$$

quando $\Delta x, \Delta t \rightarrow 0$. O termo τ^n ou $\|\tau^n\|$ é chamado de **erro de truncamento local**.

Uma variação da definição de consistência uniforme utiliza a ordem (p, q) de precisão com que o erro de truncamento local tende a zero. A seguir, apresenta-se tal definição.

Definição 3.3.7 (Consistência Uniforme) *Um esquema numérico da forma $U^{n+1} = QU^n + \Delta t G^n$ possui uma ordem (p, q) de precisão a uma dada equação diferencial se*

$$\|\tau^n\| = \mathcal{O}(\Delta x^p) + \mathcal{O}(\Delta t^q),$$

em que τ^n ou $\|\tau^n\|$ é o erro de truncamento local da definição.

Dizer que um método numérico é consistente com a equação diferencial associada, significa que a solução suficientemente suave da equação diferencial é uma solução aproximada para o esquema numérico.

A consistência é necessária para a convergência de soluções aproximadas para a solução analítica, porém não é uma condição suficiente. Exemplos de métodos consistentes que não são convergentes encontram-se nas referências [82, 86]. Contudo, além do esquema ser consistente é necessário que o método também seja estável.

3.3.3 Estabilidade

De certo modo, introduzir o conceito de estabilidade de métodos numéricos remete-se a definição de convergência. Se soluções aproximadas U_i^n convergem para solução analítica u , então certamente U_i^n é limitada. Portanto, analisar estabilidade de esquemas numéricos significa investigar se o método numérico dado produz soluções limitadas em algum sentido.

Seja

$$U^{n+1} = QU^n \tag{3.22}$$

um esquema de diferenças associado a um problema de valor inicial cuja EDP seja linear e homogênea. Define-se a estabilidade de um método numérico do tipo (3.22), da seguinte maneira

Definição 3.3.8 (Estabilidade) *Um esquema numérico conforme a equação (3.22), associado a um PVI cuja EDP é linear e homogênea é chamado **estável** se existem constantes positivas Δx_0 e Δt_0 , e constantes não negativas K e β tais que,*

$$\|U^{n+1}\| \leq Ke^{\beta t} \|U^0\| \tag{3.23}$$

para $0 \leq t = (n + 1)\Delta t$, $0 < \Delta x \leq \Delta x_0$ e $0 < \Delta t \leq \Delta t_0$.

Na literatura existem outras formas de definir a estabilidade, porém todas as definições mantêm a essência do conceito de estabilidade. Uma outra definição comum é substituir a inequação (3.23) pela

$$\|U^{n+1}\| \leq K_T \|U^0\| \quad (3.24)$$

com ou sem a restrição de que $(n + 1)\Delta t \leq T$.

A definição de estabilidade de métodos numéricos é similar à definição de problema bem posto para PVI. De fato, veja a seguinte definição:

Definição 3.3.9 (Bem Posto) *O problema de valor inicial cuja a equação diferencial parcial de primeira ordem é dada por $\mathcal{L}u = 0$, é chamado de bem posto se para qualquer tempo $T \geq 0$, existe uma constante K_T tal que qualquer solução $u = u(x, t)$ satisfaz*

$$\int_{-\infty}^{+\infty} |u(x, t)|^2 dx \leq K_T \int_{-\infty}^{+\infty} |u(x, 0)|^2 dx, \quad \text{para } 0 \leq t \leq T.$$

Uma consequência importante da definição de estabilidade é que a equação (3.23) permite que o erro cometido pelo método numérico possa crescer com a evolução temporal, entretanto seu crescimento não excede a taxa de crescimento exponencial dada pela definição 3.3.8. Logo, uma caracterização de estabilidade com relação a este fato é muito comum e está representada pela seguinte proposição.

Proposição 3.3.1 *Um esquema numérico representado pela equação (3.22) é estável com relação norma $\|\cdot\|$ se, e somente se, existem constantes positivas Δx_0 , Δt_0 e constantes não negativas, K e β tal que*

$$\|Q^{n+1}\| \leq K e^{\beta t}$$

Demonstração:

A demonstração desta proposição segue diretamente da definição de norma de operadores, isto é,

$$\|T\| = \sup_{x \neq 0} \frac{\|Tx\|}{\|x\|},$$

do fato de que a equação (3.22) pode ser reescrita por $U^{n+1} = Q^{n+1}U^0$ e da definição 3.3.8. Para mais detalhes sobre a demonstração veja a referência [86] e para detalhes sobre espaços de operadores veja [46].

No caso em que a EDP é linear, a verificação da estabilidade de um método consistente com a EDP consiste em um trabalho mais simples de se fazer, do que analisar diretamente a convergência. Entretanto, no caso de EDP's não lineares e, também sistemas de EDP's, a análise de estabilidade torna-se difícil de se realizar. Porém, em ambas as situações o conceito de estabilidade tem mais recursos e resultados, quando comparado com o conceito de convergência, como veremos mais adiante.

Um importante teorema que relaciona as definições de convergência, consistência e estabilidade é enunciado a seguir.

3.3.4 Teorema de Lax-Richtmyer

Teorema 3.3.1 (Teorema de equivalência de Lax-Richtmyer) *Um esquema numérico consistente associado a uma equação diferencial parcial de um PVI bem posto é convergente se, e somente se, for estável. [82]*

Para demonstrar tal teorema, apresentaremos uma versão mais forte.

Teorema 3.3.2 (Teorema de Lax-Richtmyer) *Seja $U^{n+1} = QU^n + \Delta t G^n$ um esquema de ordem (p, q) na norma $\|\cdot\|$ de um problema de valor inicial linear e bem posto. Se o esquema é estável na norma $\|\cdot\|$, então o método numérico é convergente.*

Demonstração:

Seja $u = u(x, t)$ a solução exata do PVI. Como o esquema é consistente então

$$u^{n+1} = Qu^n + \Delta t G^n + \Delta t \tau^n,$$

com $\|\tau^n\| = \mathcal{O}(\Delta x^p) + \mathcal{O}(\Delta t^q)$. Seja $v^k = u^k - U^k$, então v^k satisfaz

$$v^{n+1} = Qv^n + \Delta t \tau^n = Q^{n+1}v^0 + \Delta t \sum_{i=0}^n Q^i \tau^{n-i}.$$

Como $v^0 = u^0 - U^0 = 0$, então

$$\|v^{n+1}\| \leq \Delta t \sum_{i=0}^n \|Q^i\| \|\tau^{n-i}\| .$$

Por hipótese o esquema é estável, logo satisfaz a proposição 3.3.1. Assim,

$$\|v^{n+1}\| \leq \Delta t K e^{\beta(n+1)\Delta t} \sum_{i=0}^n \|\tau^{n-i}\| .$$

Como o esquema é consistente, tem-se que

$$\|\tau^{n-i}\| = \mathcal{O}(\Delta x^p) + \mathcal{O}(\Delta t^q) ,$$

isto é $\|\tau^{n-i}\| \rightarrow 0$, quando $\Delta t \rightarrow 0$ e $\Delta x \rightarrow 0$.

Portanto, $\|u^{n+1} - U^{n+1}\| \rightarrow 0$, quando $\Delta t \rightarrow 0$ e $\Delta x \rightarrow 0$, ou seja, o esquema é convergente.

No caso de problemas de valor inicial com condições de contorno, apresentamos um teorema similar.

Teorema 3.3.3 (Teorema de Lax-Richtmyer) *Seja $U^{n+1} = QU^n + \Delta t G^n$ um esquema de ordem (p, q) em uma sequência de normas $\|\cdot\|_j$ de um problema de valor inicial linear e bem posto, com condições de contorno. Se o esquema é estável com relação a sequência de normas, então o método numérico é convergente.*

Um outro recurso para verificar a estabilidade de métodos numéricos envolve a aplicação da *análise de Fourier*, em estudo de condições necessárias e suficientes que resultam em esquemas estáveis. Tal uso é chamado de *análise de von Neumann* e sua aplicação requer que tanto o método numérico quanto a equação diferencial sejam lineares.

3.3.5 A análise de von Neumann

Para encontrar uma região que determina relações entre os espaçamentos Δx e Δt que mantêm o método numérico analisado estável, aplica-se a análise de estabilidade linear ou análise de von Neumann, cuja aplicabilidade depende da linearidade da equação diferencial. A ideia principal é representar a solução numérica por *séries de Fourier discretas*,

substituí-la no esquema numérico e avaliar os valores de Δx e Δt , que mantêm os modos de Fourier limitados.

Neste caso a semidiscretização da solução numérica é representada pela série de Fourier discreta no espaço [82, 86], isto é,

$$u_j(x, t) = \sum_{m=-N/2}^{N/2} \hat{u}_m(t) e^{i\omega_m j \Delta x}. \quad (3.25)$$

Pelo princípio de superposição [82, 86], podemos trabalhar apenas com um termo da série, então considera-se

$$u_j(x, t) = \hat{u}_m(t) e^{ij\theta_m}, \quad (3.26)$$

em que $\theta_m = \omega_m \Delta x$ e $m = -N/2, \dots, N/2$.

Se o operador espacial discreto L de um esquema numérico dado é escrito na forma de lei de conservação discreta, equação (3.17), então

$$L(u_{j-r}, \dots, u_{j+s}) = z(\theta_m). \quad (3.27)$$

Considerando uma semidiscretização temporal obtida de um esquema explícito, a solução no tempo $t_{n+1} = (n+1)\Delta t$ é representada por

$$u_j^{n+1} = g(\hat{z}_m) u_j^n, \quad \hat{z}_m = -\sigma z(\theta_m), \quad m = -N/2, \dots, N/2, \quad n \geq 1, \quad (3.28)$$

em que g é o fator de amplificação que depende de θ_m e

$$\sigma = \frac{\Delta t}{\Delta x}.$$

Assim, um método de discretização espacial acoplado em um método de discretização do tempo será estável se o fator de amplificação satisfizer a seguinte condição,

$$|g(-\sigma z(\theta))| \leq 1, \quad \forall \theta \in [0, 2\pi]. \quad (3.29)$$

Esta condição de estabilidade impõe um limite superior para σ que mantém o método

linearmente estável. Assim, se o domínio da variável espacial foi discretizado por um espaçamento regular Δx , a condição de estabilidade permite encontrar um valor para o espaçamento temporal Δt , que mantém o método estável.

O seguinte teorema garante o uso dos procedimentos descritos para investigar a estabilidade mostrando as condições necessárias e suficientes para se ter estabilidade de métodos numéricos.

Teorema 3.3.4 (Estabilidade Linear) *Um esquema de passo único com coeficientes constantes é estável se, e somente se, existe uma constante K (independente de θ , Δt e Δx) e, também constantes positivas Δx_0 e Δt_0 tal que*

$$|g(\theta, \Delta x, \Delta t)| \leq 1 + K\Delta t ,$$

para todo θ , $0 < \Delta t \leq \Delta t_0$ e $0 < \Delta x \leq \Delta x_0$. Se o fator de amplificação g é independente de Δx , então

$$|g(\theta)| \leq 1 .$$

A aplicação da análise de von Neumann é feita em equações diferenciais parciais lineares. Mais detalhes do estudo de estabilidade através da análise de von Neumann encontram-se em [82, 86]. Nessas referências também encontram-se detalhes para a análise de Fourier.

No caso em que a equação diferencial parcial ou o sistema de EDP's é hiperbólico, tem-se alguns resultados importantes que auxiliam na investigação da estabilidade de métodos numéricos. Uma propriedade que surge neste contexto é a *condição de Courant-Friedrichs-Lewy*, conhecida como *CFL*. A CFL de um método numérico é o valor máximo de σ que satisfaz o Teorema 3.3.4.

3.3.6 Estabilidade não linear

Em problemas em que a equação diferencial ou o sistema de equações são não lineares, dois caminhos são comumente realizados para fazer a análise de estabilidade: um realiza a

linearização das equações diferenciais e aplica a análise de von Neumann e o outro utiliza o conceito de Valor Total. Esta subsecção destina-se ao conceito de Valor Total Decrescente (TVD).

Definição 3.3.10 (Valor Total) *Seja $q : \mathbb{R} \rightarrow \mathbb{R}$, $q = q(x)$ uma função qualquer. O Valor Total de q é dado por*

$$TV(q) = \sup \sum_{i=1}^n |q(x_i) - q(x_{i-1})| \quad (3.30)$$

em que o supremo é tomado sobre todas as subdivisões da reta Real. Outra possível definição é

$$TV(q) = \limsup_{\epsilon \rightarrow 0} \left\{ \frac{1}{\epsilon} \int_{-\infty}^{\infty} |q(x) - q(x - \epsilon)| dx \right\}. \quad (3.31)$$

Se q é diferenciável, então

$$TV(q) = \int_{-\infty}^{\infty} |q'(x)| dx. \quad (3.32)$$

Se a função q estiver definida dentro de um intervalo particionado por $x_0 < x_1 < \dots < x_N$, então

$$TV(Q) = \sum_{i=1}^N |Q_i - Q_{i-1}|, \quad (3.33)$$

em que $Q_i = q(x_i)$.

Definição 3.3.11 (Valor Total Decrescente) *Um esquema é dito ser Valor Total Decrescente (TVD) se, os valores computados em quaisquer passos consecutivos t_n e t_{n+1} satisfizerem*

$$TV(Q^{n+1}) \leq TV(Q^n). \quad (3.34)$$

3.4 Dispersão e Dissipação

Independente da formulação clássica ou deslocada do método de Lax-Friedrichs, ambos os casos têm excessiva dissipação numérica para evoluções temporais que exigem muitas iterações. A *dissipação numérica* tem a propriedade de suavizar a solução numérica, o

que pode fazer com que a solução aproximada deixe de ter detalhes importantes para o entendimento do fenômeno em questão.

Um método que possui uma dissipação numérica menor do que o Lax-Friedrichs é o método Lax-Wendroff - seção 3.5.3. Com o objetivo de se obter uma maior ordem de precisão na variável temporal, o esquema de fato consegue reduzir a dissipação numérica, por outro lado, acrescenta *dispersão numérica* que pode causar oscilações na solução numérica. Conseqüentemente, o método deixa de ser estável e a convergência fica comprometida.

Quando deseja-se obter novos métodos numéricos, é necessário ter em mente que, mesmo satisfazendo o teorema de Lax que garante a convergência numérica, não é possível evitar totalmente a dissipação e dispersão numéricas devido as aproximações realizadas.

Exemplos com essas características estão na seção 3.7 e detalhes sobre dissipação e dispersão em métodos numéricos são obtidos nas referências [53, 86].

3.5 Métodos de Volumes Finitos Centrados

Na seção 3.2 foi descrita a formulação geral de métodos de volumes finitos unidimensionais. Na ocasião, obteve-se a seguinte equação exata

$$\begin{aligned} \frac{1}{\Delta x} \int_{\Omega_i} u(x, t_{n+1}) dx &= \frac{1}{\Delta x} \int_{\Omega_i} u(x, t_n) dx + \\ &- \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(u(x_{i+1/2}, t)) dt - \int_{t_n}^{t_{n+1}} f(u(x_{i-1/2}, t)) dt \right]. \end{aligned} \quad (3.35)$$

Se as integrais espaciais forem interpretadas como médias, a equação pode ser reescrita como

$$\bar{u}(x, t_{n+1}) = \bar{u}(x, t_n) - \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(u(x_{i+1/2}, t)) dt - \int_{t_n}^{t_{n+1}} f(u(x_{i-1/2}, t)) dt \right]. \quad (3.36)$$

Com a intenção de determinar valores para u nos pontos $x_{i+1/2}$'s, podemos aproximá-los por polinômios. Seja w uma aproximação polinomial por partes no tempo t_n da seguinte maneira

$$w(x, t_n) = \sum \omega_j(x) p_j(x) \quad (3.37)$$

em que o polinômio $p_j = p_j(x)$ está definido na célula $\Omega_j = (x_{j-1/2}, x_{j+1/2})$ e centrado no ponto médio x_j de Ω_j . Deseja-se que w seja uma solução aproximada de u .

Substituindo a função polinomial por partes na equação (3.36), obtêm-se

$$\bar{w}(x, t_{n+1}) = \bar{w}(x, t_n) - \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(w(x_{i+1/2}, t)) dt - \int_{t_n}^{t_{n+1}} f(w(x_{i-1/2}, t)) dt \right]. \quad (3.38)$$

Nota-se que os pontos $x_{i+1/2}$'s contêm descontinuidades, pois w é uma função polinomial por partes e cada polinômio p_j está definido no volume finito Ω_j .

Métodos de Volumes Finitos Centrados baseam-se no fato de que tais descontinuidades posicionam-se nos pontos médios de interlavos, para isto considera-se as células médias deslocadas de metade do tamanho do intervalo, isto é, para partições regulares as células médias estarão deslocadas de metade do espaçamento Δx . Contudo, os intervalos

$$I_i = (x_i, x_{i+1})$$

são as células médias deslocadas e cada uma delas tem definida a lei de conservação

$$\bar{w}(x_{i+1/2}, t_{n+1}) = \bar{w}(x_{i+1/2}, t_n) - \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(w(x_{i+1}, t)) dt - \int_{t_n}^{t_{n+1}} f(w(x_i, t)) dt \right]. \quad (3.39)$$

A equação (3.39) é uma lei de conservação em cada subintervalo $I_i \subset \Omega$. As células médias deslocadas permitem obter diretamente os fluxos nas interfaces, pois agora os extremos das células médias são os pontos x_i 's. Entretanto, surge uma questão referente ao valor médio $\bar{w}(x_{i+1/2}, t_n)$: como obtê-lo para determinar $\bar{w}(x_{i+1/2}, t_{n+1})$?

A resposta vem ao reescrever a equação (3.39) em termo dos valores \bar{w}_i^n e para fazer isso, temos dois procedimentos:

- **Primeiro**, utiliza-se os valores \bar{w}_i^n para obter $w(x, t_n)$ para todo $x \in \Omega$ através da

função polinomial por partes definida por

$$w(x, t_n) = \sum_i \omega_i(x) p_i(x), \quad \text{com } \bar{p}_i(x_i) = \bar{w}_i^n \text{ e } \omega_i = 1. \quad (3.40)$$

Assim, o valor médio de $w(x, t_n)$ definido na célula média deslocada $I_i = (x_i, x_{i+1})$ é dado por

$$\bar{w}_{i+1/2}^n = \frac{1}{\Delta x} \left[\int_{x_i}^{x_{i+1/2}} p_i(x) dx + \int_{x_{i+1/2}}^{x_{i+1}} p_{i+1}(x) dx \right]. \quad (3.41)$$

Substituindo a equação (3.41) na equação (3.39) obtém-se

$$\begin{aligned} \bar{w}_{i+1/2}^{n+1} &= \frac{1}{\Delta x} \left[\int_{x_i}^{x_{i+1/2}} p_i(x) dx + \int_{x_{i+1/2}}^{x_{i+1}} p_{i+1}(x) dx \right] + \\ &- \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(w(x_{i+1}, t)) dt - \int_{t_n}^{t_{n+1}} f(w(x_i, t)) dt \right]. \end{aligned} \quad (3.42)$$

- **Segundo**, realiza-se a evolução temporal de w_i , do tempo t_n até t_{n+1} através da equação diferencial que governa o fenômeno, no caso

$$w_t + f(w)_x = 0, \quad \text{com } w(x, t_n) = p_i(x) \text{ e } x \in I_i.$$

Esses procedimentos fazem parte de um algoritmo chamado de *REA - Reconstruct-evolve-average* que consiste dos passos de Reconstrução, Evolução e Atualização da célula Média. Essas ideias foram propostas por Godunov independentemente de contextualizar a evolução através de malha deslocada (Volumes Finitos Centrados) ou através da evolução de descontinuidades (Volumes Finitos dependente de Problema de Riemann).

Em alguns textos, métodos de volumes finitos que utilizam o algoritmo REA são chamados de métodos do tipo Godunov e diferenciam-se dos métodos de volumes centrados pelo fato de dependerem de resolvidores de Problemas de Riemann (chamados de Riemann Solvers), cuja evolução temporal toma como base a discretização Upwind.

Os métodos de volumes finitos centrado independem de Riemann Solvers (chamados de métodos centrados ou métodos livres de problemas de Riemann) cuja evolução base-a

em discretização Centrada [4, 68].

Já em outros textos, são chamados de métodos do tipo Godunov, apenas os métodos que utilizam resolvidores de Problemas de Riemann [52, 53, 88].

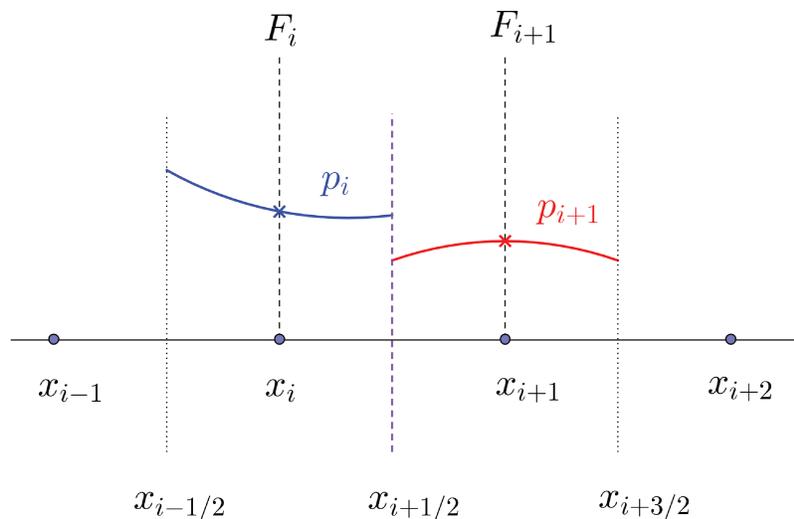


Figura 3.2: Representação gráfica de Métodos de Volumes Finitos Centrados. Cada polinômio p_i está definido em sua respectiva célula média e cada descontinuidade está localizada no ponto médio do intervalo I_i , isto é, no ponto x_i . Nota-se que os fluxos F_i são diretamente encontrados, pois a solução numérica é conhecida em todos os pontos x_i 's e esses fluxos estão representados por pontos \times .

Observações:

- 1) **Condição CFL:** As descontinuidades em métodos de volumes finitos centrados estão concentradas nos pontos $x_{i+1/2}$'s conforme representado na Figura 3.2. Tais descontinuidades possuem velocidades de propagação representadas pela letra S_k , que podem ser obtidas através dos autovalores da matriz Jacobiana. Ao realizar um passo temporal Δt de modo que a propagação das descontinuidades não atinjam os extremos de $I_i = (x_i, x_{i+1})$, os valores das grandezas definidas nos pontos x_i 's não serão afetados pelas descontinuidades que propagam-se no interior dos I_i 's. Assim, a evolução temporal das grandezas definidas nos pontos x_i 's permanecem livres de descontinuidades se o espaçamento temporal satisfizer

$$\Delta t \cdot S_i \leq \frac{1}{2} \Delta x ,$$

que, nada mais é que uma condição do tipo CFL, ver Figura 3.3. Note que esta é uma condição necessária para estabilidade do método numérico, porém não é uma condição suficiente. Métodos livres deste tipo são chamados de *métodos livres de solucionadores de problemas de Riemann*;

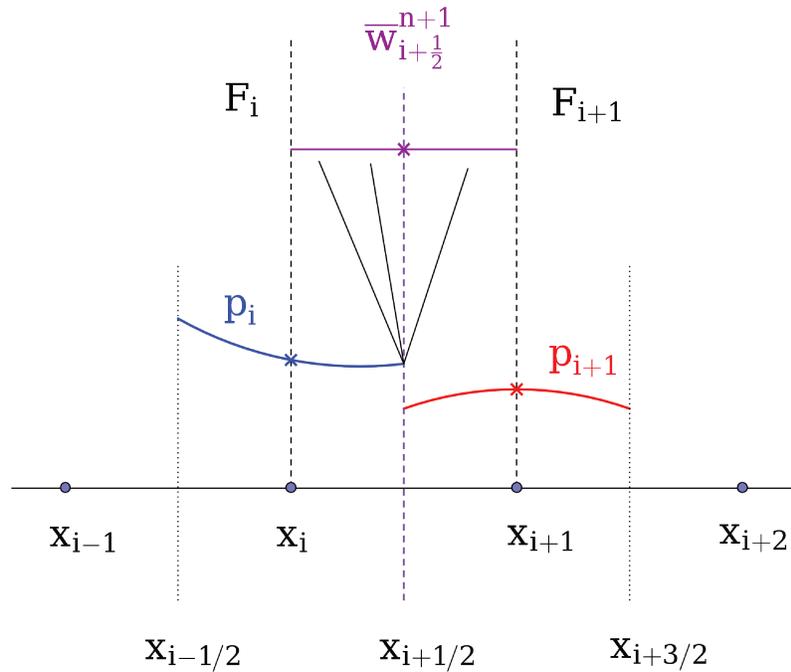


Figura 3.3: Representação de métodos de volumes finitos centrados em que a condição CFL é satisfeita, para que as evoluções de descontinuidades dentro de I_i não atinjam os extremos. As retas que saem da descontinuidade em $x_{i+1/2}$ representam a evolução temporal das descontinuidades de t_n até t_{n+1} . No caso, a solução numérica no tempo t_{n+1} é obtida em uma malha deslocada pela equação (3.42) através dos polinômios p_i e p_{i+1} definidos no tempo t_n .

2) Interpolação: Devido ao uso de interpolação polinomial por parte, tem-se

$$p_i(x_i) = U_i^n ;$$

3) Fluxos: Com a condição anterior satisfeita, o lado direito da equação (3.42) só envolve integrandos suaves;

4) Hiperbolicidade: No caso da equação diferencial ser hiperbólica, as velocidades das descontinuidades estão associadas aos autovalores da Matriz Jacobiana - definição

3.1.4, desse modo

$$S_i = \max_k |\lambda_k(U_i)|$$

em que $\lambda_k(U_i)$ são os autovalores da Matriz Jacobiana;

- 5) **Diferentes Métodos:** Basicamente, as diferenças entre os métodos de volumes finitos centrados estão nas escolhas dos polinômios p_i 's. Nota-se que não necessariamente precisamos ter sempre $\omega_i = 1$ [80].

Naturalmente, pode-se imaginar um método cujos polinômios p_i 's são constantes, isto é, com a função w constante por partes. Essa escolha recai ao conhecido método de Lax-Friedrichs, que está detalhado na próxima seção.

3.5.1 O método de Lax-Friedrichs

O Método Lax-Friedrichs no contexto de métodos de volumes finitos consiste em considerar w como uma função constante por partes - ver Figura 3.4, isto é,

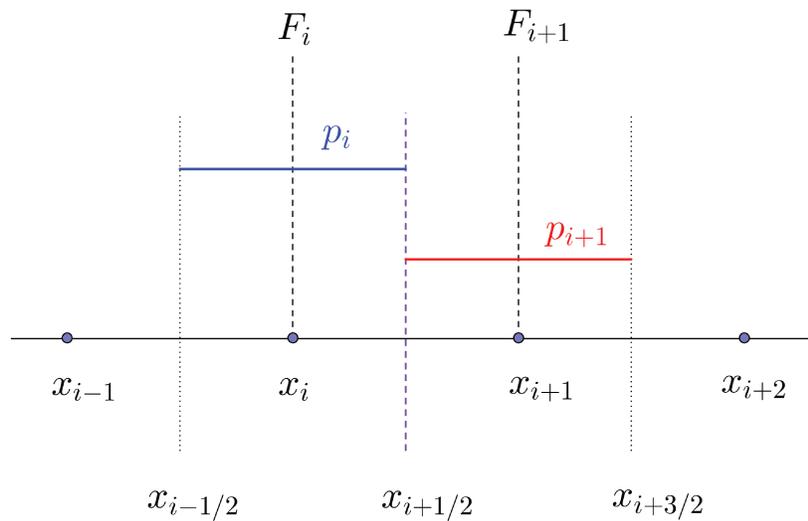


Figura 3.4: Representação gráfica do método de Lax-Friedrichs em que os polinômios p_i 's são constantes dentro da células médias Ω_i 's.

$$p_i(x) = U_i^n, \quad x \in (x_{i-1/2}, x_{i+1/2}),$$

em que $p_i(x) = U_i^0 = u(x, 0)$ com $x \in (x_{i-1/2}, x_{i+1/2})$, coincide com a condição inicial. Desse modo, a partir da equação (3.42) tem-se

$$\begin{aligned} \bar{w}_{i+1/2}^{n+1} &= \frac{1}{\Delta x} \left[\int_{x_i}^{x_{i+1/2}} U_i^n dx + \int_{x_{i+1/2}}^{x_{i+1}} U_{i+1}^n dx \right] + \\ &- \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(w(x_{i+1}, t)) dt - \int_{t_n}^{t_{n+1}} f(w(x_i, t)) dt \right]. \end{aligned}$$

Como os integrandos são suaves, podemos utilizar um método de integração para determinar tais integrais, assim aplicando a integração retangular que envolve pontos médios [47, 96] tem-se

$$\int_{t_n}^{t_{n+1}} f(w(x_{i+1}, t)) dt = \Delta t \cdot f(U_{i+1}^{n+1/2}) + \mathcal{O}(\Delta t^2)$$

e

$$\int_{t_n}^{t_{n+1}} f(w(x_i, t)) dt = \Delta t \cdot f(U_i^{n+1/2}) + \mathcal{O}(\Delta t^2),$$

esta integração também é conhecida como regra retangular do ponto médio ou simplesmente regra do ponto médio [96]. Consequentemente obtém-se,

$$U_{i+1/2}^{n+1} = \frac{1}{2} [U_{i+1}^n + U_i^n] - \frac{\Delta t}{\Delta x} [f(U_{i+1}^{n+1/2}) - f(U_i^{n+1/2})], \quad (3.43)$$

de ordem (2, 1).

A equação (3.43) possui termos que dependem do tempo intermediário $t_{n+1/2} = t_n + \frac{1}{2}\Delta t$ e para se obter a solução numérica em um tempo intermediário utiliza-se um *método do tipo predictor-corrector* ou pode-se expandir a solução em séries de Taylor para determinar uma aproximação no tempo intermediário. Contudo, definimos o método Lax-Friedrichs em malha deslocada.

Lax-Friedrichs deslocado:

$$U_{i+1/2}^{n+1} = \frac{1}{2} [U_i^n + U_{i+1}^n] - \frac{\Delta t}{\Delta x} [f(U_{i+1}^n) - f(U_i^n)]. \quad (3.44)$$

Se o objetivo é trabalhar apenas com os pontos da discretização da malha definida

pelos pontos da partição regular, um deslocamento da equação (3.44) pode ser realizado via séries de Taylor. O método Lax-Friedrichs sem malha deslocada pode ser obtido diretamente das séries de Taylor, através do Método de Diferenças Finitas [52, 53, 86, 82]. Este método é chamado de *Lax-Friedrichs Clássico* e quando aplicado em uma Lei de Conservação possui a seguinte expressão na forma conservativa.

Lax-Friedrichs Clássico:

$$U_i^{n+1} = \frac{1}{2} (U_{i-1}^n + U_{i+1}^n) - \frac{\Delta t}{2\Delta x} [F_{i+1/2} - F_{i-1/2}]$$

$$F_{i+1/2} = F(U_i^n, U_{i+1}^n) = \frac{\Delta x}{2\Delta t} (U_j^n - U_{j+1}^n) + \frac{1}{2} (f(U_j^n) + f(U_{j+1}^n)) . \quad (3.45)$$

Uma extensão natural do método de Lax-Friedrichs é considerar w linear por partes, isto é, os polinômios p_i 's sendo retas definidas em seus respectivos intervalos I_i . Na próxima seção destacamos tal método conhecido como método de Nessyahu-Tadmor.

3.5.2 O Método de Nessyahu-Tadmor

No esquema de Nessyahu-Tadmor tem-se w como uma função linear por partes [68]. Cada polinômio p_i é uma função linear definida em Ω_i que passa pelo ponto (x_i, \bar{w}_i^n) cuja inclinação é m_i , ver Figura 3.5. Desse modo, p_i possui a seguinte expressão

$$p_i(x) = \bar{w}_i^n + m_i (x - x_i) . \quad (3.46)$$

A inclinação da reta (3.46) é obtida numericamente pela série de Taylor ou pela interpolação linear,

$$w_x(x_i, t_n) = \frac{1}{\Delta x} (w_{i+1}^n - w_i^n) + \mathcal{O}(\Delta x) ,$$

assim,

$$m_i = \frac{1}{\Delta x} Dw_i , \quad (3.47)$$

em que Dw_i é uma variação conveniente da grandeza w relacionada com $(w_{i+1}^n - w_i^n)$.

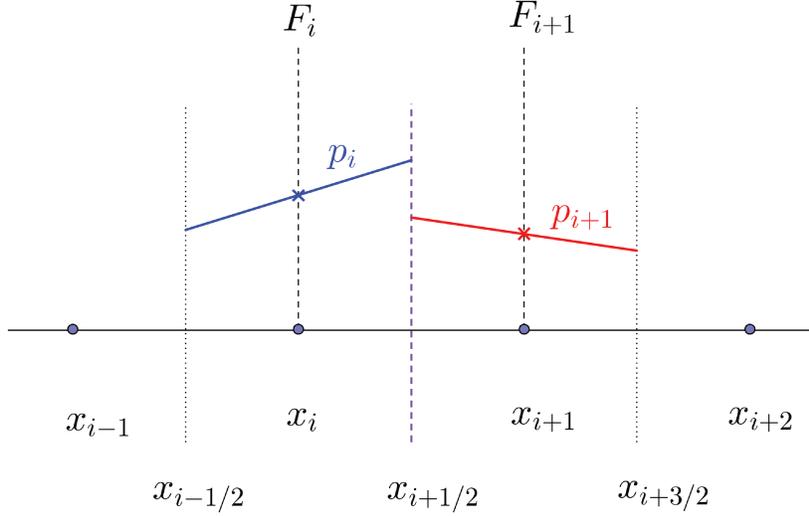


Figura 3.5: Representação gráfica do método de Nessyahu-Tadmor cujos polinômios p_i 's são retas dentro das células médias Ω_i que passam pelo ponto (x_i, \bar{w}_i^n) .

Substituindo a equação (3.47) em (3.46) tem-se a definição do polinômio p_i para ser substituído na equação (3.42). Fazendo este procedimento,

$$\begin{aligned} \bar{w}_{i+1/2}^{n+1} &= \frac{1}{\Delta x} \left\{ \int_{x_i}^{x_{i+1/2}} \left[\frac{Dw_i}{\Delta x} (x - x_i) + \bar{w}_i^n \right] dx + \int_{x_{i+1/2}}^{x_{i+1}} \left[\frac{Dw_{i+1}}{\Delta x} (x - x_{i+1}) + \bar{w}_{i+1}^n \right] dx \right\} + \\ &\quad - \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(w(x_{i+1}, t)) dt - \int_{t_n}^{t_{n+1}} f(w(x_i, t)) dt \right]. \end{aligned} \quad (3.48)$$

Resolvendo as integrais espaciais da equação (3.48), obtém-se

$$\begin{aligned} \bar{w}_{i+1/2}^{n+1} &= \frac{1}{2} (\bar{w}_{i+1}^n + \bar{w}_i^n) + \frac{1}{8} (Dw_i - Dw_{i+1}) \\ &\quad - \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(w(x_{i+1}, t)) dt - \int_{t_n}^{t_{n+1}} f(w(x_i, t)) dt \right]. \end{aligned} \quad (3.49)$$

Igualmente no método de Lax-Friedrichs, podemos resolver as integrais temporais utilizando o método retangular de integração numérica com relação ao ponto médio - veja seção 3.5.1. Assim, a equação (3.49) reduz-se à

$$\bar{w}_{i+1/2}^{n+1} = \frac{1}{2} (\bar{w}_{i+1}^n + \bar{w}_i^n) + \frac{1}{8} (Dw_i - Dw_{i+1}) - \frac{\Delta t}{\Delta x} \left[f(U_{i+1}^{n+1/2}) - f(U_i^{n+1/2}) \right]. \quad (3.50)$$

Novamente, a equação (3.50) depende do valor da solução numérica em um passo intermediário no tempo. Utilizando a série de Taylor e a lei de conservação, segue que

$$w(x_i, t_{n+1/2}) = w(x_i, t_n) + \frac{\Delta t}{2} w_t + \mathcal{O}(\Delta t^2), \quad (3.51)$$

em que $w_t = -f(w)_x$.

Contudo, obtém-se o seguinte método do tipo preditor-corretor,

$$U_{i+1/2}^{n+1} = \frac{1}{2} (U_{i+1}^n + U_i^n) + \frac{1}{8} (DU_i - DU_{i+1}) - \frac{\Delta t}{\Delta x} \left[f(U_{i+1}^{n+1/2}) - f(U_i^{n+1/2}) \right]$$

com $U_i^{n+1/2} = U_i^n - \frac{\Delta t}{\Delta x} Df(U_i^n).$

(3.52)

Para se chegar no método de Nessyahu e Tadmor [68], uma discussão sobre a variação Dw_i é necessária pois a simples definição $Dw_i = (w_{i+1}^n - w_i^n)$ pode ser desastrosa no sentido de produzir métodos extremamente oscilatórios.

A variação Dw_i está diretamente relacionada com a inclinação da reta

$$p_i(x) = \bar{w}_i^n + \frac{Dw_i}{\Delta x} (x - x_i)$$

e para se obter um método de segunda ordem não oscilatório, a inclinação $\frac{Dw_i}{\Delta x}$ deve ser uma aproximação para a derivada espacial de w , ou seja, a seguinte condição deve ser satisfeita:

$$\frac{1}{\Delta x} Dw_i = \frac{\partial}{\partial x} w(x_i, t_n) + \mathcal{O}(\Delta x). \quad (3.53)$$

A equação (3.53) fornece restrições às retas p_i 's, as inclinações das retas p_i 's devem ser uma aproximação para a derivada espacial de w , inclusive com mesmo sinal de inclinação.

Para garantir tais restrições e ainda manter o método não oscilatório [68], a derivada numérica deve satisfazer

$$0 \leq Dw_i \cdot \text{sinal}(\Delta w_{i\pm 1/2}) \leq \alpha |MM\{\Delta w_{i+1/2}, \Delta w_{i-1/2}\}| . \quad (3.54)$$

Assim, basta tomar

$$Dw_i = MM\{\Delta w_{i+1/2}, \Delta w_{i-1/2}\} , \quad (3.55)$$

em que a operação $MM\{.,.\}$ é a usual função minmod definida por

$$MM\{x, y\} = \text{minmod}\{x, y\} = \frac{1}{2} [\text{sinal}(x) + \text{sinal}(y)] \min\{|x|, |y|\} \quad (3.56)$$

e $\Delta w_{i+1/2} = w_{i+1} - w_i$.

A escolha pela equação (3.55) não é única, note que a equação tem uma constante α . Para cada valor de $\alpha \in (0, 2)$ tem-se uma função limitadora de inclinação que resulta em um método não oscilatório [53, 68, 88].

Portanto, chega-se na seguinte expressão para o método Nessyahu-Tadmor em malha deslocada.

Nessyahu-Tadmor deslocado:

$$\begin{aligned} U_{i+1/2}^{n+1} &= \frac{1}{2} (U_{i+1}^n + U_i^n) + \frac{1}{8} (DU_i - DU_{i+1}) - \frac{\Delta t}{\Delta x} \left[f(U_{i+1}^{n+1/2}) - f(U_i^{n+1/2}) \right] \\ \text{com } U_i^{n+1/2} &= U_i^n - \frac{\Delta t}{\Delta x} Df(U_i^n) , \quad DU_i^n = MM\{\Delta U_{i+1/2}^n, \Delta U_{i-1/2}^n\}, \\ Df(U_i^n) &= MM\{\Delta f(U_{i+1/2}^n), \Delta f(U_{i-1/2}^n)\} , \quad \Delta U_{i+1/2}^n = U_{i+1}^n - U_i^n, \\ \text{e } \Delta f(U_{i+1/2}^n) &= f(U_{i+1}^n) - f(U_i^n) \end{aligned} \quad (3.57)$$

Caso seja mais conveniente trabalhar com a malha sem deslocamento, o método Nessyahu-Tadmor pode ser reescrito da seguinte maneira [68]:

Nessyahu-Tadmor sem deslocamento:

$$U_i^{n+1} = \frac{1}{2} (U_{i+1}^n + U_{i-1}^n) + \frac{1}{4} (DU_{i-1} - DU_{i+1}) - \frac{\Delta t}{2\Delta x} \left[f(U_{i+1}^{n+1/2}) - f(U_{i-1}^{n+1/2}) \right]$$

$$\text{com } U_i^{n+1/2} = U_i^n - \frac{\Delta t}{\Delta x} Df(U_i^n), \quad DU_i^n = MM\{\Delta U_{i+1/2}^n, \Delta U_{i-1/2}^n\},$$

$$Df(U_i^n) = MM\{\Delta f(U_{i+1/2}^n), \Delta f(U_{i-1/2}^n)\}, \quad \Delta U_{i+1/2}^n = U_{i+1}^n - U_i^n,$$

$$\text{e } \Delta f(U_{i+1/2}^n) = f(U_{i+1}^n) - f(U_i^n) \tag{3.58}$$

Observações:

- 1) Note que, se no método Nessyahu-Tadmor os polinômios não tiverem inclinação, isto é, se a inclinação for nula, então $DU_i = 0$ e $Df(U_i) = 0$. Logo, ao substituir nas equações (3.57) e (3.58), os métodos de Lax-Friedrichs são resgatados;
- 2) A condição CFL descrita no Lax-Friedrichs permanece a mesma da seção 3.5,

$$\Delta t \cdot S_i \leq \frac{1}{2} \Delta x .$$

Entretanto, a referência [68] desenvolve um procedimento para encontrar S_i sem calcular os autovalores da matriz Jacobiana;

- 3) O método Nessyahu-Tadmor pode utilizar uma função minmod mais geral, que envolve $\alpha \in (0, 2)$, do que a função minmod exposta na equação (3.55). A partir da equação (3.54), obtém-se a seguinte função limitadora de inclinação,

$$Dw_i = MM \left\{ \alpha (\bar{w}_{i+1}^n - \bar{w}_i^n), \frac{1}{2} (\bar{w}_{i+1}^n - \bar{w}_{i-1}^n), \alpha (\bar{w}_i^n - \bar{w}_{i-1}^n) \right\}, \tag{3.59}$$

em que

$$MM\{x_1, x_2, \dots, x_n\} = \begin{cases} \min_i\{x_i\}, & \text{se } x_i > 0, \forall 1 \leq i \leq n, \\ \max_i\{x_i\}, & \text{se } x_i < 0, \forall 1 \leq i \leq n, \\ 0, & \text{caso contrário.} \end{cases}$$

- 4) A equação (3.59) faz uma escolha entre uma aproximação avançada, uma centrada e uma atrasada. Em qualquer uma das escolhas pela função minmod, a seguinte condição é satisfeita

$$\frac{Dw_i}{\Delta x} = \frac{\partial}{\partial x} w(x_i, t_n) + \mathcal{O}(\Delta x),$$

conforme a equação (3.53).

