

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

**Modelo Computacional Paralelo para a
Hidrodinâmica e para o
Transporte de Substâncias Bidimensional e Tridimensional**

por

ROGÉRIO LUIS RIZZI

Tese submetida à avaliação,
como requisito parcial para a obtenção do grau de
Doutor em Ciência da Computação

Prof. Dr. Tiarajú Asmuz Diverio
Orientador

Porto Alegre, agosto de 2002

CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Rizzi, Rogério Luis

Modelo Computacional Paralelo para a Hidrodinâmica e para o Transporte de substâncias 2D e 3D/por Rogério Luis Rizzi.-Porto Alegre: PPGC da UFRGS, 2002
254 p.: il.

Tese (doutorado) - Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR - RS, 2002. Orientador: Diverio, Tiarajú Asmuz.

1. Equações *shallow water* 3D. 2. Transporte escalar de substâncias 3D. 3. Esquema numérico semi-implícito. 4. Refinamento local. 5. Método dos fluxos limitados. 6. Decomposição de domínio. I. Diverio, Tiarajú Asmuz. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitora: Prof^a. Wrana Maria Panizzi

Pró-Reitor de Ensino: Prof. José Carlos Ferraz Hennemann

Pró-Reitor Adjunto de Pós-graduação: Prof. Jaime Evaldo Fensterseifer

Diretor do Instituto de Informática: Prof. Philippe Olivier Alexandre Navaux

Coordenador do PPGC: Prof. Carlos Alberto Heuser

Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

Agradecimentos

Gostaria de agradecer....

Ao Programa de Pós Graduação em Computação do Instituto de Informática da UFRGS, por todos os recursos disponibilizados.

A Universidade Estadual do Oeste do Paraná, cuja liberação foi fundamental para minha estada em Porto Alegre, o que possibilitou além da convivência na UFRGS, a dedicação exclusiva à pesquisa.

A CAPES, pelo fomento.

Aos professores do PPGC que compartilharam seu conhecimento.

Aos colegas de disciplinas, de trabalhos, de risadas e preocupações.

A todos os funcionários do Instituto de Informática da UFRGS, especialmente à Ida Rossi por sua dedicação em “garimpar” referências bibliográficas; ao Sr. Astrogildo, que em sua “vigilância” no prédio, dividiu sua companhia nos muitos sábados e domingos.

Ao professor Tiarajú Asmuz Diverio, por aceitar-me como orientando e compartilhar esses quatro anos de muito trabalho.

Aos amigos e colegas do Instituto de Informática dentre eles, Delcino Picinin Jr., André Luis Martinotto, Ana Paula Canal, Marcos Cassal, Marilton Aguiar, César Zeferino, Leandro Wives, Carlos Augusto Santos, Rafael Boher.

Ao amigo, companheiro de trabalho e brilhante profissional, Ricardo Vargas Dorneles, mais conhecido como “Cadinho”. Este trabalho só foi possível devido as suas inúmeras contribuições, seja discutindo “estênceis e sobreposições”, seja especificando e implementando as estruturas de dados e discutindo as muitas estratégias de solução paralela.

Aos professores Jean Leendertse, Vincenzo Casulli e Edward Gross, pela atenção, pelos conselhos e pela gentileza de enviar o material técnico, frutos de suas experiências e dedicação a área. Meu agradecimento e admiração.

As muitas contribuições, “bate-papo”, e materiais cedidos, pelos professores Alejandro Cassalas (IPH-UFRGS), Suzana Vinzon (COPPE-UFRJ), Alvaro Luiz De Bortoli e Rudnei Cunha (IM-UFRGS), Renato Simões Silva (LNCC), Dan Marchesin (IMPA), Phillippe A. O. Navaux (II-UFRGS) e Carlos de Moura (IME-UERJ).

Aos meus amigos, e colegas de trabalho, de Cascavel, Célia Kavanagh, Amarildo de Vicente, João Cândido Bracarense, Leila Franzini, Onildes Taschetto, Janice Parizotto, Cinara Ita, e tantos outros que sempre perguntavam “como andavam as coisas pelos pampas”, um abraço!

A minha família, e a minha querida esposa Claudia pelo amor, compreensão e paciência. Muito obrigado pelo seu apoio.

Sumário

Lista de Abreviaturas	07
Lista de Figuras	10
Lista de Tabelas	12
Lista de Fórmulas	14
Resumo	16
Abstract	17
1 Introdução	18
1.1 Objetivos e justificativas.....	20
1.2 Apresentação.....	24
2 Problemas de valor inicial e de fronteira	26
2.1 Introdução.....	26
2.2 Classificação e aproximações para os modelos.....	28
2.3 Modelo hidrodinâmico: ESW 3D.....	29
2.3.1 Condições de contorno e condições iniciais.....	30
2.4 Modelo hidrodinâmico: ESW 2D-IV.....	32
2.4.1 Condições de contorno e condições iniciais.....	32
2.5 Modelo de transporte escalar: ETM 3D.....	34
2.5.1 Condições de contorno e condições iniciais.....	36
2.6 Modelo de transporte escalar: ETM 2D-IV.....	36
2.7 Parametrização da turbulência.....	38
2.7.1 Classificação de modelos de turbulência.....	39
2.7.2 Modelo de turbulência de zero equações.....	41
2.7.3 Condições de contorno.....	42
2.8 Estudo de Caso: Lago Guaíba.....	42
3 Sistemas de coordenadas, valores nodais e malha	46
3.1 Malha horizontal: espaço-alternada.....	46
3.2 Malha vertical.....	48
3.2.1 Malha vertical e representação da topografia.....	51
3.3 Valores nodais e variáveis dependentes.....	54
3.4 Condições de contorno discretas.....	55
3.5 Domínios computacionais.....	56
3.5.1 Configurações geométricas e enumeração.....	56
3.5.2 Armazenamento: estruturas de dados.....	59
3.5.3 Regiões não retangulares: método ADI.....	62
3.6 Métodos adaptativos para a malha numérica.....	67
3.6.1 Refinamento local para o transporte: método-h.....	68
4 Modelos na forma discreta	72
4.1 ESW 3D na forma discreta.....	72
4.2 Condições de contorno discretas: ESW 3D.....	80
4.3 ESW 2D-IV na forma discreta.....	80
4.4 ETM 3D na forma discreta.....	82
4.5 Condições de contorno discretas: ETM 3D.....	84

4.6	ETM 2D-IV na forma discreta.....	85
4.7	Modelo de turbulência na forma discreta.....	86
5	Métodos numéricos.....	88
5.1	Método com limitação de fluxo: formulação de Sweby.....	89
5.1.1	Funções interpolantes.....	91
5.1.2	Funções interpolantes para velocidades arbitrárias.....	92
5.1.3	Esquemas de limitação de fluxo: caso multidimensional.....	93
5.1.4	Funções limitadoras de fluxo.....	95
5.2	Método beta.....	102
6	Esquemas numéricos.....	106
6.1	Esquemas semi-implícito: ESW 3D.....	106
6.1.1	Solução explícita para termos das ESW.....	114
6.1.2	Condições de contorno: supressão de termos.....	116
6.2	Esquema semi-implícito fluxo numérico: ETM 2D-IV.....	117
6.3	Esquemas semi-implícitos: método ADI.....	119
6.4	Esquema numérico para a advecção horizontal: método FLM.....	122
6.5	Esquema numérico semi-implícito: método beta.....	125
6.5.1	Condições de contorno: extrapolação.....	128
7	Solução Paralela: ambiente computacional e métodos.....	138
7.1	Plataforma computacional e ambiente de programação.....	138
7.2	Decomposição de dados.....	140
7.3	Particionamento de malha e balanceamento de carga.....	140
7.3.1	Heurísticas para o particionamento.....	141
7.3.2	Métodos e algoritmos de mapeamento.....	144
7.3.3	Heurísticas e métodos de balanceamento de carga.....	145
7.4	Métodos de decomposição de domínio.....	146
7.4.1	Métodos de Schwarz.....	147
7.4.2	Método aditivo de Schwarz: solução paralela.....	152
7.5	Métodos de solução para os sistemas de equações lineares.....	155
7.5.1	Métodos iterativos do subespaço de Krylov.....	156
7.6	Pré-condicionamento.....	157
7.6.1	Pré-condicionador polinomial.....	158
7.6.2	Pré-condicionador MDD aditivo de Schwarz.....	159
7.7	Algoritmo do gradiente conjugado.....	162
7.7.1	GC paralelo: particionamento de dados.....	162
7.8	Método Krylov-Schwarz.....	164
7.9	Algoritmo do resíduo mínimo generalizado.....	171
7.10	Método Direto: solução local e <i>pipeline</i>	173
7.10.1	Algoritmo de Thomas <i>pipeline</i>	174
8	Conclusões, contribuições e trabalhos futuros.....	180
8.1	Conclusões.....	180
8.2	Contribuições.....	182
8.3	Trabalhos futuros.....	185
Anexo 1	Propriedades numéricas e computacionais.....	186
	Monotonicidade.....	188
	Consistência e estabilidade.....	188

Difusão numérica menor que a difusão física.....	191
Acurácia espacial e temporal de alta ordem.....	193
Leis de conservação escalar.....	194
Soluções fortes e soluções fracas.....	195
Condição de entropia e unicidade da solução.....	195
Consistência e variação limitada.....	196
Soluções escalares TVD: acurácia de alta ordem.....	197
Consistência entre a continuidade e o transporte.....	198
Positividade da solução.....	200
Transporte da solução.....	201
Conservação da massa.....	202
Esquema semi-implícito.....	202
Esquemas Euleriano e semi-Lagrangeano.....	205
Esquemas de integração temporal.....	208
Componente horizontal: integração em subciclos.....	208
Integração temporal: interpolação.....	209
Integração temporal: método ADI.....	210
Anexo 2 Métricas para avaliar a qualidade numérica.....	212
Anexo 3 Métricas para avaliar o desempenho da solução paralela.....	215
Anexo 4 Funções de restrição de fluxo.....	216
Bibliografia.....	235

Lista de Abreviaturas

1D	Unidimensional
2D	Bidimensional
2D-IV	Bidimensional integrada na vertical
3D	Tridimensional
ADI	<i>Alternating Direction Implicit</i>
ARF	<i>Area preserving Flux-Form Advection Algorithm</i>
ANL	<i>Argonne National Laboratory</i>
ASD	<i>Accurate Space Derivative</i>
BBL	<i>Bottom Boundary Layer</i>
BFS	<i>Breadth First Search</i>
BCG	<i>Biconjugate Gradient</i>
CC	Condição de Contorno
CFL	Courant-Friedrichs-Levy
CI	Condição Inicial
CCP	Computação Científica Paralela
CGS	<i>Conjugate Gradient Squared</i>
CHARM	<i>Cubic-Parabolic High Accuracy Resolution Method</i>
COPPE	Coordenação do Programa de Pós-Graduação em Engenharia
CMAQ	<i>Community Multiscale Air Quality</i>
CPU	Unidade de Processamento e Controle
CSR	<i>Compressed Sparse Row</i>
CTC	Consistência do Transporte com a Continuidade
CUS	<i>Cubic Interpolation Scheme</i>
DFC	Diferença Finita Centrais (ou Centrada)
CFD	Dinâmica de Fluidos Computacional
DPL	<i>Diffusive Piecewise Linear</i>
EC	Equação da Continuidade
EC-IV	Equação da Continuidade Integrada na Vertical
ECQM	Equação da Conservação da Quantidade do Movimento
ECQM _x	Equação da Conservação da Quantidade do Movimento na direção X
ECQM _y	Equação da Conservação da Quantidade do Movimento na direção Y
EE	Equação da Energia
EI	Equação Integral
EDOs	Equações Diferenciais Ordinárias
EDPs	Equações Diferenciais Parciais
ENO	<i>Essentially Non Oscillatory</i>
ESL	Esquema Semi-Lagrangeano
ESWs	Equações <i>Shallow Water</i>
ETM	Equação do Transporte de substâncias
FFT	<i>Fourier Fast Transforms</i>
FLM	<i>Flux-Limiter Methods</i>
FLM-T	<i>Flux-Limiter Methods Trapezoidal</i>
FCT	<i>Flux Corrected Transport</i>
FLOPS	<i>Floating Point Operations per Second</i>
F-M	<i>Fiduccia-Mattheyses</i>
FOM	<i>Full Orthogonalization Method</i>
GC	Gradiente Conjugado
GCR	<i>Generalized Conjugate Residual</i>
GMCPAD	Grupo de Mat. da Computação e Processamento de Alto Desempenho

GMRES	Método Resíduo mínimo Generalizado
GPPD	Grupo de Processamento Paralelo e Distribuído
HCUS	<i>Harmonic Cubic Interpolation Scheme</i>
HQUICK	<i>Harmonic Quadratic Upwind Scheme</i>
HSFC	<i>Hilbert Space-Filling Curve</i>
IC	Fatoração de Cholesky Incompleta
I/O	Input/Output
ILU	Fatoração LU Incompleta
IMPA	Instituto de Matemática Pura e Aplicada
INPE	Instituto Nacional de Pesquisas Espaciais
ISNAS	<i>Interpolation Scheme which is Nonoscillatory for Advected Scalars</i>
K-L	<i>Kernighan-Lin</i>
LNCC	Laboratório Nacional de Computação Científica
LUS	<i>Linear Upwind Scheme</i>
MAS	Método Alternante de Schwarz
MAAS	Método Aditivo de Schwarz
MARF	<i>Area Monotone Preserving Flux-Form Advection Algorithm</i>
MB	Megabyte
MC	<i>Monotonized Central</i>
MDD	Método de Decomposição de Domínio
MDF	Método das Diferenças Finitas
MICOM	<i>Miami Isopycnic Coordinate Ocean Model</i>
MEF	Método dos Elementos Finitos
MMS	Método Multiplicativo de Schwarz
MPDATA	<i>Multidimensional Positive-Definite Advective Transport Algorithm</i>
MPE	<i>MultiProcessing Environment</i>
MPL	<i>Modified Piecewise Linear</i>
MPI	<i>Message Passing Interface</i>
MPP	<i>Massively Parallel Processors</i>
MSS	Método de Schwarz com Sobreposição
MUSCL	<i>Monotonic Upstream Scheme for Conservation Laws</i>
MVF	Método dos Volumes Finitos
M-Y	Mellor-Yamada
NCAR	<i>National Center for Atmospheric Research</i>
NKS	Newton-Schwarz-Krylov
ORB	<i>Orthogonal Recursive Bisection</i>
PCs	Computadores Pessoais
PL	<i>Piecewise Linear</i>
PPGC	Programa de Pós-Graduação em Computação
PPM	<i>Piecewise Parabolic Methods</i>
PVC	Problema de Valor de Contorno
PVI	Problema de Valor Inicial
PVIC	Problema de Valor Inicial e de Contorno
PVT	Princípio da Viscosidade Turbulenta
Q3D	Quase Tridimensional
QETE	<i>Quasi-Equilibrium Turbulent Energy</i>
QUICK	<i>Quadratic Upwind Scheme</i>
RCB	<i>Recursive Coordinate Bisection</i>
RMQ	Raiz Média Quadrada
RSB	<i>Recursive Spectral Bisection</i>
SC	Superfície de Controle
SORB	<i>Straigh Orthogonal Recursive Bisection</i>
SDP	Simétrica e Definida Positiva
SMART	<i>Sharp and Monotonic Algorithm for Realistic Transport</i>

SO	Sistema Operacional
SOR	<i>Sobre-Relaxação Sucessiva</i>
SPL	<i>Symmetric Piecewise Linear</i>
SPMD	<i>Single Program Multiple Data</i>
SRCB	<i>Straigh Recursive Coordinate Bisection</i>
STRIP	<i>Stripwise Partitioning</i>
SSTRIP	<i>Straigh Stripwise Partitioning</i>
STRIP	<i>Stripwise Partitioning</i>
TV	<i>Total Variation</i>
TVD	<i>Total Variation Diminishing</i>
UFRGS	Universidade Federal do Rio Grande do Sul
UMIST	<i>Upstream Monotonic Interpolation for Scalar Transport</i>
USPL	<i>Upstream Symmetric Piecewise Linear</i>
USR	<i>Upstream Symmetric Limiter</i>
UWMCB	<i>Upstream Weighting Multiple Cell Balance</i>
VC	Volume de Controle

Lista de Figuras

FIGURA 2.1 -	Perfil das velocidades para as ESW-IV.....	32
FIGURA 2.2 -	Imagem real do Lago Guaíba, obtida via sonar.....	45
FIGURA 3.1 -	Malha espaço-alternada Arakawa tipo C.....	47
FIGURA 3.2 -	Batimetria do Lago Guaíba representada por 100 isolinhas.....	47
FIGURA 3.3 -	Regimes de escoamento para uma dada coordenada vertical.....	49
FIGURA 3.4 -	Domínio físico e domínio em coordenadas Cartesianas.....	52
FIGURA 3.5 -	Célula 3D no sistema de coordenadas (x,y,z)	53
FIGURA 3.6 -	Localização das variáveis dependentes no plano $X-Z$	53
FIGURA 3.7 -	Domínio retangular de dimensão 4×3	57
FIGURA 3.8 -	Representação matricial para o domínio discreto.....	57
FIGURA 3.9 -	Domínio não retangular cujas fronteiras são irregulares.....	58
FIGURA 3.10 -	Domínio discretizado com estêncil de 5-pontos.....	59
FIGURA 3.11 -	Exemplo para formato CSR.....	60
FIGURA 3.12 -	Subdomínio onde se destacam as células e suas enumerações.....	61
FIGURA 3.13 -	Uma configuração do Lago Guaíba para o método ADI.....	63
FIGURA 3.14 -	Matriz 3-diagonal para uma linha genérica com CC $U-U$	63
FIGURA 3.15 -	Matriz 3-diagonal para uma coluna genérica com CC $\eta-\eta$	65
FIGURA 3.16 -	Algoritmo para interpolação linear dos níveis.....	70
FIGURA 3.17 -	Malha do transporte encaixada na malha da hidrodinâmica.....	71
FIGURA 3.18 -	Substâncias saindo na fronteira <i>outflow</i> devido a CC gradiente.....	71
FIGURA 6.1 -	Curva para a solução analítica para os níveis de água.....	110
FIGURA 6.2 -	Curva para a solução numérica para os níveis de água.....	110
FIGURA 6.3 -	Curva de níveis: solução analítica para os níveis da água.....	112
FIGURA 6.4 -	Curva de níveis: solução numérica para os níveis da água.....	112
FIGURA 6.5 -	Curvas para solução numérica para velocidades verticais.....	113
FIGURA 6.6 -	Curvas para solução numérica para velocidades verticais.....	113
FIGURA 6.7 -	Curvas para a solução numérica: velocidades horizontais.....	113
FIGURA 6.8 -	Estêncil computacional para o transporte horizontal.....	128
FIGURA 6.9 -	Coluna vertical representando as CC tipo não escoamento.....	129
FIGURA 7.1 -	Particionamento, em 23 processos, usando o algoritmo RCB.....	143
FIGURA 7.2 -	Particionamento, em 16 processos, usando a biblioteca METIS.....	143
FIGURA 7.3 -	Subdomínios gerados pelo método alternante de Schwarz.....	148
FIGURA 7.4 -	Algoritmo alternante de Schwarz.....	148
FIGURA 7.5 -	Algoritmo do MDD aditivo de Schwarz.....	152
FIGURA 7.6 -	Algoritmo aditivo de Schwarz (em paralelo).....	153
FIGURA 7.7 -	Subdomínios com mostrando CC tipo Dirichlet homogêneas.....	154
FIGURA 7.8 -	<i>Speedup</i> \times número de processos para 23 processos.....	154
FIGURA 7.9 -	Eficiência \times número de processos para 23 processos.....	155
FIGURA 7.10 -	Algoritmo para cálculo do fator incompleto, IC(0).....	160
FIGURA 7.11 -	Algoritmo para o cálculo do fator incompleto diagonal, DIC(0).....	161
FIGURA 7.12 -	Algoritmo do GC pré-condicionado paralelizado.....	164
FIGURA 7.13 -	Tempo de execução com e sem pré-condicionamento (Guaiba200).....	166
FIGURA 7.14 -	Tempo de execução com e sem pré-condicionamento (Guaiba100).....	167
FIGURA 7.15 -	<i>Speedup</i> \times número de processos para o GC (Guaiba200).....	167
FIGURA 7.16 -	<i>Speedup</i> \times número de processos para o GC (Guaiba100).....	167
FIGURA 7.17 -	Eficiência \times número de processos para o GC (Guaiba200).....	168
FIGURA 7.18 -	Eficiência \times número de processos para o GC (Guaiba100).....	168
FIGURA 7.19 -	Desempenho computacional do MAAS \times KS (Guaiba200).....	169
FIGURA 7.20 -	Desempenho computacional do MAAS \times KS (Guaiba100).....	169

FIGURA 7.21 -	Acurácia: solução monoprocessada × multiprocessada (<i>float</i>).....	170
FIGURA 7.22 -	Acurácia: solução monoprocessada × multiprocessada (<i>float</i>).....	170
FIGURA 7.23 -	Acurácia: solução monoprocessada × multiprocessada (<i>double</i>).....	171
FIGURA 7.24 -	Acurácia: solução monoprocessada × multiprocessada (<i>double</i>).....	171
FIGURA 7.25 -	Algoritmo GMRES(<i>m</i>).....	173
FIGURA 7.26 -	Passos <i>forward</i> e <i>backward</i> do algoritmo de Thomas.....	173
FIGURA 7.27 -	Algoritmo para inversão da matriz triangular inferior.....	174
FIGURA 7.28 -	Algoritmo para inversão da matriz 3-diagonal SDP.....	174
FIGURA 7.29 -	Decomposição de domínio: um sistema para cada linha.....	175
FIGURA 7.30 -	Particionamento de dados para um subdomínio regular.....	175
FIGURA 7.31 -	Escalonamento para o algoritmo de Thomas.....	176
FIGURA 7.32 -	Escalonamento para o algoritmo de Thomas <i>pipeline</i>	176
FIGURA 7.33 -	Particionamento em faixas na direção <i>Y</i>	177
FIGURA 7.34 -	Campo de velocidade para o particionamento em faixas.....	178
FIGURA 7.35 -	<i>Speedup</i> obtido, para um dado refinamento: método ADI.....	178
FIGURA 7.36 -	Eficiência obtida, dado refinamento, para o método ADI.....	179
FIGURA A1.1 -	Exemplo para oscilação numérica e difusão numérica.....	187
FIGURA A1.2 -	Posicionamento dos fluxos numéricos nas faces da célula 3D.....	203

Lista de Tabelas

TABELA 3.1 -	Exemplo de sublinhas de cálculo para método ADI: direção X.....	65
TABELA 3.2 -	Exemplo de subcolunas de cálculo para método ADI: direção Y....	66
TABELA 5.1 -	Teste com a onda quadrada e o esquema UMIST.....	100
TABELA 5.2 -	Experimento (UMIST) com $C_r=0,25$: análise da precisão.....	101
TABELA 5.3 -	Experimento (UMIST) com $C_r=0,25$: influência do perfil.....	102
TABELA 5.4	Resultados para o teste para o esquema beta.....	105
TABELA 7.1 -	Número de iterações do GC com e sem pré-condicionamento.....	166
TABELA A1.1 -	Resultados para condição CTC: dados iniciais constantes.....	200
TABELA A1.2 -	Resultados para condição CTC: dados iniciais não constantes.....	200
TABELA A4.1 -	Resultados para o teste com onda quadrada para $C_r=0,25$ (FLM)..	216
TABELA A4.2 -	Resultados para o teste com onda quadrada para $C_r=0,25$ (FLM)..	216
TABELA A4.3 -	Resultados para o teste com onda quadrada para $C_r=0,25$ (FLM)..	217
TABELA A4.4 -	Resultados para o teste com onda quadrada para $C_r=0,25$ (FLM)..	217
TABELA A4.5 -	Resultados para o teste com onda quadrada para $C_r=0,25$ (FLM)..	217
TABELA A4.6 -	Resultados para o teste com onda quadrada para $C_r=0,25$ (FLM)..	218
TABELA A4.7 -	Resultados para o teste com onda quadrada para $C_r=0,25$ (FLM)..	218
TABELA A4.8 -	Resultados para o teste com onda quadrada para $C_r=0,25$ (FLM)..	218
TABELA A4.9 -	Resultados para o teste com onda quadrada para $C_r=0,40$ (FLM)..	220
TABELA A4.10-	Resultados para o teste com onda quadrada para $C_r=0,40$ (FLM)..	220
TABELA A4.11-	Resultados para o teste com onda quadrada para $C_r=0,40$ (FLM)..	220
TABELA A4.12-	Resultados para o teste com onda quadrada para $C_r=0,40$ (FLM)..	221
TABELA A4.13-	Resultados para o teste com onda quadrada para $C_r=0,40$ (FLM)..	221
TABELA A4.14-	Resultados para o teste com onda quadrada para $C_r=0,40$ (FLM)..	221
TABELA A4.15-	Resultados para o teste com onda quadrada para $C_r=0,33$ (FLM)..	222
TABELA A4.16-	Análise da precisão dos resultados do esquema <i>upwind</i>	223
TABELA A4.17-	Análise da precisão dos resultados do esquema <i>upwind</i>	223
TABELA A4.18-	Análise da precisão dos resultados do esquema Roe-Superbee.....	224
TABELA A4.19-	Análise da precisão dos resultados do esquema Roe-Superbee.....	224
TABELA A4.20-	Análise da precisão dos resultados do esquema UMIST.....	224
TABELA A4.21-	Análise da precisão dos resultados do esquema UMIST.....	225
TABELA A4.22-	Análise dos resultados para erros relativos no esquema <i>upwind</i> ...	225
TABELA A4.23-	Análise dos resultados para erros relativos no esquema <i>upwind</i> ...	226
TABELA A4.24-	Análise dos resultados para erros relativos no esquema Roe.....	226
TABELA A4.25-	Análise dos resultados para erros relativos no esquema Roe.....	226
TABELA A4.26-	Análise dos resultados para erros relativos no esquema UMIST....	227
TABELA A4.27-	Análise dos resultados para erros relativos no esquema UMIST....	227
TABELA A4.28-	Análise da influência do perfil para o esquema Roe-Superbee.....	228
TABELA A4.29-	Análise da influência do perfil para o esquema Roe-Superbee.....	228
TABELA A4.30-	Análise da influência do perfil para o esquema Roe-Superbee.....	228
TABELA A4.31-	Análise da influência do perfil para o esquema <i>upwind</i>	229
TABELA A4.32-	Análise da influência do perfil para o esquema <i>upwind</i>	229
TABELA A4.33-	Análise da influência do perfil para o esquema <i>upwind</i>	229
TABELA A4.34-	Análise da influência do perfil para o esquema UMIST.....	230
TABELA A4.35-	Análise da influência do perfil para o esquema UMIST.....	230
TABELA A4.36-	Análise da influência do perfil para o esquema UMIST.....	230
TABELA A4.37-	Análise da influência do perfil para o esquema UMIST.....	231
TABELA A4.38-	Análise da influência do perfil para o esquema <i>upwind</i>	231
TABELA A4.39-	Análise da influência do perfil para o esquema Roe-Superbee.....	231

TABELA A4.40-	Análise da influência do perfil para o esquema Roe-Superbee.....	232
TABELA A4.41-	Análise da influência do perfil para o esquema <i>upwind</i>	232
TABELA A4.42-	Análise da influência do perfil para o esquema UMIST.....	233
TABELA A4.43-	Análise da influência do perfil para o esquema Roe-Superbee.....	233
TABELA A4.44-	Análise da influência do perfil para o esquema <i>upwind</i>	233
TABELA A4.45-	Análise da influência do perfil para o esquema UMIST.....	234

Lista de Fórmulas

FÓRMULA 2.1 -	ESW 3D: equação quantidade do movimento na direção X.....	29
FÓRMULA 2.2 -	ESW 3D: equação quantidade do movimento na direção Y.....	29
FÓRMULA 2.3 -	ESW 3D: equação da continuidade.....	29
FÓRMULA 2.4 -	ESW 3D: equação da continuidade integrada na vertical.....	29
FÓRMULA 2.5 -	ESW 2D-IV: equação quantidade do movimento na direção X....	31
FÓRMULA 2.6 -	ESW 2D-IV: equação quantidade do movimento na direção Y....	31
FÓRMULA 2.7 -	ESW 2D-IV: equação da continuidade.....	31
FÓRMULA 2.8 -	ETM 3D.....	34
FÓRMULA 2.9 -	Taxa de decaimento.....	34
FÓRMULA 2.10 -	ETM 2D-IV.....	37
FÓRMULA 4.1 -	Componente u da velocidade horizontal na forma discreta.....	73
FÓRMULA 4.2 -	Componente v da velocidade horizontal na forma discreta.....	74
FÓRMULA 4.3 -	Nível de água na forma discreta.....	75
FÓRMULA 4.4 -	Velocidade vertical w na forma discreta.....	75
FÓRMULA 4.5 -	Formulação matricial para a componente u da velocidade.....	76
FÓRMULA 4.6 -	Formulação matricial para a componente v da velocidade.....	76
FÓRMULA 4.7 -	Formulação matricial para os níveis de água.....	76
FÓRMULA 4.8 -	Formulação matricial para o vetor dos operadores em X.....	77
FÓRMULA 4.9 -	Formulação matricial para o vetor dos operadores em Y.....	78
FÓRMULA 4.10 -	Formulação para os coeficientes de viscosidade em X.....	78
FÓRMULA 4.11 -	Formulação para os coeficientes de viscosidade em Y.....	78
FÓRMULA 4.12 -	Formulação matricial para os níveis de água.....	78
FÓRMULA 4.13 -	ETM 3D na forma discreta.....	81
FÓRMULA 4.14 -	ETM 3D na forma conservativa fluxo.....	82
FÓRMULA 5.1 -	Função interpolante de van Leer.....	89
FÓRMULA 5.2 -	Função limitadora de fluxo.....	90
FÓRMULA 5.3 -	Condição suficiente para a função de restrição de fluxo.....	90
FÓRMULA 5.4 -	Condição para restrição em função do número de Péclet.....	90
FÓRMULA 5.5 -	Função limitadora de fluxo para velocidades arbitrárias.....	91
FÓRMULA 5.6 -	Funções monitoras de fluxo.....	91
FÓRMULA 5.7 -	Função fluxo numérico para face superior.....	102
FÓRMULA 5.8 -	Função fluxo numérico para face superior: esquema beta.....	102
FÓRMULA 5.9 -	Função fluxo numérico para face inferior: esquema beta.....	102
FÓRMULA 5.10 -	Função fluxo numérico para advecção vertical: esquema beta....	102
FÓRMULA 6.1 -	Sistema de equações lineares para os níveis de água.....	104
FÓRMULA 6.2 -	ESW 1D.....	107
FÓRMULA 6.3 -	Operador explícito para a componente u da velocidade.....	107
FÓRMULA 6.4 -	Operador explícito para a componente v da velocidade.....	113
FÓRMULA 6.5 -	Sistema de equações para a concentração ETM 2D-IV -.....	113
FÓRMULA 6.6 -	Sistema de equações 3-diagonal: primeiro passo de tempo.....	117
FÓRMULA 6.7 -	Sistema de equações 3-diagonal: segundo passo de tempo.....	118
FÓRMULA 6.8 -	Fluxos numéricos para as velocidades horizontais: FLM.....	122
FÓRMULA 6.9 -	Sistema de equações lineares para ETM 3D.....	124
FÓRMULA 7.1 -	MDD multiplicativo de Schwarz.....	148
FÓRMULA 7.2 -	Método aditivo de Schwarz para N subdomínios.....	149
FÓRMULA A1.1 -	Condição de estabilidade para esquema numérico das ESW.....	183
FÓRMULA A1.2 -	Condição de estabilidade para esquema beta da ETM 3D.....	184
FÓRMULA A1.3 -	Coefficiente de difusão numérica.....	185

FÓRMULA A1.4 - Coeficiente de difusão numérica para o esquema beta.....	186
FÓRMULA A1.5 - Problema de valor inicial: lei de conservação escalar.....	188
FÓRMULA A1.6 - Equação integral para solução fraca.....	189
FÓRMULA A1.7 - Equação da continuidade.....	192
FÓRMULA A1.8 - Equação do transporte escalar.....	193

Resumo

Neste trabalho desenvolveu-se e implementou-se um *modelo computacional paralelo multifísica* para a simulação do transporte de substâncias e do escoamento hidrodinâmico, bidimensional (2D) e tridimensional (3D), em corpos de água. Sua motivação está centrada no fato de que as margens e zonas costeiras de rios, lagos, estuários, mares e oceanos são locais de aglomerações de seres humanos, dada a sua importância para as atividades econômica, de transporte e de lazer, causando desequilíbrios a esses ecossistemas. Esse fato impulsiona o desenvolvimento de pesquisas relativas a esta temática.

Portanto, o objetivo deste trabalho é o de construir um modelo computacional com alta qualidade numérica, que possibilite simular os comportamentos da hidrodinâmica e do transporte escalar de substâncias em corpos de água com complexa configuração geométrica, visando a contribuir para seu manejo racional.

Visto que a ênfase nessa tese são os aspectos numéricos e computacionais dos algoritmos, analisaram-se as características e propriedades numérico-computacionais que as soluções devem contemplar, tais como a estabilidade, a monotonicidade, a positividade e a conservação da massa. As estratégias de soluções enfocam os termos advectivos e difusivos, horizontais e verticais, da equação do transporte. Desse modo, a advecção horizontal é resolvida empregando o método da limitação dos fluxos de Sweby, e o transporte vertical (advecção e difusão) é resolvido com os métodos beta de Gross e de Crank-Nicolson. São empregadas malhas com distintas resoluções para a solução do problema multifísica. O esquema numérico resultante é semi-implícito, computacionalmente eficiente, estável e fornece acurácia espacial e temporal de segunda ordem.

Os sistemas de equações resultantes da discretização, em diferenças finitas, das equações do escoamento e do transporte 3D, são de grande porte, lineares, esparsos e simétricos definidos-positivos (SDP). No caso 2D os sistemas são lineares, mas os sistemas de equações para a equação do transporte não são simétricos. Assim, para a solução de sistemas de equações SDP e dos sistemas não simétricos empregam-se, respectivamente, os métodos do subespaço de Krylov do gradiente conjugado e do resíduo mínimo generalizado. No caso da solução dos sistemas 3-diagonal, utiliza-se o algoritmo de Thomas e o algoritmo de Cholesky.

A solução paralela foi obtida sob duas abordagens. A *decomposição ou particionamento de dados*, onde as operações e os dados são distribuídos entre os processos disponíveis e são resolvidos em paralelo. E, a *decomposição de domínio*, onde obtém-se a solução do problema global combinando as soluções de subproblemas locais. Em particular, emprega-se neste trabalho, o método de decomposição de domínio aditivo de Schwarz, como método de solução, e como pré-condicionador.

Para maximizar a relação computação/comunicação, visto que a eficiência computacional da solução paralela depende diretamente do balanceamento de carga e da minimização da comunicação entre os processos, empregou-se algoritmos de particionamento de grafos para obter localmente os subproblemas, ou as partes dos dados.

O modelo computacional paralelo resultante mostrou-se computacionalmente eficiente e com alta qualidade numérica

Palavras-chave: equações *shallow water* 3D, equação de transporte escalar 3D, esquemas numéricos semi-implícito, refinamento local, método de Sweby, método de decomposição de Schwarz.

TITLE: “PARALLEL COMPUTATIONAL MODEL FOR HYDRODYNAMICS AND FOR THE SCALAR TWO-DIMENSIONAL AND THREE-DIMENSIONAL TRANSPORT OF SUBSTANCES”

Abstract

A multi-physics parallel computational model was developed and implemented for the simulation of substance transport and for the two-dimensional (2D) and three-dimensional (3D) hydrodynamic flow in water bodies. The motivation for this work is focused in the fact that the margins and coastal zones of rivers, lakes, estuaries, seas and oceans are places of human agglomeration, because of their importance for economic, transport, and leisure activities causing ecosystem disequilibrium. This fact stimulates the researches related to this topic.

Therefore, the goal of this work is to build a computational model of high numerical quality, that allows the simulation of hydrodynamics and of scalar transport of substances behavior in water bodies of complex configuration, aiming at their rational management.