- 5) A função limitadora de inclinação minmod garante que o método Nessyahu-Tadmor é TVD se a seguinte condição CFL for satisfeita,

$$\frac{\Delta t}{\Delta x} \left| \frac{f(U_{i+1}^{n+1/2}) - f(U_i^{n+1/2})}{U_{i+1}^{n+1/2} - U_i^{n+1/2}} \right| \leq \beta$$

em que $\beta \leq \frac{1}{2\alpha} (\sqrt{4 + 4\alpha - \alpha^2} - 2)$ [68]. Assim para a igualdade satisfeita, tem-se o gráfico da Figura 3.6;

- 6) A ordem do erro de truncamento local do método Nessyahu-Tadmor é dado por

$$\|\tau^n\| = \mathcal{O}(\Delta x^3) + \mathcal{O}(\Delta t^2),$$

ver referência [68] para detalhes.

3.5.3 O método de Lax-Wendroff

Na literatura, um método que vem subsequente ao método Lax-Friedrichs, com o intuito de aumentar a ordem de precisão da discretização temporal é o método Lax-Wendroff [86]. Tal método acrescenta condições no método de diferenças finitas centrado no espaço

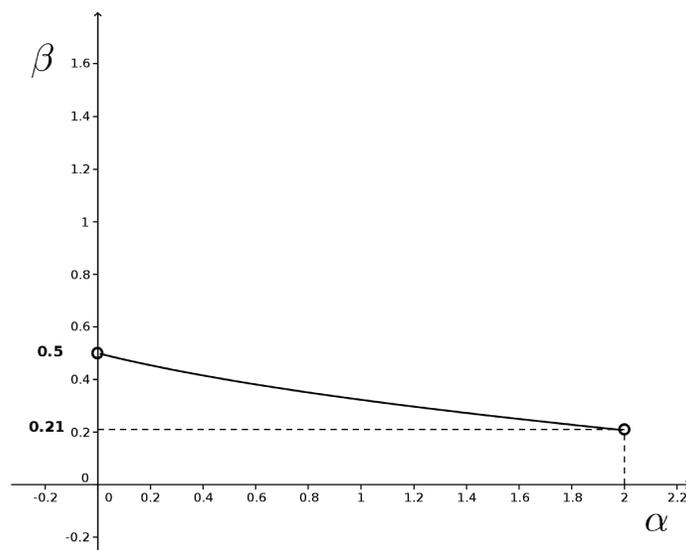


Figura 3.6: Gráfico de β em função de α cujos valores estão relacionados com a condição CFL e a função *minmod*.

e avançado no tempo na tentativa de estabilizá-lo. Uma formulação do método Lax-Wendroff do tipo preditor-corretor conhecido como Lax-Wendroff-Richtmyer para leis de conservação encontra-se na referência [53] e, a seguir, temos a definição do método.

Lax-Wendroff-Richtmyer:

$$U_{i+1/2}^{n+1/2} = \frac{1}{2} (U_i^n + U_{i+1}^n) - \frac{\Delta t}{2\Delta x} [f(U_{i+1}^n) - f(U_i^n)]$$

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x} (f(U_{i+1/2}^{n+1/2}) - f(U_{i-1/2}^{n+1/2})) .$$

O método Lax-Wendroff é dispersivo [53, 86], isto é, acrescenta oscilações na solução numérica que, conforme ocorre evolução temporal que podem crescer a ponto de tornar o método instável. Assim, na tentativa de controlar tais oscilações realiza-se uma semidiscretização espacial e depois, a integração temporal é efetuada por um método Runge-Kutta com a propriedade de ser TVD.

Lax-Wendroff-Richtmyer com Runge-Kutta 3 - TVD

Semi-discretização:

$$\frac{d}{dt} u_i = -\frac{1}{\Delta x} [F(U)_{i+1/2} - F(U)_{i-1/2}] = L(U) ,$$

em que $L(U)$ é a discretização do operador espacial.

Operador espacial (discretizado):

$$L(U) = -\frac{1}{\Delta x} \left(f(U_{i+1/2}^{n+1/2}) - f(U_{i-1/2}^{n+1/2}) \right) .$$

Runge-Kutta de ordem 3 (TVD):

$$\begin{aligned} u^{(1)} &= u^n + \Delta t L(U) \\ u^{(2)} &= \frac{3}{4}u^n + \frac{1}{4}u^{(1)} + \frac{1}{4}\Delta t L(u^{(1)}) \quad . \\ u^{n+1} &= \frac{1}{3}u^n + \frac{2}{3}u^{(2)} + \frac{2}{3}\Delta t L(u^{(2)}) \end{aligned}$$

A propriedade TVD do método Runge-Kutta definido com o Lax-Wendroff pode ser verificada na referência [67].

3.6 Métodos de Volumes Finitos do tipo Godunov

Nesta seção estão expostos alguns métodos de volumes finitos que trabalham diretamente com a célula média $\Omega_i = (x_{i-1/2}, x_{i+1/2})$ sem deslocá-la. Consequentemente, um ingrediente a mais é acrescentado ao método de volumes finitos, os chamados resolvidores de Problemas de Riemann nas interfaces $x_{i+1/2}$'s. Em cada ponto $x_{i+1/2}$ tem-se um conjunto de descontinuidades definidas e necessárias de serem resolvidas analiticamente ou numericamente, com o intuito de encontrar o fluxo numérico $F_{i+1/2}$ e só assim conseguir obter a solução numérica da lei de conservação em um passo consecutivo.

Para obter métodos numéricos com tais características voltemos na seção 3.2 onde foi descrita a formulação geral de métodos de volumes finitos unidimensionais. Na ocasião, obteve-se a seguinte equação exata

$$\begin{aligned} \frac{1}{\Delta x} \int_{\Omega_i} u(x, t_{n+1}) dx &= \frac{1}{\Delta x} \int_{\Omega_i} u(x, t_n) dx + \\ &- \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(u(x_{i+1/2}, t)) dt - \int_{t_n}^{t_{n+1}} f(u(x_{i-1/2}, t)) dt \right]. \end{aligned} \quad (3.60)$$

Interpretando as integrais espaciais como médias, a equação pode ser reescrita como

$$\bar{u}(x, t_{n+1}) = \bar{u}(x, t_n) - \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(u(x_{i+1/2}, t)) dt - \int_{t_n}^{t_{n+1}} f(u(x_{i-1/2}, t)) dt \right]. \quad (3.61)$$

Com a intenção de determinar valores para u nos pontos $x_{i+1/2}$'s, podemos aproximá-lo por polinômios. Seja w uma aproximação polinomial por partes no tempo t_n da seguinte maneira

$$w(x, t_n) = \sum \omega_j(x) p_j(x) \quad (3.62)$$

em que o polinômio $p_j = p_j(x)$ está definido na célula $\Omega_j = (x_{j-1/2}, x_{j+1/2})$ e centrado no ponto médio x_j de Ω_j . Deseja-se que w seja uma solução aproximada de u .

Substituindo a função polinomial por partes na equação (3.61), obtém-se

$$\bar{w}(x, t_{n+1}) = \bar{w}(x, t_n) - \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(w(x_{i+1/2}, t)) dt - \int_{t_n}^{t_{n+1}} f(w(x_{i-1/2}, t)) dt \right]. \quad (3.63)$$

Nota-se que nos pontos $x_{i+1/2}$'s há descontinuidades, pois w é uma função polinomial por partes e cada polinômio p_j está definido no volume finito Ω_j .

Métodos de Volumes Finitos do tipo Godunov basea-se no fato de que as descontinuidades estão posicionadas nos extremos do volume de controle, isto é, no caso unidimensional as descontinuidades estão nos extremos das células médias $\Omega_i = (x_{i-1/2}, x_{i+1/2})$, onde aplicam-se as leis de conservação em questão. Assim, sabendo o comportamento dos fluxos $f(w(x, t))$ nos pontos $x_{i+1/2}$'s, a variação da grandeza em estudo com relação a variável temporal é obtida diretamente pela lei de conservação e, então, a evolução temporal

pode ser realizada.

Neste contexto, uma pergunta que norteia métodos de volumes finitos do tipo Godunov é a seguinte: Como obter $f(w(x, t))$ nos pontos $x_{i+1/2}$'s se apenas conhecemos $w(x, t)$ nos pontos x_i ? A resposta vem do método original proposto por Godunov [29].

3.6.1 Método de Godunov de primeira ordem

O método de Godunov de primeira ordem considera o polinômio w sendo constante por partes, isto é,

$$p_i(x) = U_i^n, \quad x \in (x_{i-1/2}, x_{i+1/2}),$$

conforme a Figura 3.7.

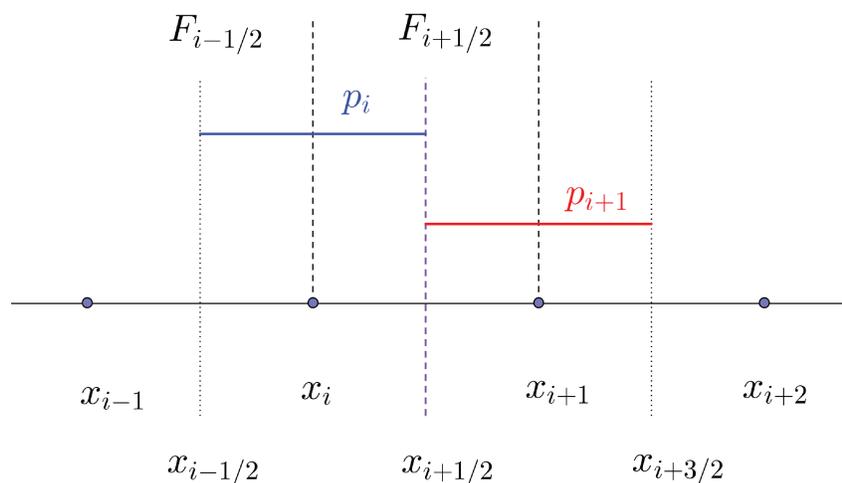


Figura 3.7: Representação gráfica do método Godunov de primeira ordem cujos polinômios p_i 's são constantes dentro da células médias Ω_i 's. Em cada intervalo $I_i = (x_i, x_{i+1})$ tem-se um problema de Riemann local.

Ao considerar w constante por partes tem-se $\bar{w}_i^n = U_i^n$ e, em cada intervalo I_i definimos um problema de Riemann local dado por

$$\begin{cases} w_t + f(w)_x = 0, & x \in (x_i, x_{i+1}) \\ w(x, t_n) = w_n(x) \end{cases} \quad (3.64)$$

em que

$$w_n(x) = \begin{cases} p_i(x) = U_i^n, & \text{se } x < x_{i+1/2} \\ p_{i+1}(x) = U_{i+1}^n, & \text{se } x > x_{i+1/2} \end{cases} . \quad (3.65)$$

Em coordenadas locais, muitas vezes o problema de Riemann em cada I_i é representado por:

$$\begin{cases} w_t + f(w)_\xi = 0, & \xi \in (-\Delta x, \Delta x) \\ w(\xi, 0) = w_0(\xi) \end{cases} \quad (3.66)$$

em que

$$w_0(\xi) = \begin{cases} U_L, & \text{se } \xi < 0 \\ U_R, & \text{se } \xi > 0 \end{cases} . \quad (3.67)$$

Representando a solução do problema de Riemann por \tilde{w} , a equação (3.63) torna-se

$$\bar{w}(x, t_{n+1}) = \bar{w}(x, t_n) - \frac{1}{\Delta x} \left[\int_{t_n}^{t_{n+1}} f(\tilde{w}(x_{i+1/2}, t)) dt - \int_{t_n}^{t_{n+1}} f(\tilde{w}(x_{i-1/2}, t)) dt \right] . \quad (3.68)$$

A equação (3.68) pode ser reescrita na forma conservativa conforme a definição 3.2.2, então tomando $x = x_i$

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x} [F_{i+1/2} - F_{i-1/2}] , \quad (3.69)$$

em que

$$U_i^n = \bar{w}(x_i, t_n) \cong \int_{\Omega_i} u(x, t_n) dx \quad \text{e} \quad F_{i+1/2} \cong \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} f(\tilde{u}(x, t)) dt .$$

Independente da utilização de solucionadores de problemas de Riemann analíticos ou numéricos, em ambos os casos são necessários conhecer ou estimar as velocidades de propagação das descontinuidades definidas pelos problemas de Riemann. Em sistemas de equações diferenciais em forma de Leis de Conservação - definição 3.1.3, as velocidades de propagação das descontinuidades estão associadas aos autovalores da matriz Jacobiana - definição 3.1.4 do seguinte problema quase linear:

$$U_t + \mathcal{A}(U).U_x = 0 ,$$

em que \mathcal{A} é a matriz Jacobiana.

Assim, métodos do tipo Godunov ou métodos Upwind do tipo Godunov necessitam de uma ferramenta a mais para se obter soluções numéricas de Leis de Conservação. Como esses métodos são dependentes de Resolucionadores de Problemas de Riemann, eles dependem da obtenção do conjunto de autovalores da matriz Jacobiana.

Estudos analíticos relacionados às velocidades de propagação de discontinuidades e sobre sistema de autovalores encontram-se nas seguintes referências: equação de Burgers [22, 35, 53, 56, 83, 88], equações de Euler - Hidrodinâmica Ideal [52, 53, 88], Magneto-hidrodinâmica [11, 90] e em fluidos relativísticos [2, 25].

Nesses estudos tais velocidades estão associadas a um conjunto de ondas chamadas de *ondas elementares*. No caso do sistema de leis de conservação ser estritamente hiperbólico, todos os autovalores são distintos e, então, todas as ondas elementares são distinguíveis. No caso de algum autovalor possuir degenerescência, a resolução de problemas de Riemann torna-se mais trabalhosa. Para problemas que envolvem Relatividade Especial ou Relatividade Geral, existem autovalores associados a ondas não físicas o que acentua ainda mais a dificuldade, pois estas ondas precisam ser removidas para se resolver os problemas de Riemann [41, 43]. Estudos sobre problemas de Riemann em Magnetohidrodinâmica Relativística encontram-se na referência [74] e ainda continuam em desenvolvimento.

Até mesmo os resolucionadores numéricos de problemas de Riemann, também chamados de problemas de Riemann aproximados, possuem dificuldades quando os autovalores não são todos distintos e/ou quando trata-se de GRMHD [25].

Uma das finalidades de resolver os problemas de Riemann é construir o fluxo numérico $F_{i+1/2}$ da equação (3.69) através dos autovalores de \mathcal{A} . Assim além de se resolver os problemas de Riemann, o fluxo numérico $F_{i+1/2}$ deve ser uma boa aproximação do fluxo $f(u(x_{i+1/2}, t))$ da lei de conservação $u_t + f(u)_x = 0$.

Diferentes fluxos numéricos caracterizam um método numérico dentro da classe de métodos do tipo Godunov. Logo, diferentes Riemann Solvers estão associados a diferentes fluxos numéricos. Exemplos mais comuns de resolucionadores de problemas de Riemann

são: de Roe [73], de Osher [21], de Harten-Lax-Leer (HLL) [31], HLL com Contato (HLLC) de Toro [88] e de Rusanov [76].

Observações:

1) **Condição CFL:** As descontinuidades em métodos de volumes finitos do tipo Godunov estão concentradas nos pontos $x_{i+1/2}$'s conforme representado na Figura 3.8. Tais descontinuidades possuem velocidades de propagação representadas pela letra S_k , que podem ser obtidas através dos autovalores da matriz Jacobiana. Ao realizar um passo temporal Δt de modo que a propagação das descontinuidades não atinja os pontos médios x_i 's, definidos por cada célula métrica Ω_i , os valores das grandezas definidas nos pontos x_i 's não serão afetados pelas descontinuidades que propagam-se no interior dos Ω_i 's. Assim, a evolução temporal das grandezas definidas nos pontos x_i 's não são afetadas pelos fluxos definidos nas extremidades de Ω_i se satisfazem a seguinte condição de CFL

$$\Delta t \cdot S_i \leq \frac{1}{2} \Delta x ,$$

ver Figura 3.8;

2) **Interpolação:** Devido ao uso de interpolação polinomial por partes, a consistência e consequentemente o erro de truncamento local são obtidos diretamente do erro de interpolação polinomial;

3) **Fluxos:** Em cada interface $x_{i+1/2}$ é necessário resolver um problema de Riemann para depois obter o fluxo $F_{i+1/2}$. Note que a evolução de cada solução numérica U_i^n depende dos fluxos $F_{i+1/2}$ e $F_{i-1/2}$, portanto dois problemas de Riemann são necessários para fazer a evolução temporal de U_i^n até U_i^{n+1} ;

4) **Hiperbolicidade estrita:** No caso do sistema de m leis de conservação ser estritamente hiperbólico, a matriz \mathcal{A} é diagonalizável. Logo, o sistema não linear pode ser substituído por um sistema quase linear:

$$\partial_t \mathbf{U} + \partial_x F(\mathbf{U}) = 0 \Leftrightarrow \partial_t \mathbf{W} + \Lambda \mathbf{W}_x = 0 ,$$

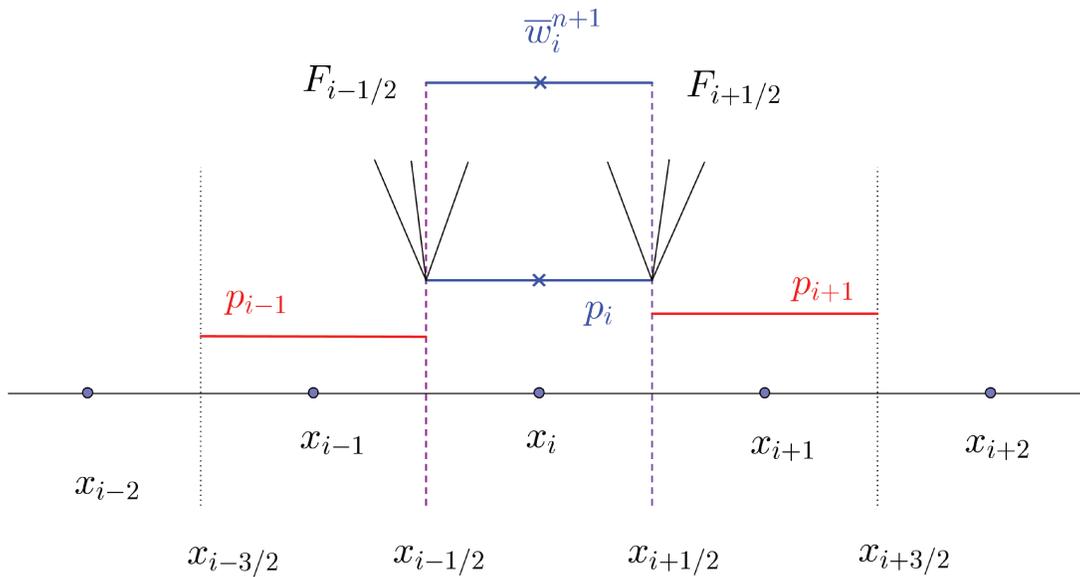


Figura 3.8: Representação de métodos de volumes finitos do tipo Godunov de primeira ordem, cuja função w é constante por partes e está representada pelos p_i 's. As evoluções das discontinuidades estão representadas nas interfaces $x_{i+1/2}$'s e para se obter os fluxos $F_{i+1/2}$'s são necessários resolver problemas de Riemann. O ponto representado por \times representa onde temos a solução numérica $\bar{w}_i^{n+1} = U_i^{n+1}$.

em que $\mathcal{A} = K\Lambda K^{-1}$, $K = [K^{(1)} | \dots | K^{(m)}]$, $\mathcal{A}K^{(i)} = \lambda_i K^{(i)}$, $\mathcal{A} = \text{diag}(\lambda_i)$ matriz diagonal e $\mathbf{U} = K\mathbf{W}$;

- 5) **Captura de Choques:** Devido ao uso dos autovalores, os métodos do tipo Godunov conseguem ter um melhor acompanhamento das descontinuidades quando comparados aos métodos livres de problemas de Riemann. Entretanto, conhecer os autovalores aumenta o custo computacional do método e restringe a aplicabilidade dos métodos tornando-os menos robustos quando comparados aos métodos centrados.
- 6) **Ondas Elementares:** Ao se utilizar um método de Godunov que necessita de resolvidores de problemas de Riemann, as estruturas físicas das ondas que se propagam no fluido são conhecidas, devido aos autovalores e autovetores e essas informações podem ser utilizadas para uma melhor descrição e entendimento físico dos fenômenos em estudo.

3.7 Comparações

Nesta seção são feitas comparações entre os métodos de volumes finitos expostos neste capítulo. O problema escolhido foi o escoamento de um fluido, dado pelas equações de Euler Unidimensional,

$$\frac{\partial}{\partial t}\rho + \frac{\partial}{\partial x}(\rho v) = 0 \quad (\text{equação da continuidade}) \quad (3.70)$$

$$\frac{\partial}{\partial t}\rho v + \frac{\partial}{\partial x}(\rho v^2 + p) = 0 \quad (\text{equação do movimento}) \quad (3.71)$$

$$\frac{\partial}{\partial t}E + \frac{\partial}{\partial x}(v(E + p)) = 0 \quad (\text{equação da energia}) \quad (3.72)$$

em que

$\rho = \rho(x, t)$	(densidade do fluido em escoamento)
$v = v(x, t)$	(velocidade do fluido)
$p = p(x, t)$	(distribuição de pressão no fluido)
$E = E(x, t)$	(energia total do fluido)
$\epsilon = \epsilon(x, t)$	(energia interna do fluido)
γ	(calor específico do fluido)

com

$$E = \frac{1}{2} (\rho v^2 + \epsilon) \quad \text{e} \quad \epsilon = \frac{p}{(\gamma - 1)\rho}.$$

Note que as equações de Euler estão escritas na forma diferencial de Leis de Conservação, definição 3.1.3. Dessa maneira, ao realizar a discretização de tais equações por algum método descrito nas Seções 3.5 e 3.6, o método numérico pode ser escrito na forma conservativa, isto é, conforme a definição 3.2.1.

3.7.1 Recursos Computacionais

As simulações computacionais foram realizadas via terminal no console do Octave versão 3.6.4 em um Linux/GNU 64bits de kernel 3.8.0 – 30 com processador intel i7 de 2,9GHz com 4 núcleos de processamentos, 6 GbRAM DDR 3 e bogomips 5786,81 (informação extraída do *cpuinfo*).

Implementamos códigos dos métodos Lax-Friedrichs, Lax-Wendroff's, Nessyahu-Tadmor e Godunov-HLL neste ambiente para obter simulações computacionais dos exemplos a seguir, que envolvem as equações de Euler que modelam escoamento de fluidos.

O método Lax-Friedrichs implementado foi o esquema que não tem a malha deslocada, o método Lax-Friedrichs Clássico, equação (3.45). Também, implementamos os métodos Lax-Wendroff-Richtmyer e o Lax-Wendroff-Richtmyer para discretização espacial e o esquema Runge-Kutta de terceira ordem TVD na discretização temporal. O método Nessyahu-Tadmor implementado foi o esquema com malha deslocada, equação (3.57) e, por fim, um representante dos esquema dependentes de solucionadores de problemas de

Riemann - o método Godunov-HLL, seção 3.7 [88].

No método Nessyahu-Tadmor com malha deslocada utilizamos a função *minmod* dada pela equação (3.56).

3.7.2 Exemplo 1 - problema de Sod

Considere um exemplo de escoamento de fluido unidimensional conhecido como problema teste de Sod [81]. Neste exemplo tem-se a formação das três classes de ondas elementares: ondas de choque, ondas de rarefação e ondas de contato. Conseqüentemente, o problema incorpora as dificuldades computacionais encontradas quando buscamos realizar implementações de métodos numéricos, com o intuito de se obter soluções em situações em que a solução analítica possui descontinuidades.

O problema de Sod consiste das seguintes condições iniciais:

$$\rho(x, 0) = \begin{cases} 1, 0 & \text{se } x < 0 \\ 0, 125 & \text{se } x > 0 \end{cases}, \quad v(x, 0) = \begin{cases} 0, 0 & \text{se } x < 0 \\ 0, 0 & \text{se } x > 0 \end{cases},$$

e

$$p(x, 0) = \begin{cases} 1, 0 & \text{se } x < 0 \\ 0, 1 & \text{se } x > 0 \end{cases}. \quad (3.73)$$

Neste exemplo, consideramos $\gamma = 1, 4$ e para as simulações representadas nas Figuras 3.11 - 3.22 utilizamos os dados da Tabela 3.1. As condições iniciais foram plotadas nas Figuras 3.9 e 3.10. Em todos os gráficos de soluções numéricas apresentamos a solução analítica de referência em cor preta.

Tabela 3.1: Dados computacionais para a obtenção das soluções do problema de Sod

Domínio	Subintervalos	Δx	CFL	Δt	passos de tempo	t_f
$[-2, 2]$	400	0,01	0,1	0,001	800	0,8

Na tabela 3.2 encontram-se informações sobre tempo de execução de cada método numérico.

Utilizando os dados da Tabela 3.1 obtivemos soluções numéricas dos métodos Lax-Friedrichs, Nessyahu-Tadmor, Lax-Wendroff-Richtmyer, LWR com Runge-Kutta de terceira ordem TVD e Godunov com HLL.

Tabela 3.2: Informações sobre a performance computacional de cada método numérico

Esquema	tempo CPU (s)	operações Mflop/s
Lax-Friedrichs	31,908	113,4
Lax-Wendroff	54,083	183,6
Godunov-HLL	81,219	-
Nessyahu-Tadmor-ST	120,790	-
LWR-RK3TVD	171,841	-

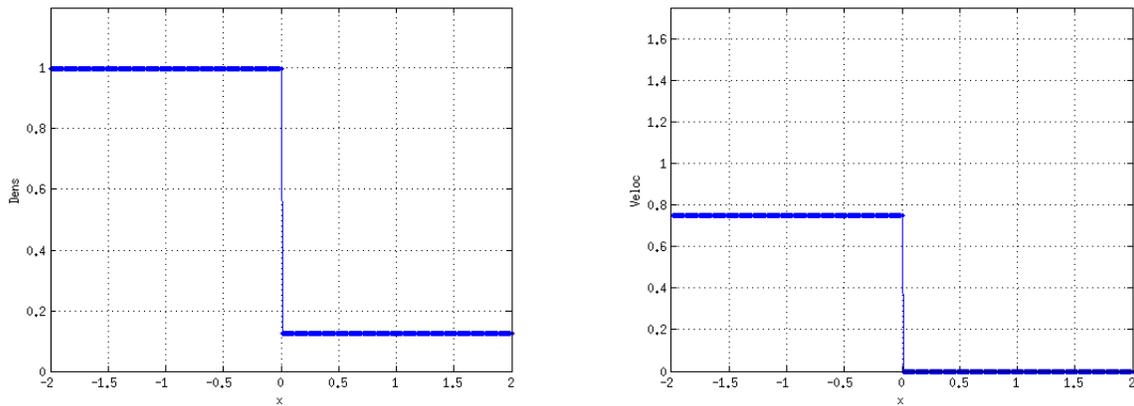


Figura 3.9: Condições iniciais: à esquerda tem-se a densidade inicial e à direita a velocidade inicial.

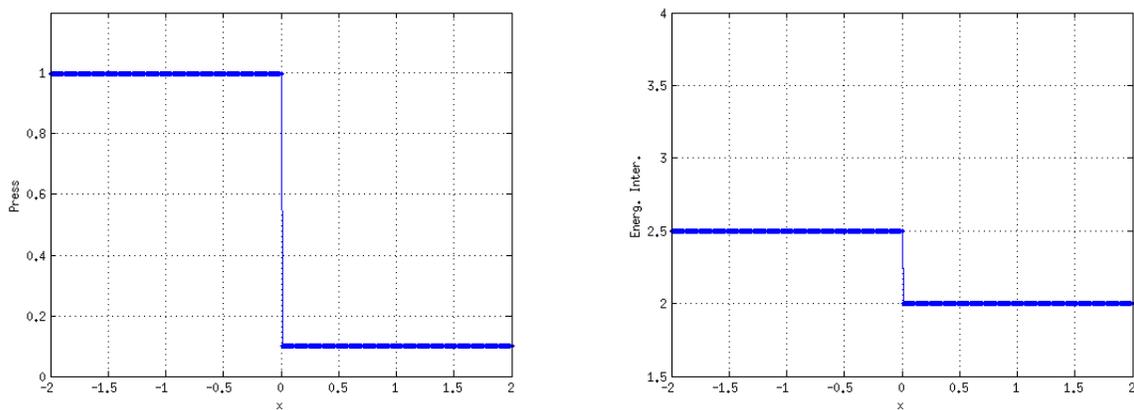


Figura 3.10: Condições iniciais: à esquerda tem-se a pressão inicial e à direita a energia interna inicial.

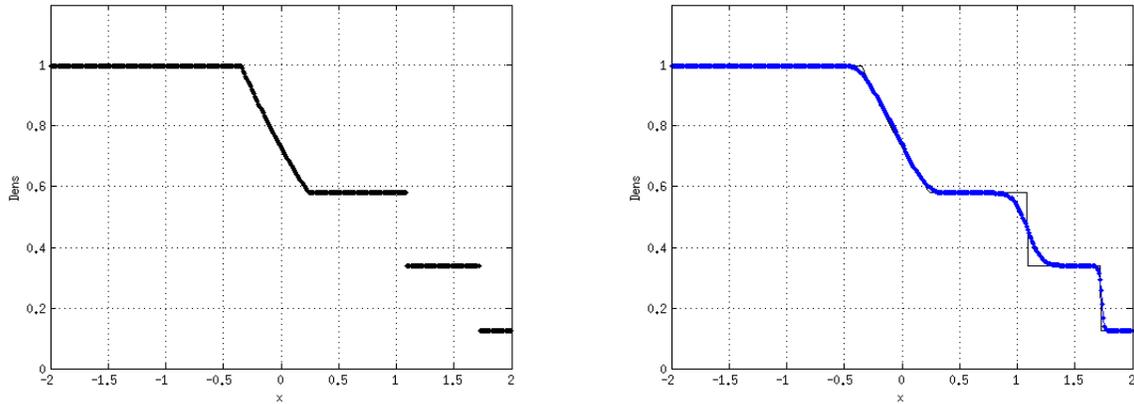


Figura 3.11: **Evolução da densidade:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov, com o solucionador de problemas de Riemann HLL.

Os gráficos à esquerda das Figuras 3.12 e 3.13 representam bem o caráter dissipativo do método Lax-Friedrichs e oscilatório do Lax-Wendroff-Richtmyer.

Nessas mesmas figuras, os gráficos à direita mostram a diminuição da dissipação numérica devido às melhorias que o método Nessyahu-Tadmor faz a partir do método Lax-Friedrichs e, também, tem-se a propriedade TVD, juntamente com método Runge-Kutta, controlando as oscilações do método Lax-Wendroff-Richtmyer.

Entretando esses benefícios resultam em custo computacional, pela Tabela 3.2 percebe-se que o método Nessyahu-Tadmor demora quase 4 vezes mais para executar um passo temporal ao se comparar com o método Lax-Friedrichs e o método LWR com RK3TVD gasta quase 3 vezes mais para realizar uma iteração do que o Lax-Wendroff.

O gráfico à direita da Figura 3.11 mostra a solução numérica do método de Godunov-HLL, que possui uma boa aproximação perto das discontinuidades, com um tempo computacional de CPU aproximadamente 2,5 vezes maior do que o tempo gasto pelo Lax-Friedrichs. Porém deve-se ressaltar que o método de Godunov precisa de mais informações sobre as equações diferenciais (sobre o sistema de autovalores e autovetores). As equações de Euler unidimensionais possuem todos os autovalores distintos, o que torna possível aplicar métodos do tipo Godunov sem acréscimos de técnicas [88].

As demais Figuras 3.14 - 3.22 apresentam as soluções obtidas por cada método numérico

e expõem os comportamentos já citados anteriormente.

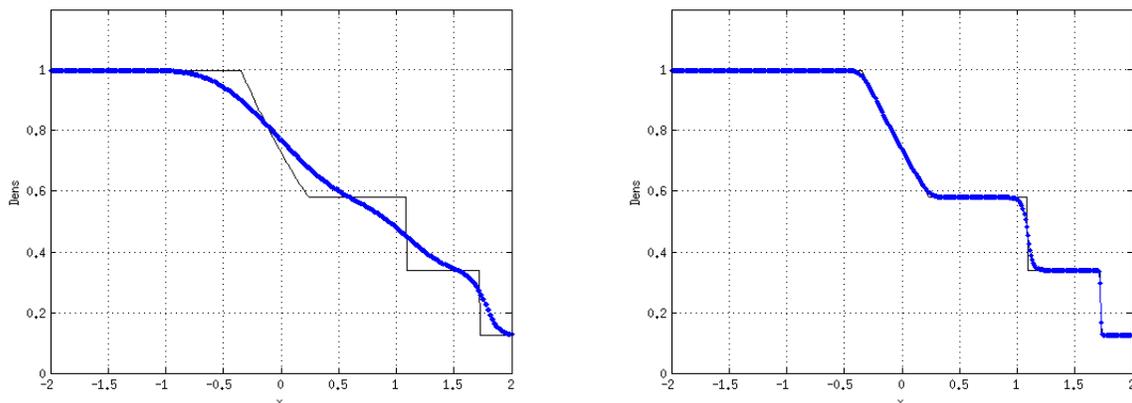


Figura 3.12: **Evolução da densidade:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

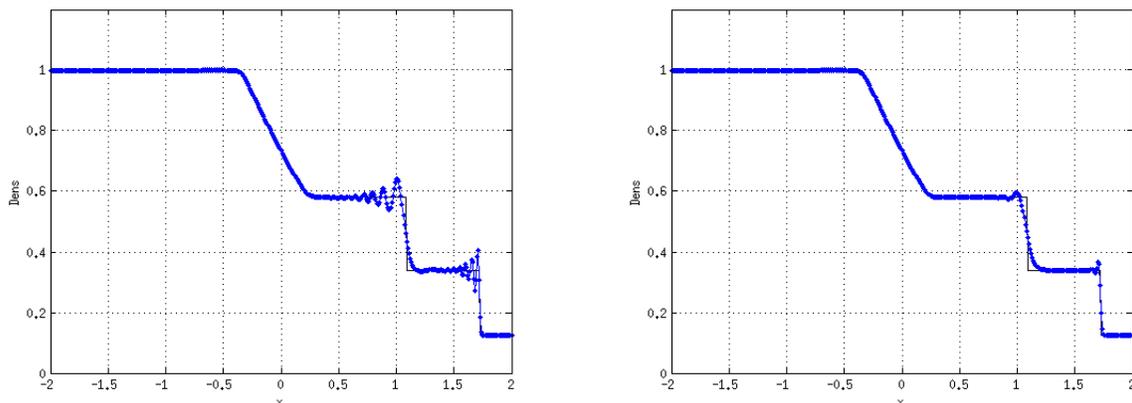


Figura 3.13: **Evolução da densidade:** à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

3.7.3 Exemplo 2 - problema de Sod

O exemplo 2 tem as mesmas condições iniciais consideradas no exemplo 1. Entretanto, a condição CFL não é fixada igualmente para todos os métodos numéricos e cada esquema trabalhará próximo do valor máximo de cada método.

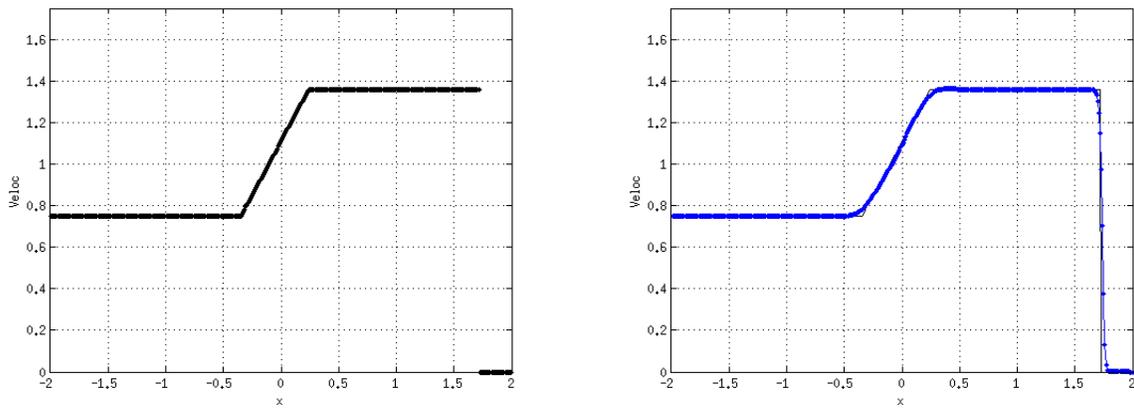


Figura 3.14: **Evolução da velocidade:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.

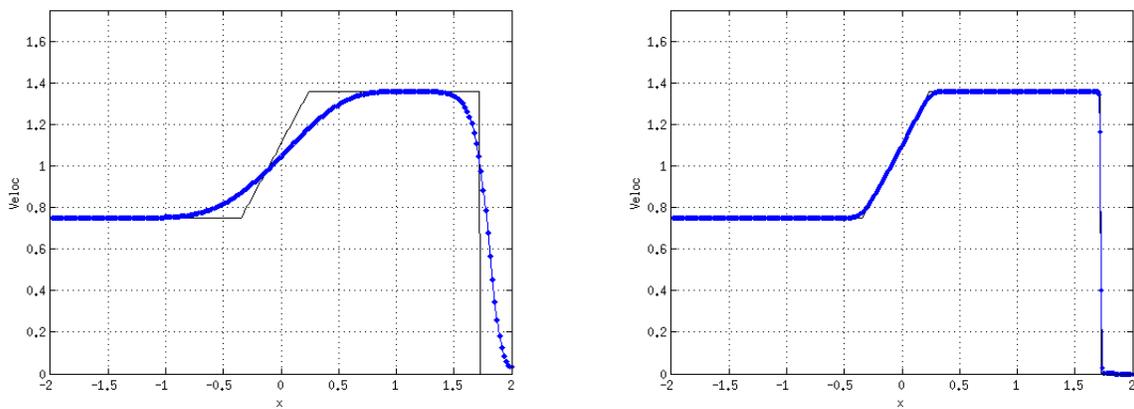


Figura 3.15: **Evolução da velocidade:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

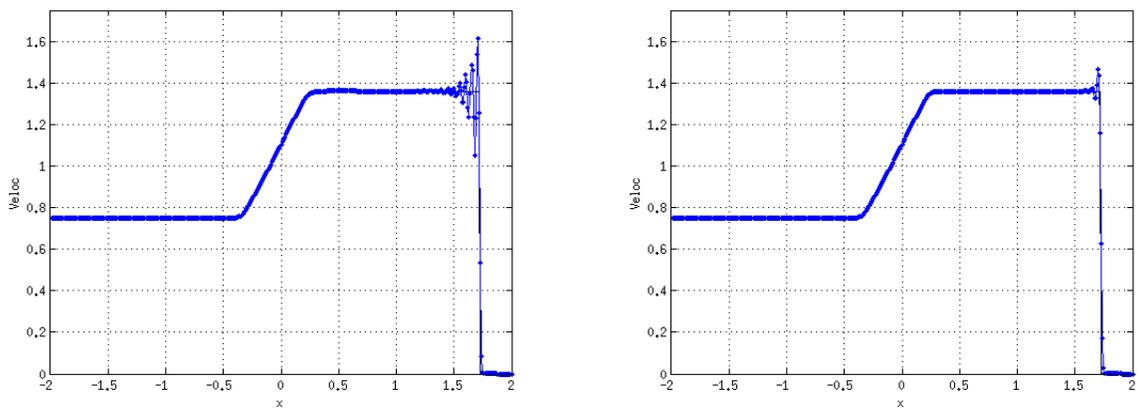


Figura 3.16: **Evolução da velocidade:** à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

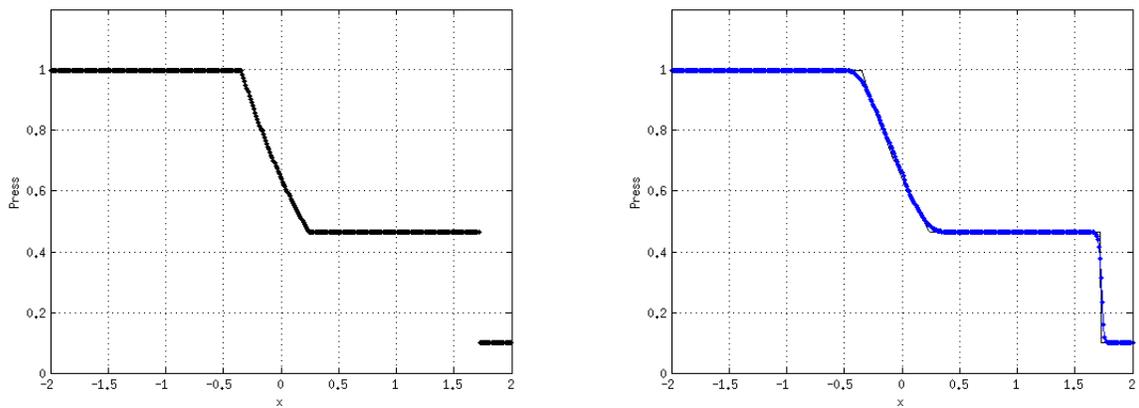


Figura 3.17: **Evolução da pressão:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.