Since the focuses of this thesis are the numerical and computational aspects of the algorithms, the main numerical-computational characteristics and properties that the solutions need to fulfill were analyzed. That is: stability, monotonicity, positivity and mass conservation. Solution strategies focus on advective and diffusive terms, horizontal and vertical terms of the transport equation. In this way, horizontal advection is solved using Sweby’s flow limiting method; and the vertical transport (advection and diffusion) is solved with Gross and Crank-Nicolson’s beta methods. Meshes of different resolutions are employed in the solution of the multi-physics problem. The resulting numerical scheme is semi-implicit, computationally efficient, stable and provides second order accuracy in space and in time.

The equation systems resulting of the discretization, in finite differences, of the flow and 3D transport are of large scale, linear, sparse and symmetric positive definite (SPD). In the 2D case, the systems are linear, but the equation systems for the transport equation are not symmetric. Therefore, for the solution of SPD equation systems and of the non-symmetric systems we employ, respectively, the methods of Krylov’s sub-space of the conjugate gradient and of the generalized minimum residue. In the case of the solution of 3-diagonal systems, Thomas algorithm and Cholesky algorithm are used.

The parallel solution was obtained through two approaches. In data decomposition or partitioning, operation and data are distributed among the processes available and are solved in parallel. In domain decomposition the solution of the global problem is obtained combining the solutions of the local sub-problems. In particular, in this work, Schwarz additive domain decomposition method is used as solution method and as preconditioner.

In order to maximize the computation/communication relation, since the computational efficiency of the parallel solution depends directly of the load balancing and of the minimization of the communication between processes, graph-partitioning algorithms were used to obtain the sub-problems or part of the data locally.

The resulting parallel computational model is computationally efficient and of high numerical quality.

Key words: shallow water 3D equations, 3D equation of scalar transport, semi-implicit numerical schemes, local refinement, Sweby method, Schwarz domain decomposition method.

1 Introdução

O Grupo de Matemática da Computação e Processamento de Alto Desempenho (GMCPAD) e o Grupo de Processamento Paralelo e Distribuído (GPPD) do Programa de Pós-Graduação em Computação da Universidade Federal do Rio Grande do Sul (PPGC da UFRGS) vem trabalhando em aplicações de alto desempenho, e dentre elas está contemplado o desenvolvimento de *modelos computacionais paralelos para a simulação do escoamento e do transporte de substâncias em corpos de água*. O objetivo desses modelos, enquanto aplicação, é simular complexos processos hidrodinâmicos e de transporte de substâncias que ocorrem em corpos de água, como aqueles de lagos, regiões costeiras e estuários.

Essas questões são os principais elementos motivadores desta tese, porém, enquanto pesquisa aplicada em Ciência da Computação, mais especificamente em Computação Científica Paralela, o objetivo deste trabalho é desenvolver *esquemas numéricos e estratégias de solução paralelas* apropriados, sob o ponto de vista numérico e computacional, à construção de modelos computacional paralelos multifísica, nos quais são simulados diferentes fenômenos físicos no mesmo domínio de definição. Esse é caso da hidrodinâmica 3D e do transporte escalar de substâncias 3D, como tratado neste trabalho.

Para esta tese, foi desenvolvido um modelo paralelo para a hidrodinâmica (modo barotrópico, que é aquele que considera apenas as ondas de gravidade externa) e para o transporte de substâncias (dejetos humanos) em corpos de águas rasos, motivado pelo fato de que a disponibilização desses modelos computacionais com alta resolução e alto desempenho permite a simulação em detalhes da circulação e do transporte de substâncias nesses ambientes. Essas questões são essenciais para a compreensão das atividades ecológicas relacionadas a esses ambientes.

Para contextualizar o *modelo computacional paralelo multifísica* aqui desenvolvido em relação aos modelos já existentes, objetivando-se comparar suas características, fez-se uma ampla revisão bibliográfica, onde se encontrou 36 *modelos*. A análise mostra que as principais características *dinâmicas* e *numéricas* de grande parte desses modelos segue o mesmo padrão. A discretização é via diferenças finitas, usa-se o método ADI e o algoritmo de Thomas para obter a solução – seqüencial - dos sistemas de equações, e usa-se o sistema de coordenadas cartesianas ou curvilíneas para construir a malha horizontal. Além disso, esses *modelos* são desenvolvidos ou para o escoamento ou para o transporte de substâncias.

Exemplos desses modelos, que são semelhantes ao desenvolvido originalmente por Leendertse [LEE70] e [LEE71], e que reúnem essas características, típicas de modelos 2D integrados na vertical, são os de: Cassalas [CSL85], Silveira [SIL86] e Kaplan [KAP98]. Exemplos de modelos 2D que não seguem essas características são os de Paglieri [PAG97] e Ding [DIN98]. Nesses dois casos a advecção é discretizada usando a abordagem semi-Lagrangeano e a solução numérica é obtida em paralelo. Em [PAG97] a solução paralela é obtida via *decomposição de domínio*, e em [DIN98] a solução paralela é obtida via transformada rápida de Fourier (FFT).

Outros modelos, agora no caso 3D, são aqueles cujas principais características são: a *separação dos modos (modo interno e modo externo)*, o uso de malha curvilínea (ou malha não estruturada) na horizontal, o uso do sistema *sigma* na vertical, e cuja solução é obtida seqüencialmente. Exemplos são os modelos de: Sheng [SHE83], Gomes e Blumberg [GOM95], Fortunato [FOR96]. Alguns exemplos de modelos 3D que não usam a *separação dos modos* são os de: Vollebregt [VOL97], Vilar [VLA99], Kim e Lee [KIM94], Bonaventura e Villa [ROS2000], Zhu, Johnson, Bangalope, Huddleston e Skjellum [ZHU98], Zheng e Wang [ZHE99], Li e Zhanh [LIY96], Gross [GRO98], Cassuli [CAS92], e Leendertse [LEE89].

Destacam-se algumas particularidades de alguns desses modelos 3D de modo a permitir uma comparação mais específica com o *modelo computacional paralelo multifísica* desenvolvido nesta tese, particularmente os modelos de [SHE83], [VOL97], [ZHE98], [VLA99] e [ZHE99].

No trabalho de Sheng destaca-se, principalmente, pela discussão sistemática e abrangente das principais características dinâmicas e numéricas dos modelos hidrodinâmico e de transporte de substâncias. Além disso, outro ponto de destaque no trabalho de Sheng é a discussão das EDPs dos modelos matemáticos de modo a obter EDPs adimensionais em coordenadas sigma na vertical e em coordenadas curvilíneas na horizontal. A solução seqüencial é obtida via separação dos modos e a parametrização da turbulência é semi-empírica.

Vollebregt [VOL97] desenvolveu modelos 3D, para a hidrodinâmica e para o transporte de substâncias, e usa o sistema de coordenadas Cartesianas na vertical e coordenadas curvilíneas na horizontal. Um modelo de parametrização da turbulência $\kappa - \varepsilon$ (modelo de *duas* equações) é, também, desenvolvido e implementado. Para obter a solução paralela dos modelos emprega-se o método ADI combinado com estratégias de transposição dos dados, e métodos do subespaço de Krylov. Os algoritmos numéricos seguem a abordagem desenvolvida por Leendertse [LEE89].

O trabalho de Vilar [VLA99] contempla, também, a solução – seqüencial e via método ADI – dos modelos hidrodinâmicos e de transporte de substâncias 3D. A parametrização da turbulência é semi-empírica e semelhante aquela feita por Sheng. Destaca-se o processo de discretização, que é feito empregando o método dos volumes finitos usando uma malha não estruturada na horizontal, e de modo a permitir o uso de um sistema híbrido (como, por exemplo, sigma-Cartesiano-sigma) na vertical.

Já o trabalho de Zheng e Wang [ZHE99] se destaca por ser um modelo abrangente para o transporte multi-constituente de substâncias de modo a simular sua advecção, difusão e reações químicas em escoamento através de meios porosos (*groundwater flow*). A discretização via diferenças finitas envolve esquemas numéricos empregando as abordagens semi-Lagrangiana e TVD. A solução – seqüencial – é obtida empregados-se métodos iterativos (pré-condicionados) do subespaço de Krylov.

Em [ZHU98] desenvolve-se o modelo 3D empregando o método das diferenças finitas para a discretização das EDPs, as malhas horizontal e vertical são em coordenadas Cartesianas, a solução paralela é obtida via *decomposição de dados*. A estratégia de solução e os algoritmos numéricos seguem a abordagem desenvolvida originalmente por Casulli [CAS92].

No caso do *modelo computacional paralelo para a hidrodinâmica e para o transporte de substâncias 2D e 3D* desenvolvido nesta tese, pode-se sintetizar algumas de suas características são: discretização via diferenças finitas (forma conservativa, via fluxo numérico); malha espaço-alternada na horizontal; sistema de coordenadas Cartesianas para a malha vertical; algoritmos para calcular os fluxos são construídos usando os métodos de limitação de fluxo de Sweby ([SWE84] e o Beta de Gross [GRO98]; desenvolvimento de estratégias de solução paralela empregando o método ADI combinado com o algoritmo de Thomas na versão *pipeline* como Povitsky [POV98], a decomposição de dados, e decomposição de domínio e o método Krylov-Schwarz. A solução para o *problema multifísica* foi obtida construindo-se um refinamento local. A parametrização da turbulência é, também, semi-empírica, dado que o modelo de turbulência é do tipo *zero equações*.

Essas características tornam o *modelo computacional* desenvolvido neste trabalho distinto de todos os encontrados na literatura. Nas próximas seções segue-se uma descrição mais detalhada e técnica dos principais atributos numéricos e computacionais deste *modelo computacional multifísica*.

1.1 Objetivos e justificativas

Considerando-se que o objetivo geral é desenvolver um *modelo computacional paralelo multifísica*, os objetivos específicos deste trabalho foram, portanto, o de desenvolver *esquemas numéricos, estratégias de paralelização e de solução* apropriados, sob o ponto de vista numérico e computacional, à construção desses modelos paralelos. No decorrer das atividades de pesquisa, analisaram-se os mais importantes atributos numéricos e computacionais que modelos paralelos devem considerar, visando à qualidade numérica e ao desempenho computacional.

Com essa abordagem construíram-se modelos computacionalmente eficientes, e que oferecem soluções numéricas condizentes com a realidade física. Além disso, com o exame detalhado das propriedades dos modelos fundamentaram-se os critérios e as abordagens para a construção dos esquemas (algoritmos) numéricos. As soluções paralelas também foram desenvolvidas sob essa abordagem. O resultado final é a produção de modelos computacionais eficientes e acurados, apropriados para simulações de eventos de longa duração.

No caso dos esquemas numéricos do modelo hidrodinâmico, eles foram construídos baseados na estratégia desenvolvida por Casulli ([CAS90], [CAS92] e [CAS94]). Essa escolha decorreu do fato de que nesses trabalhos estabeleceu-se uma rigorosa base matemática para a construção dos esquemas numéricos, de modo que o algoritmo numérico resultante não depende de considerações heurísticas, acarretando a eliminação de inconsistências na solução numérica, como aquelas que surgem em modelos que empregam a técnica da divisão do *modo* (*mode splitting*).

A análise de estabilidade das equações diferenciais parciais (EDPs) que modelam a hidrodinâmica mostrou que a *celeridade* (velocidade de propagação da onda) é expressa por alguns termos constituintes das EDPs. Tais termos são os coeficientes dos gradientes de pressão barotrópica nas equações da conservação da quantidade de movimento, e as derivadas espaciais das velocidades na equação da continuidade. Portanto, esses termos devem ser discretizados implicitamente para que a estabilidade do esquema numérico não dependa da celeridade. Os demais termos constituintes das EDPs não têm essa restrição e podem ser discretizados explicitamente, de modo que o esquema numérico resultante é semi-implícito.

Com essa abordagem, o esquema numérico para a hidrodinâmica não depende das condições Courant-Friedrichs-Levy (CFL) [FLE88], e tem um mínimo grau de implicidade, de modo que o algoritmo numérico é estável e tem máxima eficiência computacional com o mínimo de custo computacional.

Note-se, porém, que o algoritmo desenvolvido nos trabalhos iniciais de Casulli ([CAS90] e [CAS92]) é numericamente estável e computacionalmente eficiente, mas ele é de primeira ordem de acurácia no tempo, o que introduz amortecimento artificial na solução numérica. Porém, esse erro de amortecimento pode ser reduzido ao mínimo quando se emprega uma média no tempo como no esquema Crank-Nicolson, como foi mostrado por Patnaik [PAT87]. Então, para obter uma acurácia temporal de segunda ordem, emprega-se o método trapezoidal ou seu caso particular, o método- θ . Tomando $\theta=0,5$ os termos dos níveis da água (gradientes de pressão barotrópica) são aproximados como uma média nos níveis de tempo n e $n+1$. O esquema de integração temporal resultante apresenta acurácia de segunda ordem, e o esquema numérico é estável e independente da velocidade da onda [CAS94].

Especificamente, aproximam-se os gradientes de elevação e a difusão vertical nas equações de conservação da quantidade de movimento, e a equação da continuidade, de modo implícito usando o método- θ . Os termos de difusão horizontal, de advecção (horizontal e vertical) e de Coriolis são aproximados explicitamente. Nessa abordagem geram-se sistemas de equações lineares cujas matrizes são simétricas definidas-

positivas (SDP). Nessa situação a existência e a unicidade da solução numérica são asseguradas.

A solução desses sistemas SDP fornece os níveis de água. As velocidades horizontais são obtidas resolvendo sistemas de equações 3-diagonal, também SDP, de dimensão igual a da quantidade de camadas. As velocidades verticais são calculadas explicitamente. Sob o ponto de vista de solução numérica, essa abordagem requer a solução de sistemas de equações lineares com matrizes *tipo 5-diagonal*, que são aqueles sistemas de equações cujas matrizes dos coeficientes têm no máximo cinco elementos por linha, e requer a solução dos sistemas 3-diagonal. Esses resultados induziram os desenvolvimentos de determinadas estratégias quando da obtenção da solução paralela.

Na solução da equação do transporte escalar de substâncias (ETM), deu-se especial atenção quando da construção dos esquemas numéricos aos dois termos mais críticos dessa EDP, que são as advecções vertical e horizontal, e o termo da difusão vertical. Isso se justifica pelo fato de que o objetivo central de esquemas advectivos verticais é simular apropriadamente a mistura vertical e, portanto, devem-se desenvolver esquemas numéricos com baixa difusão numérica, pois caso contrário, a difusão numérica obscurece a difusão física calculada pelo modelo de fechamento de turbulência. No caso da advecção, sua solução deve contemplar esquemas que não sejam suscetíveis de sofrerem de oscilações numéricas no entorno dos altos gradientes de concentração como aqueles que surgem quando da simulação de eventos como o despejo de dejetos.

Além disso, deve-se considerar nessas abordagens as estratégias para obter a solução paralela do transporte escalar tendo em vista que o modelo computacional desenvolvido neste trabalho é aquele para problemas multifísica. Esses problemas são aqueles nos quais os fenômenos simulados são divididos em diferentes fenômenos físicos no mesmo domínio ou subdomínio de definição, e que exigem diferentes escalas espaço-temporais nos algoritmos numéricos para a obtenção de suas soluções. Esse é a situação da solução para a hidrodinâmica 3D e da solução para o transporte escalar de substâncias 3D.

Nesse caso, devido as soluções já desenvolvidas para a hidrodinâmica, um esquema numérico para o transporte escalar com dependência implícita de dados somente na coluna de água facilita o desenvolvimento de estratégias na obtenção de sua solução paralela. Esses objetivos balizaram a escolha do método de solução para o transporte de substâncias, de modo que o transporte vertical seja tratado implicitamente e o transporte horizontal seja tratado explicitamente.

Para a solução da advecção vertical empregou-se o método $-\beta$ semi-implícito desenvolvido por Gross [GRO98]. A análise de estabilidade desse método mostra que ele é estável para todos os números de Courant. Além disso, tem menos erros de difusão e de dispersão que uma abordagem puramente implícita. Por esses atributos, essa abordagem é particularmente importante em esquemas advectivos verticais para representar a mistura vertical.

Para a solução da difusão vertical empregou-se o método Crank-Nicolson que é semi-implícito e incondicionalmente estável [FLE88]. O resultado é a produção de um esquema numérico semi-implícito para o transporte vertical (advecção e difusão), e que pode ser empregado em regiões com grande variação no número de Courant, visto que esquemas apropriados são aqueles que são acurados para baixos números de Courant e que tenham uma boa acurácia para altos números de Courant, já que esta é uma característica marcante em muitos corpos de água. Essa escolha não impõe uma excessiva limitação na magnitude do passo de tempo, melhorando a eficiência computacional.

O uso dos métodos beta e Crank-Nicolson gera um esquema numérico semi-implícito para o transporte vertical. Assim, uma solução explícita para o transporte horizontal, combinada com o esquema do transporte vertical, pode fornecer um esquema numérico global que é semi-implícito e tem o importante atributo computacional de gerar sistemas de equações que são 3-diagonal.

A questão central é como aproximar o transporte horizontal explicitamente sem degradar completamente a sua acurácia, visto que a solução da advecção horizontal pode sofrer de fortes oscilações numéricas no entorno dos altos gradientes de concentração se o esquema de solução não for apropriadamente construído. Essa questão é essencial e faz-se uma análise sistemática de soluções para esse problema através da especificação de uma série de funções limitadoras de fluxo considerando-se o emprego do esquema de limitação de fluxo de Sweby para a solução explícita da advecção horizontal.

Essa escolha, discutida detalhadamente no anexo 1, decorre do fato de que soluções advectivas devem ser limitadas, acuradas e estáveis e essas características dependem do método empregado. Métodos numéricos lineares somente são efetivos em regiões com baixa advecção. Por exemplo, o esquema *upwind* de primeira ordem produz soluções estáveis e limitadas, mas é pouco acurado dado que é muito difusivo em regiões com alta advecção e/ou altos gradientes. As soluções limitadas, acuradas e estáveis são obtidas empregando-se malhas altamente refinadas e/ou esquemas com alta ordem de acurácia.

Neste trabalho desenvolvem-se essas duas abordagens para obter soluções advectivas de alta qualidade numérica. Malhas refinadas são discutidas no capítulo 3, enquanto os esquemas não lineares conservativos são discutidos no capítulo 5 e empregam o método de limitação dos fluxos (FLM), de modo que as soluções tenham acurácia de no mínimo segunda ordem. Neste trabalho são analisadas e implementadas várias funções de restrição do fluxo numérico, já que essas funções que definem a qualidade dos resultados do método FLM.

Especificamente, a advecção horizontal da ETM é discretizada empregando métodos de restrição de fluxo desenvolvido por Sweby [SWE84]. Dependendo da escolha da função limitadora de fluxo os esquemas numéricos explícitos construídos sob tal abordagem obedecem à característica de serem *Total Variation Diminishing* (TVD), satisfazendo a condição de entropia, que é um critério matemático que permite obter a solução correta entre as *soluções fracas* do problema, fornecendo soluções fisicamente corretas ([OSH84], [SNO98] e [SAT98]).

No tocante à paralelização, adotou-se a abordagem *programação explícita*, na qual há um completo controle sobre as estratégias de implementação e sobre a própria implementação. Essa escolha decorre do fato de que as aplicações paralelas desenvolvidas neste trabalho visam arquiteturas tipo *clusters* de PCs, e inexistem ferramentas apropriadas para extrair o paralelismo implicitamente desse tipo de arquitetura. Além disso, em *clusters* multiprocessados, como o caso do disponível no Instituto de Informática da UFRGS, tem-se memória compartilhada, na qual os dados e as operações são globais, isto é, têm o mesmo endereçamento, e tem-se memória distribuída, na qual a comunicação entre processos é obtida através de trocas de mensagens. Assim, deve-se explorar os dois níveis de paralelismo, o intra e o internodal.

Nesses ambientes de processamento paralelo as soluções podem ser obtidas via decomposição de dados, em que as operações e os dados são distribuídos entre os processadores, ou via decomposição de domínio, em que a solução é obtida através da combinação de soluções dos subproblemas. Sob quaisquer dessas abordagens são decompostos o domínio ou os dados, de modo a identificar as células da fronteira e as células internas a cada subdomínio ou conjunto de dados. O objetivo, nesse caso, é balancear a carga computacional entre eles de modo a minimizar o tempo de

comunicação entre os processos. Essas questões são do escopo de métodos de particionamento de malha.

O problema de particionamento do domínio computacional (ou malha numérica) é visto como um problema de particionamento de grafos, quando se considera no grafo que o conjunto de vértices são as células computacionais, e as arestas são as comunicações entre essas células. Então, pode-se associar a cada nodo da arquitetura os processos ou dados, e às arestas, as comunicações ou dependências de dados. Além disso, pode-se associar aos pesos dos nodos a carga computacional, e aos pesos das arestas, a carga de comunicação.

Essa abordagem é particularmente atraente, pois apesar de o *problema de particionamento* não possuir solução ótima (é um problema NP-difícil), os algoritmos baseados em grafos têm uma firme base matemática, que pode ser explorada para estabelecer algoritmos aproximativos de solução de modo a gerar os subdomínios ou as porções de dados.

Já os métodos de decomposição de domínios para solução das EDPs constituem um conjunto de técnicas (matemáticas e computacionais) para obter a solução do problema global combinando as soluções dos subproblemas locais. Essa abordagem é apropriada, não somente do ponto de vista computacional, mas também do matemático, já que existe uma firme base matemática para os algoritmos decorrentes dessa abordagem (problemas elípticos) como discutido em [CHA94], [SMI96] e [QUA99].

Em particular, o método de decomposição de domínio aditivo de Schwarz é empregado neste trabalho sob duas abordagens. Em uma delas é como método de solução, e na outra como pré-condicionador. Nessa última abordagem tem-se o chamado método Krylov-Schwarz, que é um dos métodos mais efetivos atualmente utilizados na solução de grandes e esparsos sistemas de equações.

Os métodos de solução usados para resolver os sistemas de equações localmente ou de modo distribuído, ou seja, que foram empregados de modo seqüencial ou que se tornaram objeto de processos de paralelização, foram o método direto de Thomas e os métodos iterativos do gradiente conjugado (GC) e do resíduo mínimo generalizado (GMRES).

Quando as matrizes são 3-diagonal, o emprego do algoritmo de Thomas é inquestionável quanto à robustez e à rapidez. Ele é um método direto de ordem N , onde N é a dimensão da matriz, e não preenche a matriz. Devido a essas características, foi empregado neste trabalho para obter as soluções dos sistemas 3-diagonal, como aqueles gerados pelo do método ADI (*Alternating Direction Implicit*) no caso de modelos 2D, e como aqueles gerados pela discretização do transporte vertical no modelo ETM 3D. Devido a formulação dos algoritmos numéricos, as velocidades horizontais do modelo hidrodinâmico 3D foram obtidas via algoritmo de Cholesky.

Para resolver os sistemas de equações lineares com matrizes *tipo 5-diagonal*, que são aquelas que têm no máximo cinco elementos por linha, foram empregados métodos iterativos do subespaço de Krylov. A opção por esses métodos decorreu de análise feita a partir da literatura técnica. Métodos diretos têm a vantagem de serem robustos e de fornecerem uma solução numérica exata, exceto pelos arredondamentos decorrentes da aritmética de ponto flutuante. Porém, levam ao preenchimento da matriz destruindo sua esparsidade, o que dificulta o armazenamento e sua otimização. Além disso, apresentam dependência global nas operações, tornando complicada a sua paralelização. Já os métodos iterativos clássicos (Jacobi, Gauss-Seidel, SOR, etc.) são, geralmente, computacionalmente menos eficientes, quando comparados aos métodos iterativos do subespaço de Krylov [SAA96].

Métodos do subespaço de Krylov são particularmente eficientes quando aplicados a matrizes grandes e esparsas. A eficiência decorre do fato de que seus algoritmos são construídos sobre operações básicas em álgebra linear e, portanto, são altamente

paralelizáveis. Uma aparente desvantagem dos métodos iterativos do subespaço de Krylov é a falta de uma sólida base matemática. Porém, no caso particular em que as matrizes são diagonalmente dominantes, simétricas ou não, essa classe tem métodos como o gradiente conjugado (GC), para matrizes simétricas definidas-positivas, e o como o resíduo mínimo generalizado (GMRES), para matrizes não simétricas. Esses algoritmos convergem rapidamente e suas propriedades são bem conhecidas, sobretudo as do algoritmo do GC. Por essas razões, foram esses os métodos de solução empregados neste trabalho.

1.2 Apresentação

Note-se, portanto, que a construção de um modelo computacional paralelo para simular a hidrodinâmica e o transporte de substâncias em corpos de água abrange várias etapas, que vão desde a definição do modelo matemático até a análise e a validação dos resultados. Sendo assim, para uma melhor apresentação do trabalho realizado, são sinteticamente pontuadas, a seguir, as principais questões que devem ser levadas em conta no desenvolvimento desses modelos paralelos:

1. Os modelos matemáticos que descrevem os problemas de valor inicial e de contorno, que são constituídos por equações diferenciais parciais ou equações integrais, condições de contorno e condições iniciais;
2. A malha numérica, o método de discretização para obter a versão discreta das EDPs, e as funções interpolantes para construir os esquemas (algoritmos) numéricos;
3. Os algoritmos de particionamento de malha que devem particionar o domínio computacional de modo a obter a melhor relação carga computacional /comunicação;
4. As estratégias e técnicas de paralelização que devem considerar a decomposição do problema em subproblemas e/ou dados, de modo que a solução paralela seja consistente com a solução em um só processador;
5. Os métodos de solução apropriados para serem empregados em conjunto com as estratégias de paralelização, objetivando maximizar a eficiência computacional;
6. A plataforma em que é implementado o código computacional. Se tal plataforma é formada por *clusters* de PCs heterogêneos e multiprocessados, o balanceamento da carga e a comunicação entre os processos são questões que devem ser adequadamente consideradas;

Essas questões são discutidas ao longo deste trabalho. O capítulo 2 trata dos problemas de valor inicial e de contorno que, neste trabalho, são constituídos das equações diferenciais parciais, e das condições iniciais e de contorno. Nesse capítulo, são constituídos, classificados e discutidos todos os modelos matemáticos empregados, e são apresentadas as aproximações e considerações físicas e matemáticas desses modelos.

O capítulo 3 contempla as discussões relativas ao sistema de coordenadas, aos valores nodais, a malha numérica e o domínio discreto. Nele se discutem as características das malhas horizontal e vertical, bem como se definem as variáveis dependentes na forma discreta. Discute-se, também, como são construídas as condições de contorno superior, inferior, e laterais abertas e fechadas. Além disso, discutem-se os tipos de domínio empregados para aproximar a região, considerando-se domínios retangulares e não retangulares.

O capítulo 4 discute os modelos introduzidos no capítulo 2, porém, dessa vez, na forma discreta. Adotando-se o método das diferenças finitas, são construídos os modelos hidrodinâmico e de transporte de substâncias 2D e 3D. Além disso, é construído um modesto modelo de parametrização de turbulência na forma discreta,

de modo a obter os coeficientes de difusão e de viscosidade turbulenta para os modelos 3D da hidrodinâmica e do transporte de substâncias.

O capítulo 5 aborda os métodos e técnicas de aproximação para os fluxos numéricos. A partir de uma ampla revisão bibliográfica da literatura técnica, optou-se por construir funções aproximantes para a advecção horizontal do modelo de transporte 3D, considerando-se que o método de limitação de fluxo de Sweby, dado que ele tem a desejável característica de ser TVD. Na solução da advecção vertical do modelo de transporte 3D foi empregado o método beta de Gross. Nos casos de modelos 2D, a abordagem para tratar os termos advectivos foi, em geral, a *upwind* de primeira ordem, dado que o interesse nesses casos foi explorar os aspectos computacionais e obter *background* para desenvolver os modelos 3D.

No capítulo 6, é discutida a aplicação dos métodos e técnicas de aproximação introduzidos no capítulo 5 para construir os esquemas numéricos para a obtenção das soluções das EDPs do escoamento nos casos 2D e 3D. Também se discutem os esquemas numéricos para a solução das EDPs do transporte escalar, nos casos 2D e 3D. Todas as condições de contorno implementadas são apresentadas e analisadas.

O capítulo 7 contempla as questões relativas ao paralelismo – métodos e estratégias – e ao ambiente computacional. Nele são discutidos paradigmas de programação, a plataforma computacional e os ambientes de programação empregados neste trabalho. Duas abordagens foram empregadas para a obtenção da solução paralela: a decomposição de dados e a decomposição de domínio. Além disso, faz-se uma discussão sobre o problema de particionamento de malha numérica, bem como do procedimento de refinamento local para a solução detalhada do transporte de substâncias. Apresentam-se os métodos e as estratégias empregadas para a solução dos sistemas de equações lineares resultantes dos procedimentos de discretização e das funções interpolantes. Discute-se sobre os métodos iterativos do subespaço de Krylov, pré-condicionadores algébricos e sobre o algoritmo de Thomas. Além disso, é discutido o método aditivo de Schwarz, como “resolvedor” (*solver*) e como pré-condicionador. Neste último caso, obtém-se o método de Krylov-Schwarz, que também é empregado na obtenção da solução paralela para os sistemas de equações lineares.

O capítulo 8 trata especificamente da apresentação das discussões e considerações finais decorrentes deste trabalho, bem como os apontamentos e comentários sobre trabalhos futuros.

O anexo 1 discorre sobre as propriedades e características numérico-computacionais que modelos computacionais de escoamento de superfície livre e de transporte escalar devem considerar quando do seu desenvolvimento e implementação. Discutem-se em detalhes a monotonicidade, a estabilidade, a consistência entre a continuidade e o transporte, a difusão numérica e a difusão física, a positividade, o transporte da solução, a conservação de massa, a construção de esquemas semi-implícitos e esquemas de integração temporal. Nos anexos 2 e 3 são apresentadas, respectivamente, as principais métricas para a avaliação da qualidade numérica e do desempenho computacional desses tipos de modelos computacionais paralelos. No anexo 4 discorre-se sobre os resultados, tabelados, de várias funções de restrição de fluxo que foram empregadas para obter a solução numérica da advecção horizontal da ETM 3D.

2 Problemas de valor inicial e de contorno

Para realizar simulações de fenômenos físicos, deve-se inicialmente traduzi-los em modelos matemáticos capazes de fornecer uma descrição precisa da natureza dos eventos, quando se consideram seus aspectos essenciais. Esses problemas são, geralmente, expressos através de equações diferenciais parciais (EDPs) (ou equações integrais), de condições iniciais (CI) e de condições de contorno (CC), e formam um problema de valor inicial e de contorno (PVIC).

Em um PVIC para problemas evolutivos, as equações são definidas em $(x, y, z, t) \in \Omega \subseteq \mathbb{R}^3 \times (0, T)$; as CI são definidas em $(x, y, z) \in \Omega$; e as CC, em $(x, y, z) \times t \in \partial\Omega \times [0, T]$, onde Ω e $\partial\Omega$ denotam, respectivamente, o interior e a fronteira do domínio, e T denota um dado tempo. A especificação das CC e CI são, nesse caso, imprescindíveis na simulação.

Condições de contorno tipo Dirichlet, Neumann e Cauchy definem o tipo da interação do escoamento ou do transporte, sob a ação de forças, na interface interior-exterior do domínio. As CC podem ser abertas, quando do caso de fronteiras fictícias que delimitam o interior e o exterior do domínio, nas quais as perturbações podem entrar e sair do domínio. Outras CC são as fechadas, situação em que não é permitido às perturbações entrar ou sair do domínio.

Para resolver numericamente esse PVIC contínuo, empregam-se métodos de discretização e funções interpolantes para aproximar as derivadas de modo a obter uma versão discreta, resolvível computacionalmente. A solução é obtida através da construção de esquemas (algoritmos) numéricos que são definidos em uma malha computacional, de modo que os valores das variáveis dependentes associados a esses pontos são obtidos a partir do cálculo dos sistemas de equações que fornecem esses valores no nível de tempo $t + \Delta t$ a partir do conhecimento dos valores no nível t , onde Δt é o passo de tempo.

2.1 Introdução

A dinâmica dos fluidos lida com o comportamento de fluidos em repouso ou em movimento. São muitos os conceitos necessários para oferecer uma visão geral a título de introdução a ela. Alguns desses conceitos são expostos a seguir, objetivando definir uma terminologia básica que é empregada ao longo do texto. Discussões a respeito desses conceitos podem ser encontrados em [BTC90], [FOX98], [RIZ98] e [APS2001].

Fluido é a matéria que se deforma continuamente sob a aplicação de uma tensão de cisalhamento não importando quão pequena ela possa ser. Assim, os fluidos compreendem as fases líquidas e gasosas das formas físicas nas quais a matéria existe. Uma distinção entre fluidos e sólidos pode ser feita comparando seus comportamentos. Ao contrário do fluido um sólido pode se deformar, porém não continuamente, quando uma tensão de cisalhamento lhe é aplicada.

Um escoamento é o transporte de substâncias em um fluido em que existe um deslocamento relativo de suas diversas partes [MAC76]. O escoamento pode ou não ter fronteiras. Se qualquer propriedade do fluido não varia com o tempo, o escoamento é dito estacionário (ou permanente).

Um escoamento é governado pelos princípios fundamentais da Mecânica e da Termodinâmica Clássica. A aplicação desses princípios fornece o modelo matemático que é escrito na forma de equações diferenciais ou equações integrais (EI). A abordagem integral permite a presença de descontinuidades na região, e isso tem particular importância em escoamentos com ondas de choque (escoamentos compressíveis). A forma diferencial assume que as propriedades do escoamento são

diferenciáveis, logo contínuas. Isso indica que a forma integral, por poder tratar expressões descontínuas, é indispensável na construção de soluções fracas, como discutido no anexo 1. Quando as quantidades do meio são contínuas e suficientemente diferenciáveis (suaves), pode-se mostrar que as duas formas são equivalentes.

Em geral, os textos sobre Mecânica dos Fluidos adotam os termos sistema e volume de controle (VC) para definir a região de trabalho. Um sistema é definido como uma quantidade de massa fixa e identificável; as fronteiras do sistema separam-no do ambiente à volta e podem ser fixas ou móveis, porém não há transferências de massa através das mesmas. Um VC é um volume arbitrário em uma região através da qual o fluido escoar. A fronteira geométrica, isto é, o contorno da região, é chamada superfície de controle (SC). Ela pode ser real ou fictícia, e pode estar em repouso ou em movimento.

Quanto ao enfoque, os princípios fundamentais que são aplicados ao estudo da Mecânica dos Fluidos podem ser formulados em termos de VCs infinitesimais ou finitos. Por infinitesimal, entende-se que a abordagem é via elemento diferencial de área ou de volume, e por finito, entende-se uma dada região finita do plano ou do espaço.

As grandezas (ou quantidades) físicas como comprimento, tempo, massa e temperatura são as dimensões, isto é, as variáveis espaciais e temporal do problema. Suas grandezas mensuráveis podem ser subdivididas em dois grupos: as primárias (ou básicas) e secundárias (ou derivadas das primárias), que são aquelas cujas dimensões são expressas em termos das primárias. As quantidades físicas devem ser dimensionalmente homogêneas, ou seja, devem estar no mesmo sistema de unidades.

Na Mecânica dos Fluidos dos meios contínuos, não se considera a estrutura molecular do fluido, apesar de ele ser composto por moléculas em constante movimento. Isso se deve ao fato de que, para a maioria das aplicações tecnológicas, o interesse está nos efeitos médios ou macroscópicos de muitas moléculas, pois são os elementos perceptíveis e mensuráveis. Então, pode-se tratar um fluido como matéria contínua (hipótese do contínuo), deixando de lado o comportamento molecular. Além disso, são encontradas as forças superficiais, que atuam nas fronteiras de um meio através do contato direto.

As forças desenvolvidas sem contato físico e distribuídas por todo o volume do fluido são denominadas forças de campo. As tensões produzidas em um meio resultam das forças que atuam em alguma parte desse meio. A tensão, uma grandeza tensorial de segunda ordem, é um modo de descrever o modo pelo qual as forças atuantes nas fronteiras do meio são transmitidas através dele.

Os fluidos podem ser classificados, de modo geral, de acordo com a relação entre a tensão de cisalhamento aplicada e a taxa de deformação. Como um fluido é a matéria que se deforma continuamente sob a ação de uma tensão de cisalhamento, na ausência dela não há deformação. Fluidos nos quais a tensão de cisalhamento é diretamente proporcional à taxa de deformação são chamados de Newtonianos. Não-Newtonianos são aqueles nos quais essa tensão não é diretamente proporcional à taxa de deformação.