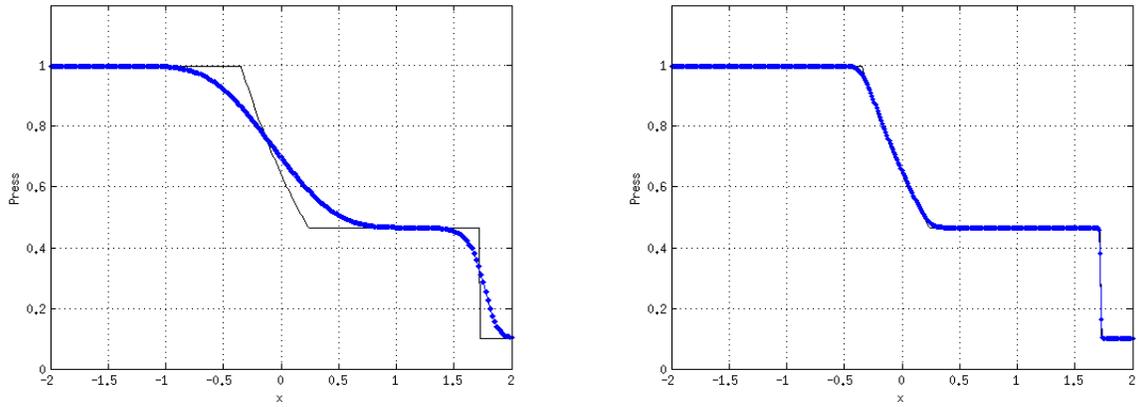


Figura 3.18: **Evolução da pressão:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

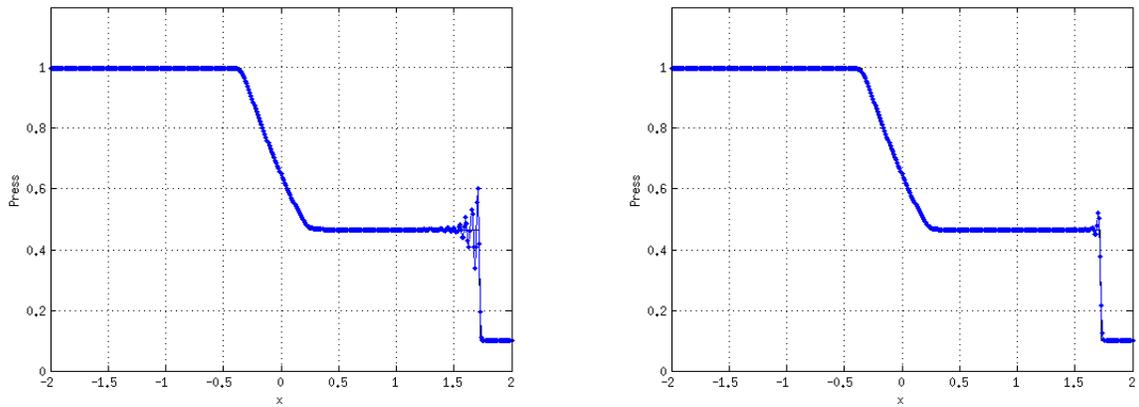


Figura 3.19: **Evolução da pressão:** à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

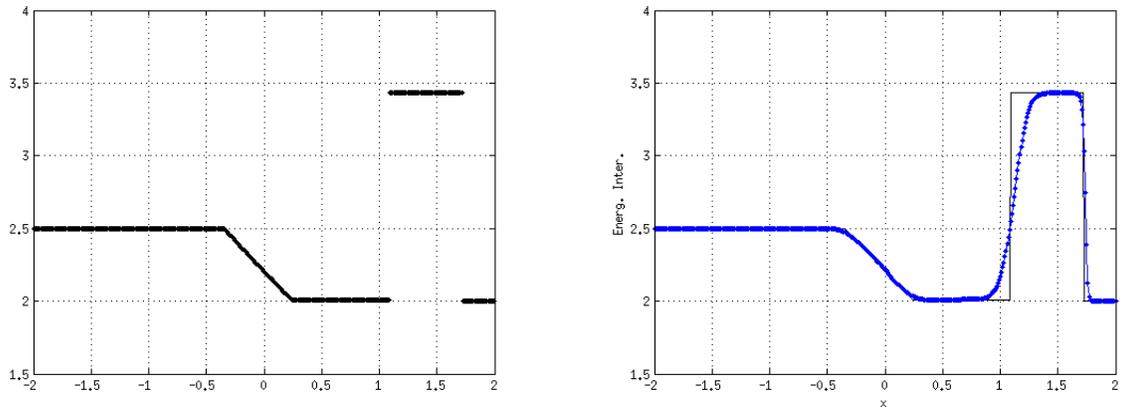


Figura 3.20: **Evolução da energia interna:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o resolucionador de problemas de Riemann HLL.

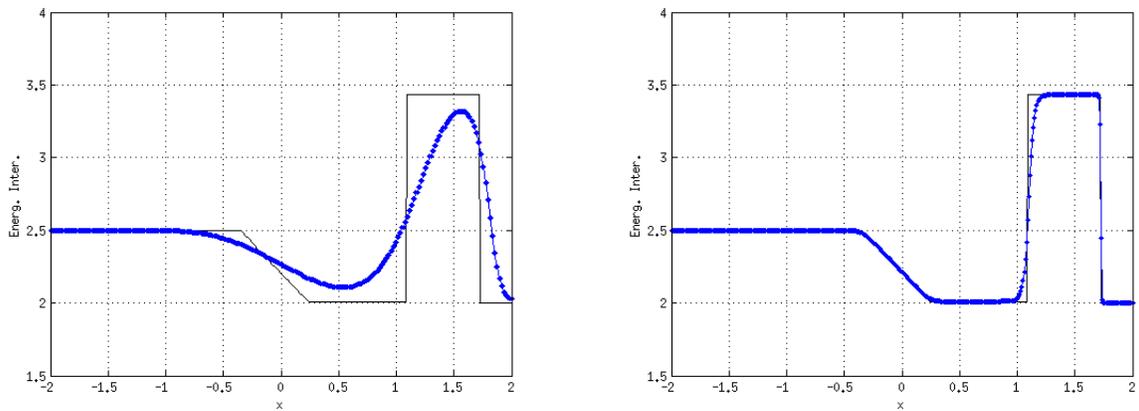


Figura 3.21: **Evolução da energia interna:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

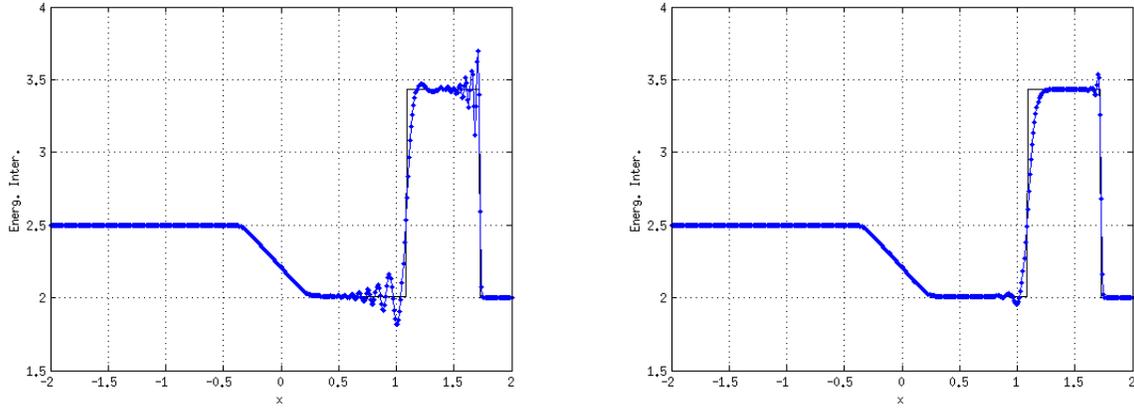


Figura 3.22: **Evolução da energia interna:** à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

Neste exemplo mantém-se o problema de Sod no domínio $[-2, 2]$, com 400 subdivisões, isto é $\Delta x = 0,01$ e com o objetivo de obter a solução numérica próximo do tempo final de 0,8.

Após as simulações obtivemos os dados da Tabela 3.3, com as informações da CFL próximo do máximo mantendo a estabilidade de cada método; quantidade de iterações para se chegar perto do tempo final de 0,8 e o tempo de CPU gasto por cada método. Esta comparação mostra que o método Lax-Wendroff com RK3TVD precisou de menos passos para atingir o tempo final, enquanto que o método Nessyahu-Tadmor levou mais passos para realizar uma evolução temporal semelhante.

Tabela 3.3: Informações sobre a performance computacional de cada método numérico

Esquema	CFL	passos de tempo	tempo final	tempo CPU (s)
Lax-Friedrichs	0,35	229	0,8015	8,9852
Lax-Wendroff	0,35	229	0,8015	14,415
Godunov-HLL	0,35	229	0,8015	22,675
LWR-RK3TVD	0,40	200	0,80	41,675
Nessyahu-Tadmor-ST	0,15	532	0,7980	78,311

Na Figura 3.23 foram plotados os gráficos da solução exata de referência e da solução numérica obtida pelo método Godunov-HLL.

Comparando as figuras que contêm as soluções numéricas dos métodos Lax-Friedrichs

e Nessyahu-Tadmor, Figuras 3.24, 3.27, 3.30 e 3.33, percebe-se que o método Nessyahu-Tadmor continua menos dissipativo do que o método Lax-Friedrichs, mesmo precisando de mais iterações para se chegar no mesmo tempo final. Entretanto, a diferença entre o tempo gasto de CPU aumenta tornando o método de Lax-Friedrichs quase 9 vezes mais rápido.

Observando as Figuras 3.25, 3.28, 3.31 e 3.34, que constituem as simulações feitas pelos métodos Lax-Wendroff-Richtmyer e LWR com RK3TVD, induz-se que a propriedade TVD suaviza a solução numérica diminuindo a oscilação do método Lax-Wendroff. Por outro lado observa-se que tal propriedade acrescenta dissipação numérica na solução aproximada.

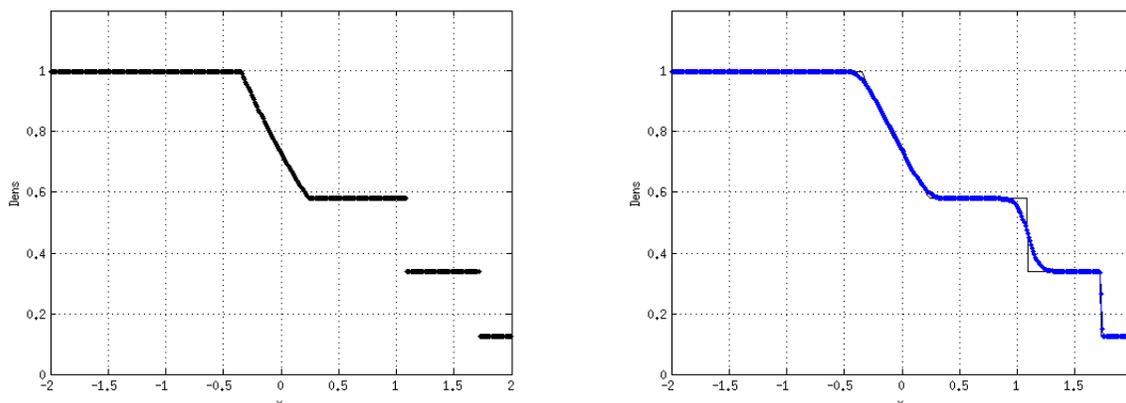


Figura 3.23: **Evolução da densidade:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo método de Godunov com o solucionador de problemas de Riemann HLL.

3.7.4 Exemplo 3 - problema de Woodward & Colella

Este exemplo descreve o escoamento de fluido unidimensional conhecido como problema teste de Woodward & Colella [94]. O destaque é a grande diferença de ordem de grandeza na pressão inicial, o que traz um ambiente encontrado em problemas de formação de jatos relativísticos. Um outro interesse em realizar este exemplo é o fato de que o fluido se desenvolverá à direita, isto é, no sentido contrário do problema de Sod - exemplos 1 e

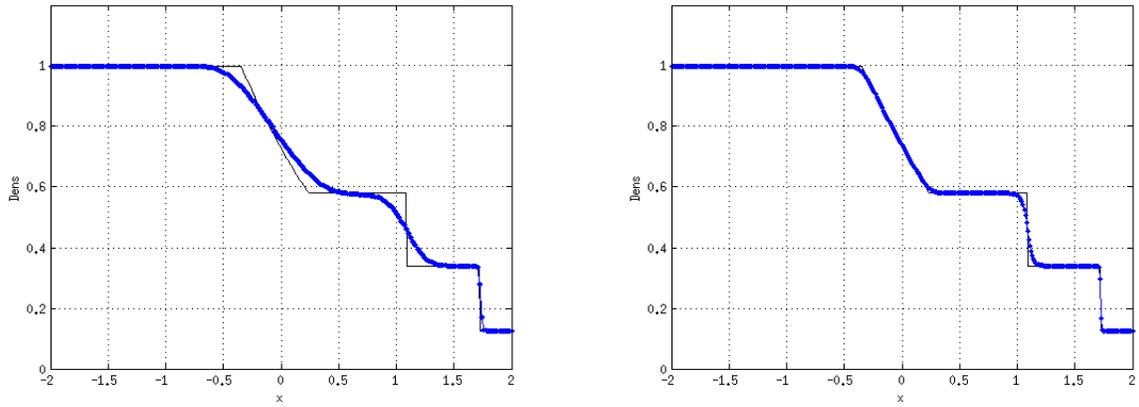


Figura 3.24: **Evolução da densidade:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

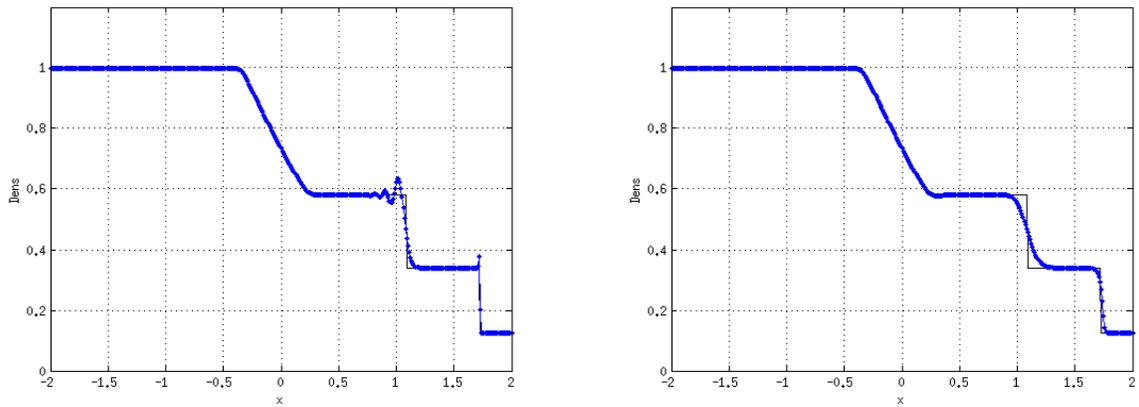


Figura 3.25: **Evolução da densidade:** à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

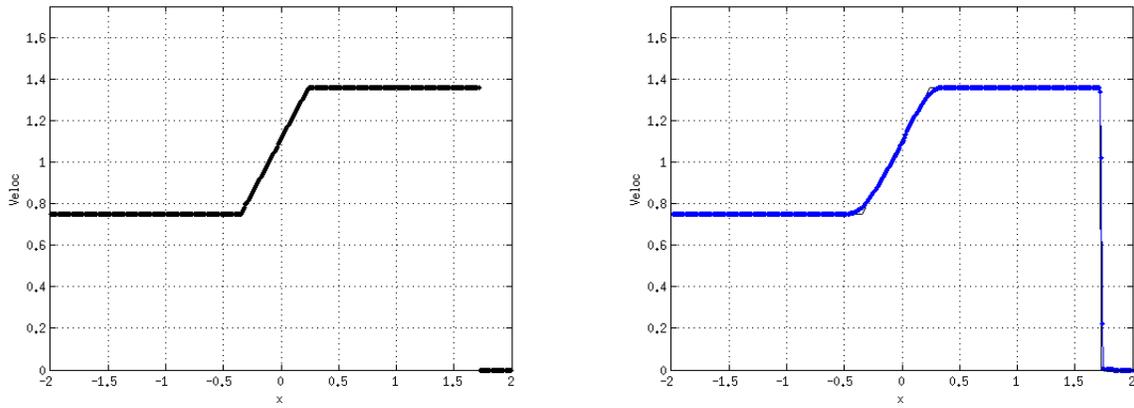


Figura 3.26: **Evolução da velocidade:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.

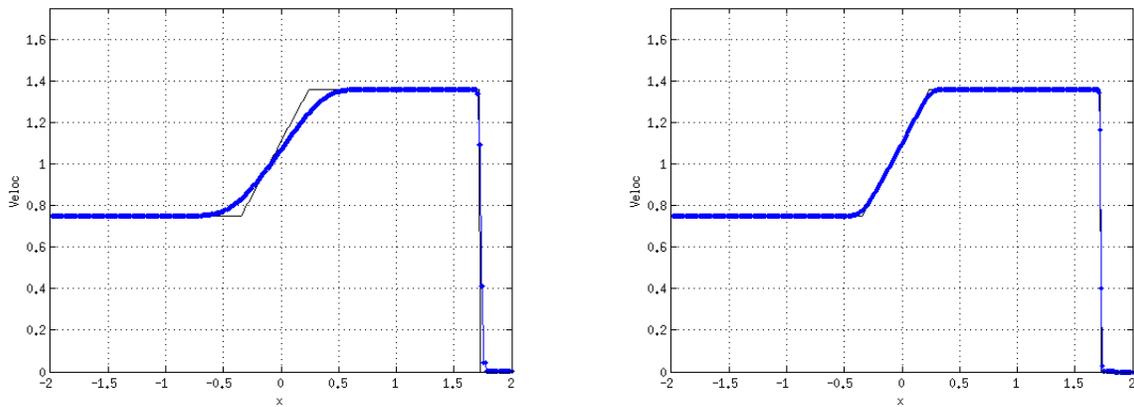


Figura 3.27: **Evolução da velocidade:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

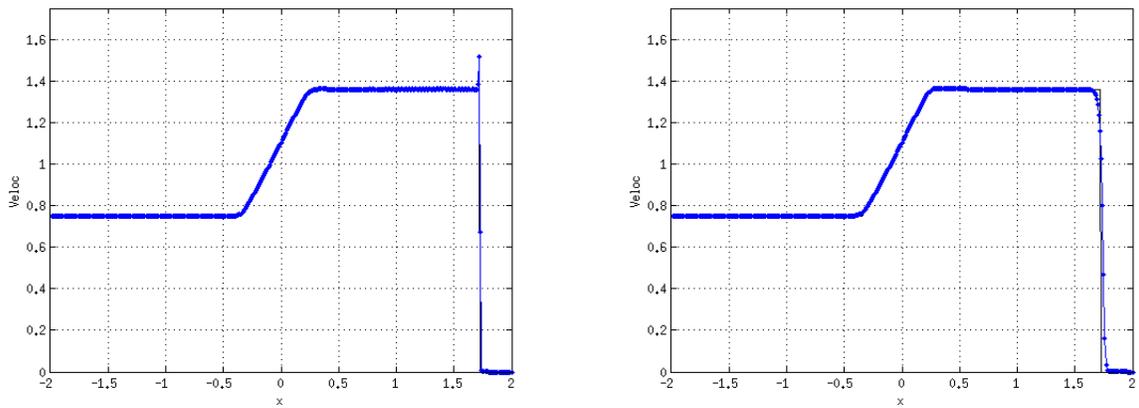


Figura 3.28: **Evolução da velocidade:** à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

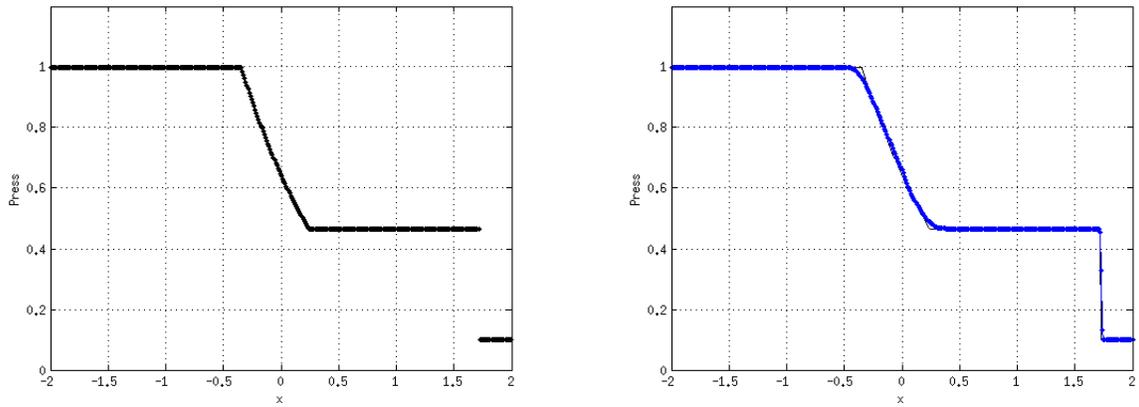


Figura 3.29: **Evolução da pressão:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.

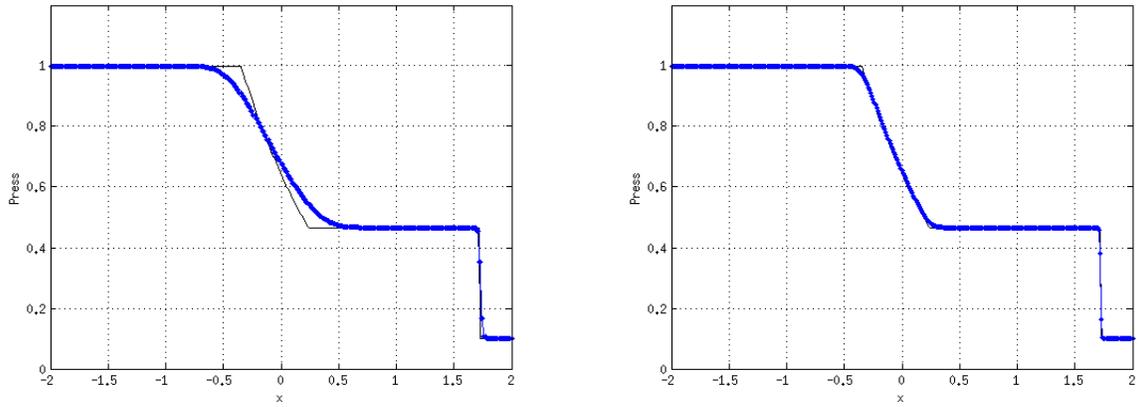


Figura 3.30: **Evolução da pressão:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

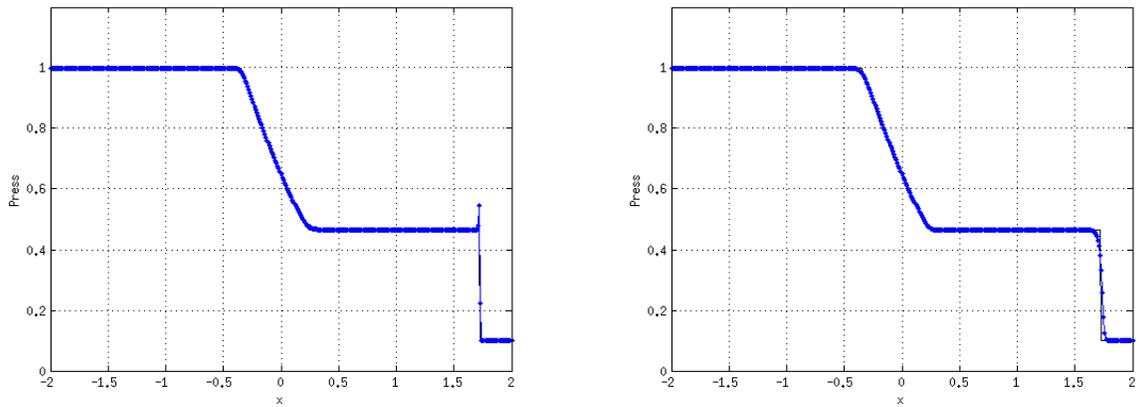


Figura 3.31: **Evolução da pressão:** à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

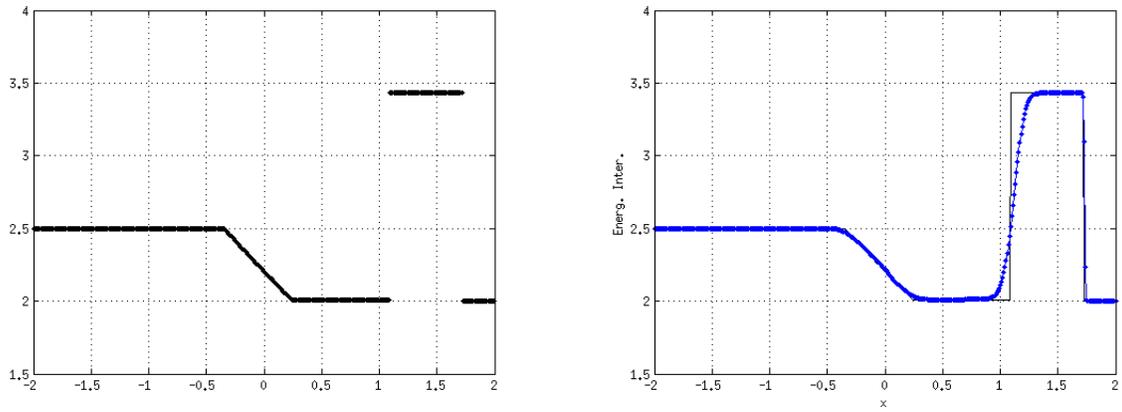


Figura 3.32: **Evolução da energia interna:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.

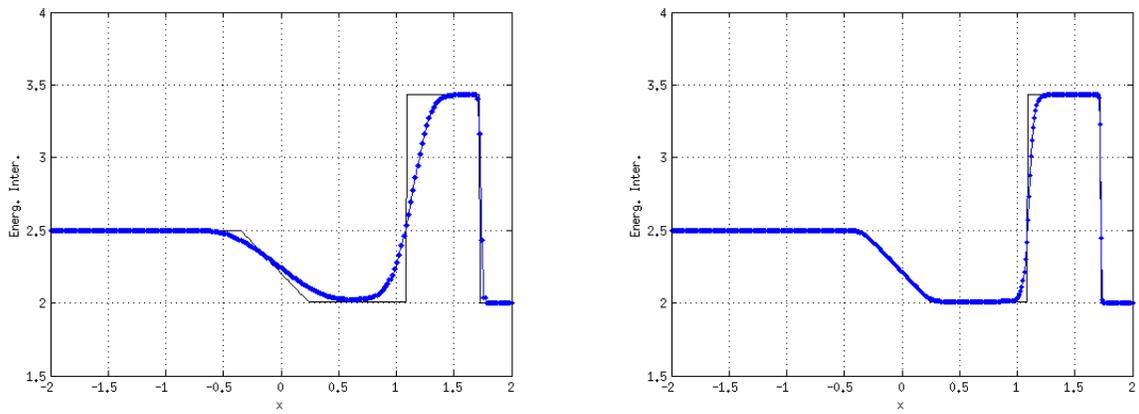


Figura 3.33: **Evolução da energia interna:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

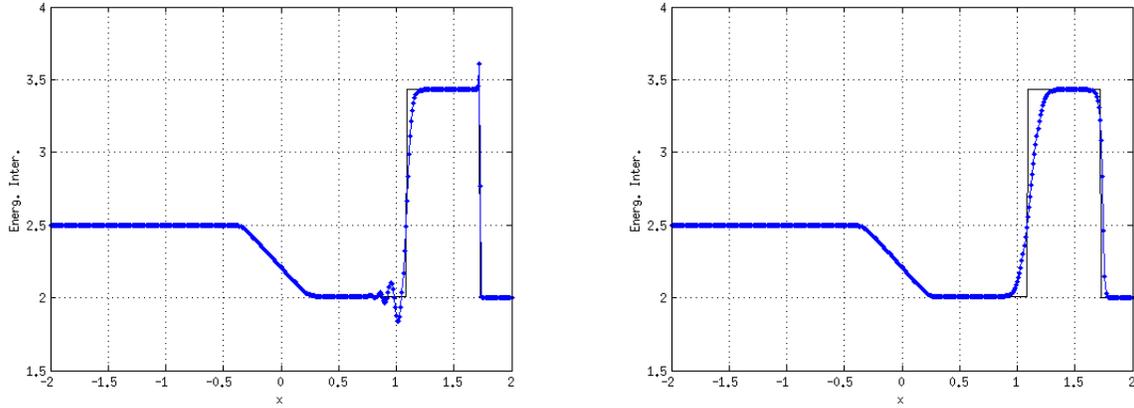


Figura 3.34: **Evolução da energia interna:** à esquerda expõe-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

2, assim os dois problemas testam os códigos dos métodos em várias situações possíveis tornando-os mais confiáveis.

O problema de Woodward & Colella à direita consiste das seguintes condições iniciais:

$$\rho(x, 0) = \begin{cases} 1,0 & \text{se } x < 2 \\ 1,0 & \text{se } x > 2 \end{cases}, \quad v(x, 0) = \begin{cases} 0,0 & \text{se } x < 2 \\ 0,0 & \text{se } x > 2 \end{cases},$$

e

$$p(x, 0) = \begin{cases} 0,01 & \text{se } x < 2 \\ 1.000,00 & \text{se } x > 2 \end{cases}. \quad (3.74)$$

Consideramos $\gamma = 1,4$ e para as simulações representadas pelas Figuras 3.37 - 3.48 utilizamos os dados da Tabela 3.4. As condições iniciais estão plotadas nas Figuras 3.35 e 3.36. Em todos os gráficos de soluções numéricas mostramos a solução analítica de referência em cor preta. Na Tabela 3.5 encontram-se informações sobre o desempenho computacional de cada método, quando aplicados ao problema em questão.

Tabela 3.4: Dados computacionais para a obtenção das soluções do problema de Woodward & Colella

Domínio	Subintervalos	Δx	t_f
[0,4]	800	0,005	0,04

Utilizando os dados da Tabela 3.4 encontramos soluções numéricas dos métodos Lax-

Tabela 3.5: Informações sobre a performance computacional de cada método numérico

Esquema	CFL	passos de tempo	tempo final	tempo CPU (s)
Lax-Friedrichs	0,02	400	0,04	32,089
Godunov-HLL	0,02	400	0,04	80,471
LWR-RK3TVD (*)	0,02	400	0,04	166,76
Nessyahu-Tadmor-ST	0,008	1.000	0,04	754,99
Lax-Wendroff (**)	0,0001-0,04	-	-	-

(*) método estável para os valores 0,01 e 0,02, e instável para 0,005 e 0,04.

(**) método não estabilizado entre 0,0001 e 0,04.

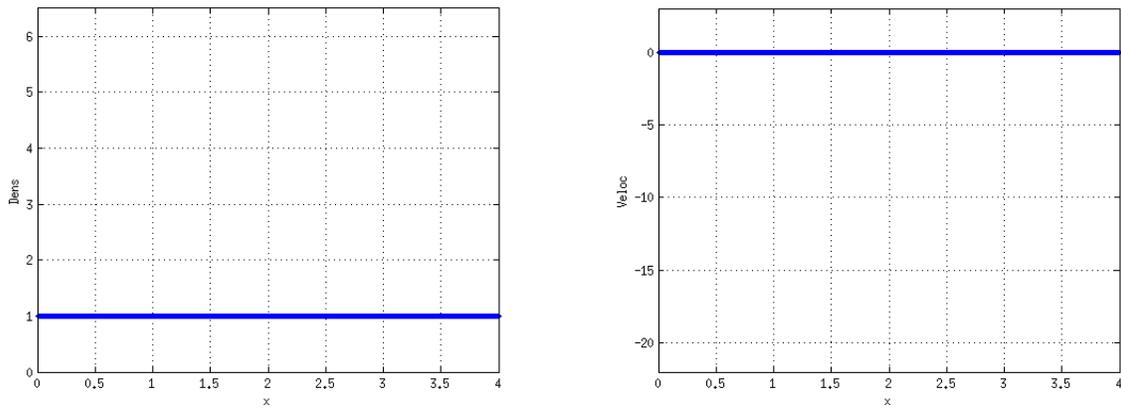


Figura 3.35: Condições iniciais: à esquerda tem-se a densidade inicial e à direita a velocidade inicial.

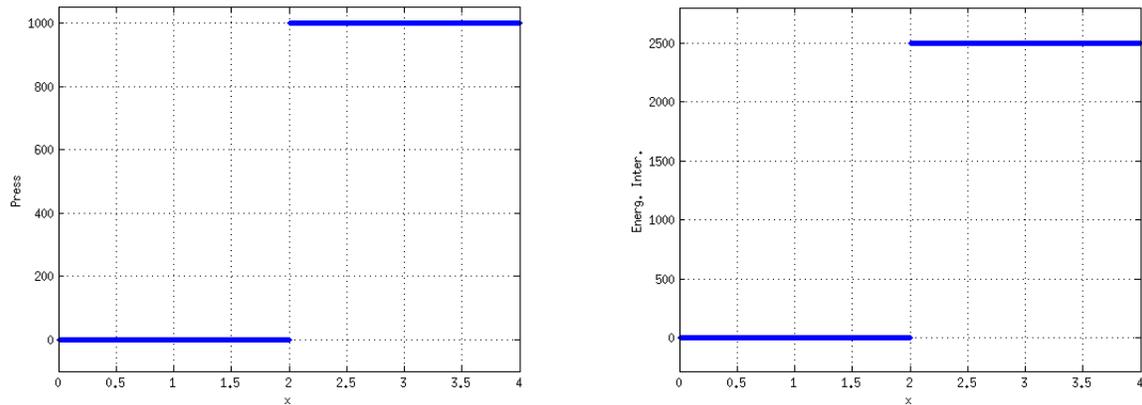


Figura 3.36: Condições iniciais: à esquerda tem-se a pressão inicial e à direita a energia interna inicial.

Friedrichs, Nessyahu-Tadmor, Lax-Wendroff-Richtmyer, LWR com Runge-Kutta de terceira ordem TVD e Godunov com HLL.

Os gráficos à esquerda das Figuras 3.38, 3.41, 3.44 e 3.47 mostram as soluções encontradas pelo Método Lax-Friedrichs. Apesar do caráter dissipativo, as soluções numéricas são boas aproximações e tal caráter neste exemplo não ficou tão evidenciado quando comparado ao problema de Sod - Exemplo 1. A dissipação numérica foi mais evidenciada no gráfico da densidade - Figura 3.38.

Nas mesmas figuras, os gráficos à direita mostram as soluções obtidas pelo método Nessyahu-Tadmor e tal método permaneceu menos dissipativo mesmo precisando de mais iterações quando comparado ao Lax-Friedrichs - veja Tabela 3.5. Confrontando os métodos Nessyahu-Tadmor e Godunov-HLL - figuras 3.37, 3.40, 3.43 e 3.46 - as soluções gráficas possuem comportamentos semelhantes. O método Nessyahu-Tadmor precisou de um tempo de execução maior que o do método Godunov-HLL - ver Tabela 3.5, porém ressalta-se que o método Godunov-HLL necessita dos autovalores da matriz Jacobiana e em sistemas que não são estritamente hiperbólicos, as dificuldades de aplicação aumentam. Embora as soluções apresentem grandes semelhanças, destacam-se as soluções obtidas pelo método Nessyahu-Tadmor, pois foi possível perceber uma melhor representação do pico que aparece no gráfico da densidade, mesmo sem precisar de informações a mais como no caso do método Godunov-HLL.

As soluções obtidas pelos métodos Lax-Wendroff's encontram-se nas Figuras 3.39, 3.42, 3.45 e 3.48. Nos gráficos à esquerda têm-se as soluções obtidas pelo método Lax-Wendroff-Richtmyer, cuja evolução temporal não atingiu o tempo final devido as grandes oscilações. Na tentativa de estabilizá-lo variou a CFL entre os valores de 0,0001 até 0,04, entretanto não foi possível obter uma solução estável. As Figuras 3.39, 3.42, 3.45 e 3.48 possuem soluções, com oscilações utilizando o método LWR com os valores $CFL = 0,0001$, 1200 iterações e tempo final de 6.10^{-4} , com um pouco mais de iterações a solução é ilimitada obtendo-se um *overflow*.

Nota-se que, mesmo o método Lax-Wendroff-Richtmyer sendo consistente e condicionalmente estável - seção 3.5.3, não foi possível obter uma solução estável devido a propagação de grandes oscilações nas proximidades da descontinuidade inicial da pressão.

A modificação realizada no método Lax-Wendroff utilizando o Runge-Kutta 3 TVD para a discretização, foi capaz de controlar a grande oscilação inicial e estabilizar a solução numérica, veja as Figuras 3.39, 3.42, 3.45 e 3.48, gráficos à direita. Modificando a CFL para 0,01 realizamos a simulação com o método LWR-KK3TVD até o tempo final de 0,04 com 800 iterações, neste caso a oscilação inicial continuou controlada, porém com uma amplitude maior do que a solução obtida com a CFL de 0,02 - ver Figuras 3.49 e 3.50. De fato as soluções foram controladas, simulações com tempo final maior foram realizadas e tal oscilação não aumentou de tamanho.

O custo computacional desses métodos encontram-se na Tabela 3.5. O método Nessyahu-Tadmor demorou 20 vezes mais para executar a mesma evolução temporal realizada pelo método Lax-Friedrichs e o método LWR com RK3-TVD gastou quase 4 vezes mais para realizar a evolução que o Lax-Friedrichs fez. Por outro lado, o método Nessyahu-Tadmor foi o que melhor representou a evolução da descontinuidade inicial da pressão e as demais descontinuidades subsequentes às condições iniciais. O grande aumento no tempo de execução do método Nessyahu-Tadmor está vinculado à função *minmod*, pois sua implementação computacional depende de tomadas de decisões - comandos IF's.

O gráfico à direita da Figura 3.37 mostra a solução numérica do método de Godunov-HLL. As demais Figuras 3.40 - 3.48 apresentam as soluções obtidas por cada método numérico e expõem os comportamentos já citados anteriormente.

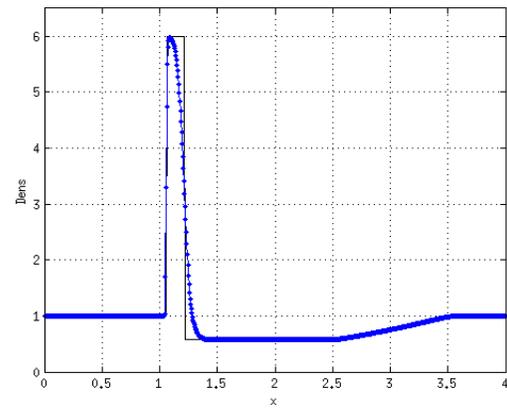
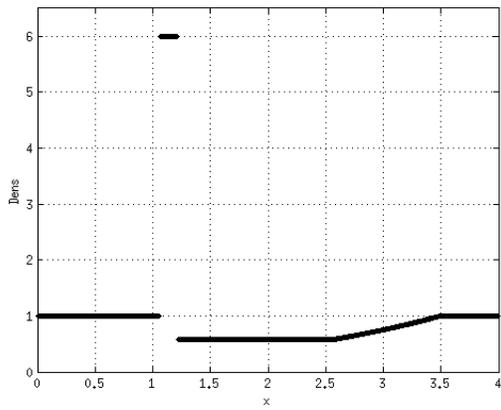


Figura 3.37: **Evolução da densidade:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.