A resistência à deformação que um fluido Newtoniano oferece quando submetido à ação da tensão de cisalhamento é uma constante que depende das propriedades do fluido sob influência do meio físico, e é proporcional à taxa de deformação. Essa constante é a viscosidade. Uma classificação dos fluidos sob esse ponto de vista pode ser encontrada em [FOX98].

Em problemas definidos no escopo da Mecânica Newtoniana, as equações (EDPs ou EIs) empregadas são as da continuidade (EC), da conservação da quantidade de movimento (ECQM) e da energia (EE), que são declarações matemáticas de três

princípios físicos fundamentais: a massa é conservada, a quantidade de movimento é conservada, e a energia é conservada.

Esses princípios, escritos na forma matemática, dão origem às chamadas equações *completas*, que não se resumem apenas às equações conservação da quantidade de movimento para um escoamento viscoso. O sistema completo que abrange as equações da continuidade, da conservação da quantidade de movimento e da energia é chamado de equações de Navier-Stokes [AND95]. Ele pode ser escrito nos sistemas de referência fixos ou não, nas abordagens integral e diferencial, nas formas conservativas e não conservativas [HIR92]. Se o escoamento é invíscido (não viscoso), o sistema é chamado de equações de Euler.

Quando o fluido é viscoso e incompressível, pode-se considerar a (quase) desvinculação dos processos mecânicos dos térmicos no escoamento. Deixam de existir os efeitos térmicos com origem em expansões internas do fluido. Um elo de ligação que permanece entre a parte dinâmica e a térmica do escoamento é a viscosidade molecular e, então, pode-se admitir que a viscosidade molecular é constante [SIL86]. É possível, portanto, simular o escoamento mecânico de um fluido viscoso incompressível somente com as equações da conservação da quantidade de movimento e da continuidade.

Usando a terminologia adotada em modelos atmosféricos, o enfoque desta tese está na *parte dinâmica* dos modelos hidrodinâmicos e de transporte escalar de substâncias em corpos de água rasos, com particular atenção aos seus aspectos numéricos e computacionais. As equações que modelam tais fenômenos são chamadas, respectivamente, de equações *shallow water* (ESW) e de equação do transporte escalar de substâncias (ETM). Quanto à *parte física*, a única questão abordada é a parametrização (simplificada) da turbulência.

Especificamente, no tocante à parte *dinâmica*, os processos físicos considerados pelo modelo hidrodinâmico são a taxa de variação local, a taxa advectiva, a aceleração de Coriolis, os gradientes de pressão barotrópica, os atritos nas interfaces água-sedimento e água-superfície livre. Quanto ao transporte de sedimentos, os processos físicos envolvidos são a taxa local e a taxa advectiva da variação de concentração. Essas quantidades físicas são calculadas pelos modelos de escoamento e de transporte de substâncias.

Quanto à parte *física*, os processos físicos envolvidos são, para o modelo hidrodinâmico, a viscosidade cinemática turbulenta nas direções horizontal e vertical. No caso do modelo de transporte, os processos físicos envolvidos são a difusão turbulenta nas direções horizontal e vertical. Essas quantidades físicas são calculadas pelo modelo de turbulência ou obtidas através de expressões semi-empíricas. A discussão sobre essa ou aquela escolha é feita na seção 2.7.2.

2.2 Classificação e aproximações para os modelos

Os modelos matemáticos podem ser classificados de acordo com as características dinâmicas e numéricas [SHE83]. Quando de acordo com a dinâmica, a dimensão espacial é a característica de classificação. Assim, eles podem ser unidimensional (1D), bidimensional (2D), tridimensional (3D) ou quase 3D (Q3D). Modelos 3D são os baseados nas *equações completas*. Modelos Q3D são aqueles modelos nos quais o corpo de água é dividido em várias camadas horizontais, sendo a camada superior a única com espessura variável no tempo devido às alterações da superfície livre. A integração entre as camadas é realizada pela equação da continuidade. Nesse modelo, são desprezados os efeitos inerciais e de atrito verticais. Esse é o caso das ESW 3D que, a rigor, são modelos Q3D.

Quanto aos modelos 2D, existem basicamente dois tipos: o integrado na vertical e o integrado na horizontal. O integrado na horizontal é restrito a corpos de água

fortemente estratificados com domínio suficientemente regular. Nesse caso, as EDPs são obtidas integrando-se na largura do corpo de água, e as variáveis dependentes são representativas da largura da seção transversal [SCE91]. O integrado na vertical é o mais popular e é discutido detalhadamente nos trabalhos de Leendertse ([LEE67], [LEE70] e [LEE71]). Outra abordagem, que considera aspectos horizontais e verticais, pode ser vista em [WUJ95].

Uma discussão mais qualitativa sobre a hidrodinâmica e o transporte pode ser encontrada em [WEI92], [ZHE95] e [TUC98]. E uma discussão detalhada sobre os aspectos físicos e matemáticos da modelagem da qualidade da água pode ser vista em [ORL83]. Neste trabalho, descrevem-se, porém, objetivamente, as aproximações físicas consideradas na dedução dessas EDPs, que são:

1. *incompressibilidade do escoamento*: sendo a água um fluido com variação total da densidade muito menor que a unidade, pode-se desacoplar as equações de continuidade e de conservação da quantidade de movimento da equação da energia, mostrando que é possível obter a equação do transporte de calor da equação da energia [VLA99];
2. *fluido Newtoniano*: nesse caso considera-se que os tensores de forças são linearmente proporcionais ao tensor de deformação e pode-se considerar que a viscosidade molecular é constante como em [CHE93] e [VLA99];
3. *aproximação hidrostática*: sendo as escalas horizontais muito maiores que as verticais, pode-se considerar que as acelerações verticais são desprezadas quando relativas às outras que intervêm na equação do movimento na direção vertical. Uma análise dimensional das ordens de magnitude das forças envolvidas mostra que as contribuições devem-se à gravidade e à pressão. Assim, a equação da quantidade de movimento vertical reduz-se ao balanço hidrostático das forças;
4. *aproximação de Boussinesq*: considera que as acelerações dos fluxos são menores que as da gravidade e, portanto, as variações de densidade são importantes quando afetam os termos de gravidade e não o são nos termos de inércia [VLA99]. Ou seja, assume-se que a densidade é constante, exceto nos termos baroclínicos [CHE93]. Nesse caso, a integração da equação de estado, que depende somente da concentração da substância (e da temperatura), incorpora ao modelo hidrodinâmico a variação de densidade;
5. *decomposição de Reynolds*: devido ao caráter turbulento do fluxo, as equações de Navier Stokes não podem ser resolvidas nem analítica nem numericamente, pois as incógnitas são de magnitude instantâneas. Uma solução para esse problema é a decomposição proposta por Reynolds (promediação de Reynolds), que consiste em considerar as magnitudes como sendo compostas de uma parte relativamente lenta e outra turbulenta, que leva em conta as perturbações da primeira, obtendo as equações de Navier-Stokes em valores médios temporais (ou promediadas);
6. *fechamento de turbulência*: ao decompor e promediar as equações obtêm-se modelos de fechamento da turbulência cujos termos regulam as variações do campo de flutuações. Esses modelos são resolvidos como os de fluxos turbulentos com termos de viscosidade, em que se supõe que são proporcionais ao gradiente da magnitude transportada, e cujo fator de proporcionalidade é a viscosidade e a difusão turbulenta horizontal e vertical, que são obtidas através de modelos de turbulências.

2.3 Modelo hidrodinâmico: ESW 3D

O modelo matemático de escoamentos de superfície livre com densidade constante, escrito em variáveis primitivas, e que pode ser obtido a partir das equações de Navier-Stokes assumindo a promediação de Reynolds e o escoamento como sendo

hidrostático, é chamado de equações *shallow water* (ESW) ([LEE89], [CAS90] e [WEI92]).

Sob o ponto de vista matemático, as ESW formam um sistema de EDPs hiperbólicas não-lineares para um fluido incompressível com superfície livre. Esse sistema é composto pelas equações da conservação da quantidade de movimento para as velocidades horizontais (ECQM_x e ECQM_y) e pela equação da continuidade (EC), e é escrito como:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} = -g \frac{\partial \eta}{\partial x} + \nu_h \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + \frac{\partial}{\partial z} \left(\nu_v \frac{\partial u}{\partial z} \right) + \Phi v \quad (2.1)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} = -g \frac{\partial \eta}{\partial y} + \nu_h \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) + \frac{\partial}{\partial z} \left(\nu_v \frac{\partial v}{\partial z} \right) - \Phi u \quad (2.2)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (2.3)$$

em que:

1. $u = u(x, y, z, t)$ e $v = v(x, y, z, t)$ são as velocidades horizontais nas direções X e Y , e $w = w(x, y, z, t)$ é a velocidade vertical na direção Z ;
2. $\eta = \eta(x, y, t)$ é o nível de água;
3. $\Phi = 2\omega \sin\phi$ é a força de Coriolis, onde ω é a velocidade angular da Terra e ϕ é a latitude;
4. $\nu_h = \nu_h(x, y, z, t)$ é o coeficiente de viscosidade cinemática turbulenta horizontal;
5. $\nu_v = \nu_v(x, y, z, t)$ é o coeficiente de viscosidade cinemática turbulenta vertical;
6. g é a aceleração gravitacional, e t é o tempo.

Integrando a equação da continuidade (ECIV) sobre a coluna de água e usando a condição de contorno cinemática na superfície livre, obtém-se a equação para a integração entre as camadas, dada por:

$$\frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x} \left(\int_{-h}^{\eta} u dz \right) + \frac{\partial}{\partial y} \left(\int_{-h}^{\eta} v dz \right) = 0 \quad (2.4)$$

sendo $h = h(x, y)$ a profundidade da água medida a partir de um dado nível, como aquele determinado pela superfície da água não perturbada, em que a profundidade total é dada por $H(x, y, t) = h(x, y) + \eta(x, y, t)$.

2.3.1 Condições de contorno e condições iniciais

As condições de contorno na superfície livre, isto é, na interface água-vento, em $z = \eta$, são obtidas considerando que não há escoamento atravessando essa interface, sendo a difusão vertical, que é determinada pelos coeficientes das viscosidades cinemáticas turbulenta vertical em (2.1) e (2.2), a responsável por estabelecer as CC nessa interface. Essas CC são dadas por ([AGO94] e [GRO99a]):

$$\nu_v \frac{\partial u}{\partial z} = \tau_x^v (u_{agua} - u_{ar}) \sqrt{(u_{agua} - u_{ar})^2 + (v_{agua} - v_{ar})^2}$$

$$v_v \frac{\partial u}{\partial z} = \tau_y^v (v_{agua} - v_{ar}) \sqrt{(u_{agua} - u_{ar})^2 + (v_{agua} - v_{ar})^2}$$

em que os coeficientes de atrito na interface água-superfície livre nas direções X e Y são como:

$$\tau_x^v = \rho_{ar} C_d^v \text{ e } \tau_y^v = \rho_{ar} C_d^v$$

em que C_d^v é o coeficiente de atrito do vento na superfície da água; u_{agua} e v_{agua} são as velocidades da água na camada superior nas direções X e Y , respectivamente; onde $(u_{ar}, v_{ar}) = (\|(u, v)\| \text{ sen } \phi, \|(u, v)\| \text{ cos } \phi)$ é a velocidade do vento, ϕ é o ângulo entre a velocidade do vento com o eixo X .

As CC no leito do corpo de água, isto é, na interface sedimento-água, são obtidas considerando que não há fluxo atravessando essa interface, sendo a difusão vertical a responsável por estabelecer as CC nessa interface. Nesse caso, pode-se especificar essas CC empregando a expressão de Manning-Chezy ou um modelo de turbulência de camada limite ([CAS2000] e [APS2001]). A primeira abordagem foi considerada neste trabalho e nesse caso, as CC inferiores são dadas, em $z = -h$, por:

$$v_v \frac{\partial u}{\partial z} = \tau_x^s u \text{ e } v_v \frac{\partial v}{\partial z} = \tau_y^s v$$

em que $\tau^s = g(C_z^2)^{-1} \sqrt{u^{-2} + v^{-2}}$, onde C_z é o coeficiente de atrito de Chezy, sendo que \bar{u} e \bar{v} são as componentes da velocidade na interface sedimento-água nas direções X e Y , respectivamente. Essas componentes agregam os efeitos da rugosidade no fundo, sendo que esses efeitos podem ser incorporados nas equações, considerando que são funções do gradiente do nível da água e da velocidade da água como [LEE67]:

$$\bar{u} = \frac{\partial u}{\partial t} + g \frac{\partial \eta}{\partial x} = 0 \text{ e } \bar{v} = \frac{\partial v}{\partial t} + g \frac{\partial \eta}{\partial y} = 0$$

sendo que outras expressões para os diferentes tipos de rugosidade do leito podem ser vistas em [GRO99a] e [VLA99], onde se obtém o atrito no fundo através um perfil logarítmico, função da rugosidade e , da constante de von Karman.

Note-se que as CC podem ser dos tipos laterais abertas (*inflow* e *outflow*), laterais fechadas, superior e inferior. As CC superior e inferior são aquelas em que há deslizamento parcial e são especificadas em termos dos atritos nas interfaces da superfície livre e do leito do corpo de água, respectivamente, como feito nessa seção. Nas fronteiras laterais abertas especifica-se ou CC de Dirichlet (modelo de escoamento), definindo níveis de água ou vazão, ou a CC de Neumann (modelo de transporte), definindo que o gradiente é nulo. Nas fronteiras laterais fechadas especifica-se que não há fluxo e há deslizamento completo, indicando que não há penetração de água em paredes sólidas e que as velocidades da água nessas fronteiras são dadas pela componente tangencial da velocidade. Esses tipos de CCs são discutidos na seção 2.5.1, 3.2.1 e 3.4.

As condições iniciais para o PVIC podem ser obtidas considerando que o escoamento se encontra parado no tempo inicial e com uma dada elevação para a superfície da água. Mas essa é uma simplificação não necessária.

Note-se, também, que as equações da conservação da quantidade de movimento devem ser modificadas quando o constituinte modelado na equação do transporte é, por exemplo, o sal ou sedimentos. Nesses casos, deve-se considerar o gradiente de densidade no termo de pressão, e faz-se necessário utilizar uma equação de estado para relacionar a concentração da substância com a massa específica da água ρ e a

temperatura. Isso é necessário devido ao fato de que a densidade da água varia quando essas substâncias estão nela dissolvidas, o que altera o regime do escoamento e isso deve ser incorporado nas equações de conservação da quantidade de movimento.

Nessas condições, vale a relação $\partial p/\partial z = -\rho g$, sendo p a pressão. Essa abordagem é dita aproximação de Boussinesq, e nela a densidade é constante em todos os termos de força, exceto na força de corpo das ECQMs [CHE93]. Integrando essa expressão convenientemente, obtém-se uma nova formulação para as ECQM_x e ECQM_y como pode ser vista em [KIM94] e [GRO98].

2.4 Modelo hidrodinâmico: ESW 2D-IV

Modelos 2D podem ser integrados na vertical (2D-IV) quando o corpo de água é bem misturado ou parcialmente estratificado. Essa classe envolve os mais conhecidos modelos 2D em uso. Essa aproximação pode ser considerada quando o fluido tem densidade constante e uniforme; os movimentos ocorrem sobretudo na horizontal, onde a escala vertical H é muito menor que a horizontal L ; os efeitos das flutuações de velocidade de pequena escala estão agregados dentro dos termos de tensão e de viscosidade (promediação de Reynolds). Assim, a integração vertical de uma variável dependente qualquer ψ , que fornece seu valor médio na vertical, como ilustrado na fig. 2.1, é determinada por ([LEE71] e [CSL85]):

$$\frac{1}{H} \int_{-h}^{\eta} \psi dz = \bar{\psi}$$

onde h, η e $H = h + \eta$ são, respectivamente, a profundidade abaixo de um nível de referência, o nível da água acima desse referencial, e a profundidade total. A variável dependente ψ pode ser u, v ou s , que são, respectivamente, as velocidades horizontais nas direções X e Y da água e a concentração. Então, $\bar{\psi}$ denota, respectivamente, U, V e S que são as velocidades horizontais médias e a concentração média, variáveis dependentes de modelos hidrodinâmicos e de transporte de substâncias integrados na vertical.

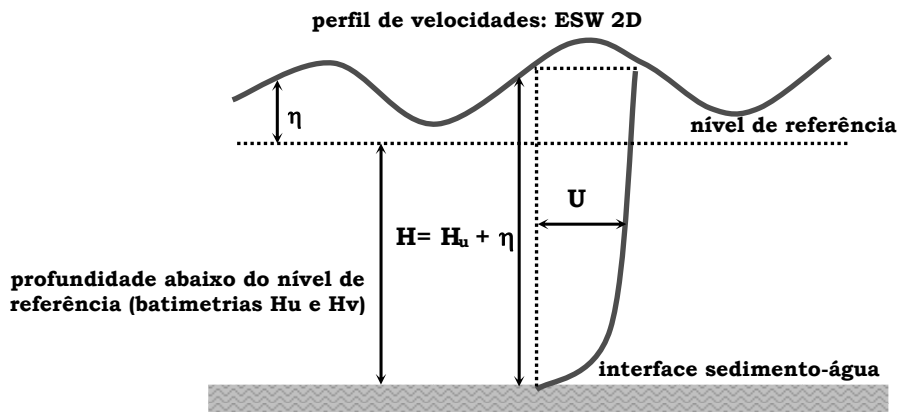


FIGURA 2.1 - Perfil das velocidades para as ESW integradas verticalmente

Após manipulações algébricas, cujos detalhes podem ser vistos em [CSL85] e [WEI92], as ESW 2D integradas verticalmente (ESW 2D-IV) são escritas como:

$$\frac{\partial U}{\partial t} + U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial y} - \Phi V = -g \frac{\partial \eta}{\partial x} + \frac{\tau_x^v}{\rho H} - \frac{\tau_x^s}{\rho H} + \varepsilon_h \left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} \right) \quad (2.5)$$

$$\frac{\partial V}{\partial t} + U \frac{\partial V}{\partial x} + V \frac{\partial V}{\partial y} + \Phi U = -g \frac{\partial \eta}{\partial y} + \frac{\tau_y^v}{\rho H} - \frac{\tau_y^s}{\rho H} + \varepsilon_h \left(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} \right) \quad (2.6)$$

$$\frac{\partial \eta}{\partial t} + \frac{\partial [(h + \eta)U]}{\partial x} + \frac{\partial [(h + \eta)V]}{\partial y} = 0 \quad (2.7)$$

em que:

1. g é a aceleração gravitacional, e $\rho = \rho_{\text{água}}$ é a densidade da água;
2. $H = h + \eta$ é a distância do fundo até a superfície livre, onde $\eta = \eta(x, y, t)$ é a elevação da água acima de um nível de referência, e $h = h(x, y)$ representa a profundidade abaixo desse nível;
3. $U = U(x, y, t)$ e $V = V(x, y, t)$ são as componentes das velocidades horizontais médias obtidas pela integração vertical nas direções X e Y ;
4. $\Phi = 2\omega \sin\phi$ é a força de Coriolis, onde ω é a velocidade angular da Terra e ϕ é a latitude;
5. ε_h é o coeficiente de viscosidade cinemática turbulenta horizontal.

2.4.1 Condições de contorno e condições iniciais

As condições de contorno inferiores são especificadas assumindo, como no modelo 3D, que não há fluxo e a difusão vertical é o único termo onde se agregam os atritos nessa interface. Desse modo considera-se o coeficiente de Chezy $C_z = 7,83 \ln(0,3H/Z_0)$, em que $Z_0 \approx 0,001m$ é a rugosidade na interface água-leito do corpo de água. Esse parâmetro é, no perfil turbulento de velocidade de Prandtl, uma medida da ordem de grandeza da espessura da camada limite. Ou seja, é uma medida que é afetada pela natureza do fluido. As CC são, para as direções X e Y , como:

$$\frac{\tau_x^s}{\rho_{\text{água}} H} = g \frac{U \sqrt{U^2 + V^2}}{C_z^2 H} \quad \text{e} \quad \frac{\tau_y^s}{\rho_{\text{água}} H} = g \frac{V \sqrt{U^2 + V^2}}{C_z^2 H}$$

Para as CC superiores, consideram-se os coeficientes de atrito na interface água-superfície livre nas direções X e Y , resultando em:

$$\frac{\tau_x^v}{\rho_{\text{água}} H} = C_d^v \rho_{\text{ar}} W_x^2 \quad \text{e} \quad \frac{\tau_y^v}{\rho_{\text{água}} H} = C_d^v \rho_{\text{ar}} W_y^2$$

onde $W_x = \|(U, V)\| \sin\phi$ e $W_y = \|(U, V)\| \cos\phi$ são as componentes da velocidade do vento nessas direções, medidas a 10 metros acima do nível da água; ρ_{ar} é a densidade do ar e C_d^v é o coeficiente de atrito na superfície da água.

Como as velocidades horizontais e o coeficiente de arrasto C_d^v são obtidos a 10 metros acima do nível da água, eles são, muitas vezes, denotados na literatura técnica por U_{10} , V_{10} e C_{10} . Existem diferentes relações empíricas para determinar o arrasto do vento na superfície da água, e sua interação e a sua contribuição na determinação do atrito na interface água-superfície livre. Algumas expressões encontradas na literatura técnica estão sistematizadas em [VLA99] e são como:

1. $C_{10} = 0.9 \cdot 10^{-3}$; $10ms^{-1} < \|(U_{10}, V_{10})\| < 23ms^{-1}$;
2. $C_{10} = (2.4 \pm 0.5)10^{-3}$; $\|(U_{10}, V_{10})\| > 10ms^{-1}$;

3. $C_{10} = 1.2 \cdot 10^{-3}; |(U_{10}, V_{10})| \approx 10 \text{ms}^{-1};$
4. $C_{10} = 1.14 \cdot 10^{-3}; 4 \text{ms}^{-1} < |(U_{10}, V_{10})| < 11 \text{ms}^{-1};$
5. $C_{10} = (0.98 + 0.14 |(U_{10}, V_{10})|) 10^{-3}.$

sendo que essas relações também se aplicam às condições de contorno discutidas na seção 2.3.1 (para o modelo 3D).

Nas ESW, considera-se que a distribuição de pressão na vertical é hidrostática, de modo que os gradientes de pressão barotrópica são substituídos pelos gradientes da superfície livre da água. A distribuição das velocidades verticais é considerada uniforme, e as substâncias dissolvidas são bem misturadas na coluna vertical de água. Sendo a coordenada vertical aproximada pela pressão, dado que a água está em equilíbrio hidrostático, desprezam-se a aceleração vertical e a componente vertical da força de Coriolis.

2.5 Modelo de transporte escalar: ETM 3D

A presença de substâncias dispersas e/ou dissolvidas em corpos de água que sejam decorrentes ou não de atividades humanas como, por exemplo, derramamento de petróleo, despejo de dejetos domésticos, despejo industrial, sedimentos, etc., e a construção de obras de acesso portuário e/ou de transporte fluvial nas margens costeiras de estuários e lagos podem resultar em danos a esses ecossistemas. É, portanto, importante a previsão, o monitoramento e a avaliação dessas ações e/ou atividades.

A modelagem computacional do transporte (advecção-difusão) e do regime hidrodinâmico (escoamento) permite avaliar o impacto de como determinadas ações podem alterar o regime hidrodinâmico, bem como a dispersão de determinadas substâncias. Este trabalho considera que o constituinte simulado é aquele formado por dejetos humanos e o interesse é obter a concentração dos coliformes fecais em regiões contaminadas. O termo de fonte da equação de transporte escalar de substâncias é uma abstração de um emissário que, em geral, é construído através de uma tubulação cuja extremidade tem difusores ([VIN91] e [CAM99]). Para uma eficiente diluição, de modo a não contaminar regiões que tenham importância ambiental e econômica, o local do despejo deve ser apropriado e considerar o regime hidrodinâmico do corpo de água.

O transporte do constituinte é realizado através de uma série de mecanismos conhecidos como *mistura* e classificados como sendo aqueles que produzem o transporte advectivo, e aqueles que produzem o transporte difusivo. O transporte advectivo é realizado pelo campo de escoamento, que é influenciado por fatores astronômicos (marés), meteorológicos (ventos, chuvas), sazonais (descarga de água) e por efeitos secundários (força de Coriolis). O transporte difusivo deve-se principalmente à turbulência do escoamento, pois a difusão molecular é menos importante [SCE91].

Note-se que os fenômenos de advecção e convecção são parte integrante das mais variadas áreas e, em particular de problemas sob o escopo da Dinâmica dos Fluidos. Alguns autores, como Vreugdenhil [VRE94], não distinguem esses fenômenos físicos, indicando que ambos significam o transporte de alguma quantidade (calor, massa, quantidade de movimento, etc.) por um escoamento, já que o processo físico é o mesmo, independentemente da causa. A distinção é feita em dinâmica dos fluidos geofísica, onde a convecção indica escoamento vertical gerado pela diferença de densidade de um fluido termicamente estratificado, enquanto a advecção denota o escoamento horizontal devido ao termo não linear da equação do movimento. Neste trabalho se emprega o termo advecção para denotar ambos casos, como Vreugdenhil.

Não considerando os fenômenos de natureza química, são basicamente dois os processos envolvidos no transporte de substâncias: a advecção e a difusão. Simplificadamente, pode-se dizer que a advecção transporta as substâncias sem dispersá-las, e que a difusão é o fenômeno pelo qual as substâncias vão se espalhando ao mesmo tempo em que a concentração do poluente vai baixando. É devido à difusão que, mesmo na ausência da circulação, a concentração das substâncias diminui ao longo do tempo.

Em geral, o estudo dessa dispersão é feito a partir da formação inicial da pluma e da sua conseqüente distribuição pelos movimentos da água. A dispersão e a configuração da pluma dependem de fatores como a especificação dos difusores, a taxa de vazão e a densidade da substância ejetada, etc. Além disso, dependem do transporte das substâncias pelos processos advectivo e difusivo turbulento, sendo esses dependentes do campo de velocidades e das características físicas da água na região de lançamento.

Levando-se em conta a ação da circulação sobre uma pluma inicial, se a fonte do afluente está localizada em um ponto (x_0, y_0, z_0) , a concentração da substância ejetada em uma célula 3D deve satisfazer o fato de que a variação total de concentração no tempo deve ser igual à variação advectiva, mais a difusão turbulenta, mais o termo de decaimento, mais o termo de fonte. Essa afirmação pode ser escrito matematicamente como ([GRO98] e [CAM99]):

$$\left. \begin{aligned} \underbrace{\frac{\partial S}{\partial t}}_1 + \underbrace{\frac{\partial(uS)}{\partial x} + \frac{\partial(vS)}{\partial y} + \frac{\partial(wS)}{\partial z}}_2 &= \underbrace{\frac{\partial}{\partial x} \left(\varepsilon_h \frac{\partial S}{\partial x} \right) + \frac{\partial}{\partial y} \left(\varepsilon_h \frac{\partial S}{\partial y} \right) + \frac{\partial}{\partial z} \left(\varepsilon_v \frac{\partial S}{\partial z} \right)}_3 + \\ &+ \underbrace{F(x_0, y_0, z_0, t)}_4 S + \underbrace{K(x, y, z, t)}_5 \end{aligned} \right\} \quad (2.8)$$

onde os indicadores numéricos expressos na parte inferior da equação 2.8 indicam que a respectiva porção da EDP representa:

1. taxa de variação local, dada pela variação temporal da concentração S em cada ponto do espaço;
2. termo advectivo horizontal e vertical, que é a variação de S produzida pelo transporte das substâncias (ou efluentes) pela circulação;
3. termo de difusão turbulenta horizontal e vertical, que é variação produzida por movimentos de subescala (movimentos turbulentos);
4. termo fonte, que representa a vazão da emissão em (x_0, y_0, z_0) ;
5. termo de decaimento (ou depleção) da substância.

Para determinar o termo (ou matriz) de decaimento $K = K(x, y, z, t)$, considera-se uma concentração inicial $S = S(x_0, y_0, z_0, t)$ em um corpo de água estático e com termo de fonte $F = F(x, y, z, t) = 0$. Nesse caso, (2.8) reduz-se a:

$$\left. \frac{\partial S}{\partial t} = K(x, y, z, t) \right\} \quad (2.9)$$

Como o decaimento, resultante de processos físicos, químicos e/ou biológicos, é proporcional à concentração da substância a cada tempo, tem-se que $K(x, y, z, t) = -kS(x, y, z, t)$, onde k é a constante de proporcionalidade e o sinal negativo indica o decaimento. Portanto, (2.9) pode ser reescrita como $\partial S(x, y, z, t)/\partial t = -kS(x, y, z, t)$, cuja solução é $S(x, y, z, t) = S(x, y, z, 0)e^{-kt}$, indicando que a concentração decai com o tempo.

Especificamente, no caso de bactérias coliformes, considera-se que no tempo $t=T_{90}$ a concentração decai para 10% do valor inicial, ou seja, o tempo necessário para que a concentração inicial decaia em 90% [ZHE95]. Nesse caso tem-se $0,1S(x,y,z,0)=S(x,y,z,0)e^{-kt}$, e portanto $k=2,3/T_{90}$.

Este trabalho considera somente o transporte escalar de uma substância, mas uma discussão detalhada sobre os procedimentos numérico-computacionais para a simulação de vários constituintes pode ser vista em [LEE70], [LEE71] e [ORL83]. A equação de transporte escalar 3D (ETM) é escrita na forma conservativa como [GRO98]:

$$\begin{aligned} \frac{\partial(S)}{\partial t} + \frac{\partial(uS)}{\partial x} + \frac{\partial(vS)}{\partial y} + \frac{\partial(wS)}{\partial z} = \\ = \frac{\partial}{\partial x} \left(\varepsilon_h \frac{\partial S}{\partial x} \right) + \frac{\partial}{\partial y} \left(\varepsilon_h \frac{\partial S}{\partial y} \right) + \frac{\partial}{\partial z} \left(\varepsilon_v \frac{\partial S}{\partial z} \right) + FS + K \end{aligned}$$

em que:

1. S é a concentração;
2. $\varepsilon_h = \varepsilon_h(x,y,z,t)$ é o coeficiente de difusão turbulenta horizontal;
3. $\varepsilon_v = \varepsilon_v(x,y,z,t)$ é o coeficiente de difusão turbulenta vertical;
4. F é o termo de fonte que é determinado, neste trabalho, em termos da concentração de um volume de despejo (diluição) como:

$$S_{fonte}^n = S_{fonte}^{n-1} + \frac{Q(m^3/s)\Delta t(s)}{V_d(m^3)\Delta z_k^n(m)} S_{fonte} \left(\frac{mpn}{100ml} \right)$$

em que Q é a fonte; Δt é o passo de tempo; V_d é o volume de diluição; Δz_k^n é a altura da k -ésima camada; S_{fonte} é a concentração de bactérias coliformes no local de despejo ($\cong 9.10^6$ nmp/100 ml, onde nmp é o número médio provável);

5. $K = 2.3/T_{90}$, onde T_{90} é o tempo para remover 90% dos organismos, isto é, a constante de decaimento.

Note-se que a presença de coliformes fecais indica que a água esteve em contato com dejetos, o que aumenta a probabilidade de estarem presentes bactérias patogênicas. Essa presença, embora não implique obrigatoriamente que uma pessoa desenvolva alguma doença se ingerir essa água, representa um risco maior para a sua saúde. A água só é considerada potável se a sua análise bacteriológica indicar ausência de coliformes fecais. A balneabilidade é permitida quando a concentração de bactérias coliformes estiver abaixo de 1000 nmp/100ml, onde nmp é o número médio provável, que é uma estimativa da densidade de coliformes fecais em uma amostra [CET2001].

2.5.1 Condições de contorno e condições iniciais

A modelagem da distribuição de concentração (transporte de substâncias) pode melhorar a compreensão do regime de transporte; ou reconstruir a evolução de uma pluma existente a partir da data de origem até o presente; ou avaliar as reações de uma pluma existente para os contaminantes presentes; ou simplesmente simular quais as sub-regiões são mais passíveis de contaminação para um dado regime de escoamento ([ZHE95] e [ZHE99]). Para efetivar essa simulação, deve-se considerar as condições iniciais (CI) do problema. No caso da ETM, essas CI são dadas por

$S = S_{t=0}(x, y, z)$, indicando uma distribuição inicial conhecida. Um caso particular é tomar simplesmente $S = S_{t=0}(x, y, z) = 0$.

A solução da ETM também requer a especificação de CC tipo de Dirichlet e/ou de Neumann. A CC de Dirichlet é definida quando a concentração é especificada nas fronteiras para um período específico de tempo agindo como uma fonte, fornecendo massa para o domínio, ou como sumidouro, retirando a massa do domínio. Nesse caso, as CC são dadas genericamente por:

$$S \equiv S(x, y, z, t); (x, y, z, t) \in \partial\Omega \times [0, T]$$

onde $\partial\Omega$ denota a fronteira do domínio computacional e $S(x, y, z, t)$ é uma concentração especificada nessa fronteira que pode variar no espaço e no tempo. Essa CC tipo Dirichlet é especificada em fronteiras *inflow*, ou seja, em fronteiras onde o fluxo está entrando no domínio computacional.

A condição de contorno tipo Neumann ocorre quando o gradiente de concentração é normal à fronteira, significando que um fluxo advectivo e dispersivo deve ser aí especificado. Porém, nas fronteiras laterais abertas *inflow* e *outflow*, onde esta última é a fronteira na qual o fluxo está saindo do domínio, pode-se assumir que o fluxo advectivo domina o fluxo dispersivo. Nesse caso as CC podem, então, ser simplificadas [ZHE95]. A CC gradiente é expressa como:

$$\frac{\partial S(x, y, z, t)}{\partial \bar{n}(x, y, z, t)} = \nabla S; (x, y, z, t) \in \partial\Omega \times [0, T]$$

onde $\bar{n}(x, y, z, t)$ é o vetor unitário normal no ponto $(x, y, z, t) \in \partial\Omega$.

Especificamente para os modelos de transporte, consideram-se condições de contorno do tipo Dirichlet e de Neumann. Na primeira, é dada uma concentração na fronteira *inflow* do domínio computacional como discutido na seção 2.5.1. No caso da CC de Neumann especifica-se uma CC tipo gradiente nulo na fronteira interna *outflow*, indicando que a taxa de variação espacial é nula. Esse tipo de CC permite que a concentração saia do domínio sem acumular-se na fronteira, pois fisicamente a CC $\nabla S|_{\partial\Omega} = 0$ indica que a massa que entra é igual à massa que sai. Essa condição pode ser definida observando que, se N_{cc} é o ponto da fronteira *outflow* e N_{ex} é o ponto da primeira célula externa, uma CC tipo gradiente nulo é como (direção X):

$$\nabla S|_{\partial\Omega} \stackrel{not}{\equiv} \frac{\partial S}{\partial x} = \frac{S_{N_{ex}} - S_{N_{cc}}}{\Delta x} = 0$$

resultando em $S_{(N_{cc})}^{n+1} = S_{(N_{ex})}^{n+1}$. Portanto, com essa CC a célula N_{ex} pode ser incorporada ao sistema de equações quando do emprego de esquemas numéricos implícitos ou semi-implícitos. Quando os esquemas numéricos são explícitos usa-se esquemas de aproximações para essas expressões em células adjacentes a essas fronteiras. No caso multidimensional, as outras componentes da CC gradiente (direções Y e Z) podem ser incorporadas através de decomposições unidimensionais na direção normal de cada uma das faces das células.

Note-se que a solução do transporte de substâncias não é somente influenciada pelas suas próprias CC, mas também pelas CC do modelo hidrodinâmico. Uma questão importante é, então, analisar como a combinação das CC das ESW e da ETM determinam o controle da taxa de entrada e saída de massa nas células das fronteiras. Mas a abordagem para essa questão não é tratada neste trabalho, porém alguma discussão pode ser vista em [ZHE95].