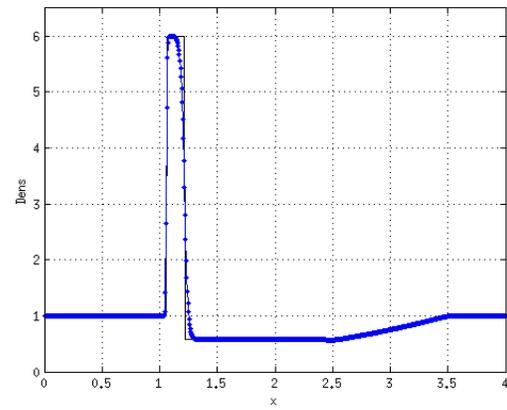
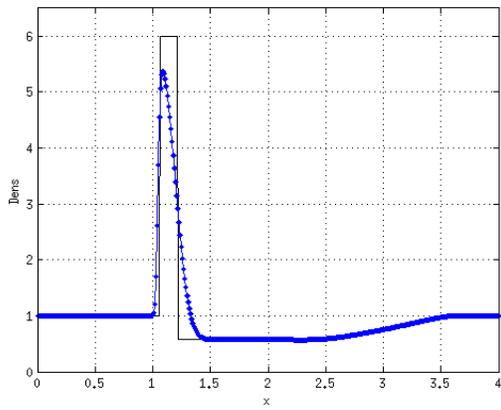


Figura 3.38: **Evolução da densidade:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

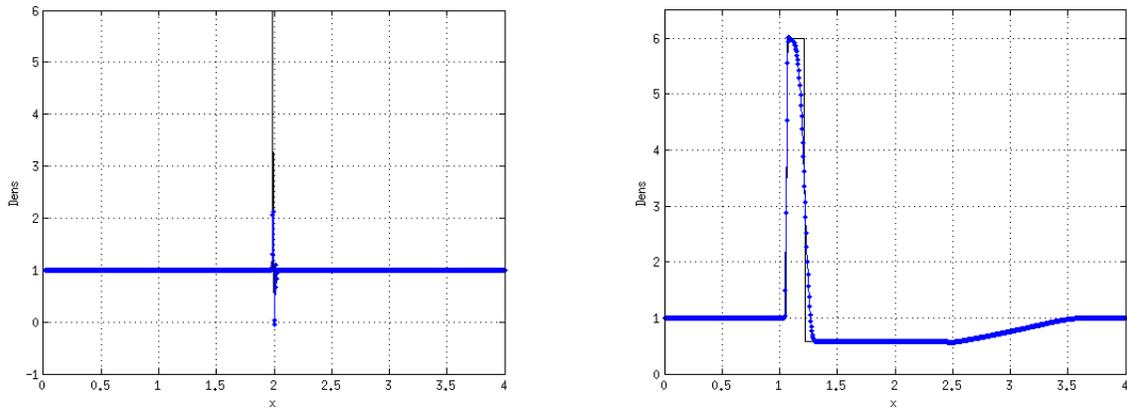


Figura 3.39: **Evolução da densidade:** à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

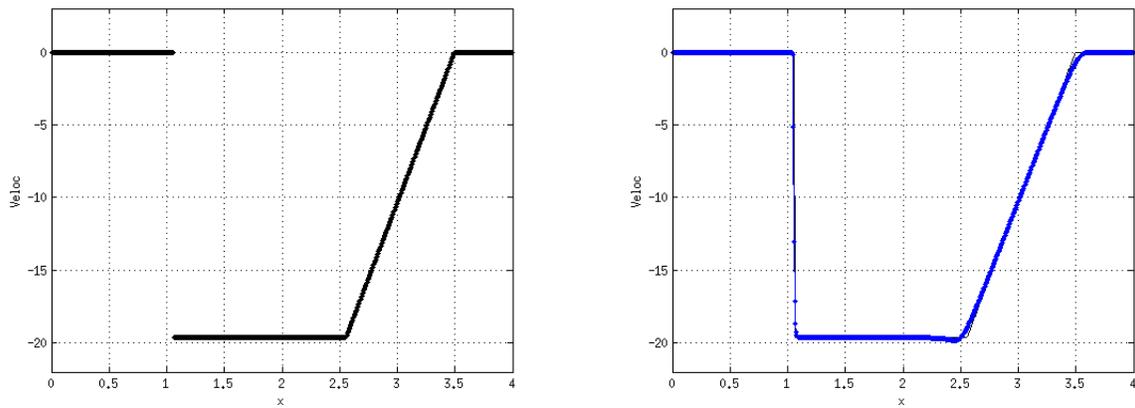


Figura 3.40: **Evolução da velocidade:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.

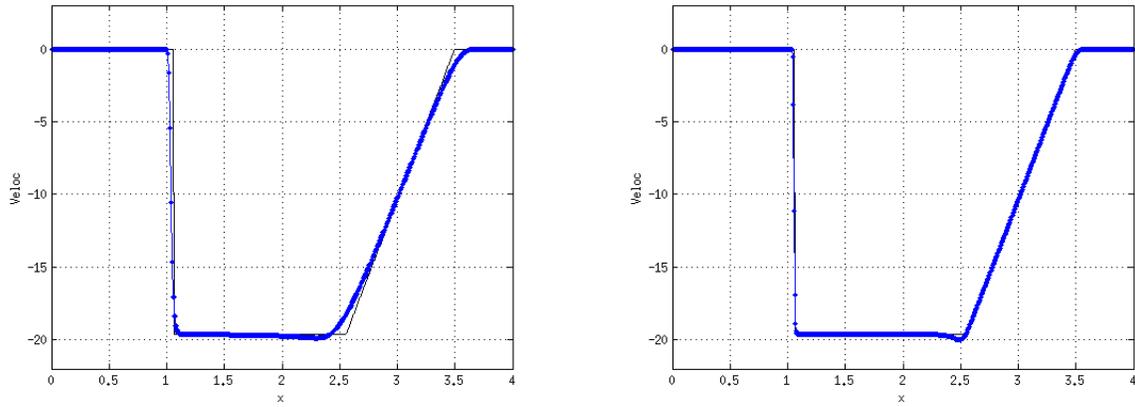


Figura 3.41: **Evolução da velocidade:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

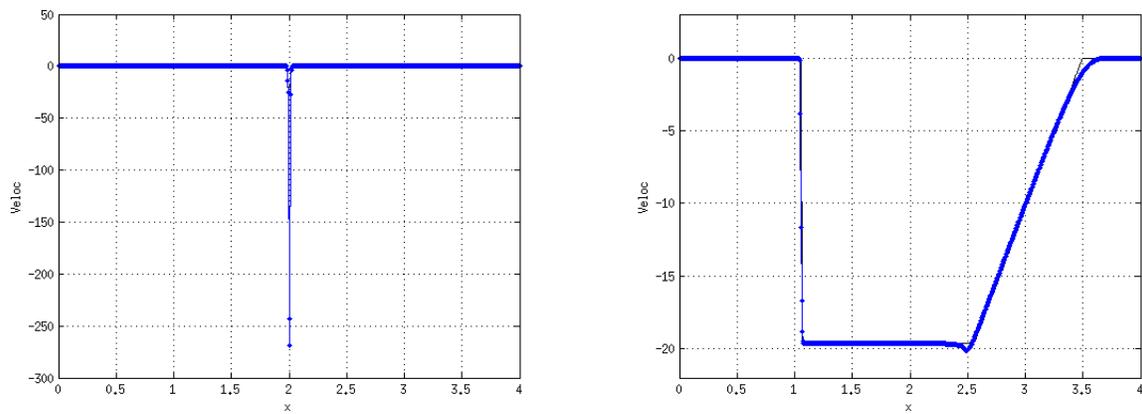


Figura 3.42: **Evolução da velocidade:** à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

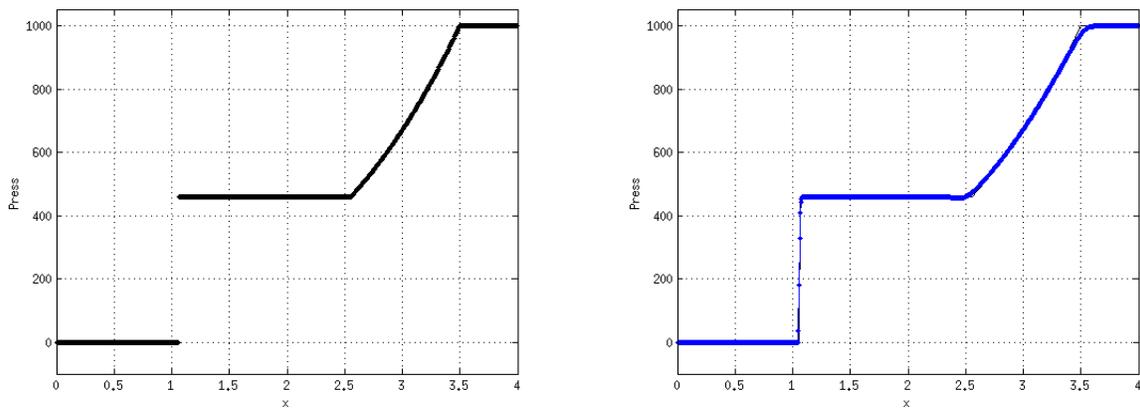


Figura 3.43: **Evolução da pressão:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o solucionador de problemas de Riemann HLL.

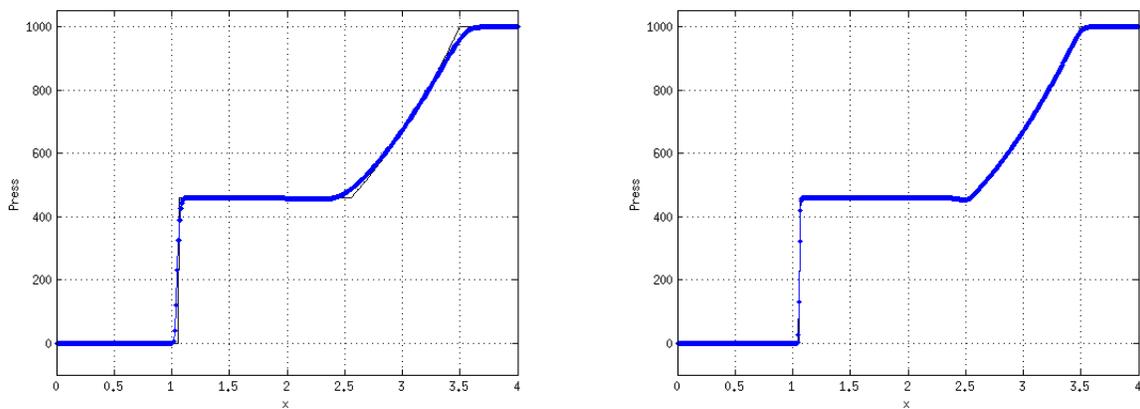


Figura 3.44: **Evolução da pressão:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

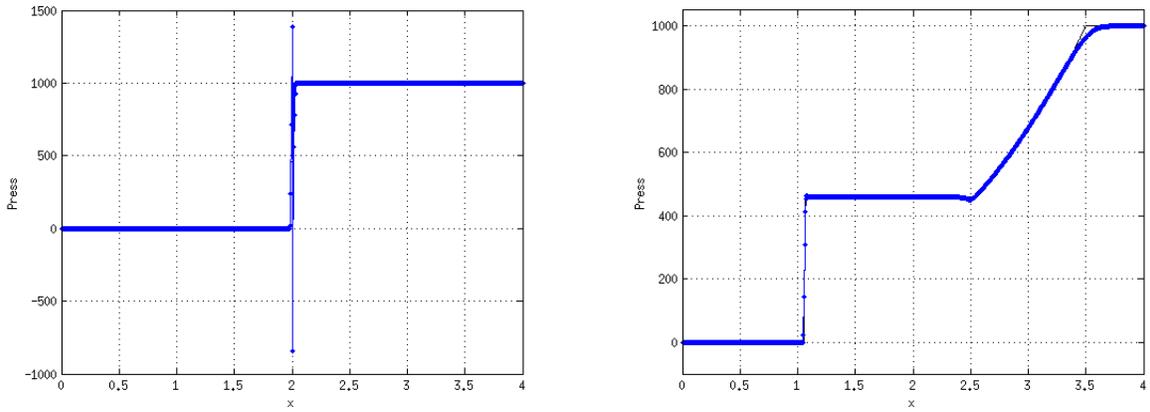


Figura 3.45: **Evolução da pressão:** à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

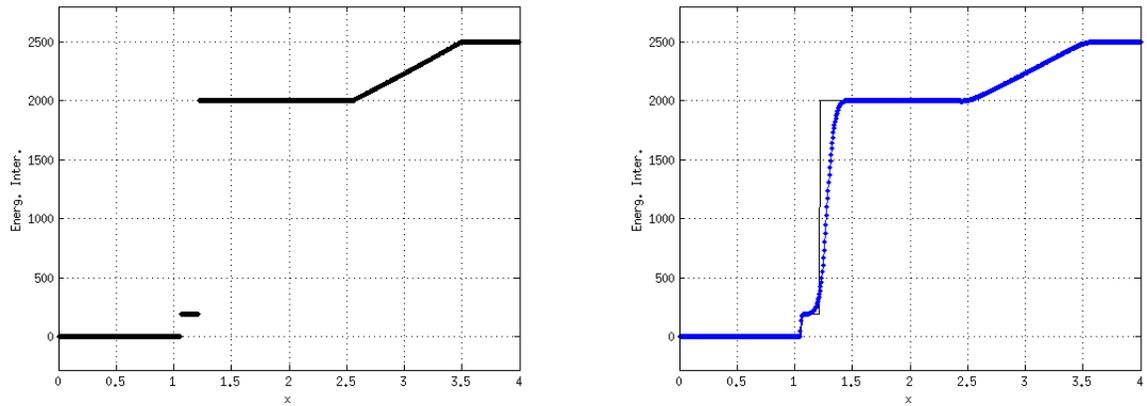


Figura 3.46: **Evolução da energia interna:** à esquerda encontra-se a solução exata de referência e à direita a solução obtida pelo Método de Godunov com o resolucionador de problemas de Riemann HLL.

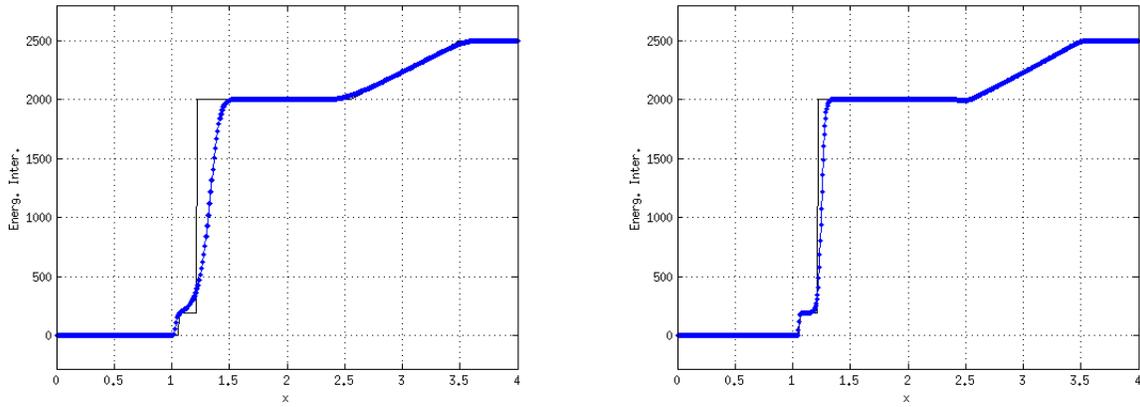


Figura 3.47: **Evolução da energia interna:** à esquerda tem-se a solução obtida pelo método Lax-Friedrichs e à direita pelo Método Nessyahu-Tadmor.

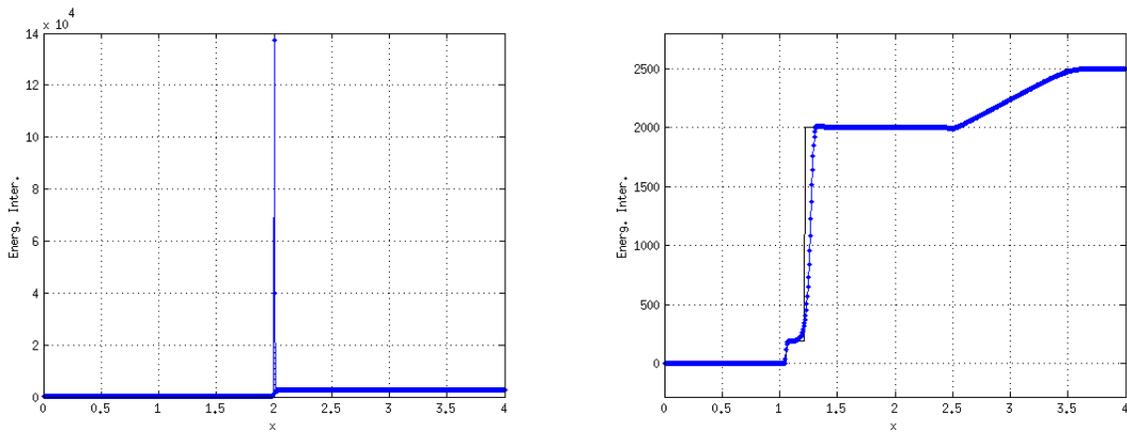


Figura 3.48: **Evolução da energia interna:** à esquerda expõem-se a solução obtida pelo método Lax-Wendroff-Richtmeyer(LWR) e à direita pelo método LWR com RK3-TVD.

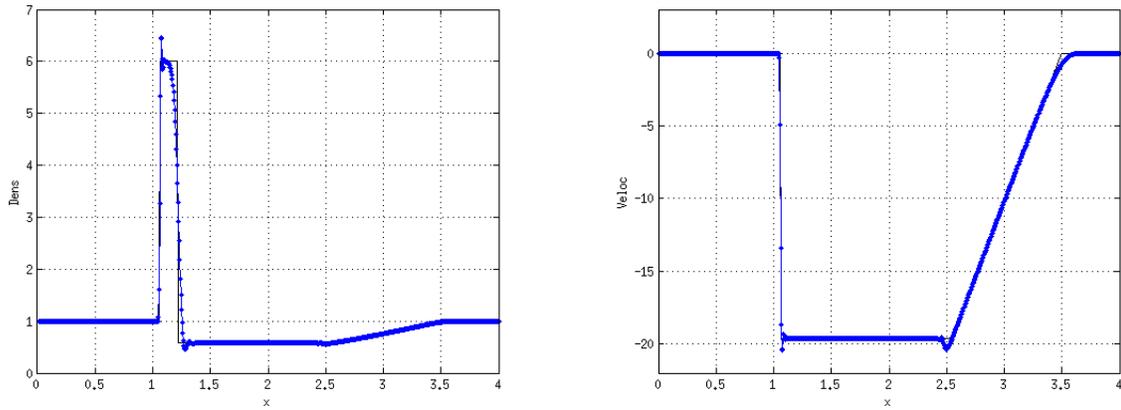


Figura 3.49: **Evolução da densidade:** à esquerda tem-se a solução da densidade e à direita da velocidade, ambas obtida pelo método LWR com RK3-TVD. Consideramos a CFL de 0,01, que apresentou uma oscilação inicial maior do que a oscilação para $CFL = 0,02$.

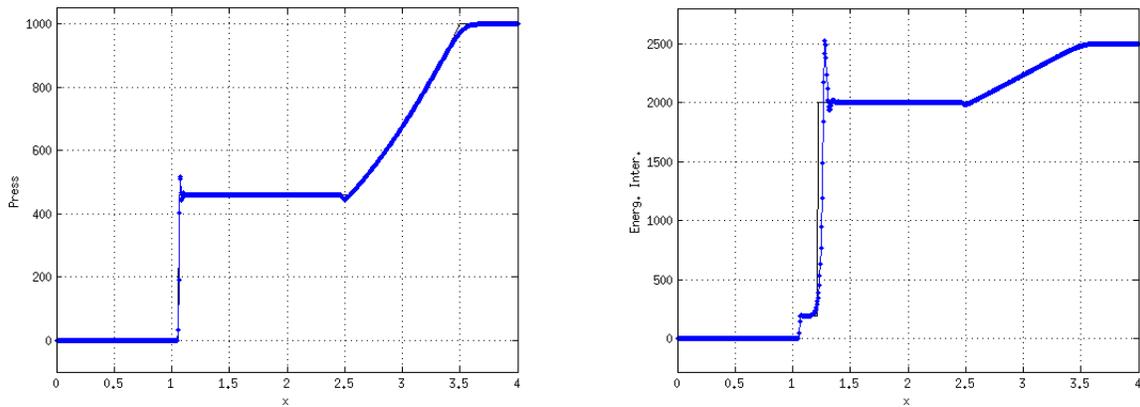


Figura 3.50: **Evolução da densidade:** À esquerda tem-se a solução da pressão e à direita da energia interna, ambas obtida pelo método LWR com RK3-TVD. Consideramos a CFL de 0,01, que apresentou uma oscilação inicial maior do que a oscilação para $CFL = 0,02$.

3.8 Métodos Numéricos com termo fonte

Leis de balanço são equações diferenciais que se assemelham com Leis de Conservação, a menos de um termo fonte. Um procedimento para obter soluções numéricas deste tipo de problema utilizando métodos de volumes finitos é separar a EDP com termo fonte em uma EDP homogênea e uma EDO relacionada com o termo fonte.

Após a separação, resolvemos numericamente a EDP homogênea entre t_n e t_{n+1} ; utilizamos a solução obtida como condição inicial da EDO e então, dentro do mesmo espaçamento temporal, resolve-se a EDO numericamente por um dos métodos Runge-Kutta, que atualiza a solução da EDP homogênea, resultando na solução da EDP com termo fonte [34]. A seguir descrevemos o procedimento.

Considere o problema unidimensional

$$u_t + f(u)_x = s(u) .$$

O processo de separar uma EDP com termo fonte em uma EDP homogênea e uma EDO, consiste em definir os seguintes problemas:

$$\begin{cases} \partial_t u + \partial_x f(u) = 0 \\ u(x, t_n) = u^n \end{cases} \Rightarrow \bar{u}^{n+1} \quad (3.75)$$

e

$$\begin{cases} \frac{du}{dt} = s(u) \\ u(x, t_n) = \bar{u}^n \end{cases} \Rightarrow u^{n+1} . \quad (3.76)$$

Primeiro, para realizar a evolução temporal da EDP com termo fonte, de um tempo t_n até t_{n+1} , obtém-se a solução numérica da EDP homogênea (3.75) representada por \bar{u}^{n+1} - neste passo aplica-se por exemplo, algum método descrito e analisado nas seções anteriores. Depois, atualiza-se a solução numérica resolvendo a EDO dentro do mesmo intervalo temporal e utilizando a solução \bar{u}^{n+1} como condição inicial. Por fim, obtém-se a solução numérica da EDP não homogênea no tempo t_{n+1} . Detalhes e características do procedimento encontram-se nas referências [34, 52, 53, 88].

3.9 Métodos Numéricos Bidimensionais

3.9.1 Métodos com separação dimensional

Uma maneira de resolver numericamente equações diferenciais parciais cuja variável espacial possui duas ou mais dimensões é substituir a EDP original por um conjunto de EDP's unidimensionais no espaço. Sendo assim, para se realizar um passo temporal é necessário resolver o conjunto de EDP's unidimensionais. Este procedimento é descrito a seguir.

Considere o seguinte problema bidimensional

$$\begin{cases} \partial_t \mathbf{U} + \partial_{x_1} \mathbf{F}_1(\mathbf{U}) + \partial_{x_2} \mathbf{F}_2(\mathbf{U}) = 0 \\ \mathbf{U}(\mathbf{x}, t_n) = \mathbf{U}_0(\mathbf{x}) = \mathbf{U}^n \end{cases} . \quad (3.77)$$

O procedimento de separar as dimensões espaciais consiste em obter o seguinte conjunto de EDP's unidimensionais no espaço:

$$\begin{cases} \partial_t \mathbf{U} + \partial_{x_1} \mathbf{F}_1(\mathbf{U}) = 0 \\ \mathbf{U}(\mathbf{x}, t_n) = \mathbf{U}_0(\mathbf{x}) = \mathbf{U}^n \end{cases} \Rightarrow \mathbf{U}^* \quad (3.78)$$

$$\begin{cases} \partial_t \mathbf{U} + \partial_{x_2} \mathbf{F}_2(\mathbf{U}) = 0 \\ \mathbf{U}(\mathbf{x}, t_n) = \mathbf{U}_0(\mathbf{x}) = \mathbf{U}^* \end{cases} \Rightarrow \mathbf{U}^{n+1} . \quad (3.79)$$

Assim, encontra-se a solução numérica U^* da EDP (3.78), por algum método unidimensional no espaço realizando-se a evolução temporal de t_n até t_{n+1} . Após este estágio, utiliza-se a solução U^* como condição inicial da EDP (3.79) para obter a solução numérica no tempo t_{n+1} , usando-se novamente um método unidimensional. Portanto, para cada passo temporal são necessários resolver dois problemas unidimensionais com um Δt que mantém o método numérico estável nas duas direções espaciais.

Na forma conservativa de métodos de volumes finitos - definição 3.2.2, o *Splitting* dimensional é representado da seguinte maneira.

$$U_{i,j}^{n+1/2} = U_{i,j}^n - \frac{\Delta t}{\Delta x} [F_{(1);i+1/2,j}^n - F_{(1);i-1/2,j}^n] \quad (3.80)$$

e

$$U_{i,j}^{n+1} = U_{i,j}^{n+1/2} - \frac{\Delta t}{\Delta y} \left[F_{(2);i,j+1/2}^{n+1/2} - F_{(2);i,j-1/2}^{n+1/2} \right] \quad (3.81)$$

Observações:

- As duas EDP's unidimensionais são evoluídas entre o espaçamento Δt ;
- $U_{i,j}^n$ é uma aproximação da solução da EDP bidimensional \mathbf{U} , no ponto (x_i, y_j) do tempo t_n , de um domínio Ω particionado regularmente $\Pi_x \times \Pi_y$ com espaçamentos Δx e Δy , nas respectivas direções x e y , isto é, $U_{i,j}^n = \mathbf{U}(x_i, y_j, t_n)$;
- Uma maneira de escolher Δt é tomar como referência o domínio de dependência e a análise de estabilidade linear (análise de von Neumann), assim

$$\Delta t = \sigma \times \min_{i,j} \left\{ \frac{\Delta x}{Sx_{i,j}^n}, \frac{\Delta y}{Sy_{i,j}^n} \right\},$$

em que, σ é a relação entre Δt e Δx obtida pela análise de estabilidade linear, Sx está relacionado com a velocidade de propagação das ondas elementares na direção x , isto é, com os autovalores da matriz jacobiana da EDP unidimensional, que envolve a variável x e Sy velocidades das ondas elementares na direção y [88];

- Esta separação de dimensão pode ocasionar oscilações na solução numérica e, por mais que o método unidimensional seja de alta ordem de precisão, a separação dimensional (*Splitting dimensional*) feita pelas equações (3.80) e (3.81) resulta em um método de primeira ordem. Para mais detalhes e *Splitting* dimensional de segunda ordem, ver [34, 53, 88].
- Comumente, opta-se pelo *Splitting* dimensional quando aplica-se métodos dependentes de Riemann Solvers, isto é, quando se tem dificuldades de aplicar a discretização bidimensional sem separar os fluxos. Por outro lado, métodos sem separação de dimensão são apropriados para Leis de Balanço.

3.9.2 Métodos sem separação dimensional

Métodos sem separação de dimensão, também chamados de Métodos *Unsplit*, fazem as discretizações dos fluxos em uma única expressão e sua evolução de t_n até t_{n+1} é feita de tal modo que todas as contribuições de cada fluxo são realizada ao mesmo tempo, isto é, em um único passo.

Assim, a discretização da EDP (3.77) na forma de métodos de volumes finitos conservativos é dada por

$$U_{i,j}^{n+1} = U_{i,j}^n - \frac{\Delta t}{\Delta x} [F_{(1);i+1/2,j}^n - F_{(1);i-1/2,j}^n] - \frac{\Delta t}{\Delta y} [F_{(2);i,j+1/2}^n - F_{(2);i,j-1/2}^n] . \quad (3.82)$$

Na forma deslocada, como no caso dos métodos Lax-Friedrichs deslocado e Nessyahu-Tadmor deslocado, tem-se

$$U_{i+1/2,j+1/2}^{n+1} = U_{i+1/2,j+1/2}^n - \frac{\Delta t}{\Delta x} [F_{(1);i+1,j}^n - F_{(1);i,j}^n] - \frac{\Delta t}{\Delta y} [F_{(2);i,j+1}^n - F_{(2);i,j}^n] . \quad (3.83)$$

3.9.3 O Método Nessyahu-Tadmor Bidimensional

Nesta seção apresenta-se o método de Nessyahu e Tadmor em um domínio bidimensional no espaço, tomando como base o que foi desenvolvido na seção 3.5 e, em especial, na subseção do método Nessyahu-Tadmor.

Considere uma partição regular do domínio espacial $\Omega = \Omega_x \times \Omega_y$ em que

$$\Omega_x : x_a = x_1 < \dots < x_i < \dots < x_{N+1} = x_b \text{ e } \Omega_y : y_a = y_1 < \dots < y_j < \dots < y_{M+1} = y_b,$$

com N subintervalos na direção x e M na direção y em que

$$\Delta x = \frac{x_b - x_a}{N} \text{ e } \Delta y = \frac{y_b - y_a}{M},$$

para quaisquer valores de i e j existentes. A i,j -ésima *célula média* ou o i,j -ésimo *volume*

finito é definido por, $\Omega_{i,j} = (x_{i-1/2}, x_{i+1/2}) \times (y_{j-1/2}, y_{j+1/2})$ em que $x_{i+1/2} = \frac{x_{i+1} + x_i}{2}$ e $y_{i+1/2} = \frac{y_{i+1} + y_i}{2}$.

Seja $u = u(x, y, t)$ uma função que representa uma grandeza física definida do domínio espacial Ω . Se u é conservada em Ω , então a lei de conservação é satisfeita em cada célula média ou volume finito $\Omega_{i,j}$, isto é,

$$\begin{aligned} \frac{d}{dt} \int_{\Omega_{i,j}} u(x, y, t) dx = & f(u(x_{i+1/2}, y_j, t)) - f(u(x_{i-1/2}, y_j, t)) + \\ & + g(u(x_i, y_{j+1/2}, t)) - g(u(x_i, y_{j-1/2}, t)) . \end{aligned} \quad (3.84)$$

Ao definir uma partição regular para a variável independente t em que $\Delta t = t_{n+1} - t_n$, um algoritmo explícito de evolução temporal de um único passo é obtido integrando (3.84), no tempo entre os valores t_n e t_{n+1} da seguinte maneira,

$$\begin{aligned} \frac{1}{\Delta x \Delta y} \int_{\Omega_{i,j}} u(x, y, t_{n+1}) dx = & \frac{1}{\Delta x \Delta y} \int_{\Omega_{i,j}} u(x, y, t_n) dx + \\ & + \frac{1}{\Delta x \Delta y} \int_{t_n}^{t_{n+1}} \int_{y_{j-1/2}}^{y_{j+1/2}} [f(u(x_{i-1/2}, y, t)) - f(u(x_{i+1/2}, y, t))] dy dt + \\ & + \frac{1}{\Delta x \Delta y} \int_{t_n}^{t_{n+1}} \int_{x_{i-1/2}}^{x_{i+1/2}} [g(u(x, y_{j-1/2}, t)) - g(u(x, y_{j+1/2}, t))] dx dt \end{aligned} \quad (3.85)$$

A equação (3.85) fornece um procedimento para encontrar os valores de u no tempo t_{n+1} , nos pontos (x_i, y_j) . Nota-se que tal equação integral é exata (não possui aproximações) e é válida para todo subintervalo de Ω .

Como o intuito é obter um **Método de Volumes Finitos Centrado** livre de Riemann Solvers, considera-se a lei de conservação (3.85) nas células médias deslocadas, isto é,

$$I_{i,j} = (x_i, x_{i+1}) \times (y_j, y_{j+1}) ,$$

assim a equação (3.85) é reescrita da seguinte forma

$$\begin{aligned}
\frac{1}{\Delta x \Delta y} \int_{I_{i,j}} u(x, y, t_{n+1}) dx &= \frac{1}{\Delta x \Delta y} \int_{I_{i,j}} u(x, y, t_n) dx + \\
&+ \frac{1}{\Delta x \Delta y} \int_{t_n}^{t_{n+1}} \int_{y_j}^{y_{j+1}} [f(u(x_i, y, t)) - f(u(x_{i+1}, y, t))] dy dt + \quad (3.86) \\
&+ \frac{1}{\Delta x \Delta y} \int_{t_n}^{t_{n+1}} \int_{x_i}^{x_{i+1}} [g(u(x, y_j, t)) - g(u(x, y_{j+1}, t))] dx dt
\end{aligned}$$

Agora, considera-se $w = w(x, y, t)$ uma função polinomial bilinear por partes em que

$$w(x, y, t_n) = \sum_i \omega_{i,j}(x, y) p_{i,j}(x, y), \quad \text{com } \bar{p}_{i,j}(x_i, y_j) = \bar{w}_{i,j}^n \text{ e } \omega_{i,j} = 1. \quad (3.87)$$

Assim, o valor médio de $w(x, y, t_n)$ definido na célula média deslocada $I_{i,j}$ é dado por

$$\begin{aligned}
\bar{w}_{i+1/2, j+1/2}^n &= \frac{1}{\Delta x \Delta y} \left[\int_{x_i}^{x_{i+1/2}} \int_{y_j}^{y_{j+1/2}} p_{i,j}(x, y) dy dx + \int_{x_{i+1/2}}^{x_{i+1}} \int_{y_j}^{y_{j+1/2}} p_{i+1,j}(x, y) dy dx \right] + \\
&\frac{1}{\Delta x \Delta y} \left[\int_{x_i}^{x_{i+1/2}} \int_{y_{j+1/2}}^{y_{j+1}} p_{i,j+1}(x, y) dy dx + \int_{x_{i+1/2}}^{x_{i+1}} \int_{y_{j+1/2}}^{y_{j+1}} p_{i+1,j+1}(x, y) dy dx \right] \quad (3.88)
\end{aligned}$$

Considerando w como uma aproximação para u e substituindo a equação (3.88) em (3.86), obtém-se

$$\begin{aligned}
\bar{w}_{i+1/2, j+1/2}^{n+1} &= \bar{w}_{i+1/2, j+1/2}^n + \\
&+ \frac{1}{\Delta x \Delta y} \int_{t_n}^{t_{n+1}} \int_{y_j}^{y_{j+1}} [f(w(x_i, y, t)) - f(w(x_{i+1}, y, t))] dy dt + \quad (3.89) \\
&+ \frac{1}{\Delta x \Delta y} \int_{t_n}^{t_{n+1}} \int_{x_i}^{x_{i+1}} [g(w(x, y_j, t)) - g(w(x, y_{j+1}, t))] dx dt .
\end{aligned}$$

Como w é uma função polinomial bilinear por partes, então

$$p_{i,j}(x) = \bar{w}_{i,j}^n + \frac{D_x w_{i,j}}{\Delta x} (x - x_i) + \frac{D_y w_{i,j}}{\Delta y} (y - y_j) , \quad (3.90)$$

em que

$$D_x w_{i,j} = MM \{ \Delta_x w_{i+1/2,j}^n , \Delta_x w_{i-1/2,j}^n \} , \text{ com } \Delta_x w_{i+1/2,j}^n = w_{i+1,j}^n - w_{i,j}^n$$

e

$$D_y w_{i,j} = MM \{ \Delta_y w_{i,j+1/2}^n , \Delta_y w_{i,j-1/2}^n \} , \text{ com } \Delta_y w_{i,j+1/2}^n = w_{i,j+1}^n - w_{i,j}^n .$$

Substituindo $p_{i,j}$ na equação (3.88), obtém-se uma expressão para $\bar{w}_{i+1/2,j+1/2}^n$. Assim,

$$\begin{aligned} \bar{w}_{i+1/2,j+1/2}^n &= \frac{1}{4} (\bar{w}_{i,j}^n + \bar{w}_{i+1,j}^n + \bar{w}_{i,j+1}^n + \bar{w}_{i+1,j+1}^n) + \\ &+ \frac{1}{16} [(D_x w_{i,j} - D_x w_{i+1,j}) + (D_x w_{i,j+1} - D_x w_{i+1,j+1})] + \\ &+ \frac{1}{16} [(D_y w_{i,j} - D_y w_{i,j+1}) + (D_y w_{i+1,j} - D_y w_{i+1,j+1})] . \end{aligned} \quad (3.91)$$

Agora falta encontrar expressões para as integrais dos fluxos da equação (3.91) para determinar o método Nessyahu-Tadmor bidimensional. Para isso, utiliza-se a regra do ponto médio para a integração temporal e a regra do trapézio para a integração espacial. Logo,

$$\int_{t_n}^{t_{n+1}} \int_{y_j}^{y_{j+1}} f(w(x_i, y, t)) dy dt \cong \frac{\Delta y \Delta t}{2} [f(w_{i,j}^{n+1/2}) + f(w_{i,j+1}^{n+1/2})] \quad (3.92)$$

e

$$\int_{t_n}^{t_{n+1}} \int_{x_i}^{x_{i+1}} g(w(x, y_j, t)) dx dt \cong \frac{\Delta x \Delta t}{2} [g(w_{i,j}^{n+1/2}) + g(w_{i+1,j}^{n+1/2})] , \quad (3.93)$$

em que

$$w_{i,j}^{n+1/2} = \bar{w}_{i,j}^n - \frac{\Delta t}{2\Delta x} D_x f_{i,j} - \frac{\Delta t}{2\Delta y} D_y g_{i,j}, \quad (3.94)$$

com $D_x f_{i,j}$ e $D_y g_{i,j}$ definidos pela função *minmod* semelhante ao caso da equação (3.90) e do método Nessyahu-Tadmor unidimensional no espaço.

Portanto, voltando na equação (3.89) chega-se na expressão procurada

$$\begin{aligned} \bar{w}_{i+1/2,j+1/2}^{n+1} &= \frac{1}{4} (\bar{w}_{i,j}^n + \bar{w}_{i+1,j}^n + \bar{w}_{i,j+1}^n + \bar{w}_{i+1,j+1}^n) + \frac{1}{16} [(D_x w_{i,j} - D_x w_{i+1,j})] + \\ &+ \frac{1}{16} [(D_x w_{i,j+1} - D_x w_{i+1,j+1}) + (D_y w_{i,j} - D_y w_{i,j+1}) + (D_y w_{i+1,j} - D_y w_{i+1,j+1})] + \\ &- \frac{\Delta t}{2\Delta x} [f(w_{i+1,j}^{n+1/2}) - f(w_{i,j}^{n+1/2})] - \frac{\Delta t}{2\Delta x} [f(w_{i+1,j+1}^{n+1/2}) - f(w_{i,j+1}^{n+1/2})] + \\ &- \frac{\Delta t}{2\Delta y} [g(w_{i,j+1}^{n+1/2}) - g(w_{i,j}^{n+1/2})] - \frac{\Delta t}{2\Delta y} [g(w_{i+1,j+1}^{n+1/2}) - g(w_{i+1,j}^{n+1/2})]. \end{aligned} \quad (3.95)$$

As equações (3.95) e (3.91) com a definição da função *minmod* - ver equação (3.90), definem o **método Nessyahu-Tadmor bidimensional** em malha deslocada. Para mais detalhes e aplicações em Magnetohidrodinâmica Clássica consulte a referência [6].

Nota-se que, se a função w for constante por partes tem-se $D_x w = D_y w = D_x f = D_y g = 0$ e então, a equação (3.95) se reduz ao **método Lax-Friedrichs bidimensional** em malha deslocada. A extensão do método Nessyahu-Tadmor para problemas hiperbólicos associados as Leis de Balanço encontra-se na referência [70].

Capítulo 4

Métodos Numéricos aplicados em problemas de GRMHD

Este capítulo expõe as razões pelas quais optamos pela implementação computacional dos métodos Lax-Friedrichs e Nessyahu-Tadmor, para realizar simulações de jatos relativísticos, desde a acreção de disco na direção de um objeto compacto central até a ejeção de matéria.

4.1 Autovalores e Autovetores

Pesquisas sobre os autovalores e autovetores de sistemas de equações de magnetohidrodinâmica, em ambiente de relatividade geral, mantêm-se em desenvolvimento tanto com o objetivo de conhecer as estruturas de ondas que se propagam no escoamento de fluidos relativísticos magnetizados, quanto para possibilitar a aplicabilidade dos métodos que necessitam de Riemann Solvers exatos.

O conhecimento de todas as ondas elementares podem fornecer informações sobre, por exemplo, qual tipo de onda mais influencia na grande liberação de energia via jatos, entre outras. Como a velocidade de cada onda elementar está associada com os autovalores e suas propriedades com os autovetores, o processo de obtenção dos autopares são fundamentais para esses estudos.

Além disso, tais informações são importantes para a implementação de métodos numéri-

cos do tipo Godunov e, também, para a construção de soluções exatas de referência de problemas unidimensionais no espaço, que servem para testar os comportamentos dos métodos numéricos desenvolvidos e mostrar a confiabilidade de novos códigos implementados.

Um artigo que estuda os autovalores, os autovetores à direita e os autovetores à esquerda baseado nas equações (2.1)-(2.4) de GRMHD e na equação de entropia foi publicado por Anile e Pennisi [1]. Este artigo apresentou um espaço de dez dimensões cujas variáveis são as quatro componentes do quadrivetor velocidade, as quatro componentes do quadrivetor eletromagnético, a pressão e a entropia. Tais autopares são inicialmente estudados no espaço-tempo de Minkowski $1+1$, isto é, com uma dimensão espacial e uma dimensão temporal.

Um estrutura local de características aplicada em métodos de Godunov dependentes de resolvidores de problemas de Riemann foi proposto por [7]. Em seu trabalho, a dificuldade maior consistiu em aplicá-los em espaços com mais dimensões devido a existência de autovalores degenerados. Uma alternativa nem sempre aplicável é utilizar o *Splitting* dimensional e, assim, para cada dimensão espacial tem-se um sistema de EDP's dependendo de apenas de uma dimensão espacial.

No caso das equações de GRMHD em espaços curvos, além dos autovalores possuem degenerescência, existem autovalores ligados a fenômenos não físicos [41, 42]. Tais referências removem as ondas não físicas e desenvolvem um método de Godunov para GRMHD baseado na estrutura local de características da relatividade especial, isto é, usa-se o fato de que espaços curvos da Relatividade Geral são localmente planos [7].

Giacomazzo e Rezzolla [27] desenvolveram procedimentos gerais para resolver analiticamente problemas de Riemann em MHD com relatividade especial, para quaisquer condições iniciais em problemas unidimensionais no espaço.

Apesar dos avanços das pesquisas sobre as estruturas de ondas elementares em GRMHD, a aplicabilidade de métodos de Godunov dependentes de problemas de Riemann exatos ou aproximados em problemas de formação de jatos desde, a acreção de disco até a sua ejeção permanecem como um grande desafio.