2.6 Modelo de transporte escalar: ETM 2D-IV

A equação que modela o transporte de constituintes dispersos em corpos de água bem misturados é a ETM 2D, que, integrada na vertical (ETM 2D-IV), é escrita na forma conservativa (ou fluxo), como em [LEE71] e [ZHE95]:

$$\left. \begin{aligned} \frac{\partial(SH)}{\partial t} + \frac{\partial(SHU)}{\partial x} + \frac{\partial(SHV)}{\partial y} \\ = \frac{\partial}{\partial x} \left(HD_x \frac{\partial S}{\partial x} \right) + \frac{\partial}{\partial y} \left(HD_y \frac{\partial S}{\partial y} \right) + KHS + FH \end{aligned} \right\} (2.10)$$

sendo, em (2.10), que:

1. S é a concentração integrada na vertical;
2. $D_x = \alpha_L \cos^2 \theta + \alpha_T \sin^2 \theta$ e $D_y = \alpha_L \sin^2 \theta + \alpha_T \cos^2 \theta$ são os componentes do tensor de dispersão nas direções X e Y ; α_L e α_T são os coeficientes de difusão longitudinal e transversal, semelhantes aos coeficientes de difusão turbulenta horizontal e vertical da ETM 3D; e θ é o ângulo formado entre o eixo X e o vetor velocidade;
3. F é o termo de fonte que é calculado, neste trabalho, em termos da concentração como [VIN91]:

$$S_{fonte}^n = S_{fonte}^{n-1} + \frac{Q(m^3/s)\Delta t(s)}{A_d(m^2)H(m)} S_{fonte} \left(\frac{mpn}{100ml} \right)$$

em que Q é a fonte; Δt é o passo de tempo; A_d é a área de diluição; H é a profundidade média; S_{fonte} é a concentração de bactérias coliformes;

A especificação das CC (de Dirichlet e de Neumann), e das condições iniciais para a ETM 2D integrada na vertical é análogo às CC da ETM 3D.

2.7 Parametrização da turbulência

Nesta seção discutem-se alguns modelos de parametrização de turbulência, ou seja, modelos de fechamento de turbulência para determinar os coeficientes de viscosidade cinemática turbulenta das $ECQM_x$ e $ECQM_y$, e os coeficientes de difusibilidade turbulenta da equação do transporte escalar de substâncias.

A turbulência caracteriza os escoamentos irregulares nos quais as grandezas físicas comportam flutuações aleatórias em relação ao tempo e ao espaço em torno de valores médios estatísticos. Em simulações de escoamentos naturais, o interesse não está nos valores das variáveis afetadas por flutuações turbulentas, e sim na influência das flutuações turbulentas do ponto de vista de um escoamento médio estatístico, no qual a turbulência é interpretada como uma perturbação. Sob o ponto de vista do escoamento médio, a turbulência é uma perturbação aleatória, rotacional, tridimensional, dissipativa, difusiva, não-linear e com diversas escalas de tempo e espaço. Esses aspectos da turbulência estão implícitos nas EDPs através das tensões aparentes com flutuações turbulentas ([SIL86] e [BUR2000]).

Um aspecto importante da influência da turbulência sobre o escoamento médio é a dissipação de energia, que influi diretamente nas outras características do escoamento. Nesse processo de dissipação, parte da energia do escoamento médio é transferida pela turbulência das maiores para as menores escalas espaciais, onde há transformação em calor pela ação das tensões viscosas. É a chamada *cascata de energia* ([SIL86] e [VIL2000]). Somente as maiores escalas espaciais interagem com o escoamento médio, e a dissipação de energia ocorre nas menores escalas (da

viscosidade), a taxa de dissipação de energia é definida pelo movimento de grande escala, e não pela viscosidade.

Essa característica da *cascata de energia* é importante, pois os principais efeitos da turbulência no escoamento médio (transporte de substâncias e de quantidade de movimento) podem ser avaliados a partir das grandes escalas espaciais. As tensões aparentes podem ser avaliadas, partindo-se de pressupostos que utilizem as características do escoamento médio. Essas tensões aparentes que surgem das EDPs integradas no tempo são conhecidas por tensões de Reynolds. Todos esses aspectos podem ser vistos, dentro de uma perspectiva e de um formalismo contemporâneos, em [BUR97], [BUR98], [BUR98a], [BUR2000], [BUR2000a] e [BUR2000b].

A partir do princípio da viscosidade turbulenta, Kolmogorov e Prandtl postularam que os coeficientes de viscosidade turbulenta, responsáveis pela troca de quantidade de movimento, e os de difusão turbulenta, responsáveis pela troca de calor e massa, são proporcionais a uma velocidade e a um comprimento de escala (*Eddy-Viscosity Models*). Outras abordagens como, por exemplo, *Beyond Eddy-Viscosity Models* e *Wall Boundary Conditions*, podem ser vistas em [APS2001].

Empregando o princípio da viscosidade turbulenta, pode-se modelar as tensões turbulentas para definir as escalas de espaço e tempo e o transporte de grandezas da turbulência. No modelo proposto por Reynolds (promediação de Reynolds) [VIL2000], a turbulência está baseada na promediação temporal das equações, que consiste em dividir o campo instantâneo de velocidade em uma componente média e outra turbulenta ([SIL86] e [VIN91]).

A modelagem das tensões de Reynolds é feita com base no conceito de viscosidade turbulenta introduzido por Boussinesq, que considerou a analogia entre os fenômenos da viscosidade e os da turbulência. Esses fenômenos proporcionam trocas de partículas fluidas com quantidades de movimento diferentes e cujo efeito no escoamento toma a forma de tensões. A diferença entre os fenômenos reside no fato de que a viscosidade, por ocorrer no nível molecular, é uma propriedade do fluido, enquanto a turbulência, por envolver partículas macroscópicas, depende também do escoamento.

Dessa interpretação, por analogia às relações de Newton e de Stokes para a viscosidade, resulta a definição de um coeficiente da viscosidade turbulenta, variável no tempo e no espaço. Assim, deve-se determinar quais os coeficientes de viscosidade e de difusão turbulenta, análogos aos coeficientes de viscosidade e de difusão do fluido. A sua obtenção é feita empregando os modelos de fechamento de turbulência das equações do escoamento e do transporte [BUR98a].

Portanto, em simulações mais acuradas, devem-se desenvolver modelos de parametrização para determinar, para as ESW, os coeficientes da viscosidade cinemática turbulenta horizontal ν_h , e os de viscosidade cinemática turbulenta vertical ν_v . No caso da ETM, deve-se obter os coeficientes de difusão turbulenta horizontal ϵ_h e os de difusão turbulenta vertical ϵ_v , que estão agregados aos modelos matemáticos do escoamento e do transporte como mostrado nas seções anteriores.

2.7.1 Classificação de modelos de turbulência

Os modelos de parametrização de turbulência podem ser definidos a partir do princípio da viscosidade turbulenta, que permite obter os coeficientes de troca turbulenta em função das propriedades médias do fluido [VIL2000]. Uma classificação de modelos de turbulência é a de Rodi [ROD80], que tem por base o número de equações de transporte de grandezas turbulentas. Assim há modelos de *zero equações* (ou nenhuma equação), de *uma equação*, de *duas equações*, e de *duas e meia equações*:

1. *os modelos de zero equações* obtêm os coeficientes de viscosidade turbulenta sem o emprego de equações de transporte, mas com parametrizações *ad hoc* ou semi-empíricas. Quando o coeficiente de viscosidade é considerado constante, a modelagem da turbulência deixa de existir, e a utilização do coeficiente de viscosidade justifica-se apenas em razão de manter a estabilidade numérica da solução. Quando o coeficiente é considerado variável, a abordagem mais comum para parametrizar a turbulência é adaptar o conceito de comprimento de mistura de Prandtl ao escoamento ([VIL2000] e [APS2001]).
2. *os modelos de uma equação* de transporte consideram, geralmente, a equação de transporte da energia cinética turbulenta. Nos casos dos modelos que empregam o conceito de viscosidade turbulenta, o coeficiente é determinado com base no valor da energia cinética κ e em uma escala espacial. No caso de modelos que não utilizam o conceito de viscosidade turbulenta, as tensões de Reynolds são avaliadas diretamente em função dos valores transportados da energia cinética turbulenta, como pode ser visto em [APS2001].
3. *os modelos de duas equações* resolvem uma equação para a energia cinética e outra equação para a taxa de dissipação da turbulência. Essas equações consideram, portanto, a escala de velocidade da energia cinética turbulenta e a escala espacial, que pode incorporar a vorticidade, a frequência e a dissipação correspondentes às flutuações turbulentas. Modelos que utilizam uma equação de transporte para a energia cinética turbulenta κ e outra para a dissipação da energia ε são do tipo κ - ε , e predizem bons resultados [BUR98a]. Implementações podem ser vistas em [VOL97], [MEI97] e [VIL2000].
4. *os modelos de duas e meia equações* consideram, além das equações de transporte do modelo de duas equações, o desvio de certos parâmetros que nos modelos anteriores, são considerados constantes. Um modelo de fechamento de turbulência dessa classe é o *Quasi-Equilibrium Turbulent Energy* (QETE), que é uma versão daquele de Mellor e Yamada ([MEL74] e [MEL82]), com a incorporação das funções de estabilidade desenvolvida por Galperin [GAL88]. Outros conjuntos de funções de estabilidade foram desenvolvidas por Rodi [ROD80]; Hossain [HOS80]; Kantha e Clayson [KAN94]; Burchard e Baumert [BUR95] e Canuto [CAN2000].

Modelos de turbulência de duas ou mais equações são os mais adequados para simular processos de turbulência e de mistura complexos. Suas funções de estabilidade têm um papel determinante na qualidade das predições desses modelos, no que concerne aos aspectos da validade física, eficiência computacional, robustez numérica e predictabilidade. Mas, em geral, empregam-se versões 1D ou 2D, pois modelos de turbulência 3D não são populares, ainda, pois os parâmetros desses modelos são difíceis de serem determinados, e porque ainda não se compreende bem como um fechamento completo da turbulência pode influenciar nos resultados do modelo de transporte/escoamento [VIL2000].

A turbulência está associada diretamente com a escala do problema e, portanto, deve-se diferenciar o cálculo dos coeficientes da direção horizontal dos da direção vertical, uma vez que os espaçamentos na horizontal são muito maiores que os espaçamentos na vertical. Além disso, em simulações hidrodinâmicas consideram-se os fenômenos de mesoescala, pois os de microescala são filtrados e incluídos na parametrização da turbulência [VLA99].

Sob essa situação, muitas vezes consideram-se constantes os coeficientes de viscosidade e de difusão do transporte horizontal, dado que a turbulência tem diferentes importâncias no padrão do escoamento e nos processos de mistura horizontais e verticais.

Neste trabalho consideram-se, por simplicidade, modelos de *zero equações* para obter os coeficientes de viscosidade e de difusão das EDPs. Porém, para uma melhor representação da turbulência, está sendo desenvolvido um modelo de *duas e meia equações* que considera o modelo de parametrização de Mellor-Yamada [MEL82] e as funções de estabilidade de Canuto [CAN2000].

2.7.2 Modelo de turbulência de zero equações

Na modelagem da turbulência deve-se levar em conta que as escalas dos fenômenos estudados são muito maiores na horizontal que na vertical refletindo, portanto, nas escolhas das respectivas malhas horizontal e vertical. Assim, aqueles fenômenos de tamanho inferior ao passo horizontal devem ser filtrados e incluídos nos coeficientes de viscosidade turbulenta horizontal como fenômenos de submalha. Nesse caso, como em [VLA99], a parametrização para a viscosidade cinemática turbulenta horizontal para as ESW 3D é determinada por:

$$\nu_h = \alpha L^{4/3}$$

onde a constante α varia entre $0,002 \text{ cm}^{2/3}\text{s}^{-1}$ e $0,01 \text{ cm}^{2/3}\text{s}^{-1}$, e toma-se $L \equiv \Delta x = \Delta y$.

No caso da ETM 3D, os coeficientes de difusibilidade turbulenta horizontal têm menos importância e podem ser considerados constantes. Nesse caso os valores são tais que, para $0 \leq k \leq 10$, tem-se:

$$\varepsilon_h \approx k$$

Para obter os coeficientes da viscosidade cinemática turbulenta vertical para as ESW emprega-se a mesma abordagem de [SHE83] e de [VLA99], na qual o problema de fechamento da turbulência reduz-se ao conhecimento da viscosidade turbulenta vertical. Nesse modelo, emprega-se a hipótese de Prandtl que relaciona a viscosidade turbulenta vertical com o gradiente médio das velocidades através do comprimento de mistura vertical L_m , como:

$$\nu_v = L_m^2 \sqrt{\left[(\partial u / \partial z)^2 + (\partial v / \partial z)^2 \right]}$$

O comprimento de mistura representa uma média da escala dos vórtices que interagem com os fluxos e dependem da existência de uma camada limite no fundo e na superfície. Pode ser obtido como $L_m = H \left[\xi(h/H) \Psi(R_i) \right]$ onde a estratificação é quantificada através do número de Richardson R_i , onde H é a profundidade total, e h é a distância até o fundo do corpo de água e onde $\Psi(R_i) = \exp(-\alpha R_i)$, com $\alpha \approx 0,8$. Algumas expressões para estimar a função $\xi(h/H)$, e uma comparação entre as parametrizações para o comprimento da mistura resultantes da escolha dessas funções, podem ser vistas no capítulo 3 de [VLA99]. Na seção 4.7 considera-se, para efeitos de implementação neste trabalho, uma expressão alternativa a L_m , como aquela empregada por [KIM94]. Uma discussão a respeito da física envolvida na parametrização semi-empírica da turbulência é como feita em [SHE83].

Os coeficientes de difusibilidade turbulenta vertical para a ETM podem ser estimados, considerando, conforme [SHE83] e [VLA99], uma expressão como:

$$\varepsilon_v = \nu_v \beta(R_i)$$

onde $\beta(R_i) = \gamma \exp(-\gamma' R_i)$, onde $\gamma' = 1,2$ e $\gamma = 1,4$. Quando o corpo de água é fortemente estratificado tem-se $R_i > 0,8$; quando $0,08 < R_i < 0,8$, ele é parcialmente estratificado; e quando $R_i < 0,08$, o corpo de água é bem misturado.

2.7.3 Condições de contorno

O modelo de parametrização de turbulência deve levar em conta as condições de contorno do modelo hidrodinâmico e do modelo de transporte de substâncias, pois neles são fornecidos as CC do modelo de parametrização de turbulência, já que estas CC são análogas às empregadas nesses modelos.

Note-se que as CC da ETM são obtidas considerando-se que não há fluxo na superfície livre e na interface água-sedimento, já que não há água e/ou sedimentos saindo ou entrando nessas fronteiras. As CC para as ECQMs são obtidas observando que, analogamente, não há fluxo nessas interfaces. As expressões específicas para essas CC são as discutidas nas seções 2.3.1 e 2.4.1, para o caso do escoamento, e na seção 2.5.1 para o transporte escalar. Desse modo, na seção 4.6 especificam-se as condições de contorno para o modelo de turbulência discreto empregando o modelo discutido na seção 2.7.2 e usando as CC estabelecidas para o modelo de escoamento.

2.8 Estudo de caso: Lago Guaíba

A principal motivação desta tese, enquanto aplicação, é a de simular os complexos processos hidrodinâmicos e de transporte de substâncias que ocorrem em corpos de água, particularmente aqueles de lagos, regiões costeiras e estuários, dado que é essencial conhecer a hidrodinâmica local que controla o transporte de substâncias do corpo de água. Por exemplo, nos sistemas estuarinos os processos de mistura, circulação e estratificação são regidos por influências ditadas pela geometria, salinidade e padrões de circulação da região costeira adjacente, e basicamente por três forçantes: a descarga de água doce, as correntes de maré e a transferência de quantidade de movimento através do cisalhamento do vento na superfície livre ([CAM99] e [MAN2001]).

Nesse último forçante há um acoplamento do atrito entre o vento e o corpo de água, onde a energia do vento é convertida para correntes através do atrito entre o ar em movimento e a superfície da água. A energia cinética da água na superfície é transferida verticalmente para baixo através do atrito dentro da coluna da água. Outros fatores importantes para determinar a hidrodinâmica são a inclinação da água (gradiente de pressão horizontal) e o efeito de Coriolis.

O gradiente de pressão horizontal age, colocando em movimento o fluido em direção às regiões de baixa pressão, e surge a partir dos gradientes espaciais da pressão na superfície da água (o gradiente de pressão barotrópico) e a pressão no interior do corpo de água (gradiente de pressão baroclínico). O modo barotrópico não leva em conta os gradientes de salinidade e de temperatura.

No caso dos forçantes eles condicionam a circulação gravitacional (modo barotrópico e modo baroclínico), a circulação residual e as correntes geradas pelo vento, sendo que geralmente um desses padrões de circulação predomina em um determinado sistema, mas os dois ou os três tipos podem ocorrer simultaneamente ou sazonalmente em um mesmo estuário [MAN2001]. Isso significa que nesses corpos de água, o modelo dinâmico deve considerar os movimentos rápidos, causados pela ondas de gravidade externa (barotrópica), e os movimentos lentos, causados pela ondas de gravidade internas (baroclínica) de correntes.

No modo baroclínico de movimento, as correntes fluem num sentido na camada superficial e no sentido oposto na camada inferior. Tal estruturação ocorre principalmente durante intrusões de massas de água, sendo que a camada superficial é a de domínio da água costeira, e a profunda é o domínio da água do mar. No caso de ausência de variação de densidade o gradiente de pressão horizontal é causado pela inclinação da superfície. Esta componente do gradiente de pressão é chamada de gradiente de pressão barotrópico. A variação de densidade (efeitos termais) contribui para o gradiente de pressão. Esta componente do gradiente é chamada de gradiente de

pressão baroclínico. Os gradientes de pressão barotrópicos e baroclínicos formam o gradiente de pressão horizontal.

Note-se, porém, que nos sistemas estuarinos o transporte de substâncias pode nem sempre ocorrer na mesma direção do fluxo principal de água, devido aos complexos processos advectivos e difusivos de mistura. A dispersão e advecção de massas de água e de suas propriedades decorre basicamente de processos que envolvem o movimento oscilatório das marés, gradientes de densidade, cisalhamento lateral entre massas de diferentes velocidades de corrente, atrito com o fundo, e difusão turbulenta [MAN2001].

Além disso, os processos de erosão e ressuspensão apresentam particular importância na ciclagem de sedimentos em estuários, pois promovem um aumento da carga de sedimento em suspensão (aumento do gradiente vertical), de forma a intensificar o transporte advectivo e atuando na manutenção da zona de máxima turbidez estuarina [DYE88]. Também durante um ciclo de maré (marés enchente e vazante) pode-se observar uma forte relação entre as velocidades das correntes e o transporte e as concentrações de sedimentos em suspensão [MAN2001]. E, os efeitos episódicos de alta energia como, por exemplo, tempestades ou períodos de extremada vazão, apresentam particular importância, pois podem movimentar, em um curto período de tempo, uma carga sedimentar superior à verificada durante longos períodos em condições normais [DYE88].

Para obter quantidades físicas como a elevação da superfície livre, velocidade do escoamento, salinidade, coeficientes de viscosidade cinemática turbulenta vertical, coeficientes de difusão turbulenta vertical, pressão hidrodinâmica, coeficiente de atrito no leito (*bottom shear stress*), coeficiente de atrito do vento (*wind shear stress*) é necessário calcular, com o modelo hidrodinâmico (ESW), os processos como: a aceleração local (inércia), aceleração advectiva, aceleração de Coriolis, gradiente de pressão barotrópica, gradiente de pressão baroclínica, pressão hidrostática e não hidrostática, difusão turbulenta nas direções horizontal, viscosidade turbulenta vertical influenciada pela estratificação da densidade, atrito no leito do corpo de água, atrito do vento.

No caso da equação do transporte escalar de substâncias (ETM) é necessário calcular os processos: da taxa local da variação de concentração, taxa advectiva da variação da concentração, difusão turbulenta nas direções horizontal, viscosidade turbulenta vertical influenciada pela estratificação da densidade, viscosidade turbulenta horizontal é parametrizada usando coeficientes constantes.

A aplicação específica para o modelo computacional multifísica aqui desenvolvido, o estudo de caso, é o Lago Guaíba. A opção por utilizar o Lago Guaíba como estudo de caso deve-se ao fato de que existem alguns dados da geometria e de parâmetros físicos para definir as condições iniciais e de contorno do problema de valor inicial e de contorno. Além disso, existem modelos seqüenciais como o de Casalas [CSL85], que estão calibrados e serviram de *benchmark* para orientar os primeiros modelos paralelos desenvolvidos e implementados. Note-se, porém, que como o modelo computacional somente foi verificado por experimentos numéricos específicos, não sendo, portanto, feita a calibração e validação do modelo, aqui não são apresentados resultados específicos da simulação de eventos do Lago Guaíba.

O Lago Guaíba se situa entre o chamado delta do Jacuí e a Lagoa dos Patos, que é o mais extenso corpo de água do sistema lagunar. Banha toda a região metropolitana de Porto Alegre e tem uma área de aproximadamente 468 km² e profundidade média de 4,0m. Recebe contribuição das águas do delta do Jacuí (formado pela confluência dos rios Jacuí, Caí, Sinos e Gravataí). Com uma extensão de aproximadamente 50 km, o Lago Guaíba deságua na Lagoa dos Patos e apresenta seções de até 15 km de largura ([SIL86] e [MEN98]). Situado entre os meridianos 50° e 55° Oeste e paralelos 28° e 35° Sul, ele é importante para o transporte fluvial, a irrigação e o abastecimento de água

das cidades da região, mas sofre constantes despejos de poluentes industriais e domésticos.

O Lago Guaíba, a Lagoa dos Patos e a Lagoa Mirim são corpos de água com características bidimensionais que, interligados, formam o sistema costeiro lagunar da Bacia de Sudeste. As Lagoas dos Patos e Mirim possuem áreas aproximadas de 10.000 e 4.200 km², e profundidades médias da ordem de 7,0m, respectivamente. O regime hidrodinâmico desse sistema lagunar, no qual o Guaíba está inserido, é complexo tanto na época de estiagem como na época de cheias. No período das cheias a Lagoa dos Patos recebe, em média, 3.000m³/s do Lago Guaíba e no período de secas a contribuição média é de 800m³/s. A fig. 2.2 ilustra uma imagem realística do Lago Guaíba. É importante destacar que, apesar de o modelo computacional ser aplicado ao Lago Guaíba, ele independente da aplicação.

Assim, um modelo computacional de alta resolução para regiões altamente complexas, como a mostrada na fig. 2.2, permite a simulação de vários cenários como, por exemplo, os fenômenos de transporte de sedimentos como areia, sal, óleo, entre outros. Desse modo, os modelos hidrodinâmicos e de transporte de substâncias podem ser empregados para o planejamento de atividades portuárias, de dragagens de canais e de obras de engenharia, etc. Especificamente, pode-se empregar esses modelos para [CAM99]:

1. o estudo da batimetria do corpo de água a fim de verificar as possíveis variações que o regime hidrodinâmico sofrerá quando da modificação desta batimetria em função de atividades como, por exemplo, de aprofundamento de canais de navegação;
2. a simulação da intensidade da poluição provocada por derramamentos de produtos químicos quando de acidentes com navios, oleodutos, atividades portuárias, etc.;
3. a simulação de pluma de poluentes ao longo do tempo para identificar as principais regiões potencialmente atingidas;
4. a simulação dos efeitos de lançamentos de emissários para a definição de pontos de lançamentos, além da verificação do desempenho de emissários submarinos de esgotos, entre outros;
5. a simulação de problemas que possam surgir do transporte de sedimentos e áreas de erosão em canais de navegação e pontos de lançamentos de lodo e/ou de resíduos sólidos;
6. o estudo das alterações nas correntes fluviais e/ou marítimas decorrentes de obras portuárias ou costeiras, tais como molhes, cais, rampas, etc.

Portanto, dado suas potencialidades econômicas, as margens dos estuários, lagos e regiões costeiras são densamente povoadas e sofrem influência direta ou indireta de diversas atividades antrópicas como, por exemplo, urbanas, recreativas, portuárias, industriais, pesqueiras, etc., desenvolvidas ao seu redor ou ao longo de sua bacia de drenagem. Um manejo racional demanda o entendimento dos vários processos ecológicos, físicos e químicos envolvidos com a hidrodinâmica e o transporte de substâncias nesses corpos de água.

Essas questões são os principais elementos motivadores desta tese, enquanto aplicação. Note-se, porém, que enquanto pesquisa aplicada em Computação Científica Paralela, o objetivo deste trabalho é desenvolver *esquemas numéricos e estratégias de solução paralelas* apropriados, sob o ponto de vista numérico e computacional, à construção de modelos computacional paralelos multifísica. Nesses modelos diferentes fenômenos físicos, dependentes ou não entre si, são simulados no mesmo domínio de definição, e de modo que suas escalas espaço-temporais são, geralmente, distintas. Esse é a situação da hidrodinâmica 3D e do transporte escalar de substâncias 3D como tratado neste trabalho.

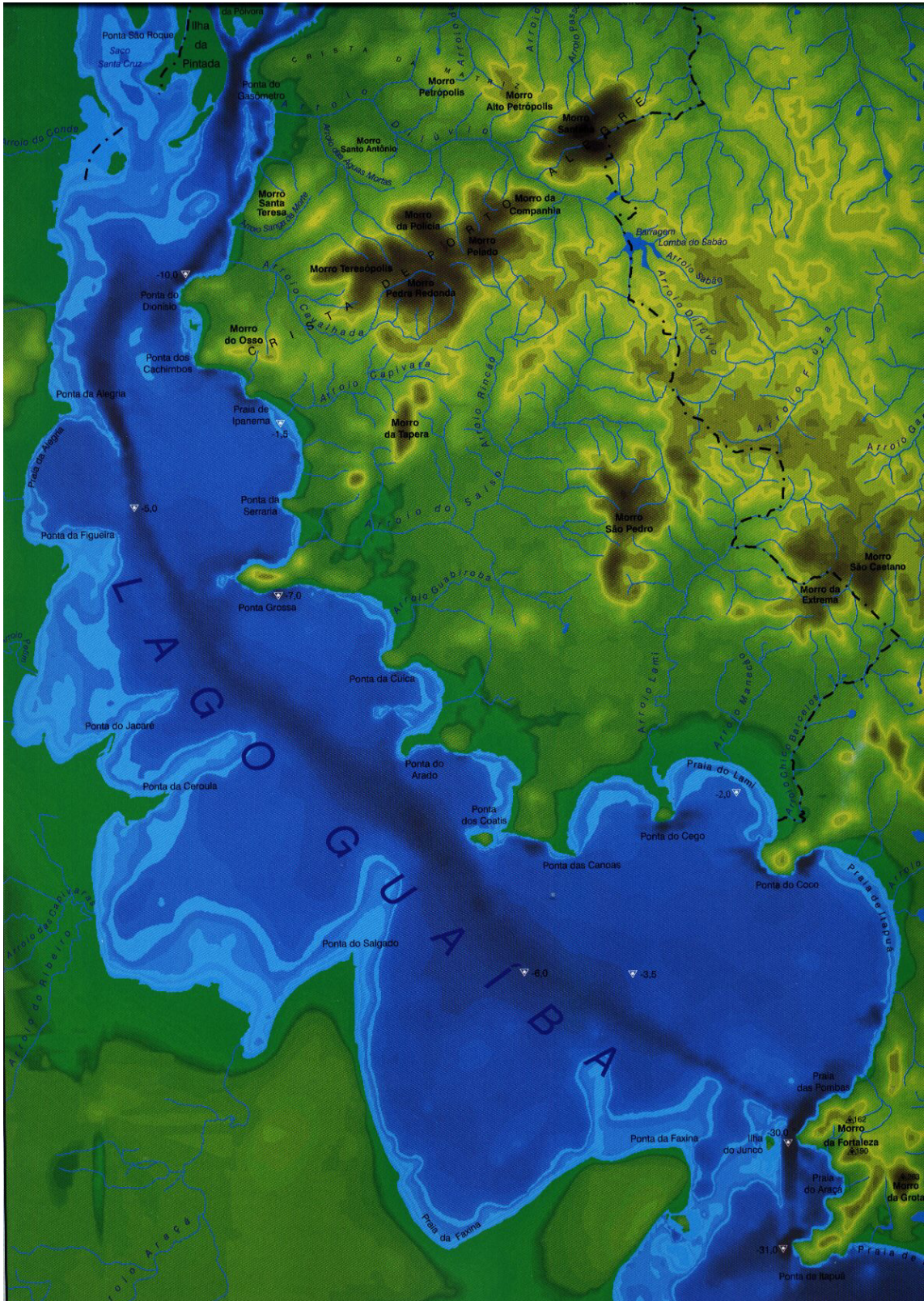


FIGURA 2.2 - Imagem do Lago Guaíba obtida via varreduras feitas por sonar (extraído de [EAS98])

3 Sistemas de coordenadas, valores nodais e malha

A malha na qual as equações são efetivamente resolvidas é chamada de domínio computacional ou malha numérica. Transformações (mapeamentos algébricos) podem transformar o domínio físico arbitrário em um domínio computacional regular. Esses mapeamentos são obtidos quando se empregam os métodos de transformações e de geração de malhas ([THM86], [KNU93] e [MAL95]). Alguns modelos hidrodinâmicos com malhas construídas em coordenadas generalizadas são os de [SHE83], [SHE89] e [CHU96].

Em uma abordagem paralela, a malha é particionada e distribuída entre os processadores disponíveis. Essa divisão do domínio é efetuada por algoritmos de particionamento que partem do princípio que o domínio computacional pode ser visto como um grafo. Aplicando-se algoritmos de particionamento de grafos pode-se reduzir as comunicações entre os subproblemas e balancear a carga entre os processadores. Além disso, quando se considera a plataforma disponível necessita-se, no caso de uma arquitetura heterogênea, de estratégias para mapear o grafo da aplicação sobre o grafo da arquitetura.

Essas questões foram estudadas na dissertação de Elias César Araújo de Carvalho, que desenvolveu seu trabalho no GMCPAD do PPGC-UFRGS, “Algoritmos e *Software* para Particionamento de Domínios e Mapeamento em *Clusters* Heterogêneos” [CAR2002], [CAR2002a] e são, também, discutidas no capítulo 7. Nesse capítulo, mais especificamente na seção 3.5, se caracterizam os procedimentos que estão relacionados com a geração e o armazenamento do domínio (dos subdomínios particionados) e cujos sistemas de equações são gerados de modo distribuídos.

3.1 Malha horizontal: espaço-alternada

Malhas numéricas são caracterizadas tendo em vista a posição relativa das variáveis dependentes em relação a essas malhas. São possíveis vários arranjos dessas variáveis, mas, para sistemas de coordenadas ortogonais, dois deles são os mais comuns: o arranjo co-localizado e o alternado (ou desencontrado ou espaço-alternado) ([MAL95], [MES98] e [BRL2000]).

A escolha pelo uso de malha espaço-alternada deve-se ao fato de que as EDPs tratadas neste trabalho são as equações *shallow water* (ESW) e a equação do transporte de substâncias (ETM). Para essas EDPs, pode-se mostrar, como em [MES76], [WAS86] e [MES98], que a malha de Arakawa tipo *C* é apropriada para a construção de esquemas conservativos via abordagem fluxo numérico. Além disso, ao desacoplar as variáveis que, então, não necessitam ser definidas no mesmo ponto, reduz-se o tempo de processamento mantendo, o erro de truncamento como em malhas não alternadas. Uma desvantagem da malha espaço-alternada, em relação a malha com arranjo co-localizado, é a complexidade adicional na implementação computacional, sobretudo para problemas 3D e para suas condições de contorno.

As variáveis dependentes, cujas posições relativas na malha de Arakawa *C* são como mostrado na fig. 3.1, têm os seguintes significados:

1. u e v são as velocidades horizontais da água nas direções X e Y ;
2. η é o nível da água acima de um dado valor de referência.

Além disso, as batimetrias $h = H_u = H_v$, que são as profundidades da água abaixo de um certo nível de referência, são posicionadas, respectivamente, com as componentes u e v da velocidade horizontal. Já o coeficiente de Chezy C_z é posicionado no centro da célula, com o nível de água. A concentração de substâncias S é posicionada no centro da célula.

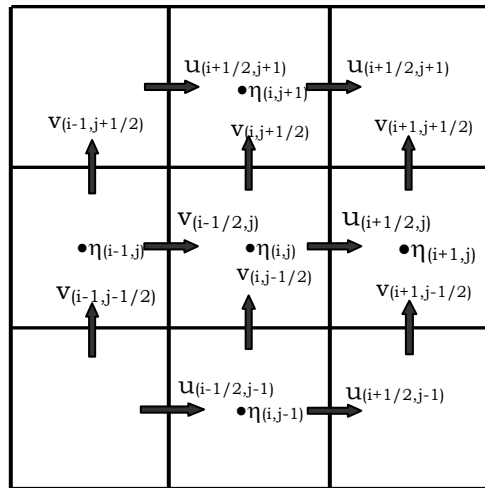


FIGURA 3.1 - Malha espaço-alternada Arakawa tipo C

Apresenta-se a seguir uma ilustração referente à aplicação ao Lago Guaíba dos modelos computacionais desenvolvidos neste trabalho, objetivando destacar que a topografia do leito do Lago Guaíba é complexa, indo de profundidades de 18m até poucos centímetros (5 cm no nosso estudo de caso). Uma representação para essa complexa batimetria é aquela mostrada na fig. 3.2, que é gerada por uma malha de 880 pontos na direção X por 660 pontos na direção Y . Todos os pontos são eqüidistantes, pois tomou-se na malha horizontal nesse exemplo como $\Delta x = \Delta y = 50\text{m}$ (arquivo Guaiba_50, que é aquele arquivo onde se toma $\Delta x = \Delta y = 50\text{m}$ na discretização da geometria).

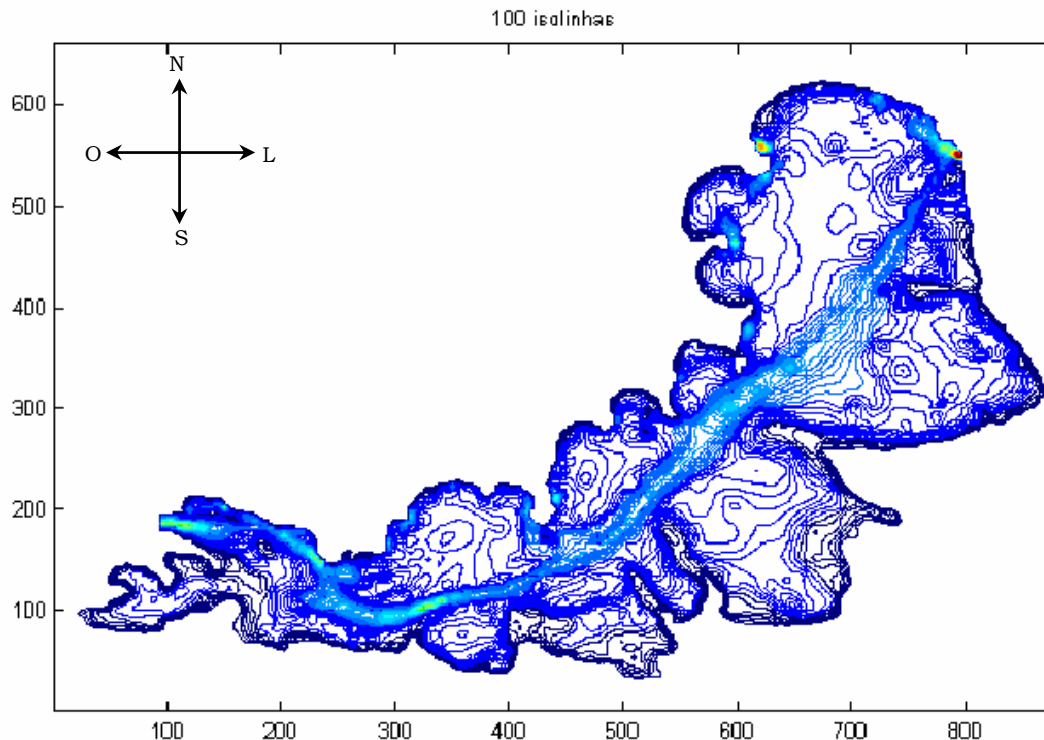


FIGURA 3.2 - Batimetria do Lago Guaíba representada por 100 isolinhas

A batimetria, mostrada na fig. 3.2, foi obtida a partir de uma malha de pontos resultante de processos de varreduras por sonar, combinados com procedimentos de interpolação para obter os pontos não coletados no processo de varredura. Essa imagem foi gerada utilizando o pacote IDRISE®, que é um sistema de informação

geográfica para processamento de imagens desenvolvido pela *Graduate School of Geography* da *Clark University*. Seu representante no Brasil é o Centro de Recursos Idrise, que está vinculado ao Centro de Ecologia do Instituto de Biociências da UFRGS [EAS98]. A fig. 3.2 foi gerada empregando a função *contour* do MATLAB®, a partir da malha de pontos gerada pelo IDRISE®.

Note-se que, quando da aplicação do modelo computacional ao Lago Guaíba, é necessário, para efeitos de cálculo, definir a localização e a orientação da malha de cálculo na superfície terrestre. Fixou-se a latitude média de 30° sul e o ângulo de 242° (sentido horário) entre o eixo OY da malha e o norte geográfico [SIL86]. Mas o indicador de orientação (pontos cardeais) da fig. 3.2 é somente para chamar a atenção para a diferença de orientação dessa figura em relação as outras figuras análogas como, por exemplo, as fig. 2.2, 3.12, 7.2, 7.3 e 7.5.

3.2 Malha vertical

A malha numérica da fig. 3.1 refere-se somente ao plano horizontal, mas em modelos 3D deve-se definir uma malha vertical. Nesse caso, a obtenção dessa malha vertical impõe que se escolha entre:

1. malhas fixas em coordenadas $-z$, como em [LEE89], [CAS92] e [KIM94];
2. malhas sigma (coordenadas $-\sigma$) que seguem o terreno, como em [SHE83], [SON94], [SHE95], [LIY96], [MEL98a] e [VIL2000];
3. malhas r – adaptativas, que redefinem os nodos de acordo com critérios de adaptatividade, como em [FOR99];
4. malhas *isopycnal* (coordenadas $-\rho$), que são as geradas por camadas com densidade uniforme, como em [CAS97], [BLE98] e [CON98];
5. malhas fixas em coordenadas $-\eta$ (coordenadas eta), como em [BLA94] e [KAL99], que aproximam a topografia “em blocos”.

A escolha por esse ou aquele sistema de coordenadas vertical decorre da necessidade de desenvolver esquemas numéricos que considerem o caráter multidimensional do corpo de água e que incorporem a topografia do leito desse corpo de água. Alguns dos problemas que aparecem na simulação do escoamento multinível devem-se à incorporação da topografia do terreno ao modelo, sendo que os problemas dependem da escolha da coordenada vertical introduzida nas equações dos modelos [MES84].

Três das coordenadas verticais mais populares são coordenada $-z$, a coordenada $-\rho$ e a coordenada $-\sigma$. A fig. 3.3 representa três regimes de escoamento associados com uma dada coordenada vertical. A camada de superfície (misturada) é representada usando coordenadas $-z$. O interior do corpo de água é representado usando coordenadas $-\rho$; e a camada do fundo é representada usando coordenadas $-\sigma$. A discussão neste trabalho a respeito desses regimes e de suas representações através dos sistemas de coordenadas (x, y, z) , (x, y, σ) e (x, y, ρ) segue aquela feita por [VLA99] e por [GRI2000].

No primeiro regime, há uma camada de misturas de superfície que geralmente é turbulenta e dominada pela transferência de quantidade de movimento, calor, água doce, etc. Esse regime geralmente é do tipo *bem misturado* na vertical através de processos convectivos turbulentos 3D, e os processos envolvidos requerem física não hidrostática que necessita de alto grau de resolução vertical e horizontal para boa representação.

No segundo regime, ocorrem processos de transporte de substâncias (*traces*) no interior do corpo de água, que ocorrem predominantemente ao longo de direções de

densidade constante. Nesse regime as propriedades das massas de água tendem a ser preservadas ao longo das escalas espaciais e temporais.

O fundo do corpo de água é representado no terceiro regime e age como um forte forçante para acelerar as correntes de água, o qual influencia diretamente o balanço dinâmico. Em um corpo de água não estratificado, o escoamento geralmente segue linhas com constantes Φ/H , onde Φ é a força de Coriolis, e H , a profundidade. No caso de ser estratificado, a escala vertical efetiva é reduzida, e o controle sobre a dinâmica é reduzido. Também existem regiões nas quais a intensidade das correntes (*overflows*) e os processos da camada de contorno turbulenta no fundo (*Bottom Boundary Layer - BBL*) agem para determinar as características da massa de água.

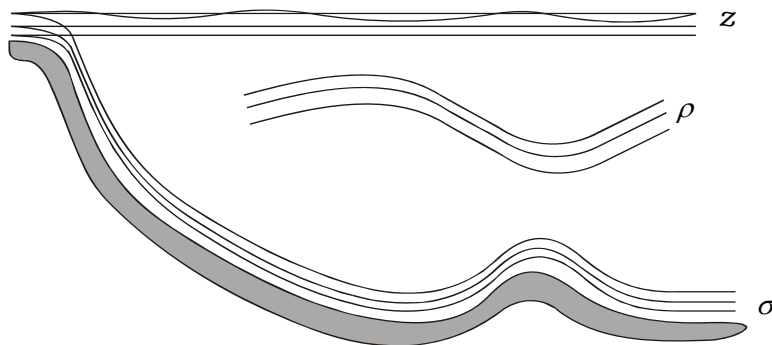


FIGURA 3.3 - Três regimes de escoamento associados com uma dada coordenada vertical (adaptado de [GRI2000])

Outra consideração que é relevante para a escolha das coordenadas é relacionada com a importância do balanço hidrostático e geostrófico para fluidos geofísicos. Visando a obter acurácia numérica e simplicidade analítica, é usual descrever esses balanços com um conveniente conjunto de coordenadas.

Uma escolha para a coordenada vertical é a coordenada $-z$, que representa a distância vertical de uma superfície de referência em $z = 0$ até $z = -h(x, y)$, que é o leito do corpo de água. Algumas das vantagens dessa classe decorrem, [GRI2000]:

1. da simplicidade da discretização numérica que permite que esses modelos sejam usados logo após seu desenvolvimento inicial;
2. do fato de que, para fluidos de Boussinesq, o gradiente de pressão horizontal pode ser facilmente representado;
3. da simplicidade e da acurácia na representação da *equação do estado* para a água, que é não linear e importante na determinação das propriedades da massa da água;
4. do fato que a camada de mistura da superfície (*surface mixed layer*) é naturalmente parametrizada usando coordenadas $-z$;
5. e que, em geral, coordenadas $-z$ fornecem um modelo útil para representar processos diabáticos.

Algumas das desvantagens de modelos $-z$, ou seja, de modelos em coordenadas $-z$, são:

1. o fato de que a advecção e a difusão de substâncias dissolvidas (*tracer*) ao longo de superfícies de densidade inclinada no interior de corpo de águas não são bem representadas;
2. a representação e a parametrização da camada de contorno turbulenta no fundo (BBL) é não natural;
3. a representação da topografia é difícil e não natural.

Outra escolha para a coordenada vertical são coordenadas $-\rho$ *isopycnal* que consideram a densidade potencial referenciada com uma dada pressão. Em um corpo de água adiabático estratificado estável, a densidade potencial é conservada materialmente e definida como uma camada monotônica de fluido. Algumas das vantagens dessa classe de coordenadas são ([CAS97] e [GRI2000]):

1. o transporte de substâncias no interior do corpo de água tem uma forte tendência para ocorrer ao longo das direções definidas pela densidade potencial localmente referenciadas (ou seja, direções neutras), em vez de através. Assim, esses modelos são apropriados para representar a dinâmica desses regimes, pois as *isopycnais* são razoavelmente paralelas às direções de densidade constante (neutras), significando que essa coordenada é apropriada para escoamentos estáveis e estratificados;
2. a topografia do fundo é representada de modo a evitar a necessidade de distinguir o fundo do lado como feito por modelos $-z$;
3. as pesquisas atuais sobre a física de *overflows* sugerem que modelos $-\rho$ podem fornecer alguma vantagem sobre modelos $-z$ na representação desses processos;
4. em fluidos adiabáticos, o gradiente de pressão horizontal pode ser facilmente representado. Além disso, para fluidos adiabáticos, o volume (fluido de Boussinesq) ou massa (para fluido não Boussinesq) entre *isopycnais* é conservado.

Algumas desvantagens de modelos $-\rho$, ou seja, de modelos em coordenadas $-\rho$, são:

1. a má representação dos efeitos de equações não lineares;
2. o fato de que a coordenada $-\rho$ gera um modelo inapropriado para a representação de camadas misturadas de superfície ou BBL, pois essas camadas de contorno são geralmente não estratificadas.

Uma terceira escolha para a coordenada vertical é a coordenada $-\sigma$ que segue o terreno (*terrain-following*), introduzida por Phillips em 1957 [MES98]. Essa coordenada é monotônica e define um mapeamento (transformação biunívoca) entre a profundidade e o sistema coordenadas $-\sigma$. Algumas das vantagens dessa escolha são [GRI2000]:

1. fornecem uma representação suave para a topografia do corpo de água, com isolinhas coordenadas concentradas em regiões onde processos BBL são mais importantes, fornecendo um modelo natural para parametrizar processos BBL;
2. os efeitos termodinâmicos associados com a equação do estado são bem representados;

Algumas desvantagens de modelos $-\sigma$, ou seja, de modelos em coordenadas $-\sigma$, são:

1. as camadas de superfície misturadas (*surface mixed layer*) não são tão bem representadas como em coordenadas $-z$. Isso decorre do fato de que as distâncias verticais entre os pontos da malha são, geralmente, crescentes na direção de regiões continentais e, portanto, levando a camadas de superfícies que têm potencialmente menos resolução vertical que a ideal;
2. como em modelos $-z$, a representação da advecção e da difusão ao longo de superfícies de densidade inclinada no interior do corpo de água não é boa. Além disso, é difícil a “re-orientação” da física em escala submalha ao longo de direções de densidade constante (neutras), pois quando a topografia é muito íngreme, as superfícies em coordenadas $-\sigma$ podem cruzar as direções de densidade constante (neutras) em grandes ângulos;

3. modelos $-\sigma$ também não são bons para representar de modo acurado o gradiente de pressão horizontal, dado que as superfícies σ – constante não são, geralmente, horizontais. Assim, o gradiente de pressão horizontal, que é perpendicular à direção vertical local definida pela gravidade, tem uma projeção e cruza a superfície coordenada $-\sigma$. Isso provoca o aparecimento de forças de pressão puramente numéricas que produzem correntes não físicas.

Note-se, então, que não há esquemas de discretização do gradiente de pressão e do cálculo do geopotencial pela equação hidrostática que estejam livres desse erro e que não resulte em um falso gradiente de pressão. Para superar parte desses problemas, Mesinger [MES84] definiu um novo sistema de coordenadas, as coordenadas $-\eta$, no qual as montanhas consistem em malhas tridimensionais tipo blocos (*grid boxes*). Veja em [MES84] ou [BLA94] a expressão matemática e a representação geométrica dessa coordenada.

Nesse sistema, as coordenadas verticais são definidas de tal modo que as superfícies permanecem em elevações fixas nos lugares onde tocam e definem a superfície do solo. O sistema em coordenadas $-\eta$ é construído a partir de um sistema de pressão de referência que considera a coordenada $-\sigma$ e as coordenadas x e y . Assim, quando a pressão é função somente da altura geométrica z , e até um nível que inclui as montanhas mais altas, essa função é igual à pressão de referência, e as superfícies de coordenada $-\eta$ são horizontais.

Então, para qualquer escolha razoável do esquema numérico, não haverá erro no gradiente de pressão. Essa é uma característica que somente pode ser obtida no sistema coordenada $-\sigma$ para condições muito restritas, nas quais a inclinação das superfícies coordenadas definidas coordenadas $-\eta$ será pequena, e erros do gradiente de pressão serão reduzidos [MES84].

Portanto, modelos em coordenada $-\eta$ têm como vantagem o fato de que as superfícies coordenadas são aproximadamente horizontais, reduzindo os erros nos cálculos de variáveis obtidas a partir de derivadas horizontais, já que esses erros são significativos em regiões de montanhas íngremes. A topografia do modelo é representada em forma de degraus cujo topo coincide com a interface das camadas. A altura de cada degrau é obtida através do valor médio das alturas máximas da topografia dentro de cada quadrado de grade. Cada degrau possui um ponto de massa no centro e quatro pontos de velocidade localizados nos vértices [KAL99]. Com essa escolha, a malha vertical é a Arakawa tipo *E*.

3.2.1 Malha vertical e representação da topografia

A representação da topografia do fundo em modelos ρ e σ é acurada e não impõe distinção entre o fundo e os lados do corpo de água, permitindo a completa representação do contorno. O uso de modelos em coordenadas $-z$ impõe essa distinção uma vez que são usadas células retangulares com espessura fixa no fundo. Porém, algumas representações da topografia de modelos $-z$ podem superar alguns dos obstáculos ao uso de coordenadas $-z$ para representar topografias complexas. Essas consideram, além da tradicional representação em células cheias (*full cell*), duas abordagens mais apropriadas que são as células cortadas (*shaved cell*) e as células parciais (*partial cell*) [GRI2000].

Na abordagem célula cheia (*full cell*) as espessuras verticais de todas as células são independentes da latitude e da longitude e, portanto, as células são completas. Na abordagem célula parcial (*partial cell*), as espessuras verticais das células podem variar de acordo com a topografia. Nesse caso, as alturas podem não preencher completamente as células, mas continuam sendo retangulares. Na abordagem *shaved*

cell, as células do fundo são lineares por partes com a topografia, permitindo diferentes alturas para os diferentes lados das células.

A abordagem *full cell* é a que produz mais erros. A *partial cell* simplifica a abordagem computacional. Além disso mostra-se que essa abordagem representa a topografia melhor que a abordagem *full cell*, mas não tão bem como a *shaved cell*. Mas duas importantes características são mantidas tanto pelas células cortadas e como pelas células parciais.

A primeira característica refere-se ao refinamento da resolução horizontal. Quando se refina a resolução horizontal sem alterar a resolução vertical, não há melhora na representação da topografia via células cheias, mas a representação via células parciais ou cortadas é significativamente melhorada. A segunda característica refere-se ao refinamento da resolução vertical. Quando a resolução vertical é refinada, mas a resolução horizontal é fixa, os modelos que usam células parciais mostram menos variações quando comparados com aqueles com células cheias. Isso significa que a simulação com células parciais é mais robusta que aquela com células cheias. E a robustez do modelo com esses refinamentos na resolução vertical é importante, devido a sensibilidades em processos *overflow*.

Assim, devido à “natureza $-\sigma$ ” das células parciais ou cortadas, analogamente ao emprego da coordenada $-\sigma$, aparecem dois termos quando do cálculo dos gradientes de pressão horizontal. Porém, em contraste com modelos em coordenadas $-\sigma$, os problemas gerados por esses termos são menores em modelos $-z$. Uma análise desses termos, e de seus resultados, mostra que o erro do gradiente de pressão para células parciais de modelos em coordenadas $-z$ pode ser desprezado.

Assim, o desenvolvimento e a implementação de modelos em “células parciais” ou em “células cortadas” em coordenadas $-z$ melhora a representação das complexas topografias. Além disso, se considerada a questão computacional, a abordagem em células parciais é particularmente interessante, dado que concilia a simplicidade computacional com uma boa representação da topografia. Portanto, nos modelos 3D desenvolvidos, a escolha recaiu na coordenada $-z$, onde o sistema de coordenadas definido é o (x, y, z) , com células parciais. Com essa escolha, a topografia é reduzida a uma série de paralelepípedos, e o fundo contínuo é representado no plano $X-Z$ como nas fig. 3.4 (a) e (b) e cuja localização das variáveis dependentes pode ser vista na fig. 3.6.

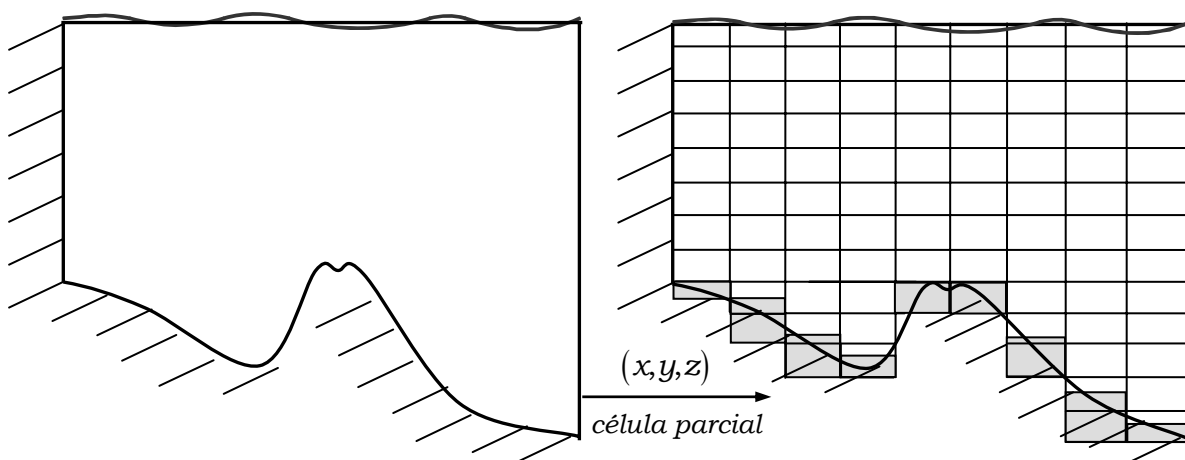


FIGURA 3.4 - (a) Domínio físico; (b) domínio computacional em coordenadas Cartesianas no plano $X-Z$

Assim, neste trabalho empregou-se nas discretizações que a malha 3D é a Arakawa C em coordenadas $-z$. O posicionamento das variáveis nessas células e, portanto, como representado na fig. 3.5. Em função dessa escolha, a malha 3D consiste de células definidas por $(\Delta x, \Delta y, \Delta z_k^n)$. Se as alturas das faces da célula não são completamente formadas de água, como no caso de algumas células adjacentes ao leito e/ou da superfície livre, tomam-se como espaçamento vertical as alturas da coluna de água nessas faces.

Em uma célula 3D, é necessário especificar seis vetores velocidades, um para cada face, cada qual com três componentes de velocidade (direção X , direção Y , direção Z). Assim, foram especificadas as dezoito componentes, usando as condições de deslizamento completo, de deslizamento parcial e a condição que não há penetração. Para especificar essas CC, observa-se que, na malha horizontal, os níveis de água são definidos no centro da célula, mas as componentes (u, v, w) da velocidade estão definidas, respectivamente, na face direita, de trás, e superior da célula, conforme ilustrado pela fig. 3.5.

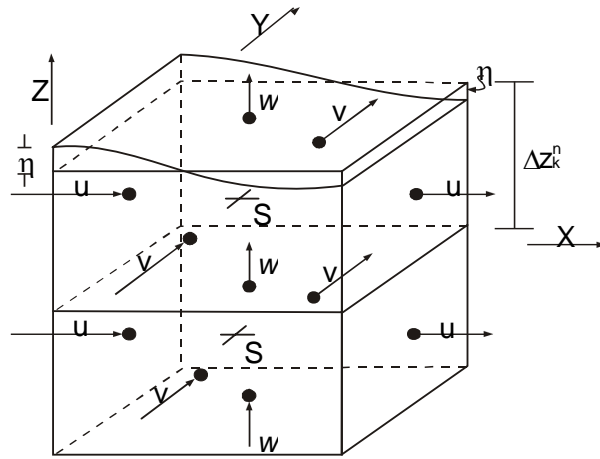


FIGURA 3.5 - Célula 3D no sistema de coordenadas (x, y, z) e considerando os espaçamentos $(\Delta x, \Delta y, \Delta z_k^n)$ (adaptado de [LEE89])

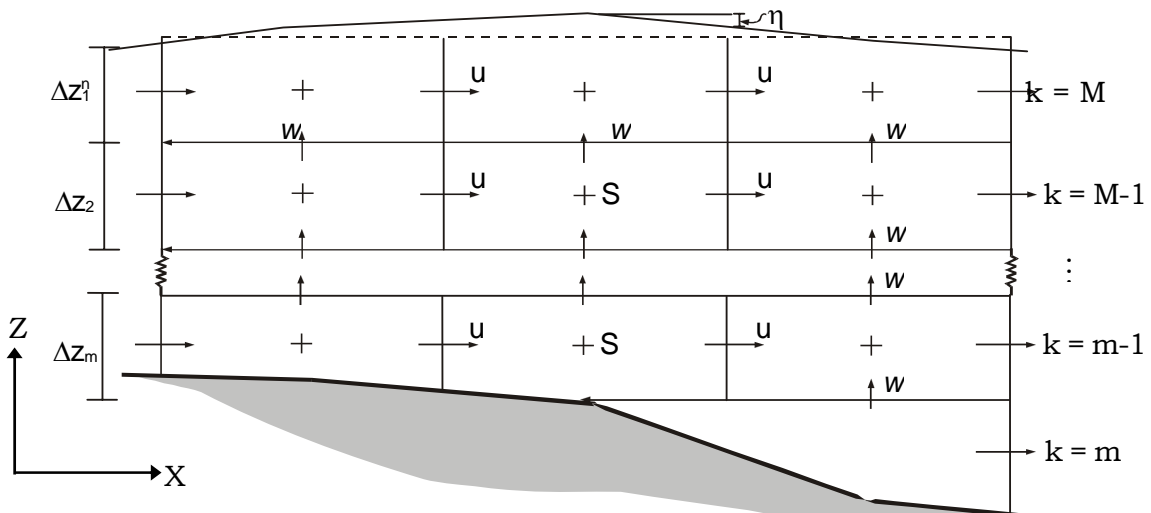


FIGURA 3.6 - Localização das variáveis dependentes no plano $X-Z$

Note-se que o espaçamento vertical na última camada (a primeira camada de cima para baixo na coluna de água) é denotado por Δz_k^n e depende da posição da camada k em relação à superfície livre que pode variar no nível de tempo t^n . Nesse caso, no cálculo dos espaçamentos entre as diversas camadas que compõem o vetor $\Delta Z = (\Delta z_M^n, \dots, \Delta z_m^n)^T$, deve-se considerar o nível de água. Para melhorar a estabilidade numérica, os espaçamentos Δz_M^n e Δz_m^n , das camadas superior e inferior, são menores que os espaçamentos das camadas interiores. Assim, denotando os espaçamentos nas camadas superior e inferior por Δz_M^n e Δz_m^n , a altura total por H_T , o número de camadas por N , e a razão entre as camadas por D , as alturas das camadas são tais que $\Delta z_M^n = \Delta z_m^n = \Delta z_h / D$, e são calculadas como:

$$\frac{\Delta z_h}{D} + \frac{\Delta z_h}{D} + \Delta z_h (N - 2) = H_T, \text{ ou seja, } \Delta z_h = \frac{DH_T}{2 + D(N - 2)}$$

Neste trabalho tomou-se $D = 3$, de modo que a primeira e a última camada tem $1/3$ da altura das camadas interiores. Quando o modelo tem apenas duas camadas as alturas são iguais. Dependendo do número de camadas especificadas as espessuras dessas camadas variam de alguns metros até 0,05m, que é a profundidade mínima especificada neste trabalho. Essa profundidade mínima não é, a rigor, necessária mas é definida para evitar instabilidades numéricas que podem surgir quando o regime do escoamento é não permanente. Nesse caso, dado que a última camada varia no espaço e no tempo, pode-se ter situações em que células podem ter alturas negativa. Uma solução para esse problema é impor para as células cuja profundidade mínima seja menor que, por exemplo, 10^{-4} m, sejam consideradas células secas, e implementar algoritmos para essas áreas passíveis de secarem ou de serem alagadas, como discutido na seção de trabalhos futuros.

3.3 Valores nodais e variáveis dependentes

No caso 3D, as variáveis dependentes discretas são definidas considerando que: N_x é o número de células da j -ésima coluna, percorrida na i -ésima linha e na k -ésima camada; N_y é o número de células da i -ésima linha, percorrida na j -ésima coluna e na k -ésima camada; N_z é o número de camadas verticais; n é o nível de tempo. Assim, as posições dos valores nodais nas células são como:

$$\left. \begin{aligned} x_i &\stackrel{\text{def}}{=} i\Delta x, \quad 1 \leq i \leq N_x \\ y_j &\stackrel{\text{def}}{=} j\Delta y, \quad 1 \leq j \leq N_y \\ z_k &\stackrel{\text{def}}{=} k\Delta z_k^n, \quad 1 \leq k \leq N_z \\ t^n &\stackrel{\text{def}}{=} n\Delta t, \quad n \in \square \end{aligned} \right\}$$

Os valores nodais do nível de água, da concentração das substâncias e das componentes das velocidades horizontal e vertical são definidos por:

$$\begin{aligned} \eta_{(i,j,k)}^n &\stackrel{\text{def}}{=} \eta(x_{i-1/2}, y_{j-1/2}, z_{k-1/2}, n\Delta t) \\ &= \eta\left(\left(i - \frac{1}{2}\right)\Delta x, \left(j - \frac{1}{2}\right)\Delta y, \left(k - \frac{1}{2}\right)\Delta z, n\Delta t\right) \end{aligned}$$

$$\begin{aligned}
S_{(i,j,k)}^n &\stackrel{def}{=} S\left(x_{i-\frac{1}{2}}, y_{j-\frac{1}{2}}, z_{k-\frac{1}{2}}, n\Delta t\right) \\
&= S\left(\left(i-\frac{1}{2}\right)\Delta x, \left(j-\frac{1}{2}\right)\Delta y, \left(k-\frac{1}{2}\right)\Delta z, n\Delta t\right) \\
u_{\left(i+\frac{1}{2}, j, k\right)}^n &\stackrel{def}{=} u\left(x_i, y_{j-\frac{1}{2}}, z_{k-\frac{1}{2}}, n\Delta t\right) \\
&= u\left(i\Delta x, \left(j-\frac{1}{2}\right)\Delta y, \left(k-\frac{1}{2}\right)\Delta z, n\Delta t\right) \\
v_{\left(i+\frac{1}{2}, j, k\right)}^n &\stackrel{def}{=} v\left(x_{i-\frac{1}{2}}, y_j, z_{k-\frac{1}{2}}, n\Delta t\right) \\
&= v\left(\left(i-\frac{1}{2}\right)\Delta x, j\Delta y, \left(k-\frac{1}{2}\right)\Delta z, n\Delta t\right) \\
w_{\left(i, j, k+\frac{1}{2}\right)}^n &\stackrel{def}{=} w\left(x_{i-\frac{1}{2}}, y_{j-\frac{1}{2}}, z_k, n\Delta t\right) \\
&= w\left(\left(i-\frac{1}{2}\right)\Delta x, \left(j-\frac{1}{2}\right)\Delta y, k\Delta z, n\Delta t\right)
\end{aligned}$$

As condições iniciais em $t^0=0$ podem, por simplicidade, ser definidas considerando que o escoamento encontra-se em repouso. Mas essa não é uma condição necessária. As condições iniciais são definidas por:

$$\begin{aligned}
\eta_{(i,j,k)}^{n=0} &= \eta_0\left(x_{i-\frac{1}{2}}, y_{j-\frac{1}{2}}, z_{k-\frac{1}{2}}, 0\right); 1 \leq i \leq C_j; 1 \leq j \leq L_i; 1 \leq k \leq M \\
S_{(i,j,k)}^{n=0} &= S_0\left(x_{i-\frac{1}{2}}, y_{j-\frac{1}{2}}, z_{k-\frac{1}{2}}, 0\right); 1 \leq i \leq C_j; 1 \leq j \leq L_i; 1 \leq k \leq M \\
u_{\left(i+\frac{1}{2}, j, k\right)}^{n=0} &= u_0\left(x_i, y_{j-\frac{1}{2}}, z_{k-\frac{1}{2}}, 0\right); 0 \leq i \leq C_j; 1 \leq j \leq L_i; 1 \leq k \leq M \\
v_{\left(i+\frac{1}{2}, j, k\right)}^{n=0} &= v_0\left(x_{i-\frac{1}{2}}, y_j, z_{k-\frac{1}{2}}, 0\right); 1 \leq i \leq C_j; 0 \leq j \leq L_i; 1 \leq k \leq M \\
w_{\left(i, j, k+\frac{1}{2}\right)}^{n=0} &= w_0\left(x_{i-\frac{1}{2}}, y_{j-\frac{1}{2}}, z_k, 0\right); 1 \leq i \leq C_j; 1 \leq j \leq L_i; 1 \leq k \leq M
\end{aligned}$$

onde C_j é o número efetivo de células da j -ésima coluna percorrida na i -ésima linha, na k -ésima camada; L_i é o número efetivo de células da i -ésima linha percorrida na j -ésima coluna, na k -ésima camada.

Note-se que a malha 2D é um caso particular do caso 3D, pois os valores de U, V, η e S no nível de tempo n , as condições de contorno e as condições iniciais especificadas no tempo inicial $t^0=0$ são definidos simplesmente suprimindo a coordenada k e não especificando a velocidade vertical w .

3.4 Condições de contorno discretas

Neste trabalho, foram especificadas condições de contorno para as ESW e a ETM. A solução da ETM requer a especificação de CC de Dirichlet e de Neumann. Na CC de Dirichlet, especifica-se uma concentração em um período específico de tempo, o que significa o fornecimento ou a retirada de massa do domínio. Na CC de Neumann, o gradiente é normal à fronteira, impondo a necessidade de nela se especificar um fluxo. Como nas fronteiras laterais abertas assume-se que o fluxo advectivo domina o fluxo dispersivo (conforme discutido na seção 2.5.1), considera-se na fronteira *inflow* CC tipo constante (Dirichlet) e fronteira *outflow* CC tipo gradiente nulo (Neumann). Esta última CC permite que a concentração saia do domínio sem acumular-se na fronteira, como pode ser visto na fig. 3.18.

No caso das ESW, as condições de contorno podem ser do tipo laterais abertas (*inflow* e *outflow*), que neste trabalho, são definidas em termos de níveis de água ou

vazão. Outras fronteiras são: laterais fechadas, superior e inferior. Nas fronteiras laterais fechadas considera-se que não há fluxo (*no flux condition*) e há deslizamento (*full-slip condition*). Na fronteira inferior considera-se que não há fluxo e há deslizamento parcial (*partial-slip condition*), dado pelos termos de atrito na interface água-sedimento. No caso da fronteira superior, também se considera o deslizamento parcial, e a CC é dada pelos termos de atrito na interface água-superfície livre. Assim, as CC são definidas empregando as notações:

1. as faces em relação à célula são denotadas por: U = superior; L = inferior; S = sul; N = norte; E = leste; W = oeste;
2. as velocidades em paredes sólidas são dos tipos não escoamento (NF), onde a velocidade normal é nula, ou deslizamento (D), ou deslizamento parcial (DP).

Dessas definições e/ou notações, em respeito à célula 3D, como nas fig. 3.5 e 3.6, nessas fronteiras laterais fechadas, inferior (interface sedimento-água) e CC superior (superfície livre), deve-se definir, respectivamente:

$$\begin{aligned} E(u_{\alpha_D}, v_{\alpha_D}, w_{\alpha_D}) &\equiv (NF, D, D) , & W(u_{\alpha_E}, v_{\alpha_E}, w_{\alpha_E}) &\equiv (NF, D, D) \\ S(u_{\alpha_F}, v_{\alpha_F}, w_{\alpha_F}) &\equiv (D, NF, D) , & N(u_{\alpha_T}, v_{\alpha_T}, w_{\alpha_T}) &\equiv (D, NF, D) \\ L(u_{\alpha_I}, v_{\alpha_I}, w_{\alpha_I}) &\equiv (DP, DP, NF) \text{ e } & U(u_{\alpha_S}, v_{\alpha_S}, w_{\alpha_S}) &\equiv (DP, DP, NF) \end{aligned}$$

Assim, as velocidades discretas (u, v, w) que não estão localizadas nas fronteiras leste, oeste, norte, sul, superior e inferior podem ser calculadas considerando a localização das variáveis dependentes para as velocidades tangenciais nas fronteiras e a localização das variáveis dependentes para as velocidades normais nas fronteiras. Detalhes podem ser vistos em [RIZ2002a]. Outras CC como, por exemplo, reflectiva, absorvente, periódica, etc. podem ser vistas em [PIE84], [ZON97], [MAT99] e [RIZ99].

Note-se que nas células internas adjacentes às fronteiras do domínio, deve-se obter aproximações diferenciadas para os termos constituintes das equações, pois nessas células esses termos não são aproximados como o são nas outras células internas aos domínios. Pode-se suprimir apropriadamente alguns termos das equações e/ou empregar fórmulas de extrapolação, como em [AND95], [HUN95] e [HUN96], considerando-se o estêncil computacional. No capítulo 6, trata-se detalhadamente dessas questões.

3.5 Domínios computacionais

O método de discretização adotado neste trabalho para obter as EDPs na forma discreta foi o das diferenças finitas, e os esquemas numéricos foram construídos sobre a malha espaço-alternado a equiespaçada mostrada na fig. 3.1. Essas escolhas determinam uma certa restrição ao tratamento mais realístico, que é dado quando do emprego de elementos finitos ou de volumes finitos sobre malhas não estruturadas, para regiões com geometria e batimetria complexa. Contudo, a técnica de discretização de *domínios gerais* via diferenças finitas, como discutida em [GER85] e [CUM2000], fornece uma boa representação para regiões arbitrárias e, por esse motivo, ela é empregada neste trabalho.

3.5.1 Configurações geométricas e enumeração

Domínios retangulares discretos são como representados pela fig. 3.7. Nela se pode notar que os valores em negrito nas células sombreadas são os índices associados com os valores nodais internos, enquanto os valores não negritados são índices associados com os valores nodais na fronteira do domínio.

de serem *interna* e *aberta* sejam especificadas CC de Dirichlet, e na fronteira contorno e *aberta* (a face “F” da célula (3,1)) seja especificada a CC de Neumann, considerando-se o gradiente nulo.

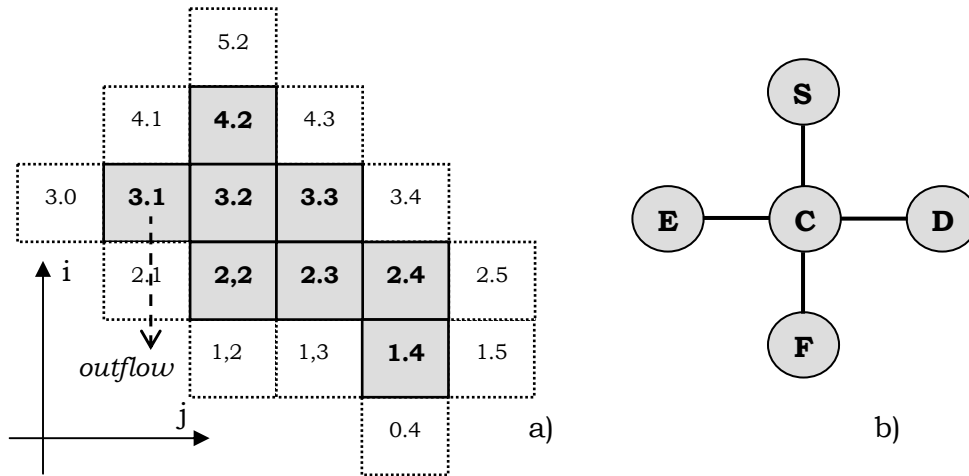


FIGURA 3.9 - Domínio não retangular com CC tipo Dirichlet em todas as faces, exceto a face F da célula (3,1), onde se tem CC tipo Neumann

Nesse caso, para montar o sistema de equações um procedimento varre todas as células testando se as fronteiras são dos tipos *interna* ou *contorno* e, se são do tipo *aberto* ou *fechado*. No caso do exemplo em discussão, identifica-se quais são as faces *interna* e *aberto* entre os tipos “F”, “S”, “D” e “E” como denotadas pelo estêncil mostrado na fig. 3.9. b). No caso específico da célula (4,2) ela tem como fronteiras as células (4,1), (4,3) e (5,2). Os valores das variáveis nessas células são conhecidos no passo de tempo anterior, e portanto, podem ser empregadas para estabelecer a CC de Dirichlet. Considerando a linha específica dessa célula, o vetor dos termos independentes fica como:

$$B_{4,2}^n - E_{4,2}\phi_{4,1}^n - D_{4,2}\phi_{4,3}^n - S_{4,2}\phi_{5,2}^n$$

sendo os demais casos análogos. Note-se que, neste trabalho, denotam-se os coeficientes “F”, “S”, “D”, “E” e “C” do sistema de equações com uma ordenação relativa à célula “C”. Assim, $E_{(4,2)}$ indica o coeficiente à esquerda de $C_{(4,2)}$; $D_{(4,2)}$ indica o coeficiente à direita de $C_{(4,2)}$ e $S_{(4,2)}$ indica o coeficiente acima de $C_{(4,2)}$. Os coeficientes gerados são como aqueles discutidos no capítulo 6.