Problemas desse tipo são bidimensionais ou tridimensionais no espaço e ao aplicar

métodos do tipo Godunov com Riemann Solvers, dependem do *Splitting* dimensional, isto traz restrições à aplicabilidade desses métodos, pois nesses casos, as magnitudes dos autovalores não podem ter grandes ordens de diferença.

Entretanto, no processo de transição entre acreção de matéria via disco e sua ejeção há uma variação brusca de pressão nas proximidades do objeto compacto central e neste momento os sistemas de EDP's separados, para cada dimensão espacial, possuem autovalores de diferentes ordens de magnitudes, com grandes diferenças. Tal situação é exatamente o caso em que métodos que utilizam separações dimensionais podem falhar, conseqüentemente, métodos *Unsplitting* são altamente recomendados e esses métodos podem ser formulados em esquemas de volumes finitos centrados.

Além disso, uma dificuldade adicional é que o modelo em questão não é uma lei de conservação hiperbólica, mas sim uma lei de balanço.

Três fatos interessantes de se notar são:

- Desde o primeiro artigo de Koide, Shibata e Kudoh em 1999 [38] até suas pesquisas recentes sobre a formação de jatos relativísticos [40] - 2011, o método utilizado é o Lax-Wendroff adicionado de viscosidade artificial com uma discretização temporal TVD, com *Splitting* dimensional e limitadores de inclinação para evitar as oscilações geradas pela separação de dimensões espaciais;
- Os recentes códigos em GRMHD mais utilizados não incluem problemas de formação de jatos desde a acreção até a ejeção;
- Os métodos numéricos aplicados em GRMHD investigados nos principais códigos e artigos utilizam o recurso de separar as dimensões espaciais (*Splitting* dimensional).

Um dos objetivos dessa tese é exatamente desenvolver um código totalmente novo e voltado para o problema de formação de jatos relativísticos implementando outros métodos de volumes finitos centrados, além do esquema Lax-Wendroff TVD utilizado por Koide *et al* [38, 40].

Na seção seguinte, apresentamos alguns códigos mais utilizados recentemente para a modelagem de problemas em hidrodinâmica e magnetohidrodinâmica relativística.

4.2 Códigos em Magnetohidrodinâmica Relativística

Alguns dos códigos mais recentemente aplicados na modelagem de problemas em hidrodinâmica e magnetohidrodinâmica relativística são expostos nessa seção.

4.2.1 HARM

Um dos primeiros códigos disponíveis que envolvem SRMHD e GRMHD foi o **HARM** - *High Accuracy Relativistic Magnetohydrodynamics* (Magnetohidrodinâmica Relativística de Alta Precisão) [26].

O código HARM disponibiliza os métodos de Godunov-HLL com diferentes possibilidades de fluxos numéricos e limitadores de inclinação para extrapolações lineares para o centro das células médias, também tem-se opções de fluxos que preservam a divergência nula do campo magnético. Problemas bidimensionais utilizam-se do *splitting* dimensional e no código original não há discretização tridimensional espacial. A escrita é feita em *linguagem C* com a possibilidade de paralelismo via *MPI - Message Passing Interface* (interface de troca de informações entre processadores).

Os exemplos e aplicações que o código HARM disponibiliza são [26, 24]:

- a) *Ondas Elementares*: captura e propagação de ondas não lineares unidimensionais no espaço de fluidos magnetizados com velocidades relativísticas (relatividade especial);
- b) *Transporte*: este exemplo modela um disco que modifica sua densidade conforme sua rotação relativística;
- c) *vórtice de Orszag-Tang*: uma versão relativística (relatividade especial) do clássico problema de MHD;
- d) *Escoamento de Bondi*: este problema modela acreção de matéria com simetria esférica na métrica de Schwarzschild, tanto para fluidos neutros quanto para fluidos magnetizados;
- e) *Acreção em métrica de Kerr*: a modelagem deste problema considera o estado estacionário de uma acreção de fluido magnetizado dentro de uma órbita marginalmente

estável, em particular, este problema mostra a interação entre o fluido magnetizado em órbita equatorial e a rotação do buraco negro de Kerr;

- f) *Toros de plasma em equilíbrio*: este exemplo simula órbitas estáveis de toros de plasma que rodeiam um buraco negro central.

4.2.2 `cosmos++`

O programa `cosmos++` foi desenvolvido com base no método numérico conhecido como esquema de Viscosidade Artificial [3]. Como o próprio nome diz, o método acrescenta uma viscosidade artificial com o intuito de diminuir a dissipação numérica.

As equações implementadas pelo código são as equações de GRMHD multidimensional com possibilidade de refinamento e adaptação dos pontos da malha, conhecido como AMR - Adaptive Mesh Refinement (malha adaptativa com refinamento).

O código é feito em linguagem `C++` e utiliza os princípios de objetos orientados para criar classes de abstrações matemáticas para vetores e tensores (ambos em três ou quatro dimensões), métricas, condições de contorno, operadores e malhas não estruturadas. O programa é paralelizado via MPI.

Uma outra opção de método numérico é o esquema Kurganov-Tadmor [49], que é um método de volumes finitos centrado que utiliza informações sobre as velocidades de propagação das discontinuidades, a saber, o método utiliza o maior autovalor em módulo da matriz Jacobiana. Após a semi-discretização espacial, a integração temporal de t_n até t_{n+1} é feita através de uma sequência de N métodos de Euler (Runge-Kutta de primeira ordem) - conforme a escolha do usuário [3]. Nesta opção o código faz *Splitting* dimensional.

Os termos fontes das equações de GRMHD são acrescentados através de um *Splitting* resolvendo primeiro as equações de GRMHD sem termo fonte e depois atualiza as soluções resolvendo uma equação diferencial através da sequência de métodos de Euler.

Os exemplos que acompanham o código original são:

Hidrodinâmica Relativística

- a) Tubo de Choque;

- b) Colisão de ondas de Choque;
- c) Ondas de Explosão.

Magnetohidrodinâmica Relativística

- d) Propagação de ondas de Alfvén;
- e) Tubo de Choque Magnetosônico;
- f) Escoamento de Bondi;
- g) Toro Magnetizado orbitando um Buraco Negro.

4.2.3 whiskyMHD

A grande vantagem do código *whiskyMHD* é o fato de implementar o conjunto completo de equações de GRMHD, isto é, o modelo inclui a dinâmica do espaço-tempo. As discretizações das equações têm como base o método Godunov-HLL, com estrutura de refinamento de malha AMR e ferramentas computacionais que possibilitam paralelismo [28].

Numa versão anterior, o código *whisky*, a evolução temporal das equações é feita pelo *método de linhas* que reduz equações diferenciais parciais em conjuntos de equações diferenciais ordinárias - EDO's. Tais EDO's podem ser resolvidas pelos métodos de Runge-Kutta ou esquemas de Cranck-Nicholson iterativos [51, 84].

Além dos exemplos listados, o código disponibiliza estudos em [28]:

- a) Binários de estrelas de Nêutrons Magnetizados ou não, evolução de Estrelas de Nêutrons;
- b) Radiação Gravitacional;
- c) Escoamento de Bondi;
- d) Colapso de núcleo estelar, relativístico e magneto-rotacional;
- e) Extensão para hidrodinâmica com simetria axial;

- f) Formação de Buracos Negros e problemas de binários de Buracos Negros;
- g) Cenários com campo magnéticos forte, acreção em Buracos Negros;
- h) Cosmologia Relativística.

4.2.4 ECHO

O código Eulerian Conservative High Order - ECHO (Euleriano Conservativo de Alta Ordem) inclui o método essencialmente não oscilatório - ENO de terceira ordem e o método essencialmente não oscilatório ponderado - WENO de quinta ordem com integração temporal realizada por métodos de Runge-Kutta de múltiplos estágios. Os métodos de Godunov-HLL e o Godunov-Roe (com resolvidor de problema de Riemann desenvolvido por Roe) são implementados e existe a possibilidade de utilizar vários limitadores de suavidade [17]. Também tem-se à disposição métodos de diferenças finitas.

O programa ECHO também tem implementado o MHD clássico e em ambos os códigos têm-se *Splitting* dimensional. A escrita do código é feita em Fortran 90 com MPI, utilizando-se a estrutura modular disponível em tal linguagem.

Existe um módulo de programação que possibilita o acréscimo de termos resistivos e, também, um outro módulo que modela instabilidades em MHD.

Um exemplo que se destaca é o modelo em MHD com relatividade especial que descreve a Nebulosa do Caranguejo - Crab Nebula, mostrando a estrutura de jatos formados em Toros.

4.2.5 WHARM

Este código é um melhoramento do código HARM que acrescenta esquemas essencialmente não oscilatórios ponderados (WENO) e também, os resolvidores de problemas de Riemann Generalizados [88], em que a extrapolação dentro da célula média é feita por interpolação polinomial e não somente mais por função constante.

O programa implementa o conjunto completo das equações de GRMHD e além dos problemas já listados, o WHAM modela a evolução de jatos relativísticos em geometria cilíndrica possibilitando investigar a formação de algumas subestruturas [85].

A discretização do domínio é feita por partição regular com possibilidade de escolher regiões onde pretende-se apenas refinar a malha. O esquema WENO implementado é o de quinta ordem para regiões suaves com *Splitting* dimensional e Runge-Kutta de terceira ordem TVD para a integração temporal. Em regiões próximas de descontinuidade, o algoritmo é capaz de modificar o tamanho do estêncil para evitar ao máximo propagações de oscilações na solução numérica.

Um outro método adicionado ao código é o método Parabólico por partes - PPM de Woodward & Colella [93] que é uma extensão do método de Godunov dependente de Riemann Solvers.

4.2.6 CoCoA/CoCoNuT

Este código é destinado a exemplos de colapso de núcleo estelar em GRMHD com dinâmica do espaço-tempo (equações de Einstein) [24, 98].

A diferença deste código é que além dos métodos do tipo Godunov dependentes de Riemann Solvers, o programa possui métodos espectrais implementados.

Nele são apresentados vários exemplos de colapso de núcleo e esta característica fornece o nome para o código: Colapso de Núcleos com novas tecnologias - CoCoNuT (Core Collapse with Nu (=new) Technology) e Colapso de Núcleos com simetria axial - CoCoA (Core Collapse in Axisymmetry).

4.2.7 GRHydro

Apesar desse código ser totalmente independente do *whiskyMHD*, o *GRHydro* pode ser acoplado ao código *whiskyMHD* [66]. Esse algoritmo possui diversos pacotes de ferramentas que complementam o código tais como: *einstein toolkit* - um conjunto de códigos de Relatividade Geral que possibilita diversas métricas, AMR, MPI com estrutura para bons escalonamentos na casa dos milhares de núcleos de processamentos, entre outros [66].

Além dos exemplos citados, o código apresenta a modelagem de estrela de nêutrons esférica magnetizada obtida via equação de TOV - Tolman-Oppenheimer-Volkhoff [13] e o

colapso estelar de um núcleo estelar magnetizado com espaço-tempo dinâmico. Também, tem-se o problema de um rotor relativístico.

4.2.8 Outros Códigos

Códigos em Magnetohidrodinâmica Relativística continuam em pleno desenvolvimento também em outros grupos além dos que implementaram os códigos mencionados. A referência [24] possui outros códigos comentados, porém todos utilizam métodos *Splitting* dimensional.

Uma implementação em Magnetohidrodinâmica com Relatividade Especial baseada em métodos de Galerkin foi realizada por Dumbser e Zanotti [19]. Os exemplos estudados foram: tubos de choque; Rotor; explosão cilíndrica e vórtice de Orszag-Tang.

4.2.9 Observações

Nos códigos expostos neste capítulo verifica-se uma ênfase à métodos do tipo Godunov dependentes de resolvedores de problemas de Riemann, que aplicam *Splitting* dimensional para estender a aplicabilidade de métodos unidimensionais no espaço para problemas em dimensões maiores.

Implementações que efetuam *Splitting* dimensional podem gerar oscilações na solução numérica e uma tentativa de controlá-las é aplicar limitadores de inclinação.

Também percebe-se a ênfase em métodos que estende os métodos de Godunov, os métodos do tipo MUSCL - Monotone Upstream-centered Schemes for Conservation Laws, dependentes de Riemann Solvers. Além disso, outros métodos que podem ter uma formulação de volumes finitos centrados são implementados na versão do tipo Godunov dependentes de Riemann Solvers, tais como: ENO's e WENO's.

4.3 Métodos Implementados

Um dos objetivos desta tese foi a implementação de métodos numéricos de volumes finitos centrados, isto é, livres de Riemann Solvers, com o intuito de não realizar *Splitting*

dimensional. Métodos que não separam as dimensões espaciais são chamados de métodos *Unsplitting*.

Os métodos implementados foram o Lax-Friedrichs, Lax-Wendroff com Runge-Kutta de terceira ordem TVD e o Nessyahu-Tadmor em um problema de formação de jatos relativísticos, desde a acreção de disco até a extensão da ejeção de matéria, através das equações de GRMHD. A descrição da implementação computacional é feita no capítulo seguinte, a linguagem de programação utilizada foi o Fortran 90 e o código foi totalmente desenvolvido durante o doutorado, sem o uso de pacotes prontos.

Capítulo 5

Implementações Computacionais

Este capítulo apresenta as equações implementadas e a estrutura do código desenvolvido em Fortran 90. As equações que modelam o problema de formação de jatos relativísticos foram especificadas no capítulo 2 e na seção 5.1, tais equações são reescritas com o intuito de suas implementações.

As seções seguintes tratam das equações complementares do modelo em estudo, impõem as condições iniciais e de contorno necessárias para o método numérico e, também, para a modelagem do problema.

Por fim, a última seção é destinada aos arquivos e aos algoritmos criados para a obtenção de um programa capaz de realizar simulações de cenários astrofísicos de formação de jatos relativísticos.

5.1 As equações Implementadas

As equações que foram implementadas no código desenvolvido nesta tese estão expostas nesta seção. A partir das equações do capítulo 2, juntamente com algumas hipóteses chegou-se nas equações de interesse, que estão nos algoritmos escritos.

Além das hipóteses e equações apresentadas no capítulo 2, consideramos o problema em estudo com **simetria axial**, isto é, o problema não possui variação com relação a variável azimutal ϕ . Isto faz com que o problema seja bidimensional no espaço, ou seja, as variáveis espaciais independentes são a radial r e a polar θ . Assim, com $x_1 = r$, $x_2 = \theta$

e a hipótese de simetria axial, as equações de GRMHD se reduzem à

Equação da Massa:

$$\frac{\partial D}{\partial t} = -\frac{1}{J} \left\{ \frac{\partial}{\partial x^1} (h_0 h_2 h_3 D v_1) + \frac{\partial}{\partial x^2} (h_0 h_3 h_1 D v_2) \right\} \quad (5.1)$$

Equações do Movimento:

$$\frac{\partial P_1}{\partial t} = -\frac{1}{J} \left\{ \frac{\partial}{\partial x^1} (h_0 h_2 h_3 T^{11}) + \frac{\partial}{\partial x^2} (h_0 h_3 h_1 T^{12}) \right\} + S_2 \quad (5.2)$$

$$\frac{\partial P_2}{\partial t} = -\frac{1}{J} \left\{ \frac{\partial}{\partial x^1} (h_0 h_2 h_3 T^{21}) + \frac{\partial}{\partial x^2} (h_0 h_3 h_1 T^{22}) \right\} + S_3 \quad (5.3)$$

$$\frac{\partial P_3}{\partial t} = -\frac{1}{J} \left\{ \frac{\partial}{\partial x^1} (h_0 h_2 h_3 T^{31}) + \frac{\partial}{\partial x^2} (h_0 h_3 h_1 T^{32}) \right\} + S_4 \quad (5.4)$$

Equação da Energia:

$$\begin{aligned} \frac{\partial \epsilon}{\partial t} = & -\frac{1}{J} \frac{\partial}{\partial x^1} [h_0 h_2 h_3 (P_1 - D v_1)] + \\ & -\frac{1}{J} \left\{ \frac{\partial}{\partial x^2} [h_0 h_3 h_1 (P_2 - D v_2)] + \right\} + S_5 \end{aligned} \quad (5.5)$$

Equações do Campo Magnético:

$$\frac{\partial B_1}{\partial t} = -\frac{h_1}{J} \left\{ \frac{\partial}{\partial x^2} (h_0 h_3 E_3) \right\} \quad (5.6)$$

$$\frac{\partial B_2}{\partial t} = -\frac{h_2}{J} \left\{ -\frac{\partial}{\partial x^1} (h_0 h_3 E_3) \right\} \quad (5.7)$$

$$\frac{\partial B_3}{\partial t} = -\frac{h_3}{J} \left\{ \frac{\partial}{\partial x^1} (h_0 h_2 E_2) - \frac{\partial}{\partial x^2} (h_0 h_1 E_1) \right\} \quad (5.8)$$

Equações - Termo Fonte:

$$S_2 = h_0 \{(\epsilon + D) G_{01} + G_{12}T^{21} + G_{13}T^{31} - G_{21}T^{22} - G_{31}T^{33}\} \quad (5.9)$$

$$S_3 = h_0 \{(\epsilon + D) G_{02} + G_{23}T^{32} + G_{21}T^{12} - G_{32}T^{33} - G_{12}T^{11}\} \quad (5.10)$$

$$S_4 = h_0 \{(\epsilon + D) G_{03} + G_{31}T^{13} + G_{32}T^{23} - G_{13}T^{11} - G_{23}T^{22}\} \quad (5.11)$$

$$S_5 = h_0 \{G_{01}P_1 + G_{02}P_2 + G_{03}P_3\} \quad (5.12)$$

Apesar das reduções nas equações de GRMHD devido a simetria axial, no programa foram implementados todos os fluxos, com o intuito de facilitar futuras alterações que incluem a terceira variável espacial. Entretanto, o armazenamento das variáveis dependentes foram feitos por *array's* bidimensionais. O código é discutido na seção 5.3, juntamente com a apresentação de sua estrutura.

5.2 Equações Complementares

Para a realização de simulações, as equações da seção 5.1 não são suficientes, pois estas dependem da métrica que representa o objeto compacto central, que no caso é um buraco negro de Schwarzschild. Uma outra necessidade é obter o campo de velocidade, o campo de pressão, o campo elétrico para conseguir construir as componentes do tensor momento-energia e os fluxos das equações diferenciais. Também, para as equações diferenciais são necessários as condições iniciais e de contorno adequadas, que complementam a modelagem do problema em estudo. A seguir, indicaremos quais expressões que foram utilizadas para tal objetivo.

A gravitação pré-determinada está representada pela métrica de Schwarzschild, que foi exposta na seção 2.2. Com ela têm-se os coeficientes da métrica que são necessários para obter os coeficientes $G_{\mu\nu}$, que aparece no termo fonte e acrescenta a gravitação nas equações diferenciais, e para fazer as correções nos fluxos das EDP's devido à curvatura

imposta pelo buraco negro de Schwarzschild. Tal métrica possui o conjunto de coordenadas $(c.t, r, \theta, \phi)$ conhecido como coordenadas de Boyer-Lindquist.

Além da métrica de Schwarzschild, uma transformação foi aplicada na coordenada radial transformando-a na coordenada radial tortoise (coordenada tartaruga) definida por

$$x = \ln \left(\frac{r}{r_s} - 1 \right) .$$

Ao aplicá-la, uma malha uniforme neste novo sistema de coordenadas, a largura da malha radial é proporcional a $(r - r_s)$ no sistema de coordenadas de Boyler-Lindquist. Assim, a largura da malha torna-se cada vez menor. Logo, uma partição regular na coordenada tartaruga implica em uma malha não uniforme em coordenadas Boyler-Lindquist. Este recurso deixa a malha mais fina próxima do buraco negro, que é justamente a região crítica onde ocorre a transição entre disco e jato, e torna a malha mais larga no extremo oposto.

Essa transformação é muito importante, caso contrário seriam necessários muitos pontos na malha em coordenadas Boyler-Lindquist para estabilizar o método numérico ou ainda recorrer a técnicas de refinamento e adaptação de pontos da malha, os chamados *Adaptive Mesh Refinement - AMR*. Logo, consegue-se uma alternativa para estabilizar os métodos em malhas com menos pontos e sem usar recursos *AMR*, que tornam os métodos mais caros e extensos.

Portanto, no código desenvolvido tem-se implementado:

Gravitação pré-determinada: métrica de Schwarzschild

$$ds^2 = -\alpha^2 dt^2 + \frac{1}{\alpha^2} dr^2 + r^2 d\theta^2 + r^2 \sin^2 \theta d\phi^2$$

em que

$$\alpha = \sqrt{1 - \frac{r_s}{r}}, \quad \text{com } c = G = 1, \quad M = \frac{1}{2} r_s \text{ (massa BH)}$$

$$h_0 = \alpha, \quad h_1 = \frac{1}{\alpha}, \quad h_2 = r, \quad h_3 = r \sin \theta,$$

r_s - raio de Schwarzschild - unidade de referência;

Expressões que envolvem as componentes da métrica:

$$G_{\mu\nu} = -\frac{1}{h_\mu h_\nu} \left(\frac{\partial}{\partial x^\nu} h_\mu \right), \quad \text{para } \mu, \nu = 0, 1, 2, 3, 4,$$

e

Coordenada Tartaruga:

$$x = \ln \left(\frac{r}{r_s} - 1 \right) .$$

Nota-se que, a motivação para realizar a transformação para a coordenada tartaruga é puramente numérica.

Outras duas relações importantes foram implementadas, a equação de estado composta por um sistema algébrico de duas equações com duas incógnitas [20, 63] - Seção 2.4,

$$\begin{aligned} x(x+1) [\Gamma a x^2 + (2\Gamma a - b)x + \Gamma a - b + d \frac{\Gamma}{2} y^2]^2 = \\ = (\Gamma x^2 + 2\Gamma x + 1)^2 [\tau^2 (x+1)^2 + 2\sigma y + 2\sigma x y + \beta^2 y^2] , \end{aligned}$$

$$\begin{aligned} [\Gamma(a - \beta^2)x^2 + (2\Gamma a - 2\Gamma\beta^2 - b)x + \Gamma a - b + d - \beta^2 + \frac{\Gamma}{2}y] y \\ = \sigma(x+1)(\Gamma x^2 + 2\Gamma x + 1) , \end{aligned}$$

em que

$$x = \gamma - 1, \quad y = \gamma(\mathbf{v} \cdot \mathbf{B}), \quad a = D + \epsilon, \quad b = (\Gamma - 1)D, \quad d = (1 - 0,5\Gamma)B^2,$$

$$\tau = P, \quad \beta = B \text{ e } \sigma = \mathbf{B} \cdot \mathbf{P},$$

com as aproximações iniciais do método Newton-Raphson sendo as soluções x e y obtidas no passo temporal anterior.

A condição *frozen-in* que fornece uma expressão para o campo elétrico impondo que a matéria em estudo é livre de forças eletromagnéticas, isto é,

Campo Elétrico:

$$\mathbf{E} = -\mathbf{v} \times \mathbf{B}$$

Por fim, o **tensor momento-energia** é dado por

$$\mathbf{T} = p\mathbf{I} + (e + p)\gamma^2\mathbf{v}\mathbf{v} - \mathbf{B}\mathbf{B} - \mathbf{E}\mathbf{E} + \frac{1}{2}(B^2 + E^2)\mathbf{I} .$$

Contudo, faltam as condições iniciais e de contorno para completar a discretização das equações implementadas no código.

Condições Iniciais:

As condições iniciais devem modelar o disco de acreção descrito na seção 2.3, juntamente com a magnetosfera e o buraco negro central. Assim, implementamos uma estrutura modular em Fortran 90 especialmente para lidar com tal etapa, que é a construção das equações da seção 2.3.

O buraco negro de Schwarzschild está implementado na estrutura modular, dentro da subrotina chamada de métrica.

No caso das condições de contorno, nos raios interno e externo, a matéria deve continuar com sua dinâmica além do domínio. Isso faz com que nessas regiões, as condições de contorno seja do tipo radiação. Agora, nos contornos $\theta = 0$ e $\theta = \frac{\pi}{2}$ têm-se uma simetria do tipo espelho, isto é, condições de contorno do tipo reflexão.

Condições de Contorno (Direção Radial): Raios interno ($r = 1, 1r_s$) e externo ($r = 20r_s$) - radiação.

Condições de Contorno (Direção Angular): $\theta = 0$ e $\theta = \frac{\pi}{2}$ - reflexão.

A condição de contorno $r = 20r_s$ delimitada pela abertura do disco, também possibilitou uma subrotina que implementa a alimentação do disco de acreção, ou seja, uma condição de contorno do tipo Dirichlet.

5.3 O código

O programa desenvolvido foi todo escrito em Fortran 90 utilizando o compilador *gfortran* versão 4.7.3 em um Linux/GNU. Na estrita do código utilizou-se a estrutura de programação modular que o Fortran 90 disponibiliza para a formação de conjuntos de subrotinas, que podem ser chamadas totalmente ou parcialmente conforme as necessidades específicas.

Neste contexto, um recurso interessante para o controle das variáveis declaradas nas subrotinas dos módulos de programação é o comando `INTENT`, que determina se a variável declarada é de entrada (IN), de saída (OUT) ou de entrada e saída (INOUT). Em uma subrotina, uma variável declarada com a intenção apenas de entrada (variável de entrada) não pode ser alterada ao longo da estrita da subrotina. Caso isso ocorra, o compilador alegará um erro facilitando um controle das variáveis.

Um outro controle causado pelo comando `INTENT` é que, as alocações de memória das variáveis de entrada e das variáveis INOUT estão fora da subrotina em questão, no caso das variáveis de saída, sua alocação é feita na própria subrotina e suas características são exportadas para fora da subrotina, onde foi definida.

A estrutura de bloco principal chamada de `PROGRAM` está escrita no arquivo *programa_principal.f90*. Assim, o código disponibiliza os seguintes módulos de programação - escritos sem acentuação - conforme o nome do arquivo com extensão **.f90* :

armazenamentos

Este módulo possui uma subrotina chamada de *arquivos_saida* que salva as variáveis relacionadas com a solução numérica em arquivos abertos nas unidades especificadas pelo programa. Os arquivos de saída após uma evolução temporal são:

- *campo.dat* - armazena a intensidade do campo magnético;
- *densidade.dat* - armazena a densidade;
- *energia.dat* - armazena a energia tota;
- *malha_x1.dat* - armazena as coordenadas r dos pontos da malha;

- *malha_x2.dat* - armazena as coordenadas θ dos pontos da malha;
- *pressao.dat* - armazena a intensidade da pressão;
- *velocidade.dat* - armazena a intensidade da velocidade;

atualizacoes

Neste módulo está a subrotina *grandezas_atualizadas* que atualiza as variáveis campo elétrico, campo de velocidades e o tensor momento-energia através da equação de estado e da condição *frozen-in*. Nessa subrotina utiliza-se o módulo responsável por resolver a equação de estado através do método Newton-Raphson, que depende das variáveis densidade, momento, energia total e campo magnético para retornar o campo de velocidade e o fator de Lorentz em cada ponto da malha.

condicoes_fronteras

O conjunto de subrotinas desse módulo disponibiliza as condições de contorno utilizadas. Existem 4 subrotinas de condições de contorno sendo que todas possuem condições de reflexão nos extremos do intervalo em que θ está definida, sendo assim, o que difere de uma subrotina para a outra são as condições nos extremos do intervalo em que r está definido. Assim, as subrotinas são:

- *condicoes_contorno* - raios interno e externo condições de radiação;
- *condicoes_contorno2* - raios interno e externo extrapolação da solução numérica;
- *condicoes_contorno3* - raio interno radiação e raio externo extrapolação;
- *condicoes_contorno4* - fronteiras radiais da magnetosfera radiação e fronteira radial do disco extrapolação (disco alimentado);

condicoes_iniciais

Este módulo possui duas subrotinas que definem as condições iniciais do problema em estudo. As subrotinas utilizam os valores de entrada definidos no módulo valores iniciais. As duas subrotinas são:

- *cond_iniciais* - condições iniciais definidas sem coordenada tartaruga;
- *cond_iniciais2* - condições iniciais definidas em coordenadas tartaruga;

discretizacoes

As duas subrotinas desse módulo definem a malha computacional, os coeficientes da métrica, algumas combinações dos coeficientes da métrica e o tensor G . Os nomes das subrotinas são: *malha* e *metrica*.

EOS

Este módulo possui a subrotina que resolve o sistema de equações que forma a equação de estado EOS através do método de Newton-Raphson [71, 75], cujo nome é *newton_metEOS*.

ferramentas

No módulo ferramentas possuem as subrotinas necessárias para realizar produtos vetoriais, produtos escalares, normas, interpolações (extrapolações) pelo método de Lagrange [71, 75] e equações algébricas pelo método de Newton-Raphson.

fluxos

Neste módulo estão definidos os fluxos das equações diferenciais implementadas na subrotina chamada de *fluxos_EDPs*.

metodos_EDPs

As três subrotinas deste módulo são destinadas aos métodos numéricos disponíveis neste código:

- *nes_tad* método Nessyahu-Tadmor sem separação dimensional;
- *lax_fried* método Lax-Friedrichs sem separação dimensional;
- *lax_wend* método Lax-Wendroff com RK3-TVD *Splitting*;

programa _principal

Este bloco consiste do programa principal que faz as chamadas dos blocos de programação em ordem adequada para realizar a evolução temporal da solução numérica. Após o tempo final atingido, o programa chama a subrotina *arquivos_saida* para armazenar a solução final. Note que no início do programa tem-se o comando USE que chama os módulos utilizados no programa principal.

Cada subrotina que necessite de algum módulo ou parte de módulo de programação, também possui o comando USE no início da escrita, antes das declarações das variáveis.

Na Figura 5.1, no retângulo com duplo traço na vertical estão os nomes dos módulos de programação utilizados pelo programa principal. Tais módulos definem as subrotinas pré-determinada que podem ser utilizadas chamando-as pelo comando CALL.

Em seguida, o programa chama as subrotinas que preparam o cenário inicial do fenômeno em estudo deixando-o pronto para efetuar o laço de repetição principal.

Dentro deste laço, os fluxos das equações diferenciais (5.1)-(5.8) são calculados para que a subrotina *metodos_numericos* é chamada para obter soluções aproximadas. As soluções obtidas são das equações sem termo fonte, assim o próximo passo do programa é atualizar as soluções resolvendo as equações diferenciais ordinárias (Seção 3.8) definidas pelos termos (5.9)-(5.12).

Nas próximas duas subrotinas, o programa prepara as condições de contorno e atualiza as variáveis necessárias para iniciar um próximo passo temporal. A última subrotina armazena tais variáveis mantendo o último cenário da evolução temporal salvo.

solucao _numerica

Neste módulo está a subrotina *metodos_numericos* que é destinada a chamar os métodos numéricos para resolver cada equação diferencial.

termos _adicionais

O termo fonte é acrescentado através da subrotina *fonte*, onde está implementado o método de Euler com N passos.

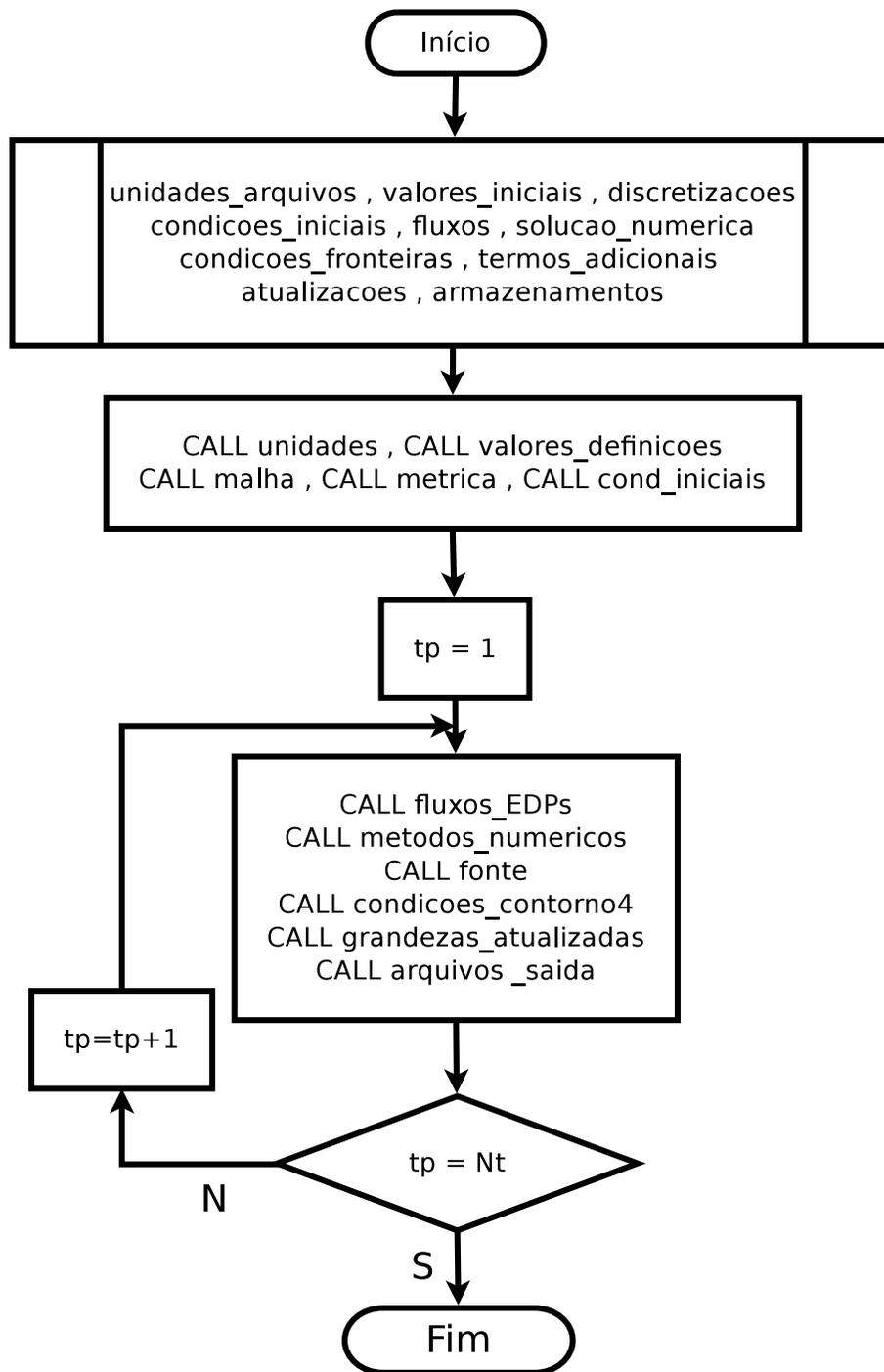


Figura 5.1: Fluxograma do programa principal.

unidades_arquivos

Este módulo possui a subrotina *unidades* responsável por abrir ou criar os arquivos de entrada e saída de dados.

valores_iniciais

Na subrotina *valores_definicoes* deste módulo estão definidos os limites do domínio computacional, o número σ que relaciona Δt , com Δr e $\Delta \theta$, quantidade de subintervalos em cada intervalo computacional, a definição do número π e a quantidade de interações que o laço principal realizará para chegar no tempo final desejado.

Capítulo 6

Simulações

Este capítulo foi destinado à exposição dos gráficos das simulações de jatos relativísticos provenientes da execução do código desenvolvido em Fortran 90. Primeiro, um exemplo é executado utilizando o Método Lax-Friedrichs Bidimensional e sem separação dimensional e depois um segundo exemplo é simulado com o uso do Método Nessyahu-Tadmor Bidimensional sem separação dimensional.

6.1 Exemplo 1

O primeiro exemplo realiza a evolução temporal de um disco fino de acreção magnetizado, em rotação com relação ao eixo vertical e em queda na direção do buraco negro central de Schwarzschild, imersos em uma magnetosfera de baixa densidade com relação ao disco. Para isto, utilizou-se o método Lax-Friedrichs bidimensional sem separação de dimensões espaciais com intuito de testar o código, verificar tendências das soluções numéricas e expor algumas características do método numérico e do fenômeno em estudo.

A seguir, estão expostos valores utilizados nas simulações:

- **Domínio Espacial:** $[1, 1.20] \times [0, \pi/2]$, isto é $r \in [1, 1.20]$ e $\theta \in [0, \pi/2]$ em radianos;
- **Subintervalos:** 630×630 , assim $\Delta r = 18,9/210$ e $\Delta\theta = \pi/420$;
- **Espaçamento Temporal:** $\Delta t = 0,1\Delta r$;

- **Constantes - Matéria:** Constante relacionada com calor específico - $\Gamma = 5/3$ e entalpia - $H = 1,3$;
- **Abertura do Disco:** $\delta = 0,125$, isto é, o disco ocupa um oitavo da abertura do domínio;
- **Densidade do Disco:** 401 vezes maior que a densidade da magnetosfera;
- **Borda interna do disco:** $r_D = 3r_S$, três vezes o raio do BH;

Na seção 2.3 foi descrito o disco de acreção, uma propriedade inicial é que o disco está em rotação com o perfil da velocidade Kepleriana. Sendo assim, a velocidade da borda interna é dado por

$$v_K = \frac{c}{\sqrt{2(3-1)}} \Rightarrow v_K = 0,5c,$$

isto é, a velocidade angular da borda interna é de 50% da velocidade da luz. Logo, o tempo necessário para a borda interna completar uma volta é de aproximadamente 12,6 unidades de tempo.

A borda externa do disco está localizado em $20r_S$ e sua velocidade Kepleriana é de aproximadamente de 16% da velocidade da luz. Assim, seu período é de 775 unidades de tempo.

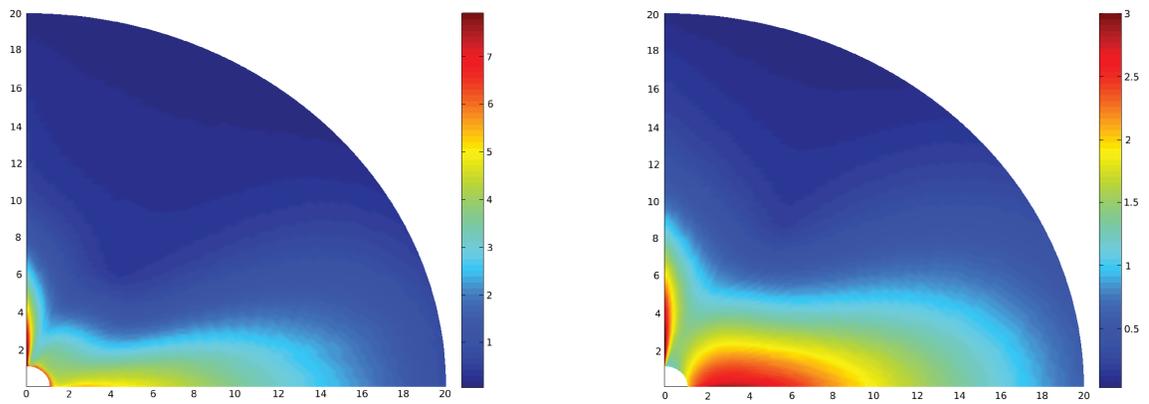


Figura 6.1: **Densidade:** à esquerda tem-se a evolução da densidade no tempo final de 11,8 e à direita no tempo final em 19,6.

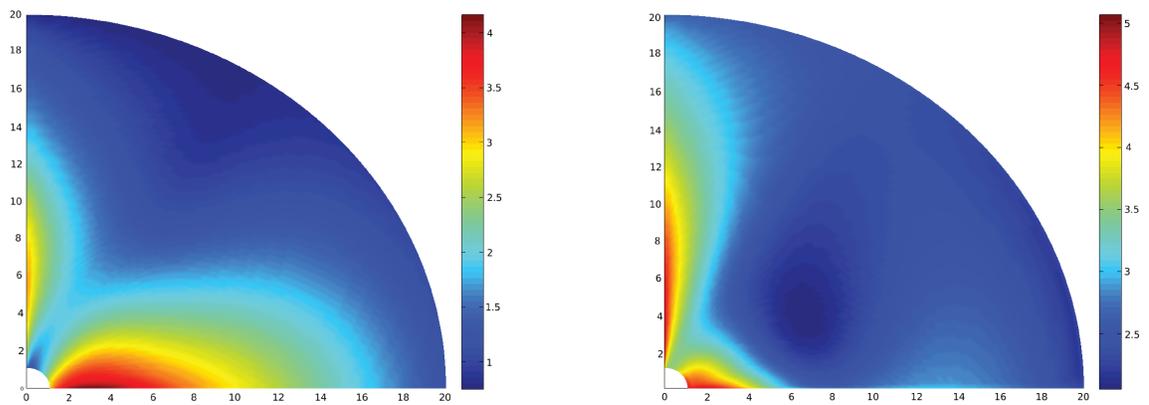


Figura 6.2: **Densidade:** à esquerda tem-se a evolução da densidade no tempo final de 39,3 e à direita no tempo final em 58,9.

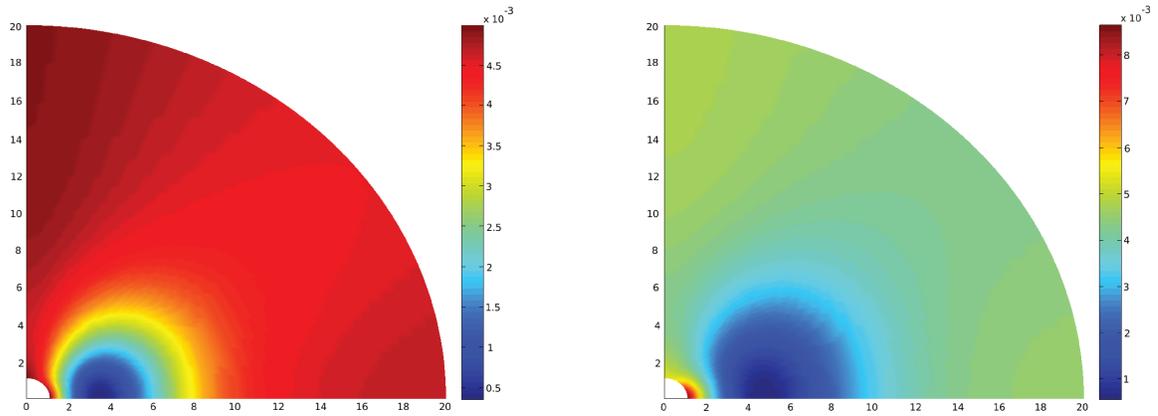


Figura 6.3: **Intensidade do Campo Magnético:** à esquerda tem-se a evolução da intensidade do campo magnético no tempo final de 11,8 e à direita no tempo final em 19,6.