As células que sejam do tipo *contorno aberto* e cuja CC seja do tipo gradiente nulo são tratadas observando que para essa CC indica que a taxa de variação espacial da quantidade física envolvida é nula na face entre o limite do domínio computacional, como discutido na seção 2.5.1. Assim, para uma variável ϕ , tem-se $\partial\phi/\partial y \square (\phi_{(i,j)} - \phi_{(i-1,j)}/\Delta y) = 0$ e, portanto, $\phi_{(i,j)} = \phi_{(i-1,j)}$. Especificamente para o exemplo acima tem-se $\phi_{(3,1)} = \phi_{(2,1)}$, e nesse caso, na montagem da matriz dos coeficientes tem-se:

$$C_{3,1}\phi_{(3,1)}^{n+1} + F_{3,1}\phi_{(3,1)}^{n+1} = (C_{3,1} + F_{3,1})\phi_{(3,1)}^{n+1}$$

significando que se deve adicionar ao coeficiente C da célula (3,1), o coeficiente “F” da célula (3,1) (note-se que o coeficiente “F” tem sinal negativo, como pode ser visto na seção 6.1 e, portanto, deve-se aqui considera-lo com o sinal “+”), sendo que nesse caso esse termo não entra no vetor dos termos independentes. A matriz dos coeficientes para esse exemplo é como dada na fig. 3.10.

1. *val*: armazena os valores da matriz da esquerda para direita e de cima para baixo;
2. *col_ind*: armazena a coluna da qual os valores contidos em “val” foram obtidos na matriz;
3. *row_ptr*: armazena os ponteiros que indicam quantos valores não nulos cada linha possui. A primeira posição desse vetor recebe o valor “1”, a segunda posição recebe o valor da posição anterior somado com o número de elementos não nulos da primeira linha do vetor e assim sucessivamente.

$\begin{pmatrix} 10 & 0 & 0 & 0 & -2 & 0 \\ 3 & 9 & 0 & 0 & 0 & 3 \\ 0 & 7 & 8 & 7 & 0 & 0 \\ 3 & 0 & 8 & 7 & 5 & 0 \\ 0 & 8 & 0 & 9 & 1 & 3 \\ 0 & 4 & 0 & 0 & 2 & -1 \end{pmatrix}$	a)
---	----

val	10	-2	3	9	3	7	8	7	3...9	3	4	2	-1
col_ind	1	5	1	2	6	2	3	4	1...5	6	2	5	6
row_ptr													
	1	3	6	9	13	17	20						

b)

FIGURA 3.11 - Formato CSR, a) exemplo para uma matriz esparsa $A_{6 \times 6}$; b) estrutura de dados empregada para armazenar os elementos não nulos da matriz exemplificada no item a)

Neste formato não considera qualquer informação sobre a estrutura da matriz e armazena os elementos não nulos em posições contíguas na memória. Porém, o formato CSR não é computacionalmente muito eficiente para matriz de dimensões modestas, pois necessita de endereçamento indireto para quaisquer operações. Se n é o número de elementos não nulos da matriz, esse formato é econômico, pois, em vez de armazenar N^2 (onde N é a dimensão da matriz) elementos, necessita-se de $n + N + 1$ posições.

Como o procedimento de geração da matriz é feito após o particionamento do domínio, as partes do sistema de equações (e as submatrizes locais) são geradas em paralelo e já estão distribuídas. Desse modo, cada processo possui a sua parte do domínio ou da matriz global, e sua parte do vetor das incógnitas, necessários para calcular o produto matriz por vetor, e as demais operações em álgebra linear, durante a solução iterativa via algoritmo do gradiente conjugado.

Note-se, porém, que como foi usado o estêncil de 5-pontos para construir os esquemas numéricos que geram os sistemas de equações SDP que são resolvidos via GC, é necessário um estêncil (halo) de uma célula nas fronteiras dos subdomínios. A geração e o armazenamento desses estênceis devem ser feitos considerando-se que as células dos estênceis são utilizadas em duas situações.

Na primeira situação objetiva-se a assegurar a continuidade da solução nas fronteiras artificiais quando da obtenção da solução paralela via decomposição de dados. Nesse caso considera-se para essas células dos estênceis que as CC são *outflow* e do tipo gradiente nulo de modo que essas células são adicionadas ao termo “C” na matriz dos coeficientes, como discutido anteriormente.

Na segunda situação os estênceis são necessários para obter os pré-condicionadores via decomposição de domínio. Nesse caso, o uso do MDD aditivo de Schwarz requer que, a cada iteração do GC, os valores dos estênceis entre as células adjacentes sejam trocadas a fim de estabelecer CC homogêneas.

Porém, dado que os domínios são não retangulares, as células do estêncil estão distribuídas de modo não ordenado entre os processos deve-se estabelecer uma estratégia a fim facilitar o armazenamento das células que são adjacentes internas às fronteiras do subdomínio (“estêncil ou sobreposição interna”), e que são as células que necessitam receber a identificação e o conteúdo das outras células que, são adjacentes internas às fronteiras dos outros subdomínios (“estêncil ou sobreposição externa”). A fig. 3.12 que ilustra um exemplo para as células internas e para os estênceis interno e

externo do subdomínio em destaque, de modo que estão identificadas as enumerações das células e seus conteúdos.

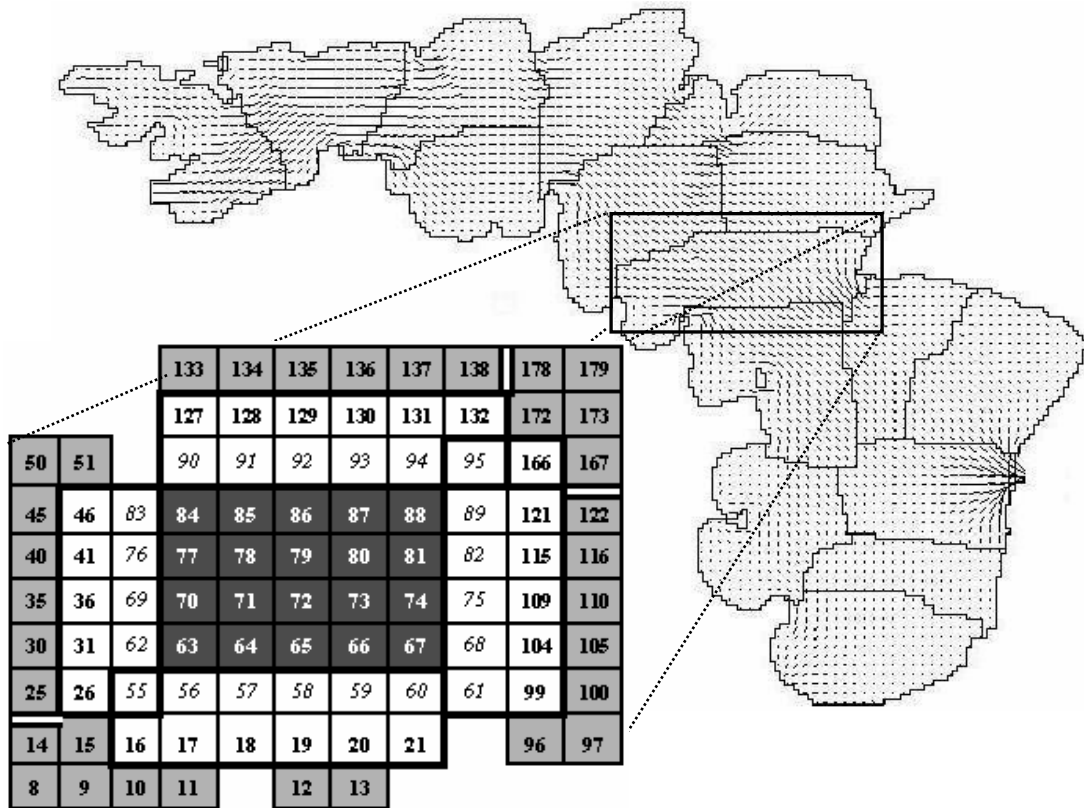


FIGURA 3.12 - Subdomínio gerado pelo particionamento gerado pela biblioteca METIS, onde se destacam as células internas e suas enumerações, e as células das fronteiras e suas enumerações

Isso significa armazenar as células, ou seja, os elementos da matriz e do vetor das incógnitas, necessárias para calcular o produto matriz por vetor, e caracterizar as comunicações para a solução paralela, via decomposição de dados, do sistema de equações, e para obter o pré-condicionador, via decomposição de domínio. Nesse último caso, o pré-condicionador é obtido de modo distribuído, e essa abordagem gera o método Krylov-Schwarz de solução, discutido na seção 7.8.

A solução adotada foi aquela discutida em [SAA2002] e [PIC2002a], onde se tem uma estrutura de armazenamento baseada na enumeração das células do domínio, que é global mas gerada localmente para cada subdomínio, como ilustrado na fig. 3.12. A estratégia baseia-se no fato de que na operação produto matriz por vetor, as células da matriz que são multiplicadas por células do vetor que não são estêncis interno, são aquelas que pertencem a algum estêncil externo. Desse modo, armazena-se separadamente e em forma de vetores, para cada subdomínio, as partes da matriz que pertencem aos estêncis. Assim, as partes da matriz correspondentes a esses estêncis estão associadas aos vetores recebidos na comunicação, e a um vetor contendo a enumeração da posição onde os resultados das operações matriciais devem ser colocados. A ordem em que estes vetores são recebidos é aquela definida pela numeração do estêncil externo, e a posição onde o resultado é colocado é aquela definida pela numeração do estêncil interno.

Assim, a multiplicação matriz esparsa por vetor é dividida em duas partes. Ou seja, $Ax = Ax_{cel_int} + Ax_{cel_est}$, onde Ax_{cel_int} denota a parte desse produto matriz \times vetor em que apenas se consideram as células internas ao domínio (células computacionais

locais), e onde Ax_{cel_est} denota a parte desse produto matriz por vetor referente aos estênceis (ou sobreposições) externos que pertencem aos processos vizinhos (células computacionais que constituem as fronteiras adjacentes externas ao subdomínio).

Essa estrutura facilita a incorporação dos pré-condicionadores baseados em decomposição de domínio, bem como a especificação das CCs tipo *outflow* na abordagem decomposição de dados. Note-se, portanto, que o produto matriz por vetor em paralelo consiste nas trocas dos “estênceis internos e externos” entre os subdomínios, na realização dos produtos matriz por vetor local (células internas), e a realização dos produtos matriz por vetor externo (células das células adjacentes externas aos subdomínios). Além disso, como as comunicações (troca dos estênceis) e os produtos matriz por vetor local são independentes, eles podem ser sobrepostos, fazendo-se essas comunicações e computações simultaneamente usando, por exemplo, troca de mensagens e *threads*.

3.5.3 Regiões não retangulares: método ADI

A técnica *Alternating Direction Implicit* (ADI) reduz a solução de um problema multidimensional à solução de problemas unidimensionais. O caso 2D é formado a partir de duas “subsoluções”, uma para cada direção coordenada. O caso 3D é formado a partir de três “subsoluções”, uma para cada direção coordenada. Essas soluções são obtidas empregando o algoritmo de Thomas para resolver os sistemas de equações lineares através de uma varredura linha a linha e outra coluna a coluna (problema 2D).

Essa abordagem foi extensamente desenvolvida e aplicada na solução de problemas nos trabalhos pioneiros de Leendertse ([LEE67], [LEE70] e [LEE71]), pois permite combinar a eficiência computacional do algoritmo de Thomas (de $O(N)$), com as boas propriedades de estabilidade do esquema numérico (semi-implícito), além de permitir uma boa representação de ilhas, istmos e outras conformações irregulares. Isto porque a malha pode ser alocada e resolvida de modo a englobar somente a região de interesse que é definida por conjuntos das linhas e das colunas, como pode ser observado na fig. 3.13.

A malha computacional é vista como uma coleção de células, em cada uma das quais, define-se o contorno físico para cada um de seus lados, que pode ser fechado, aberto interno e aberto no contorno da região. O arranjo desses três tipos produz nove espécies de células e a especificação de oito tipos de condições de contorno, que são incorporadas aos sistemas de equações lineares, como feito em alguns dos modelos 2D, em nossos trabalhos, e por [CSL85], [SIL86] e [KAP98].

As duas meias soluções são obtidas através de varreduras, a cada meio passo de tempo, nas direções X e Y . Os sistemas de equações gerados para a varredura na direção X , que é feita no nível de tempo $[n, n + \frac{1}{2}]$, são 3-diagonal. Tais sistemas de equações são, para uma linha genérica j , formados a partir do acoplamento das equações da continuidade e da conservação da quantidade do movimento para a direção X , que são definidas, respectivamente, nas localizações (i, j) e $(i + \frac{1}{2}, j)$.

Nessa situação, os domínios formados por uma coleção de linhas e uma coleção de colunas permitem a representação de domínios não retangulares sem um tratamento especial. Para cada uma dessas linhas e colunas é formado um sistema de equações, sendo necessário, portanto, especificar condições de contorno para cada um desses sistemas.

TABELA 3.1 - Sublinhas de cálculo para o caso da configuração geométrica da fig. 3.13. Solução do problema 2D via método ADI na direção X

ordem de cálculo	linha geral	coluna inicial	coluna final	tipo do contorno	ordem da matriz
1	2	4	8	U-U	7
2	3	3	8	U-U	9
3	4	2	8	U-U	11
4	4	10	11	U-U	1
5	5	1	9	U-U	15
6	5	9	13	U-U	7
7	6	1	13	U-U	23
8	7	1	12	U-U	21
9	8	1	13	η -U	24
10	9	1	14	η -U	26
11	10	1	13	η -U	24
12	10	37	40	U- η	4
13	11	1	12	η -U	22
14	11	36	39	U-U	5
15	12	2	13	U-U	21
16	12	14	16	U-U	3
17	12	35	39	U-U	7
18	12	39	40	U-U	1
19	12	41	43	U-U	3
20	13	2	14	U-U	23
21	13	14	17	U-U	5
22	13	18	20	U-U	3
23	13	21	22	U-U	1
24	13	34	45	U-U	21
25	14	2	17	U-U	29
26	14	17	23	U-U	11
27	14	27	30	U-U	5
28	14	34	44	U-U	19
29	15	2	23	U-U	41
30	15	25	31	U-U	11
31	15	32	39	U-U	13
32	15	39	41	U-U	3
33	16	2	24	U-U	43
34	16	25	40	U-U	29
35	17	2	39	U-U	73
36	18	2	36	U-U	67
37	19	2	11	U-U	17
38	19	12	35	U-U	45
39	20	2	9	U-U	13
40	20	12	32	U-U	39
41	21	12	32	U-U	39
42	22	11	31	U-U	39
43	23	11	19	U-U	15
44	23	19	27	U-U	15
45	24	13	19	U-U	11
46	24	22	26	U-U	7
47	25	13	29	U-U	11
48	25	22	25	U-U	5
49	26	14	17	U-U	5

TABELA 3.2 - Subcolunas de cálculo para o caso da configuração geométrica da fig. 3.13. Solução do problema 2D via método ADI na direção Y

ordem de cálculo	coluna geral	linha inicial	linha final	tipo do contorno	ordem da matriz
1	2	4	11	V-V	13
2	3	3	20	V-V	33
3	4	2	20	V-V	35
4	5	1	20	V-V	37
5	6	1	20	V-V	37
6	7	1	20	V-V	37
7	8	1	20	V-V	37
8	9	4	20	V-V	31
9	10	4	19	V-V	29
10	11	3	19	V-V	31
11	12	4	18	V-V	27
12	12	21	23	V-V	3
13	13	4	6	V-V	3
14	13	7	10	V-V	5
15	13	11	23	V-V	23
16	14	8	9	V-V	1
17	14	12	25	V-V	25
18	15	11	26	V-V	29
19	16	11	26	V-V	29
20	17	12	26	V-V	27
21	18	13	25	V-V	23
22	19	12	25	V-V	5
23	10	12	23	V-V	21
24	21	13	23	V-V	19
25	22	12	23	V-V	21
26	23	13	25	V-V	23
27	24	15	25	V-V	19
28	25	16	25	V-V	17
29	26	14	24	V-V	19
30	27	14	23	V-V	17
31	28	13	22	V-V	17
32	29	13	22	V-V	17
33	30	13	22	V-V	17
34	31	14	22	V-V	15
35	32	15	21	V-V	11
36	33	14	19	V-V	9
37	34	14	19	V-V	9
38	35	13	19	V-V	13
39	36	11	18	V-V	13
40	37	10	17	V-V	13
41	38	9	17	V-V	15
42	39	9	15	V-V	11
43	39	15	17	V-V	3
44	40	11	16	V-V	9
45	41	12	14	V-V	3
46	41	14	15	V-V	1
47	42	11	14	V-V	5
48	43	11	14	V-V	5
49	44	12	14	V-V	3
50	45	12	13	V-V	1
51	40	9	10	V-V	1

Se o tipo da CC for, por exemplo, U-U, em vez de η -U, a ordem da matriz 3-diagonal aumenta de uma unidade. E se o tipo da condição de contorno for U-U, em vez de U- η , a ordem da matriz 3-diagonal aumenta de uma unidade. As CC η -U e U- η foram especificadas para um caso prático, como aquele da simulação da hidrodinâmica 2D para o Lago Guaíba, cujos resultados gráficos podem ser vistos nas fig. 7.33 e 7.34.

Note-se que na tab. 3.1 a expressão *linha geral* indica qual é a linha de cálculo para cada matriz gerada, sendo que as expressões *coluna inicial* e *coluna final* indicam onde a matriz inicia e termina no domínio computacional, formando todas as sublinhas de cálculo. O mesmo acontece com as expressões *linha geral*, *linha inicial* e *linha final* da tab. 3.2.

A *ordem de cálculo* indica a ordem de solução seqüencial dos sistemas de equações cujas dimensões são dadas pela *ordem da matriz*. Mas pode-se obter uma solução paralela para a abordagem ADI em dois meios passos de tempo, de modo que ocorram varreduras simultâneas em cada subdomínio. O sincronismo e a troca de dados ficam apenas restritos às fronteiras de cada subproblema. O problema da solução paralela, quando se emprega o método ADI para se resolver os sistemas 3-diagonal, é a intrínseca dependência de dados no segundo meio passo de tempo. Essas questões são abordadas nas seções 7.10 e 7.10.1.

Uma discussão detalhada sobre o efetivo emprego do método ADI e do algoritmo de Thomas, bem como da geração dos sistemas de equações a partir do acoplamento da equação da continuidade e das equações da conservação da quantidade de movimento (abordagem de Leendertse), podem ser vistos nos trabalhos de Cassalas [CSL85], Silveira [SIL86] e em [RIZ2002a].

3.6 Métodos adaptativos para malha numérica

Muitos problemas podem ser resolvidos, eficiente e acuradamente, utilizando-se malhas fixas. Entretanto, existem aqueles nos quais o uso dessas malhas é ineficiente sob o ponto de vista computacional. Exemplos são os problemas ditos multifísica, que são aqueles nos quais os fenômenos simulados são divididos em diferentes fenômenos físicos que ocorrem no mesmo domínio de definição, e exigem diferentes escalas espaço-temporais para a obtenção de suas soluções.

A classe de problemas multifísica abrange os problemas abordados neste trabalho, que é a modelagem computacional da hidrodinâmica e do transporte de substâncias em corpos de água. Devido às diferentes naturezas das dimensões da malha das ESW elas dependem: das dimensões horizontais envolvidas; da profundidade média dos corpos de água; do fato de a água ser do tipo misturada; e do nível de detalhes desejado. No caso do Lago Guaíba, ela varia de $\Delta x = \Delta y = 1000m$ até $50m$, onde Δx e Δy são os espaçamentos nas direções X e Y. O espaçamento vertical Δz pode variar de poucos centímetros, no fundo e na superfície, até alguns metros no caso de águas mais profundas como em [UMG96], [ZHE99] e [ROS2000].

Na solução da ETM, deve-se considerar pelos menos três questões. A primeira é o fato de que a sub-região contaminada é, em geral, menor que o domínio de definição, não necessitando, portanto, de uma malha refinada globalmente. As dimensões dessa sub-região dependem de fatores como condição meteorológica, clima, período do dia, tipo de água (doce ou salgada).

A segunda questão diz respeito à importância de obter detalhes da contaminação e indicar quais as sub-regiões mais contaminadas e que sofrem restrições de balneabilidade e captação da água. O terceiro fator é numérico. Em malhas refinadas, o número de Péclet é tal que as oscilações numéricas da solução são menos importantes, pois, como o número de Péclet depende do espaçamento da malha, a construção de malhas refinadas é um importante fator para que as oscilações sejam minimizadas ou suprimidas. Por essas razões, a malha da ETM tem, nas experiências

realizadas as dimensões $\Delta x = \Delta y$ variam de 100m a 25m, e Δz varia dependendo do número de camadas.

Para uma solução acurada e eficiente, esses problemas que exigem diferentes escalas espaciais, existem os chamados métodos adaptativos, que podem ser classificados em três grupos ([HAW91], [MOO92] e [FOR99]):

1. métodos $-h$: definidos quando se adicionam mais pontos à malha. Um exemplo dessa classe são os refinamentos locais;
2. métodos $-p$: definidos quando se aumenta a ordem das funções de interpolação;
3. métodos $-r$: surgem da movimentação dos pontos da malha.

Atualmente, uma variedade de estratégias tem sido desenvolvida na busca das soluções, usando os métodos $-r$. Nesses métodos, a malha é redistribuída dinamicamente à medida que a solução evolui no tempo, usando, para isso, uma função indicadora da atividade espacial em cada sub-região. Esse procedimento dá origem ao chamado princípio da equidistribuição, que pode ser visto em detalhes em [HAW91], [SMT92], [BAI92], [NEI94] e [HUA94a].

Essa abordagem é um tema atual, e muitas das estratégias usadas ainda são empíricas e tratadas de modo heurístico. No caso de diferenças finitas não existem critérios matemáticos rigorosos para a distribuição dos nodos e para a análise da convergência de malhas móveis. O teorema de Lax não é aplicável nesse caso. São intensas as pesquisas para a solução r – adaptativa de EDPs em domínios 3D, dada a importância de construir esquemas em coordenadas generalizadas ou em malhas não estruturadas ([HUA94], [HUA97], [LIS98], [HUA99] e [CAS2000]).

Uma alternativa viável para compatibilizar a construção de modelos 3D computacionalmente eficientes com a necessidade de alto refinamento é usar o método $-h$, definindo malhas refinadas localmente, como em [MOO92] e [ZEE93]. Essa foi a abordagem adotada neste trabalho.

3.6.1 Refinamento local para o transporte: método-h

O refinamento local é uma estratégia que permite que a malha seja concentrada em sub-regiões nas quais ocorrem maiores atividades. Nessas sub-regiões, pode-se obter soluções acuradas, sem o custo computacional de uma malha globalmente refinada. Porém, essa escolha tem as desvantagens de:

1. necessitar de uma estrutura de dados sofisticada, particularmente quando a estratégia de solução envolve o paralelismo em arquitetura de memória distribuída, como no caso deste trabalho;
2. exigir procedimentos de interpolação para obter as condições de contorno para as soluções locais, o que pode degradar essas soluções se esses procedimentos não forem adequados;
3. degradar a acurácia da solução global, que pode ser sensível à combinação “interpolação + soluções locais”.

Para malhas regulares, como no caso deste trabalho, o refinamento local é obtido através de interpolações, definindo um subdomínio encaixado no domínio maior. Note-se que essa escolha é possível, ou seja, pode-se construir duas malhas, uma para o escoamento e outra para o transporte de substância. Isso porque as EDPs desses fenômenos podem ser resolvidas de forma desacoplada, já que a variação na concentração do constituinte não causa variação na densidade da água, ou seja, não modifica o escoamento. Quando o gradiente de concentração é significativo nas equações de quantidade de movimento, os modelos devem ser resolvidos de modo acoplado ([SCE91], [ZHE95] e [ZHE99]).

Quanto às desvantagens mencionadas, elas podem ser superadas, definindo estruturas de dados convenientes e processos de interpolação e de solução adequados

para obter as soluções locais. No caso deste trabalho as estruturas de dados, tanto para a malha da hidrodinâmica, quanto para a malha do transporte de substâncias, são semelhantes.

Para a malha da hidrodinâmica, há um vetor de apontadores (*pointers*) para linhas de células (*lista_linhas*). As linhas do domínio da hidrodinâmica começam na linha 0, e o número da linha é o número da posição do vetor *lista_linhas*. Para acessar a quadricula de linha *lin* e coluna *col*, a referência é feita através de $(*lista_linhas[lin])/col$. Nas versões em que se emprega o método ADI (matrizes 3-diagonal), é gerada uma matriz para cada linha, e as linhas começam em colunas variáveis, sendo a coluna inicial de cada linha dada pelo vetor *lis_col_inic*.

No transporte, utiliza-se uma estrutura semelhante à da hidrodinâmica. Emprega-se um vetor de *pointers* para linhas refinadas com coluna inicial dada por um vetor *lis_col_inic_t*. A malha de transporte e a malha da hidrodinâmica têm a mesma origem, ou seja, a linha 0 da malha de transporte e de hidrodinâmica são coincidentes, permitindo que, a partir de uma linha em uma das malhas, seja possível calcular a linha da outra malha.

O procedimento de montagem da malha da hidrodinâmica se inicia com a leitura, pelo processo 0, do arquivo com a descrição do domínio. Esse arquivo contém, para cada linha de cálculo do domínio, a coluna inicial, a coluna final e as condições de fronteira à esquerda e à direita. Contém também, para cada coluna de cálculo, as linhas inicial e final e as condições de fronteira superior e inferior. A partir dos dados desse arquivo, é gerada uma lista encadeada na qual cada nodo contém informações sobre uma linha de cálculo. Uma vez lido o arquivo de descrição, o processo 0 efetua o particionamento do domínio entre os processos e envia para cada um deles a descrição de seu subdomínio. Cada processo, recebendo esses dados, cria também uma lista de linhas de cálculo e uma lista de colunas de cálculo, que descrevem seu subdomínio. A partir dessa lista, cada processo aloca as linhas de seu subdomínio conforme a estrutura descrita acima.

A criação da malha refinada do transporte de substâncias utiliza também uma lista de linhas de cálculo semelhante à utilizada na criação da malha da hidrodinâmica. Para a criação da lista de linhas de cálculo do transporte, a lista de linhas de cálculo da hidrodinâmica é percorrida, e as linhas que cobrem, mesmo que parcialmente, a área que será refinada, geram linhas de cálculo de transporte. Uma vez gerada a lista de linhas de cálculo do transporte, a malha refinada é alocada de modo semelhante à malha da hidrodinâmica. Após a alocação da malha, a mesma é inicializada. A concentração é zerada em toda a malha, e as velocidades em cada célula são calculadas por interpolação, a partir das células correspondentes na malha da hidrodinâmica.

Durante a simulação, a cada ciclo de hidrodinâmica, é montado um sistema de equações que, resolvido, resulta nos novos valores da velocidade e/ou nível de cada célula. O procedimento que monta o sistema chama o resolvidor apropriado e, após resolvido o sistema, os valores das variáveis na malha do domínio são atualizados. Antes de iniciar o cálculo do transporte de substâncias, os valores das velocidades e nível nas células da malha refinada são obtidos por interpolação a partir das células correspondentes no domínio da hidrodinâmica. Após os valores de velocidade e nível serem atualizados, é gerado o sistema de equações para a resolução da concentração. O sistema é resolvido, e os valores da concentração resultantes são atualizados nas células. Após essa atualização, é verificado se o nível de concentração próximo a alguma das fronteiras atingiu um valor pré-determinado (10 nmp/l). No caso positivo o domínio refinado é aumentado próximo a essa fronteira, sendo alocada mais uma linha ou coluna naquela fronteira.

Quanto aos processos de interpolação e solução, as estratégias para definir o refinamento-*h* decorrem do fato de que a solução das ESWs fornece os valores das

velocidades e dos níveis que são, em cada ponto da malha e em cada nível de tempo, dados de entrada da ETM. Então a construção da malha refinada deve considerar que a pluma de substâncias varia espaço-temporalmente.

Uma alternativa de solução pode ser a de contemplar os seguintes procedimentos: resolver as ESW definidas em todo o domínio, obtendo as condições de contorno para o modelo encaixado; resolver as ESW refinadas localmente e obter os valores das velocidades e dos níveis que são dados de entrada da ETM; resolver a ETM no modelo encaixado. Porém essa estratégia exige que se resolvam dois PVIC, um global e outro local; o global fornece as CC para o local. A abordagem adotada neste trabalho, é utilizar a seguinte estratégia:

1. definir um PVIC global no qual as ESWs são resolvidas, fornecendo as velocidades e os níveis em cada ponto dessa malha;
2. definir um esquema de interpolação espacial para as velocidades e os níveis na malha refinada a partir dos obtidos na malha global;
3. resolver a ETM localmente a partir desses valores das velocidades e dos níveis.

A solução para essa abordagem é considerar que as malhas localmente refinadas são construídas através de interpolações, observando que tanto a malha global quanto a malha local são do tipo Arakawa C equiespaçada. Nessa situação, empregaram-se fórmulas tipo Newton-Gregory [GER85] para definir as expressões de interpolação objetivando obter as malhas refinadas.

Os algoritmos resultantes para as velocidades horizontais, usando interpolação linear, podem ser vistos em [RIZ2002a]. Porém, em função da estratégia desenvolvida para obter matrizes SDP (simétricas definidas-positivas), os algoritmos calculam os níveis de água η . As velocidades são recuperadas a partir do conhecimento desses níveis. Em seguida, calculam-se os valores da concentração em cada ponto da malha local. Nesse caso, o algoritmo para a interpolação linear, definida linha a linha, para a obtenção dos valores dos níveis de água η na malha refinada, ocorre conforme mostrado na fig. 3.16.

1. para $j=1$ até R faça
2. para $i=1$ até R faça
3. $\eta_{(i,j)}^n = [(i-1)/(2R-1)]\Delta\eta + \eta_L$
4. fim i
5. fim j

FIGURA 3.16 - Algoritmo para interpolação linear dos níveis

Nesse algoritmo $\Delta\eta = \eta_R - \eta_L$ é a diferença de nível na malha original, sendo, respectivamente, η_R e η_L os níveis da água à direita e à esquerda na célula original, e R , o refinamento adotado. O esquema de interpolação empregado é o bi-linear e é consistente com a formulação de diferenças finitas para a solução do escoamento a partir das velocidades obtidas nas interfaces da malha [ZHE95].

As soluções locais também necessitam de interpolação temporal, como discutido no anexo 1, pois os esquemas numéricos das ESW e a ETM têm passos de tempo diferentes. Se esses passos são denotados, respectivamente, por Δt_H e Δt_T , então $\Delta t_H = k\Delta t_T$, onde k é o fator de proporcionalidade entre os dois passos de tempo. Se R é o refinamento adotado, as malhas equiespaçadas determinam que os passos de tempo são relacionados pela expressão definida por $\Delta t_H = \Delta t_T/R$. Esses passos de tempo dependem dos limites impostos pelas aproximações construídas para cada EDP,

como tratado detalhadamente no anexo 1. A fig. 3.17 ilustra o transporte de substâncias na malha refinada encaixada na malha mais grosseira.

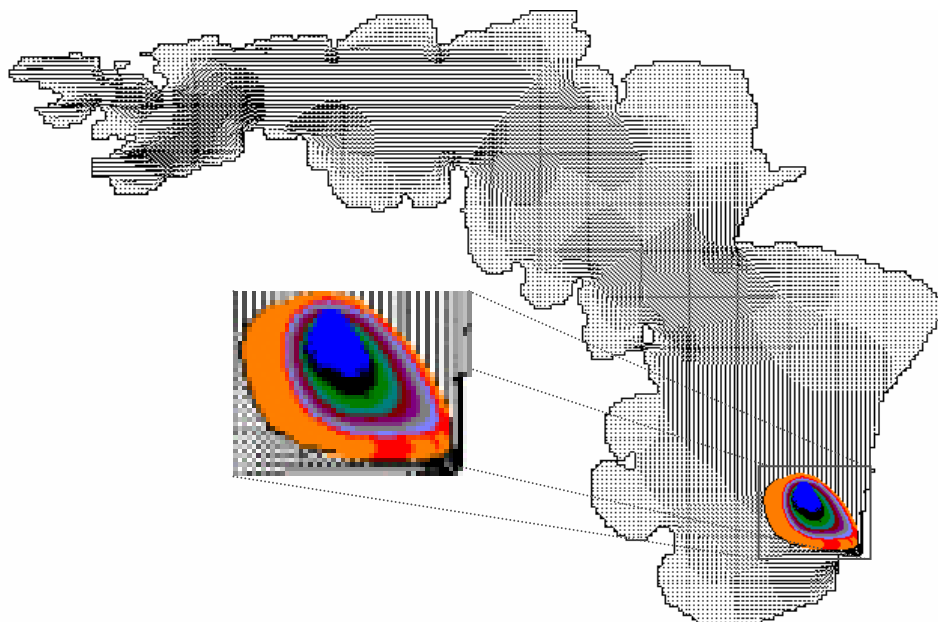


FIGURA 3.17 - Transporte de substâncias pela ETM 3D, onde se destaca a malha refinada localmente a partir da malha da hidrodinâmica (ESW 3D)

Em geral, a ETM necessita de uma malha local altamente refinada para capturar os detalhes do transporte e controlar as oscilações numéricas, dado que uma malha refinada reduz as oscilações numéricas controlando o número de Péclet, que é a métrica que determina como o transporte é dominado pela advecção. Assim, deve-se considerar a questão da acurácia do refinamento local, mas essa questão não foi analisada em detalhes neste trabalho. Deve-se analisar a acurácia do refinamento local, pois a conservatividade da massa pode requerer algoritmos de interpolação mais sofisticados como, por exemplo, a spline cúbica. Esses aspectos são objetivos de trabalhos futuros. A fig. 3.18 mostra as substâncias estão saindo do domínio computacional na fronteira *outflow*, mostrando que a concentração sai do domínio sem acumular-se na fronteira, como discutido na seção 3.4.

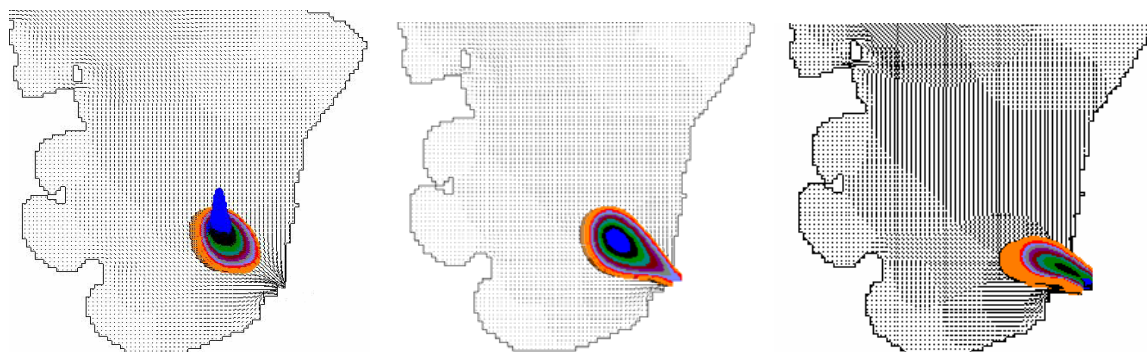


FIGURA 3.18 - Seqüência de imagens mostrando as substâncias saindo do domínio na fronteira *outflow* devido a CC gradiente nulo

Na fig. 3.18 mostra-se uma seqüência de imagens ilustrando as substâncias saído do domínio computacional. Essa solução foi obtida com um esquema numérico UMIST usando um campo de velocidades com um número de Courant constante e igual a $C_r = 0,40$. A última imagem dessa ilustração foi obtida com uma simulação executando-se, aproximadamente, 21000 ciclos do algoritmo do gradiente conjugado.

4 Modelos na forma discreta

Na definição da discretização horizontal e vertical, considerou-se que a malha definida é a Arakawa C na horizontal e a malha coordenada $-z$ na vertical, conforme ilustrado pelas fig. 3.5 e 3.6. Com essa escolha, e conforme a definição dos valores nodais e das variáveis dependentes apresentadas na seção 3.3, as componentes u , v e w das velocidades são posicionadas nas localizações $(i + \frac{1}{2}, j, k)$, $(i, j + \frac{1}{2}, k)$ e $(i, j, k + \frac{1}{2})$. Os níveis de água η e as concentrações S são posicionados na localização (i, j, k) .

A malha 3D consiste de células definidas por $(\Delta x, \Delta y, \Delta z_k^n)$, onde Δz_k^n denota que o espaçamento vertical depende da posição da camada k –ésima em relação à superfície livre, e que pode variar no nível de tempo t^n . Assim, se as alturas das faces da célula não são completamente formadas de água, como no caso das células adjacentes ao leito e/ou da superfície livre, tomam-se como espaçamento vertical as alturas da coluna de água nessas faces, conforme discutido na seção 3.2.1.

Se os espaçamentos nas direções X , Y são Δx e Δy , as alturas das faces na k –ésima camada onde $u = u_{(i+\frac{1}{2}, j, k)}$ e $v = v_{(i, j+\frac{1}{2}, k)}$ estão posicionadas são $\Delta z_{(i+\frac{1}{2}, j, k)}^n$ e $\Delta z_{(i, j+\frac{1}{2}, k)}^n$. O índice n sobrescrito nos espaçamentos verticais indica que a altura da camada da superfície depende do nível de água, que pode variar no tempo e no espaço, mostrando que Δz também varia no tempo, isto é, $\Delta z_{k=M}^n = \Delta z_M^n + \eta_{(i, j)}^n$. Suas alturas médias são definidas como:

$$\Delta z_{(i+\frac{1}{2}, j, k+\frac{1}{2})}^n = 0,5 \left(\Delta z_{(i+\frac{1}{2}, j, k+1)}^n + \Delta z_{(i+\frac{1}{2}, j, k)}^n \right) \text{ e } \Delta z_{(i, j+\frac{1}{2}, k+\frac{1}{2})}^n = 0,5 \left(\Delta z_{(i, j+\frac{1}{2}, k+1)}^n + \Delta z_{(i, j+\frac{1}{2}, k)}^n \right)$$

Além disso, $m = m_{(i+\frac{1}{2}, j, k)}$ e $m = m_{(i, j+\frac{1}{2}, k)}$ denotam os k –índices da camada inferior (interface água-sedimento); $M = M_{(i+\frac{1}{2}, j, k)}$ e $M = M_{(i, j+\frac{1}{2}, k)}$ denotam os k –índices da camada superior (interface água-superfície livre), nas direções X e Y , respectivamente, onde m varia espacialmente; e $M = M^n$ indicando que também varia espacial e temporalmente.

4.1 ESW 3D na forma discreta

As EDPs (2.1) e (2.2) são as equações da conservação da quantidade do movimento nas direções X e Y respectivamente, denotadas por $EMCM_x$ e $EMCM_y$. As EDPs (2.3) e (2.4) são as equações da conservação da massa e da conservação da massa integrada verticalmente, denotadas por ECM e $ECMIV$, respectivamente. Essas EDPs, com as condições de contorno e condições iniciais apropriadas, formam um PVIC bem posto.

Seguindo a abordagem desenvolvida por Casulli [CAS90], [CAS92], [CAS94], [CAS97] e [CAS2000], discretizam-se implicitamente os termos do gradiente de elevação da superfície livre e da difusão vertical nas ECQMs. Os termos advectivos, de difusão horizontal e da força de Coriolis são discretizados explicitamente, e a ECM e a $ECMIV$ são discretizadas implicitamente. Com essas escolhas, e considerando o método $-\theta$ para melhorar o grau de implicidade da componente vertical da velocidade e dos gradientes de elevação da superfície, resulta que essas EDPs são escritas na forma discreta como:

$$\begin{aligned} & \left(\frac{\partial u}{\partial t}\right)_{(i+\frac{1}{2},j,k)}^{n+1} + \left(u \frac{\partial u}{\partial x}\right)_{(i+\frac{1}{2},j,k)}^n + \left(v \frac{\partial u}{\partial y}\right)_{(i+\frac{1}{2},j,k)}^n + \left(w \frac{\partial u}{\partial z}\right)_{(i+\frac{1}{2},j,k)}^n = - \left(g \frac{\partial \eta}{\partial x}\right)_{(i+\frac{1}{2},j)}^{n+\theta} + \\ & + v_h \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)_{(i+\frac{1}{2},j,k)}^n + \left[\frac{\partial}{\partial z} \left(v_v \frac{\partial u}{\partial z}\right)\right]_{(i+\frac{1}{2},j,k)}^{n+1} + (\Phi v)_{(i+\frac{1}{2},j,k)}^n \end{aligned}$$

$$\begin{aligned} & \left(\frac{\partial v}{\partial t}\right)_{(i,j+\frac{1}{2},k)}^{n+1} + \left(u \frac{\partial v}{\partial x}\right)_{(i,j+\frac{1}{2},k)}^n + \left(v \frac{\partial v}{\partial y}\right)_{(i,j+\frac{1}{2},k)}^n + \left(w \frac{\partial v}{\partial z}\right)_{(i,j+\frac{1}{2},k)}^n = - \left(g \frac{\partial \eta}{\partial y}\right)_{(i,j+\frac{1}{2})}^{n+\theta} + \\ & + v_h \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right)_{(i,j+\frac{1}{2},k)}^n + \left[\frac{\partial}{\partial z} v_v \left(\frac{\partial v}{\partial z}\right)\right]_{(i,j+\frac{1}{2},k)}^{n+1} - (\Phi u)_{(i,j+\frac{1}{2},k)}^n \end{aligned}$$

$$\left(\frac{\partial u}{\partial x}\right)_{(i,j,k)}^{n+\theta} + \left(\frac{\partial v}{\partial y}\right)_{(i,j,k)}^{n+\theta} + \left(\frac{\partial w}{\partial z}\right)_{(i,j,k)}^{n+\theta} = 0$$

$$\left(\frac{\partial \eta}{\partial t}\right)_{(i,j)}^{n+\theta} + \left[\frac{\partial}{\partial x} \left(\int_{-h}^{\eta} u dz\right)\right]_{(i,j,k)}^{n+\theta} + \left[\frac{\partial}{\partial y} \left(\int_{-h}^{\eta} v dz\right)\right]_{(i,j,k)}^{n+\theta} = 0$$

Portanto, uma discretização em diferenças finitas para a ECQM_x é como:

1. o termo de avanço no tempo é $\left(\frac{\partial u}{\partial t}\right)_{(i+\frac{1}{2},j,k)}^{n+1} \approx \frac{u_{(i+\frac{1}{2},j,k)}^{n+1} - u_{(i+\frac{1}{2},j,k)}^n}{\Delta t}$;
2. o termo advectivo $\left(u \frac{\partial u}{\partial x}\right)_{(i+\frac{1}{2},j,k)}^n + \left(v \frac{\partial u}{\partial y}\right)_{(i+\frac{1}{2},j,k)}^n + \left(w \frac{\partial u}{\partial z}\right)_{(i+\frac{1}{2},j,k)}^n$ é discretizado explicitamente e agregado a um operador explícito $FU_{(i+\frac{1}{2},j,k)}^n$;
3. o gradiente de elevação da superfície é discretizado implicitamente como $\left(g \frac{\partial \eta}{\partial x}\right)_{(i+\frac{1}{2},j,k)}^{n+\theta} \approx g \frac{\theta(\eta_{(i+1,j)}^{n+1} - \eta_{(i,j)}^{n+1}) + (1-\theta)(\eta_{(i+1,j)}^{n+1} - \eta_{(i,j)}^{n+1})}{\Delta x}$;
4. o termo de difusão horizontal $v_h \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)_{(i+\frac{1}{2},j,k)}^n$ é discretizado explicitamente e agregado ao operador explícito $FU_{(i+\frac{1}{2},j,k)}^n$;
5. o termo Coriolis $(\Phi v)_{(i+\frac{1}{2},j,k)}^n \approx \Phi v_{(i+\frac{1}{2},j,k)}^{-n}$ é discretizado explicitamente e agregado ao operador $FU_{(i+\frac{1}{2},j,k)}^n$;
6. no termo da difusão vertical considera-se que a CC na superfície livre (superior), é dada pelo atrito do vento na água, e a CC inferior é definida em termos do coeficiente de Chezy. Com isso, tem-se:

$$v_{v(i+\frac{1}{2},j,M+\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,M+1)}^{n+1} - u_{(i+\frac{1}{2},j,M)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,M+\frac{1}{2})}^n} \stackrel{def}{=} \tau_x^v, \quad v_{v(i,j+\frac{1}{2},M+\frac{1}{2})} \frac{u_{(i,j+\frac{1}{2},M+1)}^{n+1} - u_{(i,j+\frac{1}{2},M)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},M+\frac{1}{2})}^n} \stackrel{def}{=} \tau_y^v$$

$$v_{v(i+\frac{1}{2},j,m-\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,m)}^{n+1} - u_{(i+\frac{1}{2},j,m-1)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,m-\frac{1}{2})}^n} \stackrel{def}{=} \tau_x^s, \quad v_{v(i,j+\frac{1}{2},m-\frac{1}{2})} \frac{u_{(i,j+\frac{1}{2},m)}^{n+1} - u_{(i,j+\frac{1}{2},m-1)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},m-\frac{1}{2})}^n} \stackrel{def}{=} \tau_y^s$$

onde τ_x^v , τ_x^s e τ_y^v , τ_y^s são, respectivamente, as condições de contorno superior e inferior nas direções X e Y . Note-se que os valores de $u_{(i+\frac{1}{2},j,M+1)}^{n+1}$, $v_{(i,j+\frac{1}{2},M+1)}^{n+1}$, $u_{(i+\frac{1}{2},j,m-1)}^{n+1}$ e $v_{(i,j+\frac{1}{2},m-1)}^{n+1}$ são fictícios e surgem para $k = M$ e $k = m$. Esses termos são substituídos pelas CC e não aparecem nas expressões finais em diferenças finitas.

Assim, na discretização do termo de advecção vertical $\left[\frac{\partial}{\partial z} \left(v_v \frac{\partial u}{\partial z} \right) \right]_{(i+\frac{1}{2},j,k)}^{n+1}$

tomam-se as k – ésimas camadas horizontais como:

- $k = M$, que indica a camada superior, discretizada como:

$$v_{v(i+\frac{1}{2},j,M+\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,M+1)}^{n+1} - u_{(i+\frac{1}{2},j,M)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,M+\frac{1}{2})}^n} - v_{v(i+\frac{1}{2},j,M-\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,M)}^{n+1} - u_{(i+\frac{1}{2},j,M-1)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,M-\frac{1}{2})}^n} \stackrel{CC superior}{=} \\ \left(\tau_x^v \right)_{(i+\frac{1}{2},j,M+\frac{1}{2})}^n - v_{v(i+\frac{1}{2},j,M-\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,M)}^{n+1} - u_{(i+\frac{1}{2},j,M-1)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,M-\frac{1}{2})}^n}$$

- $m < k < M$, que denota as camadas interiores, onde a advecção é como:

$$v_{v(i+\frac{1}{2},j,k+\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,k+1)}^{n+1} - u_{(i+\frac{1}{2},j,k)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,k+\frac{1}{2})}^n} - v_{v(i+\frac{1}{2},j,k-\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,k)}^{n+1} - u_{(i+\frac{1}{2},j,k-1)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,k-\frac{1}{2})}^n}$$

- $k = m$, que indica a camada inferior, onde a advecção discreta é como:

$$v_{v(i+\frac{1}{2},j,m+\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,m+1)}^{n+1} - u_{(i+\frac{1}{2},j,m)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,m+\frac{1}{2})}^n} - v_{v(i+\frac{1}{2},j,m-\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,m)}^{n+1} - u_{(i+\frac{1}{2},j,m-1)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,m-\frac{1}{2})}^n} \stackrel{CC inferior}{=} \\ v_{v(i+\frac{1}{2},j,m+\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,m+1)}^{n+1} - u_{(i+\frac{1}{2},j,m)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,m+\frac{1}{2})}^n} - \left(\tau_x^s \right)_{(i+\frac{1}{2},j,m)}^n u_{(i+\frac{1}{2},j,m)}^{n+1}$$

Portanto, uma discretização para ECQM_x, que fornece a componente u da velocidade horizontal, é escrita para a k – ésimas camadas, com $m \leq k \leq M$, como:

$$u_{(i+\frac{1}{2},j,k)}^{n+1} = FU_{(i+\frac{1}{2},j,k)}^n - g \frac{\Delta t}{\Delta x} \left[\theta \left(\eta_{(i+1,j)}^{n+1} - \eta_{(i,j)}^{n+1} \right) + (1 - \theta) \left(\eta_{(i+1,j)}^n - \eta_{(i,j)}^n \right) \right] + \\ \left. \begin{aligned} & + \frac{v_{v(i+\frac{1}{2},j,k+\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,k+1)}^{n+1} - u_{(i+\frac{1}{2},j,k)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,k+\frac{1}{2})}^n} - v_{v(i+\frac{1}{2},j,k-\frac{1}{2})} \frac{u_{(i+\frac{1}{2},j,k)}^{n+1} - u_{(i+\frac{1}{2},j,k-1)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,k-\frac{1}{2})}^n}}{\Delta z_{(i+\frac{1}{2},j,k)}^n} \end{aligned} \right\} \quad (4.1)$$

Analogamente, considerando todas as notações e definições anteriores, a discretização em diferenças finitas para a ECQM_y é como:

1. $\left(\frac{\partial v}{\partial t}\right)_{(i,j+\frac{1}{2},k)}^{n+1} \approx \frac{v_{(i,j+\frac{1}{2},k)}^{n+1} - v_{(i,j+\frac{1}{2},k)}^n}{\Delta t}$;
2. $\left(u \frac{\partial v}{\partial x}\right)_{(i,j+\frac{1}{2},k)}^n + \left(v \frac{\partial v}{\partial y}\right)_{(i,j+\frac{1}{2},k)}^n + \left(w \frac{\partial v}{\partial z}\right)_{(i,j+\frac{1}{2},k)}^n$ é agregado a $FV_{(i,j+\frac{1}{2},k)}^n$;
3. $\left(g \frac{\partial \eta}{\partial y}\right)_{(i,j+\frac{1}{2})}^{n+\theta} \approx g \frac{\theta(\eta_{(i,j+1)}^{n+1} - \eta_{(i,j)}^{n+1}) + (1-\theta)(\eta_{(i,j+1)}^{n+1} - \eta_{(i,j)}^{n+1})}{\Delta y}$;
4. $v_h \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right)_{(i,j+\frac{1}{2},k)}^n$ é agregado a um operador explícito $FV_{(i,j+\frac{1}{2},k)}^n$;
5. $(\Phi u)_{(i,j,k)}^n \approx \Phi \bar{u}_{(i,j+\frac{1}{2},k)}^n$ é agregado ao operador $FV_{(i,j+\frac{1}{2},k)}^n$;
6. a aproximação para a difusão vertical $\left[\frac{\partial}{\partial z} \left(v_v \frac{\partial v}{\partial z}\right)\right]_{(i,j+\frac{1}{2},k)}^{n+1}$ considera:

- $k = M$, a camada superior, que se discretiza como:

$$v_{v(i,j+\frac{1}{2},M+\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},M+1)}^{n+1} - v_{(i,j+\frac{1}{2},M)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},M+\frac{1}{2})}^n} - v_{v(i,j+\frac{1}{2},M-\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},M)}^{n+1} - v_{(i,j+\frac{1}{2},M-1)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},M-\frac{1}{2})}^n} \stackrel{CC superior}{=} \\ \left(\tau_y^v\right)_{(i,j+\frac{1}{2},M+\frac{1}{2})}^n - v_{v(i,j+\frac{1}{2},M-\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},M)}^{n+1} - v_{(i,j+\frac{1}{2},M-1)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},M-\frac{1}{2})}^n}$$

- $m < k < M$, as camadas interiores, onde a advecção vertical é como:

$$v_{v(i,j+\frac{1}{2},k+\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},k+1)}^{n+1} - v_{(i,j+\frac{1}{2},k)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},k+\frac{1}{2})}^n} - v_{v(i,j+\frac{1}{2},k-\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},k)}^{n+1} - v_{(i,j+\frac{1}{2},k-1)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},k-\frac{1}{2})}^n}$$

- $k = m$, a camada inferior, onde se toma a advecção vertical como:

$$v_{v(i,j+\frac{1}{2},m+\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},m+1)}^{n+1} - v_{(i,j+\frac{1}{2},m)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},m+\frac{1}{2})}^n} - v_{v(i,j+\frac{1}{2},m-\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},m)}^{n+1} - v_{(i,j+\frac{1}{2},m-1)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},m-\frac{1}{2})}^n} \stackrel{CC inferior}{=} \\ v_{v(i,j+\frac{1}{2},m+\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},m+1)}^{n+1} - v_{(i,j+\frac{1}{2},m)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},m+\frac{1}{2})}^n} - \left(\tau_y^s\right)_{(i,j+\frac{1}{2},m)}^n v_{(i,j+\frac{1}{2},m)}^{n+1}$$

Portanto, a discretização para ECQM_y, que fornece a componente v da velocidade horizontal, é escrita para a k -ésima camada, com $m \leq k \leq M$, como:

$$\left. \begin{aligned} v_{(i,j+\frac{1}{2},k)}^{n+1} = & FV_{(i,j+\frac{1}{2},k)}^n - g \frac{\Delta t}{\Delta y} \left[\theta(\eta_{(i,j+1)}^{n+1} - \eta_{(i,j)}^{n+1}) + (1-\theta)(\eta_{(i,j+1)}^n - \eta_{(i,j)}^n) \right] + \\ & v_{v(i,j+\frac{1}{2},k+\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},k+1)}^{n+1} - v_{(i,j+\frac{1}{2},k)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},k+\frac{1}{2})}^n} - v_{v(i,j+\frac{1}{2},k-\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},k)}^{n+1} - v_{(i,j+\frac{1}{2},k-1)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},k-\frac{1}{2})}^n} \\ & + \frac{\left(\tau_y^s\right)_{(i,j+\frac{1}{2},k)}^n v_{(i,j+\frac{1}{2},k)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},k)}^n} \end{aligned} \right\} (4.2)$$

A discretização em diferenças finitas para a ECM integrada verticalmente é:

1.
$$\left(\frac{\partial \eta}{\partial t}\right)_{(i,j)}^{n+1} \approx \frac{\eta_{(i,j)}^{n+1} - \eta_{(i,j)}^n}{\Delta t}$$
2.
$$\frac{\partial}{\partial x} \left(\int_{-h}^{\eta} u dz \right)_{(i,j,k)}^{n+\theta} \approx \frac{\theta \left(\sum_{k=m}^M u_{(i+\frac{1}{2},j,k)}^{n+1} \Delta z_{(i+\frac{1}{2},j,k)}^n - \sum_{k=m}^M u_{(i-\frac{1}{2},j,k)}^{n+1} \Delta z_{(i-\frac{1}{2},j,k)}^n \right) + (1-\theta) \left(\sum_{k=m}^M u_{(i+\frac{1}{2},j,k)}^n \Delta z_{(i+\frac{1}{2},j,k)}^n - \sum_{k=m}^M u_{(i-\frac{1}{2},j,k)}^n \Delta z_{(i-\frac{1}{2},j,k)}^n \right)}{\Delta x}$$
3.
$$\frac{\partial}{\partial y} \left(\int_{-h}^{\eta} v dz \right)_{(i,j,k)}^{n+\theta} \approx \frac{\theta \left(\sum_{k=m}^M v_{(i,j+\frac{1}{2},k)}^{n+1} \Delta z_{(i,j+\frac{1}{2},k)}^n - \sum_{k=m}^M v_{(i,j-\frac{1}{2},k)}^{n+1} \Delta z_{(i,j-\frac{1}{2},k)}^n \right) + (1-\theta) \left(\sum_{k=m}^M v_{(i,j+\frac{1}{2},k)}^n \Delta z_{(i,j+\frac{1}{2},k)}^n - \sum_{k=m}^M v_{(i,j-\frac{1}{2},k)}^n \Delta z_{(i,j-\frac{1}{2},k)}^n \right)}{\Delta y}$$

Assim, a EDP da continuidade integrada verticalmente, que é responsável pela integração entre as camadas de água, é aproximada como:

$$\left. \begin{aligned} \eta_{(i,j)}^{n+1} = & \eta_{(i,j)}^n - \frac{\Delta t}{\Delta x} \theta \left(\sum_{k=m}^M u_{(i+\frac{1}{2},j,k)}^{n+1} \Delta z_{(i+\frac{1}{2},j,k)}^n - \sum_{k=m}^M u_{(i-\frac{1}{2},j,k)}^{n+1} \Delta z_{(i-\frac{1}{2},j,k)}^n \right) \\ & - \frac{\Delta t}{\Delta y} \theta \left(\sum_{k=m}^M v_{(i,j+\frac{1}{2},k)}^{n+1} \Delta z_{(i,j+\frac{1}{2},k)}^n - \sum_{k=m}^M v_{(i,j-\frac{1}{2},k)}^{n+1} \Delta z_{(i,j-\frac{1}{2},k)}^n \right) \\ & - \frac{\Delta t}{\Delta x} (1-\theta) \left(\sum_{k=m}^M u_{(i+\frac{1}{2},j,k)}^n \Delta z_{(i+\frac{1}{2},j,k)}^n - \sum_{k=m}^M u_{(i-\frac{1}{2},j,k)}^n \Delta z_{(i-\frac{1}{2},j,k)}^n \right) \\ & - \frac{\Delta t}{\Delta y} (1-\theta) \left(\sum_{k=m}^M v_{(i,j+\frac{1}{2},k)}^n \Delta z_{(i,j+\frac{1}{2},k)}^n - \sum_{k=m}^M v_{(i,j-\frac{1}{2},k)}^n \Delta z_{(i,j-\frac{1}{2},k)}^n \right) \end{aligned} \right\} (4.3)$$

A discretização em diferenças finitas para a ECM também considera essas notações e definições, e é da seguinte forma:

1.
$$\left(\frac{\partial u}{\partial x}\right)_{(i,j,k)}^{n+\theta} \approx \frac{\theta \left(u_{(i+\frac{1}{2},j,k)}^{n+1} - u_{(i-\frac{1}{2},j,k)}^{n+1} \right) + (1-\theta) \left(u_{(i+\frac{1}{2},j,k)}^n - u_{(i-\frac{1}{2},j,k)}^n \right)}{\Delta x}$$
2.
$$\left(\frac{\partial u}{\partial y}\right)_{(i,j,k)}^{n+\theta} \approx \frac{\theta \left(v_{(i,j+\frac{1}{2},k)}^{n+1} - v_{(i,j-\frac{1}{2},k)}^{n+1} \right) + (1-\theta) \left(v_{(i,j+\frac{1}{2},k)}^n - v_{(i,j-\frac{1}{2},k)}^n \right)}{\Delta y}$$
3.
$$\left(\frac{\partial w}{\partial z}\right)_{(i,j,k)}^{n+\theta} \approx \frac{\theta \left(w_{(i,j,k+\frac{1}{2})}^{n+1} - w_{(i,j,k-\frac{1}{2})}^{n+1} \right) + (1-\theta) \left(w_{(i,j,k+\frac{1}{2})}^n - w_{(i,j,k-\frac{1}{2})}^n \right)}{\Delta z_{(i,j,k)}}$$

e, portanto, a aproximação para a equação da continuidade, que fornece as velocidades verticais, é como:

$$w_{(i,j,k+\frac{1}{2})}^{n+\theta} = w_{(i,j,k-\frac{1}{2})}^{n+\theta} - \frac{\left(u_{(i+\frac{1}{2},j,k)}^{n+\theta} \Delta z_{(i+\frac{1}{2},j,k)}^n - u_{(i-\frac{1}{2},j,k)}^{n+\theta} \Delta z_{(i-\frac{1}{2},j,k)}^n \right)}{\Delta x} \left. \vphantom{w_{(i,j,k+\frac{1}{2})}^{n+\theta}}} \right\} \quad (4.4)$$

$$- \frac{\left(v_{(i,j+\frac{1}{2},k)}^{n+\theta} \Delta z_{(i,j+\frac{1}{2},k)}^n - v_{(i,j-\frac{1}{2},k)}^{n+\theta} \Delta z_{(i,j-\frac{1}{2},k)}^n \right)}{\Delta y}$$

onde $m \leq k \leq M$, sendo que a CC na interface água-sedimento é, impondo a condição que não há escoamento, tal que $w_{(i,j,k-\frac{1}{2})}^{n+\theta} \equiv 0$. A velocidade na interface água-superfície livre é, neste trabalho, tal que $w_{(i,j,k+\frac{1}{2})}^{n+\theta} \equiv 0$.

Note-se que, nas expressões anteriores, alguns dos valores das variáveis dependentes são obtidos empregando o método $-\theta$, sendo, portanto, dadas por $\varphi^{n+\theta} = \theta\varphi^{n+1} + (1-\theta)\varphi^n$, com $0 < \theta \leq 1$. Também as alturas das faces são tais que, em geral, tem-se $\Delta z_{(i+\frac{1}{2},j,k)}^n = \Delta z_{(i,j+\frac{1}{2},k)}^n = \Delta z_{(i,j,k)}^n$, exceto nas faces verticais que interceptam a batimetria na camada inferior e/ou cruzam a superfície livre. Nesse caso, as alturas dessas células são dadas pela altura da água.

Como o cálculo dos valores das velocidades horizontais $u_{(i+\frac{1}{2},j,k)}^{n+1}$ e $v_{(i,j+\frac{1}{2},k)}^{n+1}$ é dependente dos valores dos níveis de água $\eta_{(i,j)}^{n+1}$ nos algoritmos anteriormente obtidos, deve-se resolver as expressões (4.1)-(4.3) de forma acoplada. Essa escolha acaba gerando sistemas de equações com $N_x N_y (2N_z + 1)$ equações, nos quais N_x , N_y e N_z são, respectivamente, o número efetivo de células nas três direções coordenadas, conforme definido na seção 3.3.

Porém, dado o porte desse sistema de equações, que deve ser resolvido para cada passo de tempo, essa abordagem é inconveniente. Para contornar esse problema, emprega-se a estratégia desenvolvida por Casulli em [CAS90], e faz-se a decomposição do sistema de equações (4.1)-(4.3) em um conjunto de $2N_x N_y$ sistemas 3-diagonal, cada qual com N_z equações, e um sistema *tipo 5-diagonal* de $N_x N_y$ equações. Para efetivar esse procedimento, as EDPs (4.1) a (4.3) (as ECQM_x, ECQM_y e a ECMIV) são escritas na forma matricial como [CAS92] e [CAS94]:

$$U_{(i+\frac{1}{2},j)}^{n+1} = \left(A_{(i+\frac{1}{2},j)}^n \right)^{-1} \left\{ G_{(i+\frac{1}{2},j)}^n - g \frac{\Delta t}{\Delta x} \left[\theta \left(\eta_{(i+1,j)}^{n+1} - \eta_{(i,j)}^{n+1} \right) \right] \Delta Z_{(i+\frac{1}{2},j)}^n \right\} \quad (4.5)$$

$$V_{(i,j+\frac{1}{2})}^{n+1} = \left(A_{(i,j+\frac{1}{2})}^n \right)^{-1} \left\{ G_{(i,j+\frac{1}{2})}^n - g \frac{\Delta t}{\Delta y} \left[\theta \left(\eta_{(i,j+1)}^{n+1} - \eta_{(i,j)}^{n+1} \right) \right] \Delta Z_{(i,j+\frac{1}{2})}^n \right\} \quad (4.6)$$

e

$$\eta_{(i,j)}^{n+1} = - \frac{\Delta t}{\Delta x} (1-\theta) \left[\left(\Delta Z_{(i+\frac{1}{2},j)}^n \right)^T U_{(i+\frac{1}{2},j)}^n - \left(\Delta Z_{(i-\frac{1}{2},j)}^n \right)^T U_{(i-\frac{1}{2},j)}^n \right] \left. \vphantom{\eta_{(i,j)}^{n+1}}} \right\} \quad (4.7)$$

$$- \frac{\Delta t}{\Delta y} (1-\theta) \left[\left(\Delta Z_{(i,j+\frac{1}{2})}^n \right)^T V_{(i,j+\frac{1}{2})}^n - \left(\Delta Z_{(i,j-\frac{1}{2})}^n \right)^T V_{(i,j-\frac{1}{2})}^n \right]$$

$$- \frac{\Delta t}{\Delta x} \theta \left[\left(\Delta Z_{(i+\frac{1}{2},j)}^n \right)^T U_{(i+\frac{1}{2},j)}^{n+1} - \left(\Delta Z_{(i-\frac{1}{2},j)}^n \right)^T U_{(i-\frac{1}{2},j)}^{n+1} \right]$$

$$- \frac{\Delta t}{\Delta y} \theta \left[\left(\Delta Z_{(i,j+\frac{1}{2})}^n \right)^T V_{(i,j+\frac{1}{2})}^{n+1} - \left(\Delta Z_{(i,j-\frac{1}{2})}^n \right)^T V_{(i,j-\frac{1}{2})}^{n+1} \right] + \eta_{(i,j)}^n$$

sendo que, em (4.5)-(4.7), $U_{(i+\frac{1}{2},j)}^{n+1}$, $V_{(i,j+\frac{1}{2})}^{n+1}$, $\Delta Z_{(i+\frac{1}{2},j)}^n$ e $\Delta Z_{(i,j+\frac{1}{2})}^n$ são os vetores das componentes u^{n+1} e v^{n+1} , e dos espaçamentos verticais nessas direções, respectivamente. Esses vetores têm a dimensão $1 \leq k \leq N_z$ e são escritos como:

$$U_{(i+\frac{1}{2},j)}^{n+1} = \begin{bmatrix} u_{(i+\frac{1}{2},j,M)}^{n+1} \\ u_{(i+\frac{1}{2},j,M-1)}^{n+1} \\ \vdots \\ u_{(i+\frac{1}{2},j,m+1)}^{n+1} \\ u_{(i+\frac{1}{2},j,m)}^{n+1} \end{bmatrix}, \quad V_{(i,j+\frac{1}{2})}^{n+1} = \begin{bmatrix} v_{(i,j+\frac{1}{2},M)}^{n+1} \\ v_{(i,j+\frac{1}{2},M-1)}^{n+1} \\ \vdots \\ v_{(i,j+\frac{1}{2},m+1)}^{n+1} \\ v_{(i,j+\frac{1}{2},m)}^{n+1} \end{bmatrix},$$

$$\Delta Z_{(i+\frac{1}{2},j,k)}^n = \begin{bmatrix} \Delta z_{(i+\frac{1}{2},j,M)}^n \\ \Delta z_{(i+\frac{1}{2},j,M-1)}^n \\ \vdots \\ \Delta z_{(i+\frac{1}{2},j,m+1)}^n \\ \Delta z_{(i+\frac{1}{2},j,m)}^n \end{bmatrix} \quad \text{e} \quad \Delta Z_{(i,j+\frac{1}{2},k)}^n = \begin{bmatrix} \Delta z_{(i,j+\frac{1}{2},M)}^n \\ \Delta z_{(i,j+\frac{1}{2},M-1)}^n \\ \vdots \\ \Delta z_{(i,j+\frac{1}{2},m+1)}^n \\ \Delta z_{(i,j+\frac{1}{2},m)}^n \end{bmatrix}$$

Já os vetores $G_{(i+\frac{1}{2},j)}^n$ e $G_{(i,j+\frac{1}{2})}^n$, que são expressões como em (4.8) e (4.9), agregam os operadores explícitos que são escritos da seguinte forma:

$$FU_{(i+\frac{1}{2},j,k)}^n = u_{(i+\frac{1}{2},j,k)}^n + \left(u \frac{\partial u}{\partial x} \right)_{(i+\frac{1}{2},j,k)}^n + \left(v \frac{\partial u}{\partial y} \right)_{(i+\frac{1}{2},j,k)}^n + \left(w \frac{\partial u}{\partial z} \right)_{(i+\frac{1}{2},j,k)}^n +$$

$$+ v_h \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)_{(i+\frac{1}{2},j,k)}^n + \Phi \bar{v}_{(i+\frac{1}{2},j,k)}^{-n}$$

$$FV_{(i,j+\frac{1}{2},k)}^n = v_{(i,j+\frac{1}{2},k)}^n + \left(u \frac{\partial v}{\partial x} \right)_{(i,j+\frac{1}{2},k)}^n + \left(v \frac{\partial v}{\partial y} \right)_{(i,j+\frac{1}{2},k)}^n + \left(w \frac{\partial v}{\partial z} \right)_{(i,j+\frac{1}{2},k)}^n +$$

$$+ v_h \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right)_{(i,j+\frac{1}{2},k)}^n - \Phi \bar{u}_{(i,j+\frac{1}{2},k)}^{-n}$$

nos quais $m \leq k \leq M$ são índices para a k -ésima camada, que varia de $m = 1$ até $m = N_z$.

$$G_{(i+\frac{1}{2},j)}^n = \begin{bmatrix} \Delta z_{(i+\frac{1}{2},j,M)}^n \left[FU_{(i+\frac{1}{2},j,M)}^n - g \frac{\Delta t}{\Delta x} (1-\theta) (\eta_{(i+1,j)}^n - \eta_{(i+\frac{1}{2},j)}^n) \right] + \Delta t \tau_x^v \\ \Delta z_{(i+\frac{1}{2},j,M-1)}^n \left[FU_{(i+\frac{1}{2},j,M-1)}^n - g \frac{\Delta t}{\Delta x} (1-\theta) (\eta_{(i+1,j)}^n - \eta_{(i+\frac{1}{2},j)}^n) \right] \\ \vdots \\ \Delta z_{(i+\frac{1}{2},j,m+1)}^n \left[FU_{(i+\frac{1}{2},j,m+1)}^n - g \frac{\Delta t}{\Delta x} (1-\theta) (\eta_{(i+1,j)}^n - \eta_{(i+\frac{1}{2},j)}^n) \right] \\ \Delta z_{(i+\frac{1}{2},j,m)}^n \left[FU_{(i+\frac{1}{2},j,m)}^n - g \frac{\Delta t}{\Delta x} (1-\theta) (\eta_{(i+1,j)}^n - \eta_{(i+\frac{1}{2},j)}^n) \right] \end{bmatrix} \quad (4.8)$$

e

$$G_{(i,j+\frac{1}{2})}^n = \begin{bmatrix} \Delta z_{(i,j+\frac{1}{2},M)}^n \left[FV_{(i,j+\frac{1}{2},M)}^n - g \frac{\Delta t}{\Delta y} (1-\theta) (\eta_{(i,j+1)}^n - \eta_{(i+1,j)}^n) \right] + \Delta t \tau_y^v \\ \Delta z_{(i,j+\frac{1}{2},M-1)}^n \left[FV_{(i,j+\frac{1}{2},M-1)}^n - g \frac{\Delta t}{\Delta y} (1-\theta) (\eta_{(i,j+1)}^n - \eta_{(i+1,j)}^n) \right] \\ \vdots \\ \Delta z_{(i,j+\frac{1}{2},m+1)}^n \left[FV_{(i,j+\frac{1}{2},m+1)}^n - g \frac{\Delta t}{\Delta y} (1-\theta) (\eta_{(i,j+1)}^n - \eta_{(i+1,j)}^n) \right] \\ \Delta z_{(i,j+\frac{1}{2},m)}^n \left[FV_{(i,j+\frac{1}{2},m)}^n - g \frac{\Delta t}{\Delta y} (1-\theta) (\eta_{(i,j+1)}^n - \eta_{(i+1,j)}^n) \right] \end{bmatrix} \quad (4.9)$$

e as matrizes $A_{(i+\frac{1}{2},j)}^n$ e $A_{(i,j+\frac{1}{2})}^n$ em (4.5)-(4.7) são escritas como:

$$A_{(i+\frac{1}{2},j)}^n = \begin{bmatrix} \Delta z_M^n + a_{M-\frac{1}{2}} & -a_{M-\frac{1}{2}} & \cdots & 0 \\ -a_{M-\frac{1}{2}} & \Delta z_{M-1} + a_{M-\frac{1}{2}} + a_{M-\frac{3}{2}} & -a_{M-\frac{3}{2}} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & -a_{m+\frac{1}{2}} & \Delta z_m + a_{m+\frac{1}{2}} + \Delta t \tau_x^s \end{bmatrix} \quad (4.10)$$

e

$$A_{(i,j+\frac{1}{2})}^n = \begin{bmatrix} \Delta z_M^n + a_{M-\frac{1}{2}} & -a_{M-\frac{1}{2}} & \cdots & 0 \\ -a_{M-\frac{1}{2}} & \Delta z_{M-1} + a_{M-\frac{1}{2}} + a_{M-\frac{3}{2}} & -a_{M-\frac{3}{2}} & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & -a_{m+\frac{1}{2}} & \Delta z_m + a_{m+\frac{1}{2}} + \Delta t \tau_y^s \end{bmatrix} \quad (4.11)$$

onde $a_k = v_v \Delta t / \Delta z_k^n$ em (4.10) e (4.11) em que, ou se tem $v_v = v_{v(i,j+\frac{1}{2},k+\frac{1}{2})}$ ou $v_v = v_{v(i+\frac{1}{2},j,k+\frac{1}{2})}$ que são os coeficientes de viscosidade cinemática turbulenta vertical, e ou $\Delta z_k^n = \Delta z_{(i,j+\frac{1}{2},k+\frac{1}{2})}^n$ ou $\Delta z_k^n = \Delta z_{(i+\frac{1}{2},j,k+\frac{1}{2})}^n$, respectivamente.