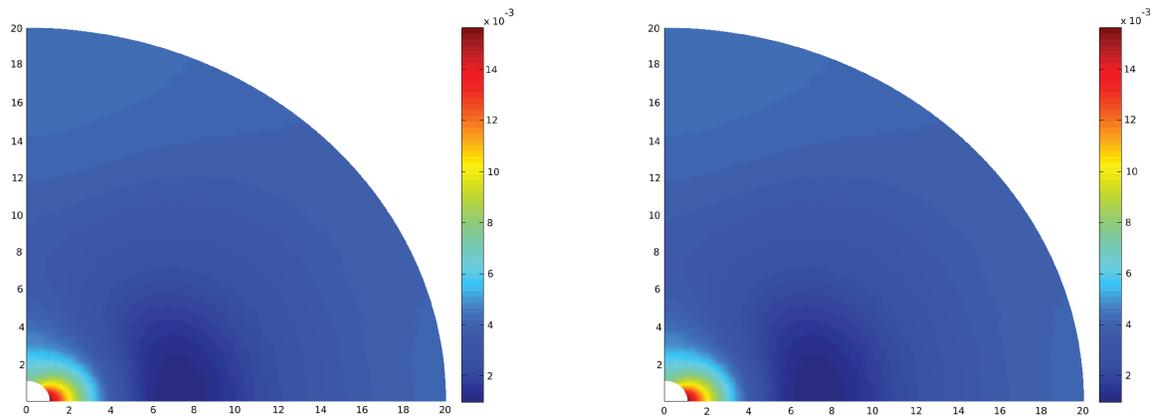


Figura 6.4: **Intensidade do Campo Magnético:** à esquerda tem-se a evolução da intensidade do campo magnético no tempo final de 39,3 e à direita no tempo final em 58,9.

6.2 Exemplo 2

Neste exemplo aplicamos o método Nessyahu-Tadmor bidimensional sem separação de dimensões espaciais, exposto na seção 3.9.3, com método de Euler de quatro estágios para o acréscimo do termo fonte. Os valores utilizados foram:

- **Domínio Espacial:** $[1, 1 - 20] \times [0, \pi/2]$, isto é $r \in [1, 1 - 20]$ e $\theta \in [0, \pi/2]$ em radianos;
- **Subintervalos:** 630×630 , assim $\Delta r = 18,9/630$ e $\Delta\theta = \pi/1260$;
- **Espaçamento Temporal:** $\Delta t = 401.0,005 \cdot \min\{\Delta r, \Delta\theta\}$;
- **Constantes - Matéria:** Constante relacionada com calor específico - $\Gamma = 5/3$ e entalpia - $H = 1,3$;
- **Abertura do Disco:** $\delta = 0,125$, isto é, o disco ocupa um oitavo da abertura do domínio;
- **Densidade do Disco:** 401 vezes maior que a densidade da magnetosfera;
- **Borda interna do disco:** $r_D = 3r_S$, três vezes o raio do BH;

Neste caso, resolveu-se normalizar os valores iniciais com relação ao valor 401 com o intuito de fazer com que inicialmente o método numérico trabalhe com valores menores. Esta normalização é chamada de fator escala e pode-se encontrar em métodos numéricos aplicados em dinâmicas de fluidos [88]. Para uma melhor visualização dos gráficos usamos o logaritmo de base 10 no eixo vertical dos valores da densidade, pressão e energia total.

A seguir têm-se os gráficos das condições iniciais.

As Figuras 6.8 - 6.11 acompanham a evolução temporal da densidade. De início tem-se apenas o disco de acreção em rotação com relação ao eixo vertical e a magnetosfera em queda em direção ao buraco negro. Depois, a simulação passa pela transição entre acreção e ejeção e a partir desse momento percebe-se a ejeção de matéria em forma de jato colimado ao longo do eixo vertical. Por fim, o jato estende-se até próximo da distância de 15 vezes o raio do buraco negro central. Nesses exemplos a unidade de referência é o raio

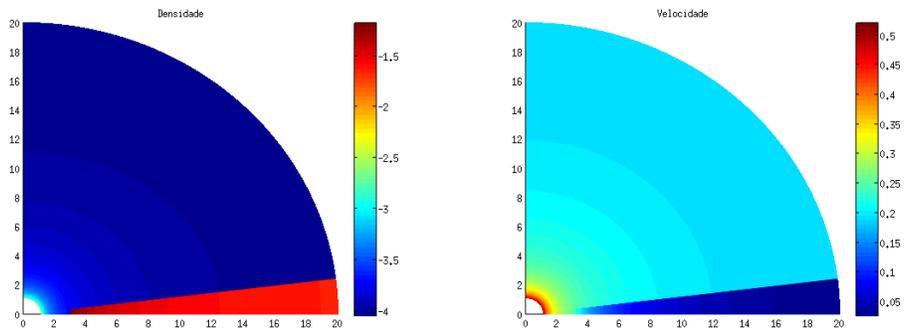


Figura 6.5: À esquerda encontra-se o gráfico da condição inicial da densidade e à direita da intensidade da velocidade.

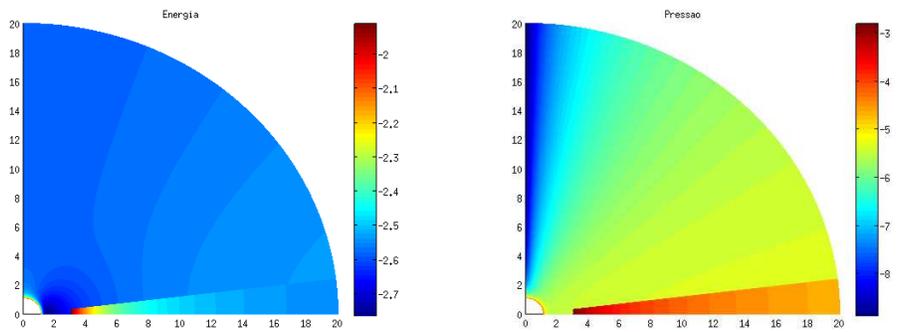


Figura 6.6: À esquerda tem-se o gráfico da condição inicial da energia total e à direita da intensidade da pressão.

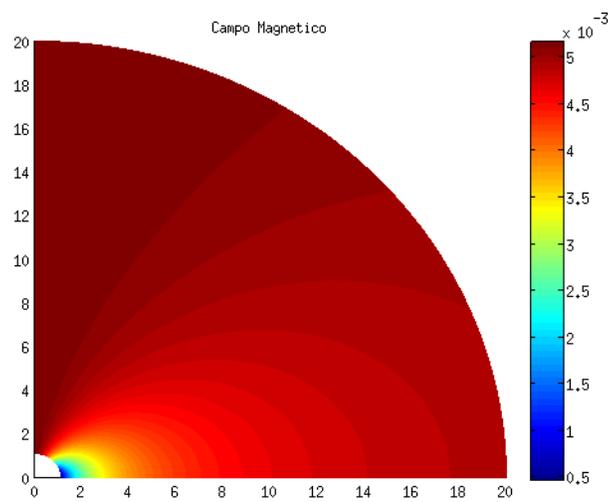


Figura 6.7: Condição inicial da intensidade do campo magnético.

do BH e nos gráficos cada unidade nos eixos verticais e horizontais corresponde a unidade de referência. O BH posiciona-se centrado na origem dos gráficos e seu raio é exatamente a unidade.

No jato colimado ejetado destaca-se a formação de subestruturas e um jato bem definido quando comparado ao jato formado no Exemplo 1.

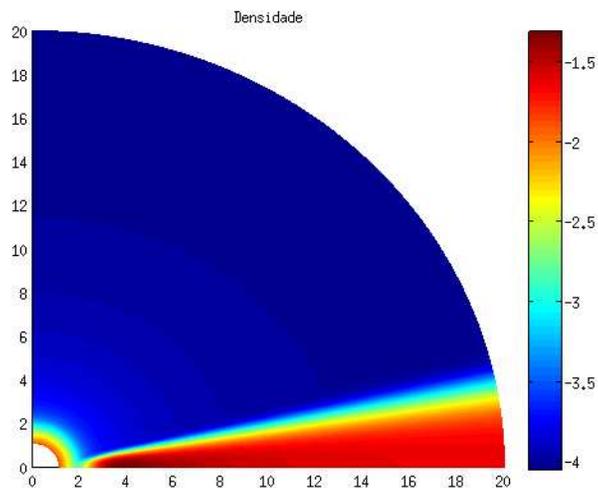


Figura 6.8: **Densidade:** solução numérica no tempo de 11,8.

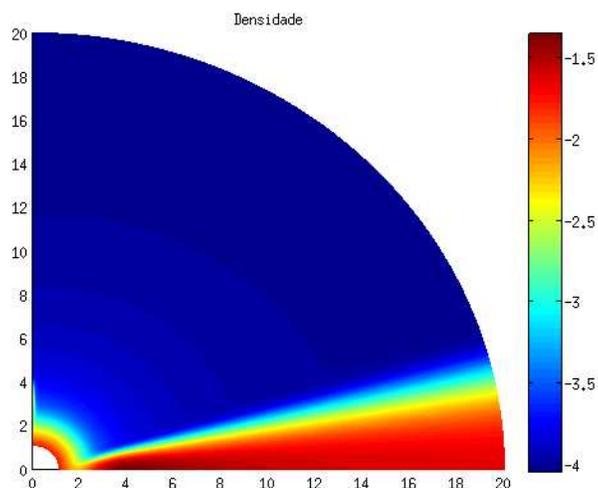


Figura 6.9: **Densidade:** solução numérica no tempo de 19,6.

As Figuras 6.12 - 6.15 mostram a evolução temporal da intensidade da velocidade. Apenas na Figura 6.15 nota-se modificações na velocidade que acompanham as subestruturas

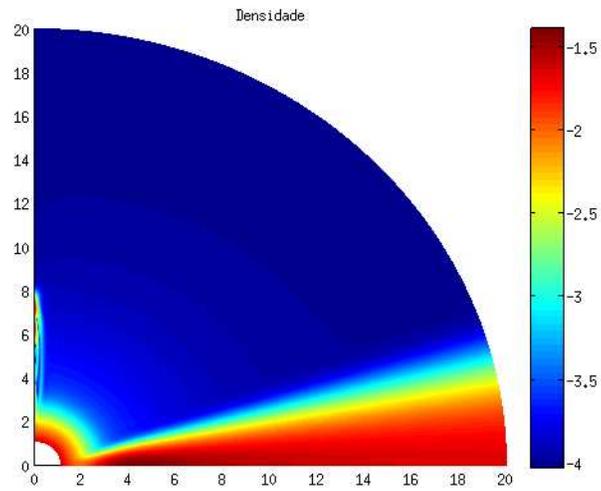


Figura 6.10: **Densidade:** solução numérica no tempo de 39,3.

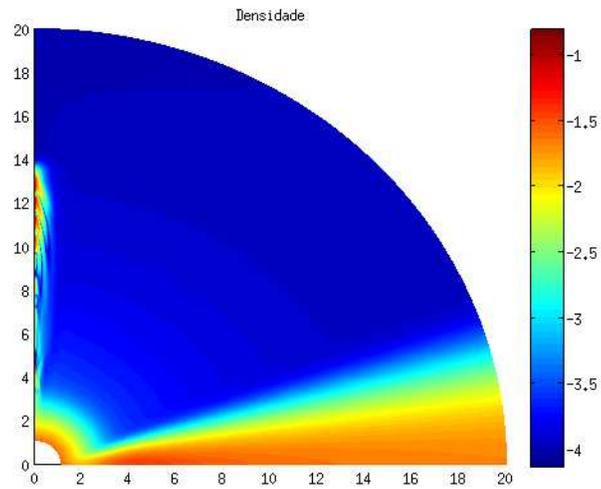


Figura 6.11: **Densidade:** solução numérica no tempo de 58,9.

turas que se destacam no jato colimado.

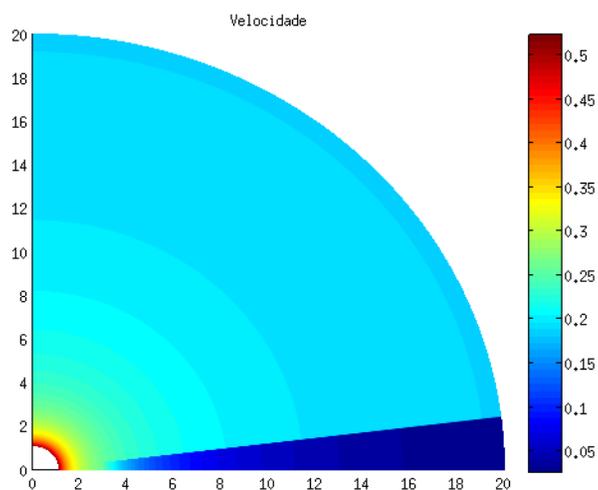


Figura 6.12: **Intensidade da Velocidade:** solução numérica no tempo de 11,8.

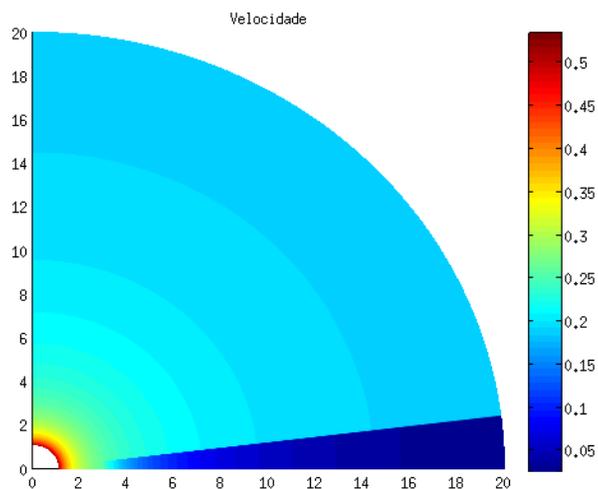


Figura 6.13: **Intensidade da Velocidade:** solução numérica no tempo de 19,6.

As Figuras 6.16 - 6.19 apresentam a evolução temporal da intensidade da pressão. Ao observar tais figuras percebeu-se uma estrutura sendo formada no disco de acreção no eixo horizontal entre os valores de 2 e 3. Esta é uma subestrutura que não fica permanente e na Figura 6.19 nota-se seu desaparecimento. Nas Figuras relacionadas com as soluções numéricas da energia total também percebe-se tal subestrutura.

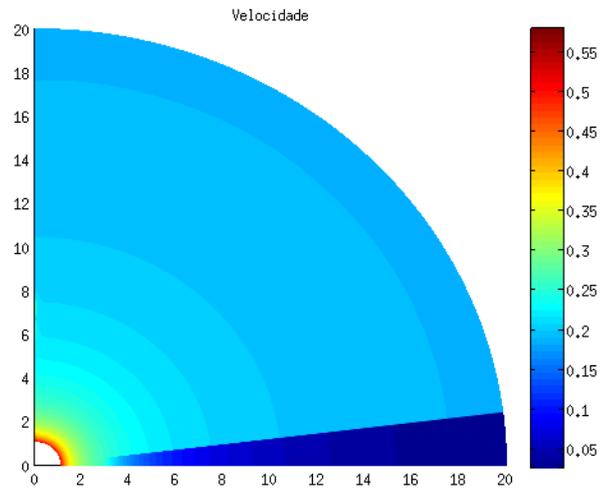


Figura 6.14: **Intensidade da Velocidade:** solução numérica no tempo de 39,3.

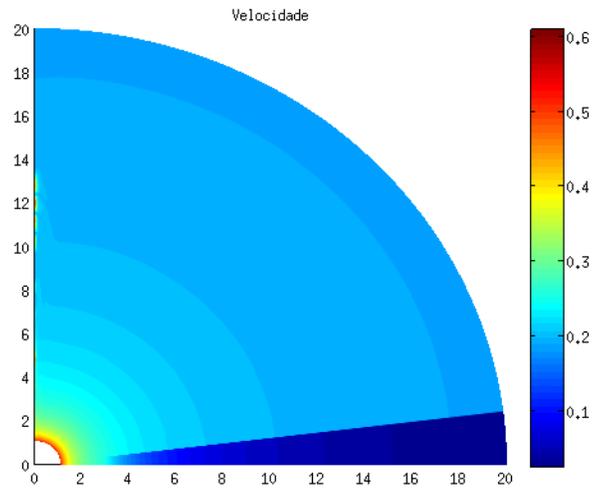


Figura 6.15: **Intensidade da Velocidade:** solução numérica no tempo de 58,9.

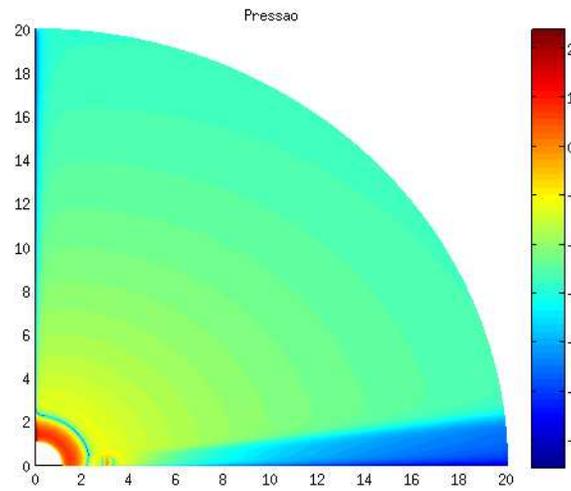


Figura 6.16: **Intensidade da Pressão:** solução numérica no tempo de 11,8 .

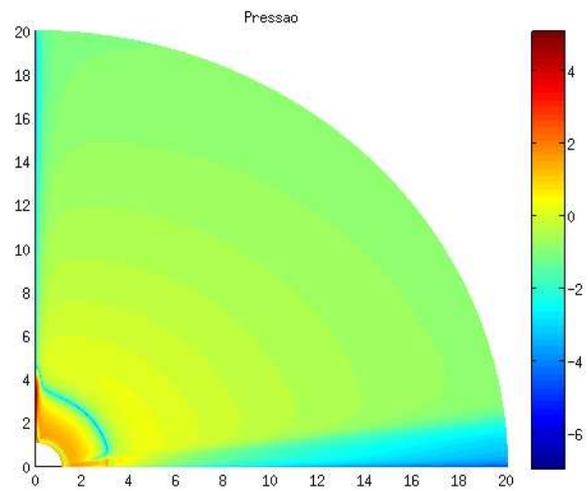


Figura 6.17: **Intensidade da Pressão:** solução numérica no tempo de 19,6 .

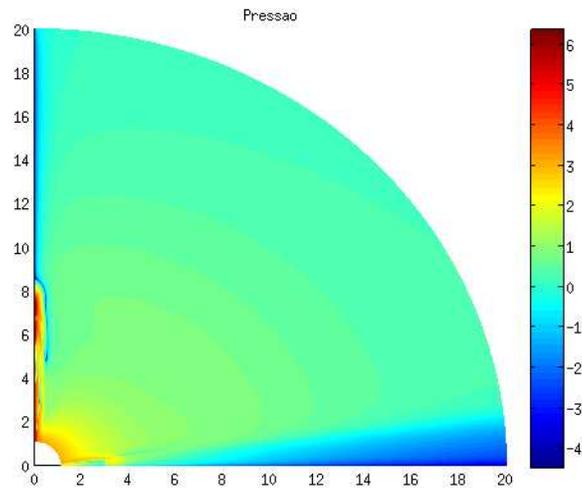


Figura 6.18: **Intensidade da Pressão:** solução numérica no tempo de 39,2 .

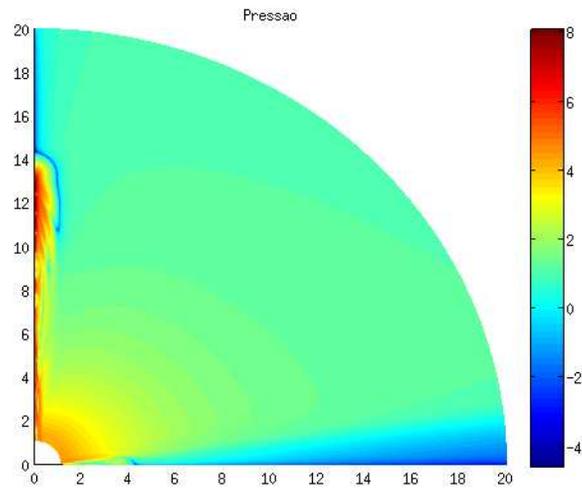


Figura 6.19: **Intensidade da Pressão:** solução numérica no tempo de 58,9 .

As Figuras 6.20 - 6.23 ilustram a evolução temporal da energia total e nesses gráficos destaca-se a formação do jato relativístico e algumas subestruturas formadas. Também, reforça-se uma subestrutura formada no disco de acreção.

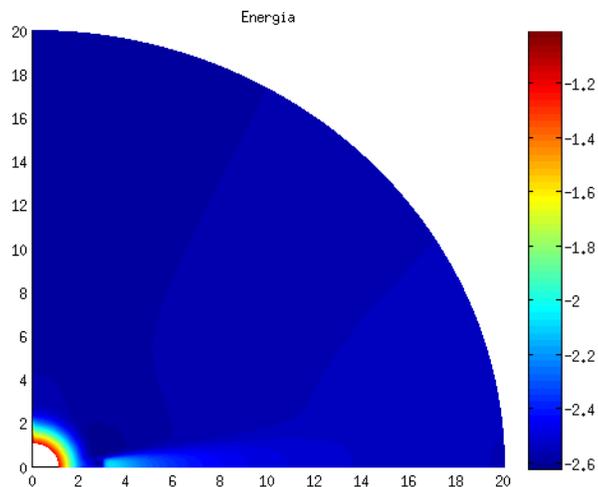


Figura 6.20: **Energia Total:** solução numérica no tempo de 11,8.

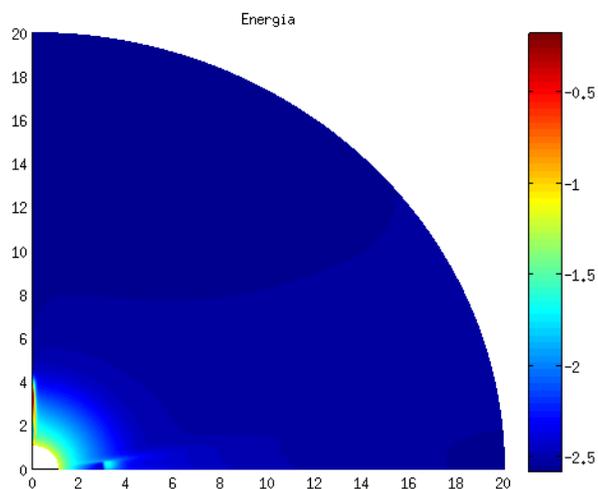


Figura 6.21: **Energia Total:** solução numérica no tempo de 19,6.

Com relação ao tempo de CPU gasto por cada método implementado, o método Lax-Friedrichs gastou aproximadamente quatro horas para obter a solução no tempo final de 58,9 enquanto que o método Nessyahu-Tadmor demorou quase 20 horas para executar tal simulação. Isto reflete a importância de se desenvolver primeiro o código em Lax-

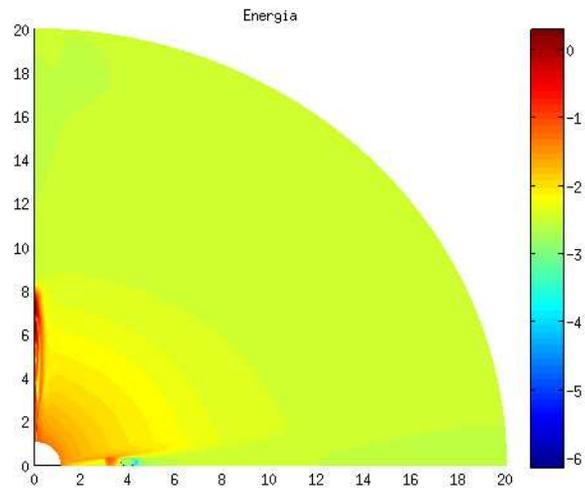


Figura 6.22: **Energia Total:** solução numérica no tempo de 39,2.

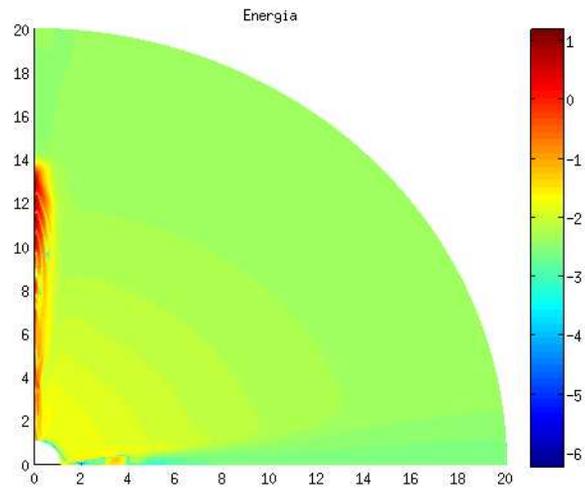


Figura 6.23: **Energia Total:** solução numérica no tempo de 58,9.

Friedrichs. Assim, depois de ganhar confiabilidade no código desenvolvido em Fortran 90, com pequenas alterações acrescentou-se o método Nessyahu-Tadmor.

Neste exemplo não foram acrescentado os gráficos da evolução temporal do campo magnético por serem muito semelhantes aos gráficos do exemplo anterior.

6.3 Exemplo 3

O exemplo 3 refere-se ao mesmo exemplo 2, porém o método aplicado é o Lax-Wendroff com Runge-Kutta de terceira ordem TVD, também, com *Splitting* dimensional e sem limitadores de inclinações. Neste caso, não foi possível obter uma solução estável e com isso verificou-se que nem sempre a propriedade TVD é capaz de estabilizar o método numérico. O módulo que contém os métodos numéricos estão em anexo e o método encontra-se disponível como opção, pois pode ser útil em algum outro exemplo.

Capítulo 7

Considerações Finais e Trabalhos

Futuros

7.1 Considerações Finais

Neste trabalho pesquisou-se métodos de Volumes Finitos Centrados, a saber: Lax-Friedrichs, Lax-Wendroff's e Nessyahu-Tadmor aplicados em problemas de formação de jatos relativísticos desde a queda de um disco de acreção, em rotação para um buraco negro (BH) central dado pela métrica de Schwarzschild, até a ejeção de fluido pelos polos do BH.

De início problematizamos o fenômeno em estudo destacando a importância na Astrofísica Moderna, a complexidade das equações envolvidas e as dificuldades em modelar, implementar e simular computacionalmente a formação de jatos relativísticos. Utilizou-se a formulação das equações de Magnetohidrodinâmica Relativística com Gravitação (GRMHD) proposta por Koide, Shibata e Kudoh [38], juntamente com aspectos fenomenológicos descritos em [8, 9, 38, 24, 87].

Depois, passou-se ao estudo de métodos de Volumes Finitos Centrados livres de resolvidores de problemas de Riemann. Após a exposição de propriedades e características associadas à convergência, consistência, estabilidade, dissipação e dispersão numéricas dos métodos de Volumes Finitos Centrados Unidimensionais, independentes de problemas de Riemann, um método dependentes de Riemann Solvers - o método Godunov desenvolvido por Harten, Lax e Leer (HLL) [31] - foi apresentado com o intuito de realizar comparações.

Implementamos computacionalmente os métodos Lax-Friedrichs clássico, Lax-Wendroff clássico, Lax-Wendroff com Runge-Kutta de três estágios e a propriedade de ser Valor Total Decrescente (TVD) no tempo, Nessyahu-Tadmor em malha deslocada e o método Godunov-HLL (extraído de [88] e adaptado em Octave) para simular escoamento de fluido ideal modelado pelas equações de Euler unidimensionais. Embora seja um problema unidimensional, o sistema de equações de Euler possui toda complexidade e desafios de se aplicar métodos numéricos para obter soluções aproximadas em escoamento de fluidos.

Nos exemplos expostos destacaram-se: o caráter dissipativo do método Lax-Friedrichs e seu baixo custo computacional; o caráter dispersivo do método Lax-Wendroff acrescentando oscilações na solução numérica que, por sua vez, de certo modo pode ser controlada se a discretização temporal for feita por um método TVD, com a ressalva de que sua região de estabilidade é mais restritiva; a boa representação de descontinuidades do método Nessyahu-Tadmor, sem precisar de informações extras, porém com um aumento no custo computacional e na restrição na relação entre Δt e Δx , e o bom comportamento do método Godunov-HLL e seu bom desempenho computacional, todavia, sua aplicabilidade é dificultada pela dependência de Riemann Solvers, autovalores e autovetores da matriz jacobiana e da decomposição dimensional (*Splitting dimensional*) em problemas de duas ou mais dimensões espaciais.

No capítulo 4 apresentamos e os principais códigos em GRMHD e alguns dos problemas que estão disponíveis em seus pacotes. Mesmo sabendo que os autovalores das equações de GRMHD possuem degenerescência [24], tais códigos têm uma tendência em utilizar métodos do tipo Godunov dependentes de Riemann Solvers com separação dimensional. Isto é, mesmo a degenerescência sendo um limitador e o Splitting dimensional sendo um colaborador com o acréscimo de oscilações na solução numérica, os códigos citados mantêm tal tendência.

Outro fato de destaque é que tais códigos não possuem em seus pacotes exemplos de formação de jatos relativísticos desde a acreção até a ejeção. Também, enfatizamos que desde o artigo pioneiro em simulação de formação de jatos, até os dias atuais [38, 40], Koide, Shibata e Kudoh utilizam em suas pesquisas o método Lax-Wendroff TVD simplificado com viscosidade artificial, decomposição dimensional (*Splitting dimensional*)

e limitadores de oscilações para evitá-las devido tal separação.

Contudo, esta tese propôs a aplicação de métodos sem separação dimensional (*métodos Unsplitting*) e livres de problemas de Riemann, para obter soluções numéricas que representam a simulação computacional de formação de jatos relativísticos, com o intuito de conseguir estabilizar a transição entre acreção e ejeção, que é uma das maiores dificuldades desse problema, por envolver fenômenos em grandes diferenças de escala. Os métodos implementados foram o Lax-Friedrichs bidimensional e o método Nessyahu-Tadmor bidimensional em malha deslocada.

Um outro objetivo foi escrever um código totalmente novo em Fortran 90, com os recursos de alocação dinâmica de memória, com a estrutura de programação modular e declaração da intenção do uso das variáveis. Tais recursos facilitam o acoplamento de futuros códigos e sua alteração, pois a intenção declarada para cada variável pode evitar alterações indevidas.

No capítulo 6 foram expostas as simulações realizadas com o método Lax-Friedrichs e o método Nessyahu-Tadmor. Ambos os métodos simularam a formação de jatos relativísticos, porém as soluções numéricas apresentadas pelo método Lax-Friedrichs apresentaram muita dissipação numérica comprometendo detalhes da estrutura do jato e do disco. Por outro lado, a simulação teve baixo custo computacional e, em um primeiro estudo, tal implementação forneceu uma ideia da dinâmica do fluido e ajudou na estruturação do programa, principalmente pelo fato de que as execuções testes do programa consumiram pouco tempo.

Implementamos ainda, o método Lax-Wendroff com Runge-Kutta de terceira ordem TVD (Valor Total Decrescente), na modelagem de jatos relativísticos e verificamos que nem sempre a propriedade TVD é suficiente para estabilizar as oscilações geradas pelo método Lax-Wendroff.

Com a aplicação do método Nessyahu-Tadmor atingimos o objetivo principal desta tese, pois conseguimos uma simulação que passa pela transição entre acreção e ejeção sem perder a estabilidade numérica, obtendo assim uma simulação mais detalhada.

Uma consequência foi a obtenção de um novo código em Fortran 90 capaz de simular a formação de jatos relativísticos, através das equações de GRMHD em duas dimensões

esaciais na métrica de Schwarzschild cuja extensão do jato atingiu 14 vezes o raio de Schwarzschild, com a possibilidade de observar subestruturas do jato quando utilizamos o método Nessyahu-Tadmor bidimensional sem separação dimensional (*Unsplitting* dimensional), com o método de Euler de quatro estágios para o termo fonte. Além disso, observou-se uma subestrutura no disco de acreção.

7.2 Trabalhos Futuros

A continuidade desta pesquisa pode caminhar para alguns dos temas seguintes:

Modelagem:

- Adaptar o código para que o modelo tenha a opção de utilizar a métrica de Kerr, cujo buraco negro central também possui rotação;
- Pesquisar o significado e a importância do acréscimo de termo resistivo nas equações de GRMHD;
- Modelar e implementar a dinâmica da geometria do espaço-tempo, isto é, considerar um fluido com auto-gravitação. Para isso, é necessário acrescentar as equações de Einstein para atuar na métrica conforme o fluido se movimenta;

Implementação Computacional:

- Acrescentar uma opção para o código ser executado em três dimensões espaciais;
- Tornar o código paralelizado via MPI (Message Passing Interface);

Métodos Numéricos:

- Investigar a viabilidade e aplicabilidade de métodos numéricos ao acrescentar nas equações de GRMHD o termo resistivo;
- Pesquisar a viabilidade do uso de Métodos do tipo Godunov nas equações de GRMHD com e sem termo resistivo, aplicados em problemas de formação de jatos;

- Fazer estudos relacionados à convergência, consistência e estabilidade de métodos numéricos para as equações de GRMHD.;

Simulações:

- Analisar as subestruturas formadas durante a evolução de simulações que envolvem discos e jatos relativísticos;
- Simular cenários que o jato formado alcance longas distâncias;
- Utilizar código em outros cenários de relevância astrofísica.

Referências Bibliográficas

- [1] A. M. Anile, S. Pennisi, *On the mathematical structure of test relativistic magnetofluid dynamics*, Ann. Inst. Henri Poincare, vol. 46, pp. 27–44, 1987.
- [2] A. M. Anile, *Relativistic fluids and magneto-fluids: With applications in astrophysics and plasma physics*, Cambridge Monographs on Mathematical Physics, (Cambridge University Press, Cambridge, U.K.; New York, U.S.A., 1989
- [3] P. Anninos, P. C. Fragile, J. D. Salmonson. *Cosmos++ : Relativistic Magnetohydrodynamics on Unstructured Grids with Local Adaptive Refinement*, Astrophys. J, vol. 635, pp. 7-23,2005.
- [4] P. Arminjon, R. Touma. *Central Volume Finite Methods with constrained transport divergence treatment for ideal MHD*, Journal of Computational Physics, vol. 204, pp 737-759, 2005.
- [5] X. Bai, J. M. Stone. *Local Study of Accretion Disks with a Strong Vertical Magnetic Field: Magnetorotational Instability and Disk Outflow*, ApJ, vol. 767 pp. 30-, 2013.
- [6] J. Balbás, E. Tadmor, C.-C. Wu. *Non-oscillatory central schemes for one- and two-dimensional MHD equations: I*, J. Comp. Phys., vol 201, pp. 261-285, 2004.
- [7] F. Banyuls, et al., *Numerical 3+1 General Relativistic Hydrodynamics: a Local Characteristic Approach*, Astrophys. J., vol. 476, pp. 221–231, 1997.
- [8] V. S. Beskin. *Magnetohydrodynamic models of astrophysical jets*. Physics - Uspekhi, 53(12), pp. 1199-1233, 2010.
- [9] V. S. Beskin. *MHD Flows in Compact Astrophysical Objects*, Springer, 2010.

- [10] R. D. Blandford, R. L. Znajek. *Electromagnetic extraction of energy from Kerr black holes*, MNRAS, vol. 169, pp. 433-456, 1977.
- [11] M. Brio, C. C. Wu. *An Upwind Differencing Scheme for the equations of Ideal Magnetohydrodynamics*, J. Comput. Phys., vol. 75, pp. 400-422, 1988.
- [12] R. D. Blandford, D. G. Payne. *Hydromagnetic flows from accretion discs and the production of radio jets*, MNRAS, vol. 199, pp. 883-903, 1982.
- [13] S. Carroll. *Spacetime: An introduction to General Relativity*
- [14] S. K. Chakrabarti. *Accretion Processes on a Black Hole*, Phys. Rep., vol. 266, pp. 229-390, 1996.
- [15] A. Ciardi, et al. *Episodic Magnetic Bubbles and Jets: Astrophysical Implications from Laboratory Experiments*, ApJ., vol. 691, pp. L147-L150, 2009.
- [16] J.-P. De Villiers, J. F. Hawley, J. H. Krolik. *Magnetically driven accretion flows in the Kerr metric. I. Models and overall structure*, Astrophys. J., vol. 599, pp. 1238-1253, 2003.
- [17] L. Del Zanna, et al. *ECHO: an Eulerian Conservative High Order scheme for general relativistic magnetohydrodynamics and magnetodynamics*, Astronomy & Astrophysics, vol. 473, n. 1, pp. 11-30, 2007.
- [18] S. S. Doeleman, et al. *Jet-Launching Structure Resolved Near the Supermassive Black Hole in M87*. Science, vol. 338, n. 6105, pp. 255-258, 2012.
- [19] M. Dumbser, O. Zanotti. *Very high order $P_N P_M$ schemes on unstructured meshes for the resistive relativistic MHD equations*, J. Comp. Phys., vol 228, pp. 6691-7006, 2009.
- [20] G. C. Duncan, P. A Hughes. *Simulations of relativistic extragalactic jets*, Astrophysical Journal, v. 436, n. 2, pp. L119-122, 1994.
- [21] B. Engquist, S. Osher. *One Sided Difference Approximations for Nonlinear Conservation Laws*, Math. Comp., vol 36(154), pp. 321-351, 1981.

- [22] L. C. Evans. *Partial Differential Equation*. American Mathematical Society.
- [23] H. Falcke, F. W. Hehl. *The Galactic Black Hole - Lectures on general relativity and astrophysics*. IOP, London, 2003.
- [24] J. A. Font. *Numerical Hydrodynamics and Magnetohydrodynamics in General Relativity*, Living Rev. Relativity, vol. 11, n. 7, 2008.
- [25] J. A. Font, F. Daigne. *The runaway instability of thick discs around black holes – I. The constant angular momentum case*, MNRAS, vol. 334, pp. 383–400, 2002.
- [26] C. F. Gammie, J. C. McKinney, G. Toth. *HARM: A numerical scheme for general relativistic magnetohydrodynamics*, Astrophys. J., vol. 589, pp. 444–457, 2003.
- [27] B. Giacomazzo, L. Rezzolla. *The exact solution of the Riemann problem in relativistic magnetohydrodynamics*, Journal of Fluid Mechanics, vol. 562, pp. 223-259, 2006.
- [28] B. Giacomazzo, L. Rezzolla. *WhiskyMHD: a new numerical code for general relativistic magnetohydrodynamics*, Class. Quantum Grav., vol. 24, n. 12, S235, 2007.
- [29] S. K. Godunov. *A Finite Difference Method for the Computation of Discontinuous Solutions of the Equations of Fluid Dynamics*, Mat. Sb., vol 47, pp. 357-393, 1959.
- [30] P. Goldreich, W. H. Julian. *Pulsar Electrodynamics*, ApJ, vol. 157, pp. 869-880, 1969.
- [31] A. Harten, P. D. Lax, B. van Leer. *On Upstream Differencing and Godunov-Type Schemes for Hyperbolic Conservation Laws*. SIAM Review, vol. 25(1), pp. 35-61, 1983.
- [32] P. Hartigan, et al, *Fluid Dynamics of Stellar Jets in real time: Third Epoch Hubble Space Telescope Images of HH 1, HH 34, AND HH 47*. The Astrophysical Journal, vol. 736, n. 29, pp. 2-20, 2011.
- [33] S. Hirose, et al. *Magnetically Driven Accretion Flows in the Kerr Metric. II. Structure of the magnetic field*, Astrophys. J., vol. 606, pp. 1083–1097, 2004.

- [34] W. Hundsdorfer, J. G. Verwer, Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations, Springer, New York, 2003.
- [35] V. Iório. *EDP - Um Curso de Graduação*. Coleção Matemática Universitária, 2a ed., Rio de Janeiro, 2005.
- [36] W. Junor, J. A Biretta, M. Livio. *Formation of the radio jet in M87 at 100 Schwarzschild radii from the central black hole*, Nature, vol. 401, 1999.
- [37] N. S. Kardashev. *Nature of the Radio Galaxy Cygnus A*, Sov. Astron., vol 7, pp. 740-747, 1964.
- [38] S. Koide, K. Shibata, T. Kudoh, *Relativistic Jet Formation from Black Hole Magnetized Accretion Disk: Method, Test and Applications of a General Relativistic Magnetohydrodynamic Numerical Code*, *The Astrophysical Journal*, 522 (1999) 727-752.
- [39] A. Khokhlov, et al. *Jets, Not Neutrinos, May Cause Supernova Explosions*, McDonald Observatory, University of Texas, Austin, 2000. Online: <http://mcdonaldobservatory.org>.
- [40] S. Koide. *General relativistic plasmas around rotating black holes*. Proceedings IAU Symposium, No 275, 2011.
- [41] A. V. Koldoba, O. A Kuznetsov, G. V. Ustyugova, *An approximate Riemann solver for relativistic magnetohydrodynamics*, Mon. Not. R. Astron. Soc., vol. 333, pp. 932-942, 2002.
- [42] S. S. Komissarov, *A Godunov-Type Scheme for Relativistic Magnetohydrodynamics*, MNRAS, vol. 303, pp. 343-366, 1999.
- [43] S. S. Komissarov. *General relativistic magnetohydrodynamic simulations of monopole magnetospheres of black holes*, Mon. Not. R. Astron. Soc., vol. 350, pp. 1431-1436, 2004.
- [44] S. S. Komissarov, et al. *Magnetic acceleration of ultra-relativistic jets in gamma-ray burst sources*, MNRAS, vol. 394, n. 3, pp. 1182-1212, 2009.

- [45] A. Koratkar, O. Blaes. *The Ultraviolet and Optical Continuum Emission in Active Galactic Nuclei: The Status of Accretion Disks*, Astronomical Society of the Pacific, vol. 111, n. 755, pp. 1-30, 1999.
- [46] E. Kreyszig. *Introductory Functional Analysis with Applications*, John Wiley & Sons, Canada, 1989.
- [47] E. Kreyszig. *Advanced Engineering Mathematics*, John & Sons, Singapore, 2006.
- [48] P. Kumar, et al. *The nature of the outflow in gamma-ray bursts*, MNRAS, vol. 376, n. 1, pp. L57-L61.
- [49] A. Kurganov, E. Tadmor. *New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection-Diffusion Equations*, J. Comp. Phys., vol. 160, pp. 241-282, 2000.
- [50] R. A. Laing. *The sidedness of jets and depolarization in powerful extragalactic radio sources*, Nature, vol. 331, pp 149-151, 1988.
- [51] G. Leiler, L. Rezzolla. *Iterated crank-nicolson method for hyperbolic and parabolic equations in numerical relativity*, Phys. Rev. D, 73:044001, 2006.
- [52] R. J. Leveque. *Numerical Methods for Conservation Laws*, Birkhäuser Verlag, 2nd ed., Basel, 1999.
- [53] R. J. Leveque. *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, New York, 2002.
- [54] E. L. Lima. *Análise Real - Volume I*, Coleção Matemática Universitária, IMPA, 6a ed., Rio de Janeiro, 2002.
- [55] M. L. Lister, et al. *MOJAVE: Monitoring of Jets in Active Galactic Nuclei with VLBA Experiments. VI. Kinematics Analysis of a Complete Sample of Blazar Jets*, The Astronomical Journal, vol. 138, pp. 1874-1892, 2009.
- [56] J. D. Logan. *An Introduction to Nonlinear Partial Differential Equations*. John Wiley & Sons, 2nd ed., New Jersey, 2008.

- [57] D. Lynden-Bell. *Galactic Nuclei as Collapsed Old Quasars*, Nature, vol. 223, pp. 690-694, 1969.
- [58] M. Massi. *Steady jets and transient jets: observational characteristics and models*, Mem. S.A.It., vol. 75, 282, 2010.
- [59] A. Mizuta, S. Yamada, H. Takabe. *Propagation and Dynamics of Relativistic Jets*, The Astrophysical Journal, vol. 606, pp. 804–818, 2004.
- [60] J. C. McKinney. *General Relativistic Magnetohydrodynamic Simulations of Jet Formation and Large-Scale Propagation from Black Hole Accretion Systems*, MNRAS, 000, pp. 1-25, 2006
- [61] J. C. McKinney, R. D. Blandford. *Stability of relativistic jets from rotating, accreting black holes via fully three-dimensional magnetohydrodynamic simulations*, MNRAS, vol. 394, pp. L126-L30, 2009.
- [62] D. L. Meier, S. Koide, Y. Uchida. *Magnetohydrodynamic production of Relativistic Jets*, Science, New Series, vol. 291, n. 5501, pp. 84-92, 2001.
- [63] A. Mignone, J. C. McKinney. *Equation of state in relativistic magnetohydrodynamics: variable versus constant adiabatic index*, MNRAS, vol. 378, n. 3, pp. 1118-1130, 2007.
- [64] I. F. Mirabel, L. F. Rodríguez. *A superluminal source in the Galaxy*, Nature, vol. 371, 46-48, 1994.
- [65] P. J. Montero, et al. *Dynamics of magnetized relativistic tori oscillating around black holes*, MNRAS, vol. 378, pp. 1101–1110, 2007.
- [66] P. Mösta, et al. *GRHydro: A new open source general-relativistic magnetohydrodynamics code for the Einstein Toolkit*, Class. Quantum Grav., vol 31, n. 1, 2014. ArXiv 2013.
- [67] M. Motamed, C. B. Macdonald, S. J. Ruuth. *On the Linear Stability of the Fifth-Order WENO Discretization*, J Sci Comput (2011) 47: 127–149.

- [68] H. Nessyahu, E. Tadmor. *Non-Oscillatory Central Differencing for Hyperbolic Conservation Laws*, Journal Computational Physics, vol. 87, n. 2, pp 408-463, 1990.
- [69] K. -I. Nishikawa, et al. *A General Relativistic Magnetohydrodynamic Simulation of Jet Formation*, The Astrophysical Journal, 625, pp. 60-71, 2005.
- [70] L. Pareschi. *Central Differencing based Numerical Schemes for Hyperbolic Conservation Laws with Relaxation Terms*, SIAM, J. Numer. Anal., vol. 39, n. 4, pp. 1395-1417, 2001.
- [71] M. J. D. Powell. *Approximation theory and methods*, Cambridge University Press, Cambridge, 2001.
- [72] R. E. Pudritz. *Jets from Accretion Discs. Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, Royal Society Publishing, Vol. 358, no. 1767, pp. 741-758, 2000.
- [73] P. L. Roe. *Approximate Riemann Solvers, Parameter Vectors and Difference Schemes*, J. Comp. Phys., vol. 43, pp. 357-372, 1981.
- [74] R. Romero, et al. *The exact solution of the Riemann problem in relativistic magnetohydrodynamics with tangential magnetic fields*, J. Fluid Mech., vol. 544, pp. 323-338, 2005.
- [75] M. A. G. Ruggiero, V. L. R. Lopes. *Cálculo Numérico - Aspectos Teóricos e Computacionais*, Makron Books, 2a edição, São Paulo, 1997.
- [76] V. V. Rusanov. *Calculation of Interaction of Non-Steady Shock Waves with Obstacles*, J. Comp. Math. Phys., USSR, vol. 1, pp. 267-279, 1961.
- [77] V. Schneider, et al. *New Algorithms for Ultra-relativistic Numerical Hydrodynamics*, J. Comp. Phys., vol. 105, n. 1, pp. 92-107, 1993.
- [78] D. Serre. *System of Conservation Laws 1: Hyperbolicity, Entropies, Shock Waves*. Cambridge University Press, virtual edition, 2003.

- [79] S. L. Shapiro, S. A. Teukolsky. *Black Holes, White Dwarfs and Neutron Stars - The Physics of Compact Objects*. John Wiley & Sons, New York, 1983.
- [80] C-W Shu, S. Osher. *Efficient implementation of essentially non-oscillatory shock capturing schemes*. J. Comp. Phys., vol. 77, pp. 439-471, 1988.
- [81] G. A. Sod. *A Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws*, J. Comp. Phys., vol. 27, pp. 1-31, 1978.
- [82] J. C. Strikwerda. *Finite Difference Schemes and Partial Differential Equations*, Wadsworth e Books, California, 1989.
- [83] M. E. Taylor. *Partial Differential Equation III - Nonlinear Equation*. Springer, New York, 1996.
- [84] S. A. Teukolsky. *Stability of the iterated Crank-Nicholson Method in Numerical Relativity*. Phys. Rev. D, 61:087501, 2000.
- [85] A. Tchekhovskoy, J. C. McKinney, R. Narayan. *WHAM: a WENO-based general relativistic numerical scheme – I. Hydrodynamics*. MNRAS, vol. 379, pp. 469–497, 2007.
- [86] J. W. Thomas, *Numerical Partial Differential Equations: Finite Difference Methods*, Springer, New York, 1995.
- [87] K. S. Thorne, R. H. Price, D. A. MacDonald. *Black Holes: The Membrane Paradigm*. Yale University Press, New Haven, 1986.
- [88] E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics - A Practical Introduction*, Springer, Germany, third edition, 2009.
- [89] R. M. Wald. *Black Hole in an uniform magnetic field*, Phys. Rev. D, 10, 1680, 1974.
- [90] M. Wesenberg. *Efficient Finite - Volumes schemes for Magnetohydrodynamic Simulations in Solar Physics*. Dissertation zur Erlangung des Doktorgrades der Fakultät für Mathematik und Physik der Albert–Ludwigs–Universität Freiburg im Breisgau, Betreuer: Prof. Dr. D. Kröner, 2003.

- [91] J. R. Wilson. *Numerical study of fluid flow in a Kerr space*, *Astrophys. J.*, vol. 173, pp. 431–438, 1972.
- [92] J. R. Wilson. *Magnetohydrodynamics near a black hole*, in Ruffini, R., ed., *First Marcel Grossmann Meeting on General Relativity*, Proceedings of the meeting held at the International Centre for Theoretical Physics, Trieste and Instituto di fisica, University of Trieste, 7 – 12 July 1975, pp. 393–413, (North-Holland, Amsterdam, Netherlands, 1977).
- [93] P. Woodward, P. Colella. *The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations*, *J. Comput. Phys.*, vol 54., pp. 174-201, 1984.
- [94] P. Woodward, P. Colella. *The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks*, *J. Comput. Phys.*, vol 54, pp. 115-173, 1984.
- [95] D. M. Worrall et al. *The inner jet of radio galaxy NGC 315 as observed with Chandra and the Very Large Array*, *MNRAS*, vol. 380, n. 1, pp. 2-14 , 2007.
- [96] C. Zarowski. *An Introduction to Numerical Analysis for Electrical and Computer Engineers*, John Wiley & Sons, Canada, 2004.
- [97] Y. B. Zeldovich, I. D. Novikov. *Relativistic Astrophysics*, Univ. of Chicago Press, Chicago, 1983.

Sites

- [98] <http://shock.uv.es:8081/COCONUT/intro.php>. Acessado em Janeiro - 2014.
- [99] NASA/IPAC Extragalactic Database <http://ned.ipac.caltech.edu/>. Accessed: November - 2013.

Apêndice A

Código

Módulo: armazenamentos.f90

```
MODULE armazenamentos

    B(3,i,j)*B(3,i,j)
    nP(i,j) = P(1,i,j)*P(1,i,j) + P(2,i,j)*P(2,i,j) +
    P(3,i,j)*P(3,i,j)
END DO
END DO

CONTAINS

SUBROUTINE arquivos_saida(m,n,tp,D,P,Et,E,B,V,GD,x1,x2)

    IMPLICIT NONE

    INTEGER :: i,j
    INTEGER, INTENT(IN) :: m,n,tp
    REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:) :: x1,x2
    REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,:) :: GD,D,Et
    REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,,:) :: V,P,B,E
    REAL, ALLOCATABLE, DIMENSION(:,:) :: nV,nP,nB

    ALLOCATE(nV(m,n),nP(m,n),nB(m,n))

    DO i=1,m,1
        DO j=1,n,1
            nV(i,j) = V(1,i,j)*V(1,i,j) + V(2,i,j)*V(2,i,j) +
            V(3,i,j)*V(3,i,j)
            nB(i,j) = B(1,i,j)*B(1,i,j) + B(2,i,j)*B(2,i,j) +
            B(3,i,j)*B(3,i,j)
        END DO
    END DO

    DO i=4,m-3,1
        WRITE(11,*) (D(i,j), j=4,n-3,1)
        WRITE(12,*) (nV(i,j), j=4,n-3,1)
        WRITE(13,*) (nP(i,j), j=4,n-3,1)
        WRITE(14,*) (Et(i,j), j=4,n-3,1)
        WRITE(15,*) (nB(i,j), j=4,n-3,1)
    END DO

    DO i=4,m-3,1
        WRITE(21,*) x1(i)
    END DO

    DO j=4,n-3,1
        WRITE(22,*) x2(j)
    END DO

END SUBROUTINE arquivos_saida

END MODULE armazenamentos
```

Módulo: atualizacoes.f90

```
MODULE atualizacoes

    IMPLICIT NONE

CONTAINS

SUBROUTINE grandezas_atualizadas(m,n,x1,x2,Gamma,GD,D,P,B,Et,V,T,E)

    USE EOS
    USE ferramentas

    IMPLICIT NONE

    INTEGER :: i,j
    INTEGER, INTENT(IN) :: m,n
    REAL, INTENT(IN) :: Gamma
    REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:) :: x1,x2
    REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,:) :: Et,D
    REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,,:) :: P,B
    REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,,:) :: E
```

```

REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,:,:): T      ! distribuicao de Stress Tensor - T
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:,:) : GD    !-----
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:,:,:) : V

REAL, ALLOCATABLE, DIMENSION(:,:) : nP,nB,nE,press,energ

ALLOCATE(T(3,3,m,n),E(3,m,n))
ALLOCATE(nP(m,n),nB(m,n),nE(m,n),press(m,n),energ(m,n))

CALL newton_metEOS(m,n,Gamma,D,P,Et,B,GD,V)

!
!-----
! campo eletrico
!-----
CALL prod_vec(m,n,V,B,E)
E = -E
!
!-----
! normas
!-----
CALL norma(m,n,P,nP)
CALL norma(m,n,B,nB)
CALL norma(m,n,E,nE)
!
!-----
! press e energ
!-----
DO i=1,m,1
DO j=1,n,1
press(i,j) = (Et(i,j) - (D(i,j)/GD(i,j)))*(Gamma-1)
energ(i,j) = (D(i,j)/GD(i,j)) + (press(i,j)/(Gamma-1))
END DO
END DO

!
!-----
! distribuicao de Energ. total Et

! distribuicao de Stress Tensor - T
!-----
DO i=1,m,1
DO j=1,n,1
T(1,1,i,j) = press(i,j) + 0.5*(nB(i,j)**2)+(nE(i,j)**2)
T(2,2,i,j) = T(1,1,i,j)
T(3,3,i,j) = T(1,1,i,j)

T(1,1,i,j)=T(1,1,i,j)+(energ(i,j)+press(i,j))*((GD(i,j)**2)*V(1,i,j)**2)
T(2,2,i,j)=T(2,2,i,j)+(energ(i,j)+press(i,j))*(GD(i,j)**2)*V(2,i,j)**2
T(3,3,i,j)=T(3,3,i,j)+(energ(i,j)+press(i,j))*(GD(i,j)**2)*V(3,i,j)**2

T(1,2,i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)*V(1,i,j)*V(2,i,j)
T(1,3,i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)*V(1,i,j)*V(3,i,j)
T(2,3,i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)*V(2,i,j)*V(3,i,j)

T(1,1,i,j) = T(1,1,i,j) - B(1,i,j)*B(1,i,j) - E(1,i,j)*E(1,i,j)
T(2,2,i,j) = T(2,2,i,j) - B(2,i,j)*B(2,i,j) - E(2,i,j)*E(2,i,j)
T(3,3,i,j) = T(3,3,i,j) - B(3,i,j)*B(3,i,j) - E(3,i,j)*E(3,i,j)

T(1,2,i,j) = T(1,2,i,j) - B(1,i,j)*B(2,i,j) - E(1,i,j)*E(2,i,j)
T(1,3,i,j) = T(1,3,i,j) - B(1,i,j)*B(3,i,j) - E(1,i,j)*E(3,i,j)
T(2,3,i,j) = T(2,3,i,j) - B(2,i,j)*B(3,i,j) - E(2,i,j)*E(3,i,j)

T(2,1,i,j) = T(1,2,i,j)
T(3,1,i,j) = T(1,3,i,j)
T(3,2,i,j) = T(2,3,i,j)
END DO
END DO
DEALLOCATE(nP,nB,nE,press,energ)

END SUBROUTINE grandezas_atualizadas

END MODULE atualizacoes

```

Módulo: condicoes_fronteras.f90

```

MODULE condicoes_fronteras

IMPLICIT NONE

CONTAINS

SUBROUTINE condicoes_contorno(m,n,D,P,Et,B,Dold,Pold,Etold,Bold)

IMPLICIT NONE

INTEGER :: i,j,k
INTEGER, INTENT(IN) :: m,n
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,:) : Dold,Etold
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,:,:) : Pold,Bold
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:,:) : D,Et
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:,:,:) : P,B

!
!-----
! Radiacao 1
!-----
DO j=4,n-3,1

D(3,j) = Dold(3,j) + D(4,j) - Dold(4,j)
D(2,j) = Dold(2,j) + D(3,j) - Dold(3,j)
D(1,j) = Dold(1,j) + D(2,j) - Dold(2,j)
Et(3,j) = Etold(3,j) + Et(4,j) - Etold(4,j)
Et(2,j) = Etold(2,j) + Et(3,j) - Etold(3,j)
Et(1,j) = Etold(1,j) + Et(2,j) - Etold(2,j)
DO k=1,3,1
P(k,3,j) = Pold(k,3,j) + P(k,4,j) - Pold(k,4,j)
P(k,2,j) = Pold(k,2,j) + P(k,3,j) - Pold(k,3,j)
P(k,1,j) = Pold(k,1,j) + P(k,2,j) - Pold(k,2,j)
B(k,3,j) = Bold(k,3,j) + B(k,4,j) - Bold(k,4,j)
B(k,2,j) = Bold(k,2,j) + B(k,3,j) - Bold(k,3,j)
B(k,1,j) = Bold(k,1,j) + B(k,2,j) - Bold(k,2,j)
END DO
END DO

!
!-----
! Radiacao 2
!-----
DO j=4,n-3,1
D(m-2,j) = Dold(m-2,j) + D(m-3,j) - Dold(m-3,j)

```

```

D(m-1,j) = Dold(m-1,j) + D(m-2,j) - Dold(m-2,j)
D(m,j) = Dold(m,j) + D(m-1,j) - Dold(m-1,j)
Et(m-2,j) = Etold(m-2,j) + Et(m-3,j) - Etold(m-3,j)
Et(m-1,j) = Etold(m-1,j) + Et(m-2,j) - Etold(m-2,j)
Et(m,j) = Etold(m,j) + Et(m-1,j) - Etold(m-1,j)
DO k=1,3,1
  P(k,m-2,j) = Pold(k,m-2,j) + P(k,m-3,j) - Pold(k,m-3,j)
  P(k,m-1,j) = Pold(k,m-1,j) + P(k,m-2,j) - Pold(k,m-2,j)
  P(k,m,j) = Pold(k,m,j) + P(k,m-1,j) - Pold(k,m-1,j)
  B(k,m-2,j) = Bold(k,m-2,j) + B(k,m-3,j) - Bold(k,m-3,j)
  B(k,m-1,j) = Bold(k,m-1,j) + B(k,m-2,j) - Bold(k,m-2,j)
  B(k,m,j) = Bold(k,m,j) + B(k,m-1,j) - Bold(k,m-1,j)
END DO
END DO
!
!-----
! Reflexao 1
!-----
DO i=4,m-3,1
  D(i,1) = D(i,7)
  D(i,2) = D(i,6)
  D(i,3) = D(i,5)
  Et(i,1) = Et(i,7)
  Et(i,2) = Et(i,6)
  Et(i,3) = Et(i,5)
DO k=1,3,1
  P(k,i,1) = P(k,i,7)
  P(k,i,2) = P(k,i,6)
  B(k,i,2) = B(k,i,6)
  B(k,i,3) = B(k,i,5)
END DO
END DO
!
!-----
! Reflexao 2
!-----
DO i=4,m-3,1
  D(i,m) = D(i,m-6)
  D(i,m-1) = D(i,m-5)
  D(i,m-2) = D(i,m-4)
  Et(i,m) = Et(i,m-6)
  Et(i,m-1) = Et(i,m-5)
  Et(i,m-2) = Et(i,m-4)
DO k=1,3,1
  P(k,i,m) = P(k,i,m-6)
  P(k,i,m-1) = P(k,i,m-5)
  B(k,i,m-1) = B(k,i,m-5)
  B(k,i,m-2) = B(k,i,m-4)
  B(k,i,m-2) = B(k,i,m-4)
END DO
END DO
END SUBROUTINE condicoes_contorno
SUBROUTINE condicoes_contorno2(m,n,D,P,Et,B,x1,x2)
USE ferramentas !, ONLY: intLagrange

```

```

IMPLICIT NONE
INTEGER :: i,j,k,grau
INTEGER, INTENT(IN) :: m,n
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:) :: x1,x2
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:,:) :: D,Et
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:,:,:) :: P,B
REAL, ALLOCATABLE, DIMENSION(:) :: x,y
grau = 3
ALLOCATE(x(grau+1),y(grau+1))
!-----
! Radiacao 1
!-----
DO j=4,n-3,1
  x(:) = x1(4:7)
  y(:) = D(4:7,j)
CALL intLagrange(x1(3),D(3,j),x,y,grau)
  x(:) = x1(3:6)
  y(:) = D(3:6,j)
CALL intLagrange(x1(2),D(2,j),x,y,grau)
  x(:) = x1(2:5)
  y(:) = D(2:5,j)
CALL intLagrange(x1(1),D(1,j),x,y,grau)
  x(:) = x1(4:7)
  y(:) = Et(4:7,j)
CALL intLagrange(x1(3),Et(3,j),x,y,grau)
  x(:) = x1(3:6)
  y(:) = Et(3:6,j)
CALL intLagrange(x1(2),Et(2,j),x,y,grau)
  x(:) = x1(2:5)
  y(:) = Et(2:5,j)
CALL intLagrange(x1(1),Et(1,j),x,y,grau)
DO k=1,3,1
  x(:) = x1(4:7)
  y(:) = P(k,4:7,j)
CALL intLagrange(x1(3),P(k,3,j),x,y,grau)
  x(:) = x1(3:6)
  y(:) = P(k,3:6,j)
CALL intLagrange(x1(2),P(k,2,j),x,y,grau)
  x(:) = x1(2:5)
  y(:) = P(k,2:5,j)
CALL intLagrange(x1(1),P(k,1,j),x,y,grau)
  x(:) = x1(4:7)
  y(:) = B(k,4:7,j)
CALL intLagrange(x1(3),B(k,3,j),x,y,grau)
  x(:) = x1(3:6)
  y(:) = B(k,3:6,j)
CALL intLagrange(x1(2),B(k,2,j),x,y,grau)
  x(:) = x1(2:5)
  y(:) = B(k,2:5,j)
CALL intLagrange(x1(1),B(k,1,j),x,y,grau)
END DO
END DO
!-----
! Radiacao 2
!-----

```

```

DO j=4,n-3,1
  x(:) = x1((m-6):(m-3))
  y(:) = D((m-6):(m-3),j)
  CALL intLagrange(x1(m-2),D(m-2,j),x,y,grau)
  x(:) = x1((m-5):(m-2))
  y(:) = D((m-5):(m-2),j)
  CALL intLagrange(x1(m-1),D(m-1,j),x,y,grau)
  x(:) = x1((m-4):(m-1))
  y(:) = D((m-4):(m-1),j)
  CALL intLagrange(x1(m),D(m,j),x,y,grau)
  x(:) = x1((m-6):(m-3))
  y(:) = Et((m-6):(m-3),j)
  CALL intLagrange(x1(m-2),Et(m-2,j),x,y,grau)
  x(:) = x1((m-5):(m-2))
  y(:) = Et((m-5):(m-2),j)
  CALL intLagrange(x1(m-1),Et(m-1,j),x,y,grau)
  x(:) = x1((m-4):(m-1))
  y(:) = Et((m-4):(m-1),j)
  CALL intLagrange(x1(m),Et(m,j),x,y,grau)
DO k=1,3,1
  x(:) = x1((m-6):(m-3))
  y(:) = P(k,(m-6):(m-3),j)
  CALL intLagrange(x1(m-2),P(k,m-2,j),x,y,grau)
  x(:) = x1((m-5):(m-2))
  y(:) = P(k,(m-5):(m-2),j)
  CALL intLagrange(x1(m-1),P(k,m-1,j),x,y,grau)
  x(:) = x1((m-4):(m-1))
  y(:) = P(k,(m-4):(m-1),j)
  CALL intLagrange(x1(m),P(k,m,j),x,y,grau)
  x(:) = x1((m-6):(m-3))
  y(:) = B(k,(m-6):(m-3),j)
  CALL intLagrange(x1(m-2),B(k,m-2,j),x,y,grau)
  x(:) = x1((m-5):(m-2))
  y(:) = B(k,(m-5):(m-2),j)
  CALL intLagrange(x1(m-1),B(k,m-1,j),x,y,grau)
  x(:) = x1((m-4):(m-1))
  y(:) = B(k,(m-4):(m-1),j)
  CALL intLagrange(x1(m),B(k,m,j),x,y,grau)
END DO
END DO
!
!-----
! Reflexao 1
!-----
DO i=4,m-3,1
  D(i,1) = D(i,7)
  D(i,2) = D(i,6)
  D(i,3) = D(i,5)
  Et(i,1) = Et(i,7)
  Et(i,2) = Et(i,6)
  Et(i,3) = Et(i,5)
DO k=1,3,1
  P(k,i,1) = P(k,i,7)
  P(k,i,2) = P(k,i,6)
  P(k,i,3) = P(k,i,5)
  B(k,i,1) = B(k,i,7)
  B(k,i,2) = B(k,i,6)
  B(k,i,3) = B(k,i,5)
END DO
END DO
!
!-----
! Reflexao 2
!-----
DO i=4,m-3,1
  D(i,m) = D(i,m-6)
  D(i,m-1) = D(i,m-5)
  D(i,m-2) = D(i,m-4)
  Et(i,m) = Et(i,m-6)
  Et(i,m-1) = Et(i,m-5)
  Et(i,m-2) = Et(i,m-4)
DO k=1,3,1
  P(k,i,m) = P(k,i,m-6)
  P(k,i,m-1) = P(k,i,m-5)
  P(k,i,m-2) = P(k,i,m-4)
  B(k,i,m-1) = B(k,i,m-5)
  B(k,i,m-2) = B(k,i,m-4)
  B(k,i,m-3) = B(k,i,m-4)
END DO
END DO
END SUBROUTINE condicoes_contorno2
SUBROUTINE condicoes_contorno3(m,n,D,P,Et,B,x1,x2)
USE ferramentas !, ONLY: intLagrange
IMPLICIT NONE
INTEGER :: i,j,k,grau
INTEGER, INTENT(IN) :: m,n
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:) :: x1,x2
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:,:) :: D,Et
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:,:,:) :: P,B
REAL, ALLOCATABLE, DIMENSION(:) :: x,y
!
!-----
! Radiacao 1
!-----
DO j=4,n-3,1
  D(3,j) = D(5,j)
  D(2,j) = D(6,j)
  D(1,j) = D(7,j)
  Et(3,j) = Et(5,j)
  Et(2,j) = Et(6,j)
  Et(1,j) = Et(7,j)
DO k=1,3,1
  P(k,3,j) = P(k,5,j)
  P(k,2,j) = P(k,6,j)
  P(k,1,j) = P(k,7,j)
  B(k,3,j) = B(k,5,j)
  B(k,2,j) = B(k,6,j)
  B(k,1,j) = B(k,7,j)
END DO
END DO
!
!-----
! Radiacao 2
!-----
DO j=4,n-3,1
  D(m-2,j) = D(m-3,j)
  D(m-1,j) = D(m-3,j)
  D(m,j) = D(m-3,j)

```

```

Et(m-2,j)= Et(m-3,j)
Et(m-1,j)= Et(m-3,j)
Et(m,j)= Et(m-3,j)
DO k=1,3,1
  P(k,m-2,j) = P(k,m-3,j)
  P(k,m-1,j) = P(k,m-3,j)
  P(k,m,j) = P(k,m-3,j)
  B(k,m-2,j) = B(k,m-3,j)
  B(k,m-1,j) = B(k,m-3,j)
  B(k,m,j) = B(k,m-3,j)
END DO
END DO
!
!-----
! Radiacao 1
!-----
! DO j=4,n-3,1
!   D(3,j) = D(5,j)
!   D(2,j) = D(6,j)
!   D(1,j) = D(7,j)
!   Et(3,j) = Et(5,j)
!   Et(2,j) = Et(6,j)
!   Et(1,j) = Et(7,j)
!   DO k=1,3,1
!     P(k,3,j) = P(k,5,j)
!     P(k,2,j) = P(k,6,j)
!     P(k,1,j) = P(k,7,j)
!     B(k,3,j) = B(k,5,j)
!     B(k,2,j) = B(k,6,j)
!     B(k,1,j) = B(k,7,j)
!   END DO
! END DO
!
!-----
! Radiacao 2
!-----
! DO j=4,n-3,1
!   D(m-2,j) = D(m-4,j)
!   D(m-1,j) = D(m-5,j)
!   D(m,j) = D(m-6,j)
!   Et(m-2,j) = Et(m-4,j)
!   Et(m-1,j) = Et(m-5,j)
!   Et(m,j) = Et(m-6,j)
!   DO k=1,3,1
!     P(k,m-2,j) = P(k,m-4,j)
!     P(k,m-1,j) = P(k,m-5,j)
!     P(k,m,j) = P(k,m-6,j)
!     B(k,m-2,j) = B(k,m-4,j)
!     B(k,m-1,j) = B(k,m-5,j)
!     B(k,m,j) = B(k,m-6,j)
!   END DO
! END DO
! DO j=4,n-3,1
!   x(:) = x1((m-6):(m-3))
!   y(:) = D((m-6):(m-3),j)
!   CALL intLagrange(x1(m-2),D(m-2,j),x,y,grau)
!   x(:) = x1((m-5):(m-2))
!   y(:) = D((m-5):(m-2),j)
!   CALL intLagrange(x1(m-1),D(m-1,j),x,y,grau)
!   x(:) = x1((m-4):(m-1))
!   y(:) = D((m-4):(m-1),j)
!   CALL intLagrange(x1(m),D(m,j),x,y,grau)
!
!-----
! Reflexao 1
!-----
DO i=4,m-3,1
  D(i,1) = D(i,7)
  D(i,2) = D(i,6)
  D(i,3) = D(i,5)
  Et(i,1) = Et(i,7)
  Et(i,2) = Et(i,6)
  Et(i,3) = Et(i,5)
  DO k=1,3,1
    P(k,i,1) = P(k,i,7)
    P(k,i,2) = P(k,i,6)
    P(k,i,3) = P(k,i,5)
    B(k,i,1) = B(k,i,7)
    B(k,i,2) = B(k,i,6)
    B(k,i,3) = B(k,i,5)
  END DO
END DO
!
!-----
! Reflexao 2
!-----
DO i=4,m-3,1
  D(i,m) = D(i,m-6)
  D(i,m-1) = D(i,m-5)
  D(i,m-2) = D(i,m-4)
  Et(i,m) = Et(i,m-6)
  Et(i,m-1) = Et(i,m-5)

```

```

        Et(i,m-2) = Et(i,m-4)
        DO k=1,3,1
            P(k,i,m) = P(k,i,m-6)
            P(k,i,m) = P(k,i,m-6)
            P(k,i,m-1) = P(k,i,m-5)
            B(k,i,m-1) = B(k,i,m-5)
            B(k,i,m-2) = B(k,i,m-4)
            B(k,i,m-2) = B(k,i,m-4)
        END DO
    END DO

END SUBROUTINE condicoes_contorno3

SUBROUTINE
condicoes_contorno4(m,n,D,P,Et,B,Dold,Pold,Etold,Bold,Din,Etin,Pin,Bin)

    IMPLICIT NONE

    INTEGER :: i,j,k
    INTEGER, INTENT(IN) :: m,n
    REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :) :: Dold,Etold,Din,Etin
    REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :, :) :: Pold,Bold,Pin,Bin
    REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:, :) :: D,Et
    REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:, :, :) :: P,B

!
!-----
! Radiacao 1
!-----
        DO j=4,n-3,1
            D(3,j) = Dold(3,j) + D(4,j) - Dold(4,j)
            D(2,j) = Dold(2,j) + D(3,j) - Dold(3,j)
            D(1,j) = Dold(1,j) + D(2,j) - Dold(2,j)
            Et(3,j) = Etold(3,j) + Et(4,j) - Etold(4,j)
            Et(2,j) = Etold(2,j) + Et(3,j) - Etold(3,j)
            Et(1,j) = Etold(1,j) + Et(2,j) - Etold(2,j)
            DO k=1,3,1
                P(k,3,j) = Pold(k,3,j) + P(k,4,j) - Pold(k,4,j)
                P(k,2,j) = Pold(k,2,j) + P(k,3,j) - Pold(k,3,j)
                P(k,1,j) = Pold(k,1,j) + P(k,2,j) - Pold(k,2,j)
                B(k,3,j) = Bold(k,3,j) + B(k,4,j) - Bold(k,4,j)
                B(k,2,j) = Bold(k,2,j) + B(k,3,j) - Bold(k,3,j)
                B(k,1,j) = Bold(k,1,j) + B(k,2,j) - Bold(k,2,j)
            END DO
        END DO

!
!-----
! Radiacao 2
!-----
        DO j=4,n-3,1
            D(m-2,j) = Din(m-2,j)
            D(m-1,j) = Din(m-1,j)
            D(m,j) = Din(m,j)
            Et(m-2,j) = Etin(m-2,j)
            Et(m-1,j) = Etin(m-1,j)
            Et(m,j) = Etin(m,j)
        END DO
    END SUBROUTINE condicoes_contorno4

END MODULE condicoes_frenteiras

        DO k=1,3,1
            P(k,m-2,j) = Pin(k,m-2,j)
            P(k,m-1,j) = Pin(k,m-1,j)
            P(k,m,j) = Pin(k,m,j)
            B(k,m-2,j) = Bin(k,m-2,j)
            B(k,m-1,j) = Bin(k,m-1,j)
            B(k,m,j) = Bin(k,m,j)
        END DO
    END DO

!
!-----
! Reflexao 1
!-----
        DO i=4,m-3,1
            D(i,1) = D(i,7)
            D(i,2) = D(i,6)
            D(i,3) = D(i,5)
            Et(i,1) = Et(i,7)
            Et(i,2) = Et(i,6)
            Et(i,3) = Et(i,5)
            DO k=1,3,1
                P(k,i,1) = P(k,i,7)
                P(k,i,1) = P(k,i,7)
                P(k,i,2) = P(k,i,6)
                B(k,i,2) = B(k,i,6)
                B(k,i,3) = B(k,i,5)
                B(k,i,3) = B(k,i,5)
            END DO
        END DO

!
!-----
! Reflexao 2
!-----
        DO i=4,m-3,1
            D(i,m) = D(i,m-6)
            D(i,m-1) = D(i,m-5)
            D(i,m-2) = D(i,m-4)
            Et(i,m) = Et(i,m-6)
            Et(i,m-1) = Et(i,m-5)
            Et(i,m-2) = Et(i,m-4)
            DO k=1,3,1
                P(k,i,m) = P(k,i,m-6)
                P(k,i,m) = P(k,i,m-6)
                P(k,i,m-1) = P(k,i,m-5)
                B(k,i,m-1) = B(k,i,m-5)
                B(k,i,m-2) = B(k,i,m-4)
                B(k,i,m-2) = B(k,i,m-4)
            END DO
        END DO
    END SUBROUTINE condicoes_contorno4

END MODULE condicoes_frenteiras

```

Módulo: condicoes_iniciais.f90

```

MODULE condicoes_iniciais
!-----
IMPLICIT NONE
CONTAINS
SUBROUTINE cond_iniciais(x1,x2,alpha,Gamma,GD,D,V,P,Et,B,E,T)
USE ferramentas
IMPLICIT NONE
INTEGER :: m,n,i,j
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:) :: x1,x2,alpha
REAL, INTENT(OUT) :: Gamma
REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,:) :: GD,D,Et
REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,:,:) :: V,P,B,E
REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,:,:,:) :: T
REAL :: H,delta,rD,EOS,cot,alphaD,gammaD,rhoD,EOSD,BD,nV2
REAL, ALLOCATABLE, DIMENSION(:) :: gammaL,veloc,VK,pressEOS,rhoEOS
REAL, ALLOCATABLE, DIMENSION(:,:) :: press,rho,energ,nE,nB,nV,nP
REAL, ALLOCATABLE, DIMENSION(:,:,:) :: EB
m = SIZE(x1)
n = SIZE(x2)
!
!-----
! Variaveis Primitivas
!-----
ALLOCATE(D(m,n),P(3,m,n),Et(m,n),B(3,m,n))
!
!-----
! Outras Variaveis de saida
!-----
ALLOCATE(V(3,m,n),E(3,m,n),T(3,3,m,n),GD(m,n))
!
!-----
! Variaveis internas
!-----
ALLOCATE(gammaL(m),veloc(m),VK(m),press(m,n),rho(m,n))
ALLOCATE(rhoEOS(m),pressEOS(m),energ(m,n))
ALLOCATE(EB(3,m,n),nE(m,n),nB(m,n),nV(m,n),nP(m,n))
!
!-----
! Constantes Fisicas
!-----
Gamma = 5.0/3.0 ! razao do calor especifico - gas politropico
H = 1.3 ! entalpia
!
!-----
! Constantes do disco de acrecao
!-----
delta = 0.125 ! abertura do disco
rD = 3.0 ! posicao de inicio do disco
! rD = 1.0 + exp(rD) ! posicao de inicio
! rD = exp(rD) + 1.0 ! do disco em coord. tortoise
!
!-----
! fator de Lorentz - gammaL
! funcao velocidade - radial - veloc
! velocidade Kepleriana - VK
! EOS - relacao entre dens e press
!-----
DO i=1,m,1
CALL raphson_newton(alpha(i),Gamma,H,gammaL(i))
veloc(i)=SQRT(1.0-(1.0/(gammaL(i)*gammaL(i))))
VK(i)=1/(SQRT((2*(x1(i)-1))))
EOS = ((Gamma-1.0)/Gamma)*((H/(alpha(i)*gammaL(i)))-1.0)
rhoEOS(i) = EOS*(1.0/(Gamma-1.0)) ! rhoffc
pressEOS(i) = EOS*rhoEOS(i) ! pressffc
END DO
print*, 'VK = '
print*,VK
print*, 'veloc = '
print*,veloc
print*, 'alpha = '
print*, alpha
rhoEOS = rhoEOS/401.0
pressEOS = pressEOS/401.0
!
!-----
! distribuicao de velocidade no dominio - V
! EOS - distribuicao de densidade - rho
! EOS - distribuicao de pressao - press
! distribuicao de energia total - energ
! considerando o disco de acrecao
!-----
DO j=1,3,1
cot = COS(x2(j))/SIN(x2(j))
DO i=1,m,1
IF ((x1(i)>rD).AND.(-0.125<cot).AND.(cot<0.125)) THEN
V(1,i,j) = 0.0
V(2,i,j) = 0.0
V(3,i,j) = VK(i)
! rho(i,j) = 101.0*rhoEOS(i)
rho(i,j) = 401.0*rhoEOS(i)
ELSE
V(1,i,j) = -veloc(i)
V(2,i,j) = 0.0
V(3,i,j) = 0.0
! rho(i,j) = rhoEOS(i)
rho(i,j) = 1.0*rhoEOS(i)
END IF
press(i,j) = pressEOS(i)
energ(i,j) = rho(i,j) + (press(i,j)/(Gamma-1.0))
END DO
END DO
j=4
DO i=1,m,1
IF ((x1(i) > rD).AND.(-0.125 < 1).AND.(1 < 0.125) ) THEN
V(1,i,j) = 0.0
V(2,i,j) = 0.0
V(3,i,j) = VK(i)
! rho(i,j) = 101.0*rhoEOS(i)
rho(i,j) = 401.0*rhoEOS(i)
ELSE
V(1,i,j) = -veloc(i)
V(2,i,j) = 0.0
V(3,i,j) = 0.0
! rho(i,j) = rhoEOS(i)
rho(i,j) = 1.0*rhoEOS(i)
END IF
press(i,j) = pressEOS(i)

```

```

      energ(i,j) = rho(i,j) + (press(i,j)/(Gamma-1.0))
END DO

DO j=5,n,1
  cot = COS(x2(j))/SIN(x2(j))
  DO i=1,m,1
    IF ((x1(i)>rD).AND.(-0.125<cot).AND.(cot<0.125)) THEN
      V(1,i,j) = 0.0
      V(2,i,j) = 0.0
      V(3,i,j) = VK(i)
      rho(i,j) = 101.0*rhoEOS(i)
      rho(i,j) = 401.0*rhoEOS(i)
    ELSE
      V(1,i,j) = -veloc(i)
      V(2,i,j) = 0.0
      V(3,i,j) = 0.0
      rho(i,j) = rhoEOS(i)
      rho(i,j) = 1.0*rhoEOS(i)
    END IF
    press(i,j) = pressEOS(i)
    energ(i,j) = rho(i,j) + (press(i,j)/(Gamma-1.0))
  END DO
END DO

!
!-----
! distribuicao do Campo Magnetico
!-----
      alphaD = SQRT(1.0-(1.0/rD))
      CALL raphson_newton(alphaD,Gamma,H,gammaD)
      EOSD = ((Gamma-1.0)/Gamma)*((H/(alphaD*gammaD))-1.0)
      rhoD = (EOSD**((1.0/(Gamma-1.0)))) ! rhoffc
      BD = 0.3*SQRT(rhoD)
      BD = 1.0*BD
      DO i=1,m,1
        DO j=1,n,1
          B(1,i,j) = BD*COS(x2(j))
          B(2,i,j) = -alpha(i)*BD*COS(x2(j))
          B(1,i,j) = BD*COS(x2(j))
          B(2,i,j) = -alpha(i)*BD*SIN(x2(j))
          B(3,i,j) = 0.0
        END DO
      END DO

!
!-----
! campo eletrico
!-----
      CALL prod_vec(m,n,V,B,E)
      E = -E
      CALL prod_vec(m,n,E,B,EB)

!
!-----
! distribuicao - fator de Lorentz - GD
! distribuicao de Densidade - D
! distribuicao de Pressao - P
!-----
      PRINT *, 'ESTOU AI'
      DO i=1,m,1
        DO j=1,n,1
          nV2 = V(1,i,j)*V(1,i,j) + V(2,i,j)*V(2,i,j) + V(3,i,j)*V(3,i,j)
          GD(i,j) = 1.0/SQRT(1.0-nV2)
          D(i,j) = GD(i,j)*rho(i,j)
          P(1,i,j) = ((energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(1,i,j))+EB(1,i,j))
          P(2,i,j) = ((energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(2,i,j))+EB(2,i,j))
          P(3,i,j) = ((energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(3,i,j))+EB(3,i,j))
        END DO
      END DO
      PRINT *, 'ESTOU AI'
      CALL norma(m,n,P,nP)
      CALL norma(m,n,B,nB)
      CALL norma(m,n,E,nE)

!
!-----
! distribuicao de Energ. total Et
! distribuicao de Stress Tensor - T
!-----
      DO i=1,m,1
        DO j=1,n,1
          Et(i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)
          Et(i,j) = Et(i,j) - press(i,j) - D(i,j)
          Et(i,j) = Et(i,j) + 0.5*((nB(i,j)**2)+(nE(i,j)**2))
          T(1,1,i,j) = press(i,j) + 0.5*((nB(i,j)**2)+(nE(i,j)**2))
          T(2,2,i,j) = T(1,1,i,j)
          T(3,3,i,j) = T(1,1,i,j)
          T(1,1,i,j) = T(1,1,i,j)+(energ(i,j)+press(i,j))* &
            & ((GD(i,j)**2)*(V(1,i,j)*V(1,i,j)))
          T(2,2,i,j) = T(2,2,i,j)+(energ(i,j)+press(i,j))* &
            & (GD(i,j)**2)*(V(2,i,j)*V(2,i,j))
          T(3,3,i,j) = T(3,3,i,j)+(energ(i,j)+press(i,j))* &
            & (GD(i,j)**2)*(V(3,i,j)*V(3,i,j))
          T(1,2,i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(1,i,j)*V(2,i,j))
          T(1,3,i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(1,i,j)*V(3,i,j))
          T(2,3,i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(2,i,j)*V(3,i,j))
          T(1,1,i,j) = T(1,1,i,j) - B(1,i,j)*B(1,i,j) - E(1,i,j)*E(1,i,j)
          T(2,2,i,j) = T(2,2,i,j) - B(2,i,j)*B(2,i,j) - E(2,i,j)*E(2,i,j)
          T(3,3,i,j) = T(3,3,i,j) - B(3,i,j)*B(3,i,j) - E(3,i,j)*E(3,i,j)
          T(1,2,i,j) = T(1,2,i,j) - B(1,i,j)*B(2,i,j) - E(1,i,j)*E(2,i,j)
          T(1,3,i,j) = T(1,3,i,j) - B(1,i,j)*B(3,i,j) - E(1,i,j)*E(3,i,j)
          T(2,3,i,j) = T(2,3,i,j) - B(2,i,j)*B(3,i,j) - E(2,i,j)*E(3,i,j)
          T(2,1,i,j) = T(1,2,i,j)
          T(3,1,i,j) = T(1,3,i,j)
          T(3,2,i,j) = T(2,3,i,j)
        END DO
      END DO
      DEALLOCATE(EB,press,rho,energ,nE,nB,nV,nP,gammaL,veloc,VK,pressEOS,rhoEOS)
      END SUBROUTINE cond_iniciais

```

```

!-----
! fator de Lorentz - gammaL
! funcao velocidade - radial - veloc
! velocidade Kepleriana - VK
! EOS - relacao entre dens e press
!-----

SUBROUTINE cond_iniciais2(x1,x2,alpha,Gamma,GD,D,V,P,Et,B,E,T)

USE ferramentas

IMPLICIT NONE

INTEGER :: m,n,i,j
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:) :: x1,x2,alpha
REAL, INTENT(OUT) :: Gamma
REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,:) :: GD,D,Et
REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,:,:) :: V,P,B,E
REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,:,:,:) :: T
REAL :: H,delta,rD,EOS,cot,alphaD,gammaD,rhoD,EOSD,BD,nV2
REAL, ALLOCATABLE, DIMENSION(:) :: gammaL,veloc,VK,pressEOS,rhoEOS
REAL, ALLOCATABLE, DIMENSION(:,:) :: press,rho,energ,nE,nB,nV,nP
REAL, ALLOCATABLE, DIMENSION(:,:,:) :: EB

m = SIZE(x1)
n = SIZE(x2)

!
!-----
! Variaveis Primitivas
!-----
      ALLOCATE(D(m,n),P(3,m,n),Et(m,n),B(3,m,n))
!
!-----
! Outras Variaveis de saida
!-----
      ALLOCATE(V(3,m,n),E(3,m,n),T(3,3,m,n),GD(m,n))
!
!-----
! Variaveis internas
!-----
      ALLOCATE(gammaL(m),veloc(m),VK(m),press(m,n),rho(m,n))
      ALLOCATE(rhoEOS(m),pressEOS(m),energ(m,n))
      ALLOCATE(EB(3,m,n),nE(m,n),nB(m,n),nV(m,n),nP(m,n))
!
!-----
! Constantes Fisicas
!-----
      Gamma = 5.0/3.0 ! razao do calor especifico - gas politropico
      H = 1.3 ! entalpia
!
!-----
! Constantes do disco de acrecao
!-----
!
!-----
! Constantes do disco de acrecao
!-----
      delta = 0.125 ! abertura do disco
      rD = 3.0 ! posicao de inicio do disco
      rD = 1.0 + exp(rD) ! posicao de inicio do disco
! em coord. tortoise
!

DO i=1,m,1
  CALL raphson_newton(alpha(i),Gamma,H,gammaL(i))
  veloc(i)=SQRT(1.0-(1.0/(gammaL(i)*gammaL(i))))
  VK(i)=1/(SQRT((2*(x1(i)-1))))
  EOS = ((Gamma-1.0)/Gamma)*((H/(alpha(i)*gammaL(i)))-1.0)
  rhoEOS(i) = EOS*(1.0/(Gamma-1.0)) ! rhoffc
  pressEOS(i) = EOS*rhoEOS(i) ! pressffc
END DO
print*,'VK = '
print*,'VK
print*,'veloc = '
print*,'veloc
print*,' alpha = '
print*,' alpha
!-----
! distribuicao de velocidade no dominio - V
! EOS - distribuicao de densidade - rho
! EOS - distribuicao de pressao - press
! distribuicao de energia total - energ
! considerando o disco de acrecao
!-----
DO j=1,3,1
  cot = COS(x2(j))/SIN(x2(j))
  DO i=1,m,1
    IF ((x1(i)>rD).AND.(-0.125<cot).AND.(cot<0.125)) THEN
      V(1,i,j) = 0.0
      V(2,i,j) = 0.0
      V(3,i,j) = VK(i)
      rho(i,j) = 101.0*rhoEOS(i)
    ELSE
      V(1,i,j) = -veloc(i)
      V(2,i,j) = 0.0
      V(3,i,j) = 0.0
      rho(i,j) = rhoEOS(i)
    END IF
    press(i,j) = pressEOS(i)
    energ(i,j) = rho(i,j) + (press(i,j)/(Gamma-1.0))
  END DO
END DO
j=4
DO i=1,m,1
  IF ((x1(i) > rD).AND.(-0.125 < 1).AND.(1 < 0.125) ) THEN
    V(1,i,j) = 0.0
    V(2,i,j) = 0.0
    V(3,i,j) = VK(i)
    rho(i,j) = 101.0*rhoEOS(i)
  ELSE
    V(1,i,j) = -veloc(i)
    V(2,i,j) = 0.0
    V(3,i,j) = 0.0
    rho(i,j) = rhoEOS(i)
  END IF
  press(i,j) = pressEOS(i)
  energ(i,j) = rho(i,j) + (press(i,j)/(Gamma-1.0))
END DO

```

```

DO j=5,n,1
  cot = COS(x2(j))/SIN(x2(j))
  DO i=1,m,1
    IF ((x1(i)>rD).AND.(-0.125<cot).AND.(cot<0.125)) THEN
      V(1,i,j) = 0.0
      V(2,i,j) = 0.0
      V(3,i,j) = VK(i)
      rho(i,j) = 101.0*rhoEOS(i)
    ELSE
      V(1,i,j) = -veloc(i)
      V(2,i,j) = 0.0
      V(3,i,j) = 0.0
      rho(i,j) = rhoEOS(i)
    END IF
    press(i,j) = pressEOS(i)
    energ(i,j) = rho(i,j) + (press(i,j)/(Gamma-1.0))
  END DO
END DO

!
!-----
! distribuicao do Campo Magnetico
!-----

alphaD = SQRT(1.0-(1.0/rD))
CALL raphson_newton(alphaD,Gamma,H,gammaD)
EOSD = ((Gamma-1.0)/Gamma)*((H/(alphaD*gammaD))-1.0)
rhoD = (EOSD**1.0/(Gamma-1.0)) ! rhoffc
BD = 0.3*SQRT(rhoD)
! BD = 10*SQRT(rhoD)
DO i=1,m,1
  DO j=1,n,1
    B(1,i,j) = BD*COS(x2(j))
    B(2,i,j) = -alpha(i)*BD*COS(x2(j))
    B(1,i,j) = BD*COS(x2(j))
    B(2,i,j) = -alpha(i)*BD*SIN(x2(j))
    B(3,i,j) = 0.0
  END DO
END DO

!
!-----
! campo eletrico
!-----

CALL prod_vec(m,n,V,B,E)
E = -E
CALL prod_vec(m,n,E,B,EB)

!
!-----
! distribuicao - fator de Lorentz - GD
! distribuicao de Densidade - D
! distribuicao de Pressao - P
!-----

PRINT *, 'ESTOU AI'
DO i=1,m,1
  DO j=1,n,1
    nV2 = V(1,i,j)*V(1,i,j) + V(2,i,j)*V(2,i,j) + V(3,i,j)*V(3,i,j)
    GD(i,j) = 1.0/SQRT(1.0-nV2)
    D(i,j)=GD(i,j)*rho(i,j)
  END DO
END DO

P(1,i,j)=((energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(1,i,j))+EB(1,i,j))
P(2,i,j)=((energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(2,i,j))+EB(2,i,j))
P(3,i,j)=((energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(3,i,j))+EB(3,i,j))
END DO
END DO
PRINT *, 'ESTOU AI'
!
!-----
! normas
!-----
CALL norma(m,n,P,nP)
CALL norma(m,n,B,nB)
CALL norma(m,n,E,nE)
!
!-----
! distribuicao de Energ. total Et
! distribuicao de Stress Tensor - T
!-----
DO i=1,m,1
  DO j=1,n,1
    Et(i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)
    Et(i,j) = Et(i,j) - press(i,j) - D(i,j)
    Et(i,j) = Et(i,j) + 0.5*((nB(i,j)**2)+(nE(i,j)**2))
    T(1,1,i,j) = press(i,j) + 0.5*((nB(i,j)**2)+(nE(i,j)**2))
    T(2,2,i,j) = T(1,1,i,j)
    T(3,3,i,j) = T(1,1,i,j)
    T(1,1,i,j) = T(1,1,i,j)+(energ(i,j)+press(i,j))* &
      & ((GD(i,j)**2)*(V(1,i,j)*V(1,i,j)))
    T(2,2,i,j) = T(2,2,i,j)+(energ(i,j)+press(i,j))* &
      & (GD(i,j)**2)*(V(2,i,j)*V(2,i,j))
    T(3,3,i,j) = T(3,3,i,j)+(energ(i,j)+press(i,j))* &
      & (GD(i,j)**2)*(V(3,i,j)*V(3,i,j))
    T(1,2,i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(1,i,j)*V(2,i,j))
    T(1,3,i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(1,i,j)*V(3,i,j))
    T(2,3,i,j) = (energ(i,j)+press(i,j))*(GD(i,j)**2)*(V(2,i,j)*V(3,i,j))
    T(1,1,i,j) = T(1,1,i,j) - B(1,i,j)*B(1,i,j) - E(1,i,j)*E(1,i,j)
    T(2,2,i,j) = T(2,2,i,j) - B(2,i,j)*B(2,i,j) - E(2,i,j)*E(2,i,j)
    T(3,3,i,j) = T(3,3,i,j) - B(3,i,j)*B(3,i,j) - E(3,i,j)*E(3,i,j)
    T(1,2,i,j) = T(1,2,i,j) - B(1,i,j)*B(2,i,j) - E(1,i,j)*E(2,i,j)
    T(1,3,i,j) = T(1,3,i,j) - B(1,i,j)*B(3,i,j) - E(1,i,j)*E(3,i,j)
    T(2,3,i,j) = T(2,3,i,j) - B(2,i,j)*B(3,i,j) - E(2,i,j)*E(3,i,j)
    T(2,1,i,j) = T(1,2,i,j)
    T(3,1,i,j) = T(1,3,i,j)
    T(3,2,i,j) = T(2,3,i,j)
  END DO
END DO
DEALLOCATE(EB,press,rho,energ,nE,nB,nV,nP,gammaL,veloc,VK,pressEOS,rhoEOS)
END SUBROUTINE cond_iniciais
END MODULE condicoes_iniciais

```

Módulo: discretizacoes.f90

```

MODULE discretizacoes
    IMPLICIT NONE
    CONTAINS
    SUBROUTINE malha(Nt,N1,N2,CFL,a1,b1,a2,b2,pi,x1,x2,D1,D2,Dt,m,n)
        IMPLICIT NONE
        INTEGER :: i
        INTEGER, INTENT(IN) :: Nt,N1,N2
        INTEGER, INTENT(OUT) :: m,n
        REAL, INTENT(IN) :: CFL,a1,b1,a2,b2,pi
        REAL, INTENT(OUT) :: D1,D2,Dt
        REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:) :: x1,x2

        m = N1+7
        n = N2+7

        ALLOCATE(x1(m),x2(n))

        D1 = (b1 - a1)/REAL(N1)
        D2 = (b2 - a2)/REAL(N2)

        print *, a1,b1,D1,D2

        Dt = CFL*min(D1,D2)

        DO i=1,m,1
            x1(i) = 0.0
            x1(i) = a1 + (i-4)*D1
            ! x1(i) = log(x1(i)-1) ! coord. radial tortoise
            ! x1(i) = exp(x1(i)) + 1.0
        END DO

        DO i=1,n,1
            x2(i) = 0.0
        END DO
        x2(i) = a2 + (i-4)*D2
    END SUBROUTINE malha

    SUBROUTINE metrica(m,n,pi,x1,x2,alpha,hs,Jacs,Gs)
        IMPLICIT NONE
        INTEGER, INTENT(IN) :: m,n
        REAL, INTENT(IN) :: pi
        REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:) :: x1,x2
        REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:) :: alpha
        REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,:,:) :: hs,Jacs,Gs
        INTEGER :: i,j
        REAL :: Jacobiano, Jac_1

        ALLOCATE(alpha(m),hs(4,m,n),Jacs(5,m,n),Gs(4,m,n))
        !
        !-----
        ! funcao lapso
        !-----
        DO i=1,m,1
            alpha(i) = sqrt(1.0-(1.0/x1(i)))
        END DO

        !
        !-----
        ! Coef. Metrica e combinacoes
        !-----
        DO i=1,m,1
            DO j=1,n,1
                hs(1,i,j) = alpha(i) ! coeficiente h0
                hs(2,i,j) = 1.0/alpha(i) ! coeficiente h1
                hs(3,i,j) = x1(i) ! coeficiente h2
                hs(4,i,j) = x1(i)*sin(x2(j)) ! coeficiente h3
                Jacobiano = hs(2,i,j)*hs(3,i,j)*hs(4,i,j) ! Jacobiano
                IF (j==4) THEN
                    Jac_1 = 1 ! rever este valor
                ELSE
                    Jac_1 = 1/Jacobiano
                END IF
                Jacs(1,i,j) = Jacobiano
                Jacs(2,i,j) = hs(2,i,j)*Jac_1
                Jacs(3,i,j) = hs(3,i,j)*Jac_1
                Jacs(4,i,j) = hs(4,i,j)*Jac_1
                Jacs(5,i,j) = Jac_1
            END DO
        END DO
        !hs = abs(hs); ! talvez voltar
        !Jacs = abs(Jacs);

        !
        !-----
        ! Coef. Tensor Metrico
        !-----
        DO i=1,m,1
            DO j=1,n,1
                Gs(1,i,j) = 0.5*(1.0/(x1(i)**2))*(1.0/(sqrt(1.0-(1/x1(i))))))
                Gs(2,i,j) = alpha(i)/x1(i)
                Gs(3,i,j) = Gs(2,i,j)
                IF (j==4) THEN
                    Gs(4,i,j) = 0
                ELSE
                    Gs(4,i,j) = (1/x1(i))*(cos(x2(j)))/sin(x2(j))
                END IF
            END DO
        END DO
    END SUBROUTINE metrica
END MODULE discretizacoes

```

Módulo: EOS.f90

```

MODULE EOS

IMPLICIT NONE

CONTAINS

SUBROUTINE newton_metEOS(m,n,Gamma,cD,P,Et,cB,GD,V)

  IMPLICIT NONE

  INTEGER, INTENT(IN) :: m,n
  REAL, INTENT(IN) :: Gamma
  REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :) :: Et, cD
  REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :, :) :: P, cB
  REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:, :) :: GD
  REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:, :, :) :: V

  INTEGER :: i,j,quant,t
  REAL :: precisao,pt1,pt2,pt3,pt4,pt5,W,W1,W2,tt1,tt2,tt
  REAL, ALLOCATABLE, DIMENSION(:) :: x,delta_x,FF
  REAL, ALLOCATABLE, DIMENSION(:, :) ::
    nB,nP,a,b,d,s,Jac,Xdef,Ydef,gdef,Vmod

  ALLOCATE(cB(3,m,n),cD(m,n))
  ALLOCATE(nB(m,n),nP(m,n),a(m,n),b(m,n),d(m,n),s(m,n))
  ALLOCATE(Jac(2,2),x(2),FF(2),delta_x(2))
  ALLOCATE(Xdef(m,n),Ydef(m,n),gdef(m,n),Vmod(m,n))

  PRINT*, 'Passei no EOS'

  !
  !-----
  ! Normas
  !-----
  DO i=1,m,1
    DO j=1,n,1
      nB(i,j) = SQRT((cB(1,i,j)**2) + (cB(2,i,j)**2) + (cB(3,i,j)**2))
      nP(i,j) = SQRT((P(1,i,j)**2) + (P(2,i,j)**2) + (P(3,i,j)**2))
    END DO
  END DO

  !
  !-----
  ! Valores conhecidos
  !-----
  DO i=1,m,1
    DO j=1,n,1
      a(i,j) = cD(i,j) + Et(i,j)
      b(i,j) = (Gamma-1)*cD(i,j)
      d(i,j) = (1 - (0.5*Gamma))*nB(i,j)*nB(i,j)
      s(i,j) = (cB(1,i,j)*P(1,i,j)) + (cB(2,i,j)*P(2,i,j)) +
        (cB(3,i,j)*P(3,i,j))
    END DO
  END DO

  Jac = 0
  PRINT*, ' Jac = ', Jac

  !
  !-----
  ! Metodo de Newton
  ! para cada (i,j)
  !-----
  precisao = 1e-3

  quant = 0

  DO i=1,m,1
    DO j=1,n,1
      !
      !-----
      ! Escolha inicial
      !-----
      x(1) = GD(i,j)
      x(2) = s(i,j)

      !
      !-----
      ! Sistema Linear%
      !-----
      ! Escolha inicial
      !-----
      !
      !-----
      ! F1 e derivadas
      !-----
      pt1 = 0
      pt1=(Gamma*a(i,j)*x(1)*x(1))+((2*Gamma*a(i,j)-b(i,j))*x(1))
      pt1=pt1+(Gamma*a(i,j)-b(i,j)+d(i,j))+0.5*Gamma*x(2)*x(2)

      pt2 = 0
      pt2=(nP(i,j)*nP(i,j)*((x(1)+1)**2))+2*s(i,j)*x(2)*(1+x(1))
      pt2=pt2+(nB(i,j)*nB(i,j)*x(2)*x(2))

      pt3 = 0
      pt3=(Gamma*x(1)*x(1))+2*Gamma*x(1) + 1

      pt4 = 0
      pt4 = (2*Gamma*a(i,j)*(x(1)+1)) - b(i,j)

      pt5 = 0
      pt5 = (2*nP(i,j)*nP(i,j)*(x(1)+1)) + (2*s(i,j)*x(2))

      FF(1) = 0
      FF(1) = (x(1)*(x(1)+2))*(pt1**2) - ((pt3**2)*pt2)

      Jac(1,1) = 0
      Jac(1,1) = 2*(x(1)+2)*pt1*pt1
      Jac(1,1) = Jac(1,1) + (2*x(1)*(x(1)+2)*pt1)*(pt4)
      Jac(1,1) = Jac(1,1) - (2*pt3*(2*Gamma*(x(1)+1))*pt2)
      Jac(1,1) = Jac(1,1) - ((pt3**2)*(pt5))

      Jac(1,2) = 0
      Jac(1,2) = 2*x(1)*(x(1)+2)*(pt1*Gamma*x(2))
      Jac(1,2)=Jac(1,2)-(pt3*pt3)*((2*s(i,j)*(x(1)+1))+2*nB(i,j)*nB(i,j)*x(2))

      !
      !-----
      ! F2 e derivadas
      !-----
      pt1 = 0
      pt1 = Gamma*(a(i,j) - (nB(i,j)*nB(i,j)))

      pt2 = 0
      pt2 = (2*pt1) - b(i,j)

      pt4 = 0
      pt4=Gamma*a(i,j)-b(i,j)+d(i,j)-(nB(i,j)*nB(i,j))+0.5*Gamma*x(2)*x(2)
    END DO
  END DO

```



```

        Jac(2,1) = ((2*pt1*(x(1)+1)) - b(i,j))*x(2) - pt5      ! Escolhendo a
                                                                ! direcao: x1, x2 ou x3
        Jac(2,2) = 0                                           !-----
        Jac(2,2) = ((pt1*x(1)*x(1)) + pt2*x(1) + pt4 ) + (Gamma*x(2)*x(2))
!
!-----
!
                                                                !if (direcao < 2)
                                                                !Vmod(i,j) = Ydef(i,j)/(gamma(i,j)*Bmod(i,j));
                                                                !!V1(i,j) = Ydef(i,j)/(gamma(i,j)*nB(i,j));
                                                                !elseif (direcao == 2)
                                                                !Vmod(i,j) = Ydef(i,j)/(gamma(i,j)*nB(i,j));
                                                                !else
                                                                !Vmod(i,j) = Ydef(i,j)/(gamma(i,j)*nB(i,j));
                                                                !Vmod(i,j) = 1 - (1/(gamma(i,j)*gamma(i,j)));
                                                                !end
                                                                END DO
                                                                END DO
                                                                GD = gdef
                                                                V(2, :, :) = (Ydef(:, :)/GD(:, :)) ! ok
                                                                ! V(2, :, :) = (Ydef(:, :)/GD(:, :))*nB(:, :) ! ok
                                                                ! V(2, :, :) = (Ydef(:, :)/GD(:, :))*nB(:, :)) ! problema no jato
                                                                ! V(2, :, :) = (Vmod(:, :)/GD(:, :))*nB(:, :)) ! problema no jato
                                                                ! V(2, :, :) = ((Ydef(:, :)/GD(:, :))*nB(:, :)) - (V(3, :, :)**2) + (V(1, :, :)**2)
                                                                ! V(2, :, :) = (Ydef(:, :)/GD(:, :)) - ((V(3, :, :)**2) + (V(1, :, :)**2)) ! ok
                                                                ! V(2, :, :) = Ydef(:, :) - (V(3, :, :)**2) - (V(1, :, :)**2)
                                                                !V(2, :, :) = ((Ydef(:, :)/GD(:, :)) - (V(1, :, :)*cB(1, :, :)+V(3, :, :)*cB(3, :, :)))/(cB(2, :, :))
                                                                !V(2, :, :) = (Ydef(:, :)/GD(:, :)) - (V(1, :, :)+V(3, :, :))
                                                                ! V(2, :, :) = Vmod(:, :) - (V(3, :, :)**2) - (V(1, :, :)**2)
                                                                ! V(2, :, :) = Vmod(:, :)
                                                                V(2, :, :) = SQRT(ABS(V(2, :, :)))
                                                                ! V(2, :, :) = Vmod(:, :) - (V(3, :, :)**2) + (V(1, :, :)**2)
                                                                ! V(2, :, :) = SQRT(ABS(V(2, :, :)))/SQRT(V(1, :, :)**2 + V(2, :, :)**2 + V(3, :, :)**2)
                                                                ! V(2, :, :) = (ABS(V(2, :, :)))/(V(1, :, :)**2 + V(2, :, :)**2 + V(3, :, :)**2)
                                                                !Vmod(Nr+1, 2) = 0.5*(Vmod(Nr+1, 1)+Vmod(Nr+1, 3)); % artificial
                                                                END SUBROUTINE newton_metEOS
!
!-----
END MODULE EOS

```

Módulo: ferramentas.f90

MODULE ferramentas

IMPLICIT NONE

CONTAINS

SUBROUTINE raphson_newton(alpha_in,Gamma,H,x_out)

IMPLICIT NONE

REAL, INTENT(IN) :: alpha_in,Gamma,H

REAL, INTENT(OUT) :: x_out

REAL :: x_new,x_old,poly_x,Dpoly_x

x_new = 1.5

poly_x = alpha_in*(Gamma-1)*(x_new**3.0) +
(2.0-Gamma)*(x_new**2) - H

DO WHILE (abs(poly_x) >= 1e-6)

x_old = x_new

poly_x = alpha_in*(Gamma-1.0)*(x_old**3.0) +
(2.0-Gamma)*(x_old**2.0) - H

Dpoly_x = 3.0*alpha_in*(Gamma-1.0)*(x_old**2.0) +
2.0*(2.0-Gamma)*x_old

x_new = x_old - (poly_x/Dpoly_x)

poly_x = alpha_in*(Gamma-1.0)*(x_new**3.0) +
(2.0-Gamma)*(x_new**2.0) - H

END DO

x_out = x_new

END SUBROUTINE raphson_newton

SUBROUTINE prod_vec(m,n,A,B,C)

IMPLICIT NONE

INTEGER m,n,i,j

REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :, :) :: A,B

REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:, :, :) :: C

ALLOCATE(C(3,m,n))

DO i=1,m,1

DO j=1,n,1

C(1,i,j) = A(2,i,j)*B(3,i,j) - A(3,i,j)*B(2,i,j)

C(2,i,j) = A(3,i,j)*B(1,i,j) - A(1,i,j)*B(3,i,j)

C(3,i,j) = A(1,i,j)*B(2,i,j) - A(2,i,j)*B(1,i,j)

END DO

END DO

END SUBROUTINE prod_vec

SUBROUTINE norma(m,n,A,nA)

IMPLICIT NONE

INTEGER m,n,i,j

REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :, :) :: A

REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:, :, :) :: nA

ALLOCATE(nA(m,n))

DO i=1,m,1

DO j=1,n,1

nA(i,j) = SQRT(A(1,i,j)**2 + A(2,i,j)**2 + A(3,i,j)**2)

END DO

END DO

END SUBROUTINE norma

SUBROUTINE intLagrange(xi,yi,x,y,n)

IMPLICIT NONE

INTEGER :: j,k

INTEGER :: n

REAL :: prod1,prod2

REAL, ALLOCATABLE, DIMENSION(:) :: L

REAL, INTENT(IN) :: xi

REAL, INTENT(OUT) :: yi

REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:) :: x,y

ALLOCATE(L(n+1))

yi = 0.0

DO k=1,n+1,1

prod1 = 0.0

prod2 = 0.0

prod1 = 1.0

prod2 = 1.0

DO j=1,n+1,1

IF(j /= k) THEN

prod1 = prod1*(xi - x(j))

prod2 = prod2*(x(k) - x(j))

END IF

END DO

L(k) = prod1/prod2

yi = L(k)*y(k) + yi

END DO

DEALLOCATE(L)

END SUBROUTINE intLagrange

END MODULE ferramentas

Módulo: fluxos.f90

```

MODULE fluxos
    IMPLICIT NONE
    CONTAINS
        SUBROUTINE fluxos_EDPs(m,n,hs,D,V,T,P,E,FD,FP,FEt,FB)
            IMPLICIT NONE
            INTEGER :: i,j
            INTEGER, INTENT(IN) :: m,n
            ! REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,) :: D
            ! REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,,:) :: V,P,E,hs
            ! REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,,:,:) :: T
            REAL, INTENT(IN), DIMENSION(:,) :: D
            REAL, INTENT(IN), DIMENSION(:,,:,:) :: V,P,E,hs
            REAL, INTENT(IN), DIMENSION(:,,:,:) :: T
            REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,,:) :: FD,FEt
            REAL, INTENT(OUT), ALLOCATABLE, DIMENSION(:,,:,:) :: FB,FP
            ! ALLOCATE(hs(4,m,n),D(m,n),V(3,m,n),P(3,m,n),E(3,m,n),T(3,3,m,n))
            ALLOCATE(FD(3,m,n),FEt(3,m,n),FB(3,3,m,n),FP(3,3,m,n))
            DO i=1,m,1
                DO j=1,n,1
                    FD(1,i,j) = hs(1,i,j)*hs(3,i,j)*hs(4,i,j)*D(i,j)*V(1,i,j)
                    FD(2,i,j) = hs(1,i,j)*hs(4,i,j)*hs(2,i,j)*D(i,j)*V(2,i,j)
                    FD(3,i,j) = hs(1,i,j)*hs(2,i,j)*hs(3,i,j)*D(i,j)*V(3,i,j)
                    FEt(1,i,j)=hs(1,i,j)*hs(3,i,j)*hs(4,i,j)*(P(1,i,j)-D(i,j)*V(1,i,j))
                    FEt(2,i,j)=hs(1,i,j)*hs(4,i,j)*hs(2,i,j)*(P(2,i,j)-D(i,j)*V(2,i,j))
                    FEt(3,i,j)=hs(1,i,j)*hs(2,i,j)*hs(3,i,j)*(P(3,i,j)-D(i,j)*V(3,i,j))
                    FB(1,2,i,j) = hs(1,i,j)*hs(4,i,j)*E(3,i,j)
                    FB(1,3,i,j) = (-1)*hs(1,i,j)*hs(3,i,j)*E(2,i,j)
                    FB(2,1,i,j) = (-1)*hs(1,i,j)*hs(4,i,j)*E(3,i,j)
                    FB(2,3,i,j) = hs(1,i,j)*hs(2,i,j)*E(1,i,j)
                    FB(3,1,i,j) = hs(1,i,j)*hs(3,i,j)*E(2,i,j)
                    FB(3,2,i,j) = (-1)*hs(1,i,j)*hs(2,i,j)*E(1,i,j)
                END DO
            END DO
        END SUBROUTINE fluxos_EDPs
    END MODULE fluxos

```

Módulo: metodos_EDPs.f90

```

MODULE metodos_EDPs
    IMPLICIT NONE
    CONTAINS
        SUBROUTINE nes_tad(m,n,D1,D2,Dt,Jacob,U,FU,GU,HU,W)
            IMPLICIT NONE
            !
            INTEGER :: i,j
            INTEGER, INTENT(IN) :: m,n
            REAL :: Umedio,conta,FRx1,FLx1,FRx2,FLx2,GRy1,GRy2,GLy1,GLy2
            REAL, INTENT(IN) :: D1,D2,Dt
            REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,) :: FU,GU,HU,Jacob
            REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,) :: U
            REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:,) :: W
            ! ALLOCATE(FU(m,n),GU(m,n),HU(m,n),Jacob(m,n),U(m,n),W(m,n))
            DO i=4,m-3,1
                DO j=4,n-3,1
                    FRx1 = FU(i+1,j-1)
                    FLx1 = FU(i-1,j-1)
                    FRx2 = FU(i+1,j+1)
                    FLx2 = FU(i-1,j+1)
                    GRy1 = GU(i-1,j+1)
                    GLy1 = GU(i-1,j-1)
                    GRy2 = GU(i+1,j+1)
                    GLy2 = GU(i+1,j-1)
                    conta = 0.25*(U(i+1,j)+U(i-1,j)+U(i,j+1)+U(i,j-1))
                    W(i,j) = conta - (Jacob(i,j))*((Dt/(4*D1))*((FRx1-FLx1) + (FRx2-FLx2)))
                    W(i,j) = W(i,j) - (Jacob(i,j))*((Dt/(4*D2))*((GRy1-GLy1) + (GRy2-GLy2)))
                END DO
            END DO
        END SUBROUTINE nes_tad
        SUBROUTINE laxfri_conservativo(m,n,D1,D2,Dt,Jacob,U,FU,GU,HU,W)

```

```

IMPLICIT NONE

INTEGER :: i,j
INTEGER, INTENT(IN) :: m,n
REAL :: Umedio,conta, FRx,FLx,FRy,FLy
REAL, INTENT(IN) :: D1,D2,Dt
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :) :: FU,GU,HU, Jacob
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :) :: U
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:, :) :: W

!      ALLOCATE(FU(m,n),GU(m,n),HU(m,n),Jacob(m,n),U(m,n),W(m,n)) !

DO i=4,m-3,1
  DO j=4,n-3,1
    FRx = (D1/(4*Dt))*(U(i,j) - U(i+1,j)) + 0.5*(FU(i,j) + FU(i+1,j))
    FLx = (D1/(4*Dt))*(U(i-1,j) - U(i,j)) + 0.5*(FU(i-1,j) + FU(i,j))
    FRy = (D2/(4*Dt))*(U(i,j) - U(i,j+1)) + 0.5*(GU(i,j) + GU(i,j+1))
    FLy = (D2/(4*Dt))*(U(i,j-1) - U(i,j)) + 0.5*(GU(i,j-1) + GU(i,j))
!
W(i,j) = U(i,j) - (Jacob(i,j))*((Dt/D1)*(FRx-FLx)) + ((Dt/D2)*(FRy-FLy))

!      W(i,j) = U(i,j) - ((Dt/D1)*(FRx-FLx))

      END DO
    END DO
  END DO

END SUBROUTINE laxfri_conservativo

SUBROUTINE FDFC4(m,n,D1,D2,Dt, Jacob,U,FU,GU,HU,W)

```

```

IMPLICIT NONE

INTEGER :: i,j
INTEGER, INTENT(IN) :: m,n
REAL :: Umedio, FX,FY
REAL, INTENT(IN) :: D1,D2,Dt
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :) :: FU,GU,HU, Jacob
REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:, :) :: U
REAL, INTENT(INOUT), ALLOCATABLE, DIMENSION(:, :) :: W

!      ALLOCATE(FU(m,n),GU(m,n),HU(m,n),Jacob(m,n),U(m,n),W(m,n))

DO i=4,m-3,1
  DO j=4,n-3,1
    FX = U(i-2,j) - 8*U(i-1,j) + 8*U(i+1,j) - U(i+2,j)
    FY = U(i,j-2) - 8*U(i,j-1) + 8*U(i,j+1) - U(i,j+2)
    Umedio = 0.25*(U(i+1,j) + U(i-1,j) + U(i,j+1) + U(i,j-1))
    W(i,j) = Umedio - (Jacob(i,j))*((Dt/(12*D1))*FX + (Dt/(12*D2))*FY)
    W(i,j) = U(i,j) - ((Dt/D1)*(FRx-FLx)) - ((Dt/D2)*(FRy-FLy))

      END DO
    END DO
  END DO

END SUBROUTINE FDFC4

END MODULE metodos_EDPs

```

Programa Principal: programa_principal.f90

```

PROGRAM programa_principal
!
!objetivo: programa principal GRMHD - jatos
!
  USE unidades_arquivos
  USE valores_iniciais
  USE discretizacoes
  USE condicoes_iniciais
  USE fluxos
  USE solucao_numerica
  USE condicoes_fronteiras
  USE termos_adicionais
  USE atualizacoes
  USE armazenamentos

  IMPLICIT NONE

  INTEGER :: Nt,N1,N2
  REAL :: CFL,a1,b1,a2,b2,pi
  INTEGER :: m,n,tp,i,j
  REAL :: D1,D2,Dt,Dt1,tempo
  REAL, ALLOCATABLE, DIMENSION(:) :: x1,x2
  REAL, ALLOCATABLE, DIMENSION(:) :: alpha
  REAL, ALLOCATABLE, DIMENSION(:, :, :) :: hs,Jacs,Gs

  REAL :: Gamma
  REAL, ALLOCATABLE, DIMENSION(:, :) :: GD,D,Et
  REAL, ALLOCATABLE, DIMENSION(:, :, :) :: V,P,B,E
  REAL, ALLOCATABLE, DIMENSION(:, :, :) :: T
  REAL, ALLOCATABLE, DIMENSION(:, :, :) :: FD,FEt
  REAL, ALLOCATABLE, DIMENSION(:, :, :, :) :: FB,FP

  REAL, ALLOCATABLE, DIMENSION(:, :) :: Dold,Etold,Din,Etin
  REAL, ALLOCATABLE, DIMENSION(:, :, :) :: Pold,Bold,Pin,Bin

  CALL unidades

  ALLOCATE(x1(m),x2(n),alpha(m),hs(4,m,n),Jacs(5,m,n),Gs(4,m,n))

  ALLOCATE(GD(m,n),D(m,n),Et(m,n),V(3,m,n),P(3,m,n),B(3,m,n),E(3,m,n))
  ALLOCATE(T(3,3,m,n))

  ALLOCATE(FD(3,m,n),FEt(3,m,n),FB(3,3,m,n),FP(3,3,m,n))

  ALLOCATE(Dold(m,n),Etold(m,n),Pold(3,m,n),Bold(3,m,n))

  ALLOCATE(Din(m,n),Etin(m,n),Pin(3,m,n),Bin(3,m,n))

  CALL valores_definicoes(Nt,N1,N2,CFL,a1,b1,a2,b2,pi)

```

```

CALL malha(Nt,N1,N2,CFL,a1,b1,a2,b2,pi,x1,x2,D1,D2,Dt,m,n)

CALL metrica(m,n,pi,x1,x2,alpha,hs,Jacs,Gs)

CALL cond_iniciais(x1,x2,alpha,Gamma,GD,D,V,P,Et,B,E,T)

! CALL arquivos_saida(m,n,tp,D,P,Et,E,B,V,GD,x1,x2)

      Din = D
      Etin = Et
      Pin = P
      Bin = B

      Dti = Dt

DO tp=1,Nt,1

!      IF(tp < 1000) THEN
!          Dt = Dti
!      ELSE
!          Dt = Dti/2.0
!      END IF

WRITE(*,*) 'iteracao ',tp

      Dold = D
      Etold = Et
      Pold = P
      Bold = B

CALL fluxos_EDPs(m,n,hs,D,V,T,P,E,FD,FP,FET,FB)

WRITE(*,*) 'fluxos_EDPs OK'

!      CALL arquivos_saida(m,n,tp,D,P,Et,E,B,V,GD,x1,x2)

CALL metodos_numericos(m,n,D1,D2,Dt,Jacs,FD,FP,FET,FB,D,P,Et,B)

WRITE(*,*) 'metodos_numericos OK'

CALL fonte(m,n,Dt,Gs,D,T,P,Et)

WRITE(*,*) 'fonte OK'

CALL condicoes_contorno(m,n,D,P,Et,B,Dold,Pold,Etold,Bold)
! jato nao alimentado

CALL condicoes_contorno2(m,n,D,P,Et,B,x1,x2)
! CALL condicoes_contorno3(m,n,D,P,Et,B,x1,x2)
! CALL condicoes_contorno4(m,n,D,P,Et,B,Dold,Pold,Etold,Bold,Din,Etin,Pin,Bin)
! jato alimentado

WRITE(*,*) 'contorno OK'

CALL grandezas_atualizadas(m,n,x1,x2,Gamma,GD,D,P,B,Et,V,T,E)

WRITE(*,*) 'atualizacoes OK'

CALL arquivos_saida(m,n,tp,D,P,Et,E,B,V,GD,x1,x2)

WRITE(*,*) 'armazenamentos OK'

CALL CPU_TIME(tempo)

WRITE(*,*) 'iteracao ',tp,' de ',Nt,', tempo em execucao ', &
& tempo, ' segundos'

!      DEALLOCATE(FD,FET,FB,FP)

if ((tp == 2000)) then
      Dt = Dt/8.0;
endif

END DO

CALL arquivos_saida(m,n,tp,D,P,Et,E,B,V,GD,x1,x2)

END PROGRAM programa_principal

```

Módulo: solucao_numerica.f90

```

MODULE solucao_numerica

IMPLICIT NONE

CONTAINS

SUBROUTINE metodos_numericos(m,n,D1,D2,Dt,Jacs,FD,FP,FET,FB,D,P,Et,B)

      USE metodos_EDPs

      IMPLICIT NONE

      INTEGER :: i,j
      REAL, ALLOCATABLE, DIMENSION(:) :: NOSSA
      INTEGER, INTENT(IN) :: m,n
      REAL, INTENT(IN) :: D1,D2,Dt
      REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,:,:) :: FD,FET,Jacs
      REAL, INTENT(IN), ALLOCATABLE, DIMENSION(:,:,:) :: FB,FP

      ALLOCATE(NOSSA(n))
      PRINT*, 'SOLUCAO NUMERICA'

      ALLOCATE(FU(m,n),GU(m,n),HU(m,n),Jacob(m,n),U(m,n),W(m,n),Zerar(m,n))
      PRINT*, 'SOLUCAO NUMERICA - 2'

      DO i=1,m,1
        DO j=1,n,1
          Zerar(i,j) = 0.0
        END DO
      END DO

```