A substituição formal de (4.5) e (4.6) em (4.7) gera uma expressão em que os níveis são as únicas incógnitas a calcular. Essa expressão é escrito como:

$$\left. \begin{aligned} & \eta_{(i,j)}^{n+1} - g \left(\frac{\Delta t}{\Delta x} \right)^2 \theta^2 \left[\Delta U^T \left(\eta_{(i+1,j)}^{n+1} - \eta_{(i,j)}^{n+1} \right) - \Delta U^T \left(\eta_{(i,j)}^{n+1} - \eta_{(i-1,j)}^{n+1} \right) \right] \\ & - g \left(\frac{\Delta t}{\Delta y} \right)^2 \theta^2 \left[\Delta V^T \left(\eta_{(i,j+1)}^{n+1} - \eta_{(i,j)}^{n+1} \right) - \Delta V^T \left(\eta_{(i,j)}^{n+1} - \eta_{(i,j-1)}^{n+1} \right) \right] \\ & = \eta_{(i,j)}^n - \frac{\Delta t}{\Delta x} (1-\theta) \left[\Delta U_{(i+\frac{1}{2},j)}^T - \Delta U_{(i-\frac{1}{2},j)}^T \right] - \frac{\Delta t}{\Delta y} (1-\theta) \left[\Delta V_{(i,j+\frac{1}{2})}^T - \Delta V_{(i,j-\frac{1}{2})}^T \right] \end{aligned} \right\} \quad (4.12)$$

onde:

$$\Delta U_{(i\pm\frac{1}{2},j)}^T = \left(\Delta z_{(i\pm\frac{1}{2},j)}^n \left(A_{(i\pm\frac{1}{2},j)}^n \right)^{-1} \Delta z_{(i\pm\frac{1}{2},j)}^n \right)^T \text{ e } \Delta V_{(i,j\pm\frac{1}{2})}^T = \left(\Delta z_{(i,j\pm\frac{1}{2})}^n \left(A_{(i,j\pm\frac{1}{2})}^n \right)^{-1} \Delta z_{(i,j\pm\frac{1}{2})}^n \right)^T$$

A expressão (4.12) manipulada adequadamente fornece a expressão matricial (6.1) em termos de coeficientes e onde os níveis de elevação da superfície são as únicas

variáveis a serem calculadas a cada passo de tempo, na solução do sistema de equações. No capítulo 6 descrevem-se detalhadamente os algoritmos numéricos para a solução efetiva do problema.

4.2 Condições de contorno discretas: ESW 3D

As CC discretas para a interface sedimento-água são obtidas, para a direção X , considerando (seção 2.5.1) que em $\tau^s = g(C_z^2)^{-1} \sqrt{\bar{u}^2 + \bar{v}^2}$ tem-se:

$$\bar{u} = u_{(i+\frac{1}{2},j,m)}^n - \frac{g\Delta t}{2\Delta x} \left(\eta_{(i+1,j)}^n - \eta_{(i,j)}^n \right) \text{ e } \bar{v} = v_{(i+\frac{1}{2},j,m)}^n - \frac{g\Delta t}{2\Delta y} \left(\eta_{(i,j+1)}^n - \eta_{(i,j)}^n \right)$$

tal que $v_{(i+\frac{1}{2},j,m)}^n = 0,25 \left(v_{(i,j+\frac{1}{2},m)}^n + v_{(i,j-\frac{1}{2},m)}^n + v_{(i+1,j+\frac{1}{2},m)}^n + v_{(i+1,j-\frac{1}{2},m)}^n \right)$. E para a direção Y tem-se:

$$\bar{v} = v_{(i,j+\frac{1}{2},m)}^n - \frac{g\Delta t}{2\Delta y} \left(\eta_{(i,j+1)}^n - \eta_{(i,j)}^n \right) \text{ e } \bar{u} = u_{(i,j+\frac{1}{2},m)}^n - \frac{g\Delta t}{2\Delta x} \left(\eta_{(i,j+1)}^n - \eta_{(i,j)}^n \right)$$

tal que $u_{(i,j+\frac{1}{2},m)}^n = 0,25 \left(u_{(i+\frac{1}{2},j,m)}^n + u_{(i-\frac{1}{2},j,m)}^n + u_{(i+\frac{1}{2},j+1,m)}^n + u_{(i-\frac{1}{2},j+1,m)}^n \right)$

4.3 ESW 2D-IV na forma discreta

Neste trabalho, foram desenvolvidos e implementados modelos 2D integrados na vertical, tanto para o escoamento quanto para o transporte de substâncias, cujas formas analíticas foram discutidas nas seções 2.4 e 2.6. Também desenvolvidos e implementados modelos 3D, para o escoamento e o transporte de substâncias, cujas formas analíticas foram discutidas nas seções 2.3 e 2.5.

Esses diferentes modelos desenvolvidos e implementados obedeceram à ordem cronológica da elaboração deste trabalho, onde inicialmente foram considerados modelos 2D e, posteriormente, modelos 3D. Porém, a forma discreta das ESW 3D com apenas uma camada, reduz-se à forma discreta das ESW 2D integradas verticalmente. Basta tomar, como em [CAS92]:

$$\begin{aligned} \Delta z_{(i+\frac{1}{2},j)}^n &= H_{(i+\frac{1}{2},j)}^n \stackrel{2D}{=} H_{u(i+\frac{1}{2},j)}^n \stackrel{def}{=} H_{u(i+\frac{1}{2},j)}^n + \eta_{(i,j)}^n, \quad \Delta z_{(i,j+\frac{1}{2})}^n = H_{(i,j+\frac{1}{2})}^n \stackrel{2D}{=} H_{v(i,j+\frac{1}{2})}^n \stackrel{def}{=} H_{v(i,j+\frac{1}{2})}^n + \eta_{(i,j)}^n \\ A_{(i+\frac{1}{2},j)}^n &= H_{(i+\frac{1}{2},j)}^n + \Delta t \left(\tau_x^s \right)_{(i+\frac{1}{2},j)}^n H_{(i+\frac{1}{2},j)}^n, \quad A_{(i,j+\frac{1}{2})}^n = H_{(i,j+\frac{1}{2})}^n + \Delta t \left(\tau_y^s \right)_{(i,j+\frac{1}{2})}^n H_{(i,j+\frac{1}{2})}^n \\ G_{(i+\frac{1}{2},j)}^n &= H_{(i+\frac{1}{2},j)}^n F U_{(i+\frac{1}{2},j)}^n + \Delta t \left(\tau_x^v \right)_{(i+\frac{1}{2},j)}^n \text{ e } G_{(i,j+\frac{1}{2})}^n = H_{(i,j+\frac{1}{2})}^n F V_{(i,j+\frac{1}{2})}^n + \Delta t \left(\tau_y^v \right)_{(i,j+\frac{1}{2})}^n \end{aligned}$$

Então, sob essas condições, as formas discretas das ECQMs 3D com uma só camada coincidem com as formas discretas das ECQMs 2D integradas na vertical. Nesse caso, as EDPs são aproximadas como:

$$U_{(i+\frac{1}{2},j)}^{n+1} = F U_{(i+\frac{1}{2},j)}^n - \frac{g\Delta t}{\Delta x} \left[\theta \left(\eta_{(i+1,j)}^{n+1} - \eta_{(i,j)}^{n+1} \right) \right] + \frac{\Delta t \left(\tau_x^v \right)_{(i+\frac{1}{2},j)}^n}{H_{(i+\frac{1}{2},j)}^n} - \frac{\Delta t g \left(\tau_x^s \right)_{(i+\frac{1}{2},j)}^n}{C_z^2 H_{(i+\frac{1}{2},j)}^n} U_{(i+\frac{1}{2},j)}^{n+1}$$

e

$$V_{(i,j+\frac{1}{2})}^{n+1} = F V_{(i,j+\frac{1}{2})}^n - \frac{g\Delta t}{\Delta y} \left[\theta \left(\eta_{(i,j+1)}^{n+1} - \eta_{(i,j)}^{n+1} \right) \right] + \frac{\Delta t \left(\tau_y^v \right)_{(i,j+\frac{1}{2})}^n}{H_{(i,j+\frac{1}{2})}^n} - \frac{\Delta t g \left(\tau_y^s \right)_{(i,j+\frac{1}{2})}^n}{C_z^2 H_{(i,j+\frac{1}{2})}^n} V_{(i,j+\frac{1}{2})}^{n+1}$$

ou seja, usando a notação como em (4.5) e (4.6), tem-se:

$$U_{(i+\frac{1}{2},j)}^{n+1} = \left(1 + \Delta t (\tau_x^s)_{(i+\frac{1}{2},j)}^n\right)^{-1} \left\{ F U_{(i+\frac{1}{2},j)}^n - g \frac{\Delta t}{\Delta x} \left[\theta (\eta_{(i+1,j)}^{n+1} - \eta_{(i,j)}^{n+1}) \right] \right\}$$

e

$$V_{(i,j+\frac{1}{2})}^{n+1} = \left(1 + \Delta t (\tau_y^s)_{(i,j+\frac{1}{2})}^n\right)^{-1} \left\{ F V_{(i,j+\frac{1}{2})}^n - g \frac{\Delta t}{\Delta y} \left[\theta (\eta_{(i,j+1)}^{n+1} - \eta_{(i,j)}^{n+1}) \right] \right\}$$

e, de modo análogo, a discretização para a ECM fornece a expressão:

$$\begin{aligned} \eta_{(i,j)}^{n+1} = & -\frac{\Delta t}{\Delta x} \theta \left[\left(H_{u(i+\frac{1}{2},j)} + \eta_{(i)}^n \right) U_{(i+\frac{1}{2},j)}^{n+1} - \left(H_{u(i-\frac{1}{2},j)} + \eta_{(i-1)}^n \right) U_{(i-\frac{1}{2},j)}^{n+1} \right] \\ & -\frac{\Delta t}{\Delta y} \theta \left[\left(H_{v(i,j+\frac{1}{2})} + \eta_{(j)}^n \right) V_{(i,j+\frac{1}{2})}^{n+1} - \left(H_{v(i,j-\frac{1}{2})} + \eta_{(j-1)}^n \right) V_{(i,j-\frac{1}{2})}^{n+1} \right] \\ & -\frac{\Delta t}{\Delta x} (1-\theta) \left[\left(H_{u(i+\frac{1}{2},j)} + \eta_{(i)}^n \right) U_{(i+\frac{1}{2},j)}^n - \left(H_{u(i-\frac{1}{2},j)} + \eta_{(i-1)}^n \right) U_{(i-\frac{1}{2},j)}^{n+1} \right] \\ & -\frac{\Delta t}{\Delta y} (1-\theta) \left[\left(H_{v(i,j+\frac{1}{2})} + \eta_{(j)}^n \right) V_{(i,j+\frac{1}{2})}^{n+1} - \left(H_{v(i,j-\frac{1}{2})} + \eta_{(j-1)}^n \right) V_{(i,j-\frac{1}{2})}^{n+1} \right] + \eta_{(i,j)}^n \end{aligned}$$

em que $H_{u(i\pm\frac{1}{2},j)}$ e $H_{v(i,j\pm\frac{1}{2})}$ são as batimetrias. A substituição formal das ECQM_x e ECQM_y na ECM fornece a expressão:

$$\begin{aligned} \eta_{(i,j)}^{n+1} - \left(\frac{\Delta t}{\Delta x} \right)^2 \theta^2 \left[\left(\frac{H_{u(i+\frac{1}{2},j)} + \eta_{(i)}^n}{1 + \Delta t (\tau_x^s)_{(i+\frac{1}{2},j)}^n} \right) \Delta_+ \eta_i - \left(\frac{H_{u(i-\frac{1}{2},j)} + \eta_{(i-1)}^n}{1 + \Delta t (\tau_x^s)_{(i-\frac{1}{2},j)}^n} \right) \Delta_- \eta_i \right] \\ - \left(\frac{\Delta t}{\Delta y} \right)^2 \theta^2 \left[\left(\frac{H_{v(i,j+\frac{1}{2})} + \eta_{(j)}^n}{1 + \Delta t (\tau_y^s)_{(i,j+\frac{1}{2})}^n} \right) \Delta_+ \eta_j - \left(\frac{H_{v(i,j-\frac{1}{2})} + \eta_{(j-1)}^n}{1 + \Delta t (\tau_y^s)_{(i,j-\frac{1}{2})}^n} \right) \Delta_- \eta_j \right] \\ = -\frac{\Delta t}{\Delta x} (1-\theta) \left[\left(H_{u(i+\frac{1}{2},j)} + \eta_{(i)}^n \right) U_{(i+\frac{1}{2},j)}^n - \left(H_{u(i-\frac{1}{2},j)} + \eta_{(i-1)}^n \right) U_{(i-\frac{1}{2},j)}^n \right] \\ - \frac{\Delta t}{\Delta y} (1-\theta) \left[\left(H_{v(i,j+\frac{1}{2})} + \eta_{(j)}^n \right) V_{(i,j+\frac{1}{2})}^n - \left(H_{v(i,j-\frac{1}{2})} + \eta_{(j-1)}^n \right) V_{(i,j-\frac{1}{2})}^n \right] \\ - \frac{\Delta t}{\Delta x} \theta \left[\left(\frac{H_{u(i+\frac{1}{2},j)} + \eta_{(i)}^n}{1 + \Delta t (\tau_x^s)_{(i+\frac{1}{2},j)}^n} \right) F U_{(i+\frac{1}{2},j)}^n - \left(\frac{H_{u(i-\frac{1}{2},j)} + \eta_{(i-1)}^n}{1 + \Delta t (\tau_x^s)_{(i-\frac{1}{2},j)}^n} \right) F U_{(i-\frac{1}{2},j)}^n \right] \\ - \frac{\Delta t}{\Delta y} \theta \left[\left(\frac{H_{v(i,j+\frac{1}{2})} + \eta_{(j)}^n}{1 + \Delta t (\tau_y^s)_{(i,j+\frac{1}{2})}^n} \right) F V_{(i,j+\frac{1}{2})}^n - \left(\frac{H_{v(i,j-\frac{1}{2})} + \eta_{(j-1)}^n}{1 + \Delta t (\tau_y^s)_{(i,j-\frac{1}{2})}^n} \right) F V_{(i,j-\frac{1}{2})}^n \right] + \eta_{(i,j)}^n \end{aligned}$$

que é completamente análoga a (4.12), tomando $k = 1$, e às notações:

$$\begin{aligned} \Delta_+ \eta_i &= \left(\eta_{(i+1,j)}^{n+1} - \eta_{(i,j)}^{n+1} \right), \quad \Delta_- \eta_i = \left(\eta_{(i,j)}^{n+1} - \eta_{(i-1,j)}^{n+1} \right), \\ \Delta_+ \eta_j &= \left(\eta_{(i,j+1)}^{n+1} - \eta_{(i,j)}^{n+1} \right) \quad \text{e} \quad \Delta_- \eta_j = \left(\eta_{(i,j)}^{n+1} - \eta_{(i,j-1)}^{n+1} \right) \end{aligned}$$

onde $\eta_{(i)}^n \stackrel{def}{=} \eta_{(i,j)}^n$ ou $\eta_{(i)}^n \stackrel{def}{=} \eta_{(i-1,j)}^n$, e $\eta_{(j)}^n \stackrel{def}{=} \eta_{(i,j)}^n$ ou $\eta_{(j)}^n \stackrel{def}{=} \eta_{(i,j-1)}^n$ dependendo da direção das componentes horizontais da velocidade. Essas duas condições são necessárias para estabelecer a propriedade CTC discutida no anexo 1.

Mas dada a importância de modelos 2D-IV e dado que eles têm uma estrutura de dados muito mais simples que os 3D, quando são implementados diretamente em duas dimensões. Além disso, as soluções paralelas apresentadas e discutidas na seção 7.10 (Thomas *pipeline*) são específicas para modelos 2D, quando das soluções obtidas empregando a abordagem ADI. E, portanto, não são passíveis de serem comparadas com as soluções de modelos 3D, mesmo que esse esteja com apenas uma camada, dado que não se implementou a abordagem ADI para o caso 3D.

4.4 ETM 3D na forma discreta

A EDP (2.9) é a equação de transporte escalar (ETM) que, de acordo com notações e definições anteriormente introduzidas, pode ser escrita na forma discreta como [GRO98], [GRO99] e [ROS2000]:

$$\left. \begin{aligned} & \underbrace{\frac{\partial(S\Delta z)}{\partial t}}_{\text{variação temporal}} + \underbrace{\frac{\partial(uS\Delta z)}{\partial x} + \frac{\partial(vS\Delta z)}{\partial y}}_{\text{adveccao horizontal}} + \underbrace{\frac{\partial(wS\Delta z)}{\partial z}}_{\text{adveccao vertical}} = \underbrace{\frac{\partial}{\partial z} \left(\varepsilon_v \frac{\partial(S\Delta z)}{\partial z} \right)}_{\text{difusao vertical}} + \\ & \underbrace{\frac{\partial}{\partial x} \left(\varepsilon_h \frac{\partial(S\Delta z)}{\partial x} \right) + \frac{\partial}{\partial y} \left(\varepsilon_h \frac{\partial(S\Delta z)}{\partial y} \right)}_{\text{difusao horizontal}} + \underbrace{\frac{KS\Delta z}{\text{decaimento}} + \frac{F\Delta z}{\text{fonte}}}_{\text{decaimento fonte}} \end{aligned} \right\} \quad (4.13)$$

onde $\varepsilon_h = \varepsilon_h(x_i, y_j, z_k, t^n)$ e $\varepsilon_v = \varepsilon_v(x_i, y_j, z_k, t^n)$ em (4.13) são os coeficientes de difusão turbulenta horizontal e de difusão turbulenta vertical, respectivamente.

A ETM discreta é obtida considerando que a advecção horizontal é interpolada via fluxo numérico nas faces; a advecção vertical é interpolada via fluxos numéricos nas faces; a difusão vertical é calculada através do método Crank-Nicolson; a difusão horizontal é calculada via diferenças centrais; a fonte e o decaimento são obtidos em termos de concentração e de uma matriz de decaimento, respectivamente, como definido na seção 2.5.

Especificamente, tomando $\varphi^{n+\theta} = \theta\varphi^{n+1} + (1-\theta)\varphi^n$ em que $0 < \theta \leq 1$, um esquema de discretização em diferenças finitas via fluxo numérico para a advecção fornece as seguintes expressões discretas para (4.13):

1. $\left(\frac{\partial(S\Delta z)}{\partial t} \right)_{(i,j,k)}^{n+1} \approx \frac{S_{(i,j,k)}^{n+1} \Delta z_{(i,j,k)}^{n+1} - S_{(i,j,k)}^n \Delta z_{(i,j,k)}^n}{\Delta t};$
2. $\left(\frac{\partial(uS\Delta z)}{\partial x} \right)_{(i,j,k)}^{n+\theta} \approx \frac{u_{(i+\frac{1}{2},j,k)}^{n+\theta} S_E \Delta z_{(i+\frac{1}{2},j,k)}^n - u_{(i-\frac{1}{2},j,k)}^{n+\theta} S_W \Delta z_{(i-\frac{1}{2},j,k)}^n}{\Delta x};$
3. $\left(\frac{\partial(vS\Delta z)}{\partial y} \right)_{(i,j,k)}^{n+\theta} \approx \frac{v_{(i,j+\frac{1}{2},k)}^{n+\theta} S_N \Delta z_{(i,j+\frac{1}{2},k)}^n - v_{(i,j-\frac{1}{2},k)}^{n+\theta} S_S \Delta z_{(i,j-\frac{1}{2},k)}^n}{\Delta y};$
4. $\left(\frac{\partial(wS\Delta z)}{\partial z} \right)_{(i,j,k)}^{n+\theta} \approx \frac{w_{(i,j,k+\frac{1}{2})}^{n+\theta} S_U \Delta z_{(i,j,k+\frac{1}{2})}^n - w_{(i,j,k-\frac{1}{2})}^{n+\theta} S_L \Delta z_{(i,j,k-\frac{1}{2})}^n}{\Delta z_{(i,j,k)}^n};$
5. $\left[\frac{\partial}{\partial z} \left(\varepsilon_v \frac{\partial(S\Delta z)}{\partial z} \right) \right]_{(i,j,k)}^{n+1} \approx \varepsilon_{(i,j,k+\frac{1}{2})} \left(S_{(i,j,k+1)}^{n+1} - S_{(i,j,k)}^{n+1} \right) - \varepsilon_{(i,j,k-\frac{1}{2})} \left(S_{(i,j,k)}^{n+1} - S_{(i,j,k-1)}^{n+1} \right) \\ + \varepsilon_{(i,j,k+\frac{1}{2})} \left(S_{(i,j,k+1)}^n - S_{(i,j,k)}^n \right) - \varepsilon_{(i,j,k-\frac{1}{2})} \left(S_{(i,j,k)}^n - S_{(i,j,k-1)}^n \right)$

$$\text{onde } \varepsilon_{(i,j,k+\frac{1}{2})} \stackrel{\text{not}}{\equiv} \frac{\varepsilon_{v(i,j,k+\frac{1}{2})}}{2\left(\Delta z_{(i,j,k+\frac{1}{2})}^n\right)^2} \text{ e } \varepsilon_{(i,j,k-\frac{1}{2})} \stackrel{\text{not}}{\equiv} \frac{\varepsilon_{v(i,j,k-\frac{1}{2})}}{2\left(\Delta z_{(i,j,k-\frac{1}{2})}^n\right)^2};$$

$$6. \left[\frac{\partial}{\partial x} \left(\varepsilon_h \frac{\partial (S\Delta z)}{\partial x} \right) \right]_{(i,j,k)}^n \approx \frac{\varepsilon_h \Delta z_{x(i,j,k)}^n}{2\Delta x^2} \left(S_{(i+1,j,k)}^n - 2S_{(i,j,k)}^n + S_{(i,j-1,k)}^n \right);$$

$$7. \left[\frac{\partial}{\partial y} \left(\varepsilon_h \frac{\partial (S\Delta z)}{\partial y} \right) \right]_{(i,j,k)}^n \approx \frac{\varepsilon_h \Delta z_{y(i,j,k)}^n}{2\Delta y^2} \left(S_{(i,j+1,k)}^n - 2S_{(i,j,k)}^n + S_{(i,j-1,k)}^n \right);$$

$$8. F\Delta z \approx \Delta t \Delta z_{(i,j,k)}^n F_{\text{fonte}}^n;$$

$$9. KS\Delta z \approx \Delta t S_{(i,j,k)}^n \Delta z_{(i,j,k)}^n K.$$

Note-se que todas as discretizações consideram a propriedade consistência da continuidade com o transporte (CTC), como discutido no anexo 1. Isso requer que os fluxos nas faces da equação da continuidade (EC) sejam aproximados de modo análogo aos fluxos nas faces da ETM, indicando que o esquema de solução da ETM deve ser tal que o nível de tempo e a posição das variáveis dependentes na malha sejam iguais aos da expressão EC discreta. As diferenças devem ser restritas aos tipos de aproximações considerados para a solução da ETM e da EC ([GRO98] e [BON2000]). Uma condição necessária para a propriedade CTC é que os espaçamentos verticais devem ser tais que:

$$\Delta z^{n+1} \stackrel{\text{def}}{\equiv} \Delta z_{(M+\frac{1}{2},j,k)}^{n+1} + \eta_{\left(i+\frac{I+1}{2},j\right)}^{n+1} \text{ e } \Delta z^{n+1} \stackrel{\text{def}}{\equiv} \Delta z_{(i,M+\frac{1}{2},k)}^{n+1} + \eta_{\left(i,j+\frac{J+1}{2}\right)}^{n+1}$$

em que $I = -\text{sign}\left(u_{(M+\frac{1}{2},j,k)}^{n+1}\right)$ e $J = -\text{sign}\left(v_{(i,M+\frac{1}{2},k)}^{n+1}\right)$, que fornecem expressões consistentes com os fluxos numéricos que são definidos para velocidades arbitrárias como detalhado no capítulo 6.

Assim, a ETM 3D é aproximada considerando uma abordagem massa conservativa (advecção na forma fluxo) como:

$$\left. \begin{aligned} S_{(i,j,k)}^{n+1} \Delta z_{(i,j,k)}^{n+1} &= S_{(i,j,k)}^n \Delta z_{(i,j,k)}^n - \\ &- \frac{\Delta t}{\Delta z_{(i,j,k)}^n} \left(\Delta z_{(i,j,k+\frac{1}{2})}^n w_{(i,j,k+\frac{1}{2})}^{n+\theta} S_U - \Delta z_{(i,j,k-\frac{1}{2})}^n w_{(i,j,k-\frac{1}{2})}^{n+\theta} S_L \right) \\ &- \frac{\Delta t}{\Delta x} \left(\Delta z_{(i+\frac{1}{2},j,k)}^n u_{(i+\frac{1}{2},j,k)}^{n+\theta} S_E - \Delta z_{(i-\frac{1}{2},j,k)}^n u_{(i-\frac{1}{2},j,k)}^{n+\theta} S_W \right) \\ &- \frac{\Delta t}{\Delta y} \left(\Delta z_{(i,j+\frac{1}{2},k)}^n v_{(i,j+\frac{1}{2},k)}^{n+\theta} S_N - \Delta z_{(i,j-\frac{1}{2},k)}^n v_{(i,j-\frac{1}{2},k)}^{n+\theta} S_S \right) \\ &+ \frac{\Delta t \varepsilon_{v(i,j,k+\frac{1}{2})}}{2\Delta z^2} \left(S_{(i,j,k+1)}^{n+1} - S_{(i,j,k)}^{n+1} \right) - \frac{\Delta t \varepsilon_{v(i,j,k-\frac{1}{2})}}{2\Delta z^2} \left(S_{(i,j,k)}^{n+1} - S_{(i,j,k-1)}^{n+1} \right) \\ &+ \frac{\Delta t \varepsilon_{v(i,j,k+\frac{1}{2})}}{2\Delta z^2} \left(S_{(i,j,k+1)}^n - S_{(i,j,k)}^n \right) - \frac{\Delta t \varepsilon_{v(i,j,k-\frac{1}{2})}}{2\Delta z^2} \left(S_{(i,j,k)}^n - S_{(i,j,k-1)}^n \right) \\ &+ \Delta t S_{(i,j,k)}^n \Delta z_{(i,j,k)}^n K + \Delta t \Delta z_{(i,j,k)}^n S_{\text{fonte}}^n \end{aligned} \right\} (4.14)$$

Na expressão (4.14) tem-se que, em geral, $\Delta z_{(i,j,k+\frac{1}{2})}^n = \Delta z_{(i,j,k-\frac{1}{2})}^n = \Delta z_{(i,j,k)}^n$ exceto, possivelmente, nas faces verticais que interceptam a batimetria na camada inferior e/ou cruzam a superfície livre na camada superior. Note-se, porém, que o

espaçamento vertical entre as k camadas deve levar em conta a variação na espessura devido ao movimento do nível de água, indicando que em $k=M$ tem-se $\Delta z_{k=M}^n = \Delta z_M^n + \eta_{(i,j)}^n$. Assim, os termos da advecção vertical e da difusão vertical em (4.14) são escritos como:

$$\frac{\partial(\omega S \Delta z)}{\partial z} \approx (\omega S)_{k+\frac{1}{2}} - (\omega S)_{k-\frac{1}{2}}$$

cuja expressão discreta em (4.14) é como $(w_{(i,j,k+\frac{1}{2})}^{n+\theta} S_U - w_{(i,j,k-\frac{1}{2})}^{n+\theta} S_L)$, e então:

$$\begin{aligned} \frac{\partial}{\partial z} \left(\varepsilon_v \frac{\partial S}{\partial z} \right) &= \left[\left(\varepsilon_v \frac{\partial S}{\partial z} \right)_S - \left(\varepsilon_v \frac{\partial S}{\partial z} \right)_F \right] = \\ &= \left(\varepsilon_{v(i,j,k+\frac{1}{2})} \frac{S_{(i,j,k+1)}^{n+1} - S_{(i,j,k)}^{n+1}}{2\Delta z^2} + \varepsilon_{v(i,j,k-\frac{1}{2})} \frac{S_{(i,j,k)}^{n+1} - S_{(i,j,k-1)}^{n+1}}{2\Delta z^2} \right) \\ &\quad - \left(\varepsilon_{v(i,j,k+\frac{1}{2})} \frac{S_{(i,j,k+1)}^n - S_{(i,j,k)}^n}{2\Delta z^2} + \varepsilon_{v(i,j,k-\frac{1}{2})} \frac{S_{(i,j,k)}^n - S_{(i,j,k-1)}^n}{2\Delta z^2} \right) \\ &= \frac{\varepsilon_{v(i,j,k+\frac{1}{2})}}{2 \left(\Delta z_{(i,j,k+\frac{1}{2})}^n \right)^2} \left(S_{(i,j,k+1)}^{n+1} - S_{(i,j,k)}^{n+1} \right) - \frac{\varepsilon_{v(i,j,k-\frac{1}{2})}}{2 \left(\Delta z_{(i,j,k-\frac{1}{2})}^n \right)^2} \left(S_{(i,j,k)}^{n+1} - S_{(i,j,k-1)}^{n+1} \right) \\ &\quad + \frac{\varepsilon_{v(i,j,k+\frac{1}{2})}}{2 \left(\Delta z_{(i,j,k+\frac{1}{2})}^n \right)^2} \left(S_{(i,j,k+1)}^n - S_{(i,j,k)}^n \right) - \frac{\varepsilon_{v(i,j,k-\frac{1}{2})}}{2 \left(\Delta z_{(i,j,k-\frac{1}{2})}^n \right)^2} \left(S_{(i,j,k)}^n - S_{(i,j,k-1)}^n \right) \end{aligned}$$

Porém, por conveniência algorítmica, a forma discreta empregada é como em (4.14). Essa conveniência deve-se ao fato de que nessa situação, pode-se escrever os algoritmos de solução em termos dos números de Courant locais $C_z = \omega \Delta t / \Delta z_k^n$, que é uma importante métrica para a análise dos efeitos da discretização espacial. Isso porque expressa a relação entre a magnitude da velocidade de fluxo, o tamanho do intervalo de tempo Δt e o tamanho da discretização espacial.

Note-se que muitas vezes o termo de difusão horizontal é desprezado, já que os coeficientes de difusão turbulenta podem ser considerados como sendo nulos, como em [GRO98], [GRO99] e [ROS2000]. Isso se justifica quando os mecanismos de transporte influenciados pelas marés são puramente advectivos, sendo que os efeitos do vento e da difusão são desprezíveis. Porém, optou-se por considerar o termo de difusão nos cálculos já que a ênfase é, neste trabalho, apenas nos aspectos numéricos e computacionais das EDPs do escoamento e do transporte.

Com essas escolhas, a discretização da ETM 3D resulta em $N_x N_y$ sistemas de equações 3-diagonal, uma para cada coluna de água, cada qual com $M - m + 1$ equações, pois o esquema numérico resultante é semi-implícito, e somente o transporte vertical é implícito. Essa é uma abordagem particularmente útil para obter a paralelização da solução.

4.5 Condições de contorno discretas: ETM 3D

As condições de contorno superior e inferior para a ETM são obtidas considerando que não existe massa atravessando as superfícies sólidas ou a superfície livre. Ou seja, nesse caso, as CC de não escoamento na superfície livre e na interface água-sedimento, são tais que:

$$w_{(i,j,M+\frac{1}{2})}^{n+\theta} S_{U(i,j,M+\frac{1}{2})} \equiv 0 \text{ e } w_{(i,j,m-\frac{1}{2})}^{n+\theta} S_{L(i,j,m-\frac{1}{2})} \equiv 0$$

Essas CC referem-se ao transporte vertical que é implícito e, portanto, essas condições são incorporadas aos sistemas de equações. Os demais termos, que são explícitos, particularmente aqueles da advecção horizontal que é aproximada com um estêncil computacional tipo 9-pontos, necessitam somente de tratamento especial em células que distam de duas células e em células que são adjacentes às fronteiras sólidas e/ou no caso das fronteiras abertas que delimitam o domínio computacional. Isso requer uma análise sistemática das diversas aproximações, questão tratada na seção 6.5.1.

4.6 ETM 2D-IV na forma discreta

Note-se que a ETM 3D discreta não se reduz, quando em uma só camada, à ETM 2D-IV, como no caso das ESW. Isso se deve ao fato de que a ETM 2D é discretizada na forma fluxo semi-implicitamente, e a ETM 3D é discretizada na forma semi-implícita de modo que em uma só camada o esquema numérico 3D torna-se completamente explícito. Mas a ETM 2D-IV é semi-implícita e é escrita na forma discreta como:

$$\begin{aligned} S_{(i,j)}^{n+1} H_{(i+\frac{1}{2},j)}^{n+1} &= S_{(i,j)}^n H_{(i+\frac{1}{2},j)}^n + S_{(i,j)}^n H_{(i,j)}^n K\Delta t + H_{(i,j)}^n S_{\text{fonte}}^n \Delta t + S_{(i,j)}^n H_{(i+\frac{1}{2},j)}^n \\ &\quad - \frac{\Delta t}{\Delta x} \left(U_{(i+\frac{1}{2},j)}^{n+1} S_E H_{(i+\frac{1}{2},j)}^{n+1} - U_{(i-\frac{1}{2},j)}^{n+1} S_W H_{(i-\frac{1}{2},j)}^{n+1} \right) \\ &\quad - \frac{\Delta t}{\Delta y} \left(V_{(i,j+\frac{1}{2})}^{n+1} S_N H_{(i,j+\frac{1}{2})}^{n+1} - V_{(i,j-\frac{1}{2})}^{n+1} S_S H_{(i,j-\frac{1}{2})}^{n+1} \right) \\ &\quad + \frac{\Delta t D_{x(i+\frac{1}{2},j)}^{n+1}}{2\Delta x^2} \left(S_{(i+1,j)}^{n+1} H_{(i+\frac{3}{2},j)}^{n+1} - S_{(i,j)}^{n+1} H_{(i+\frac{1}{2},j)}^{n+1} \right) \\ &\quad - \frac{\Delta t D_{x(i-\frac{1}{2},j)}^{n+1}}{2\Delta x^2} \left(S_{(i,j)}^{n+1} H_{(i+\frac{1}{2},j)}^{n+1} - S_{(i-1,j)}^{n+1} H_{(i-\frac{1}{2},j)}^{n+1} \right) \\ &\quad + \frac{\Delta t D_{x(i+\frac{1}{2},j)}^n}{2\Delta x^2} \left(S_{(i+1,j)}^n H_{(i+\frac{3}{2},j)}^n - S_{(i,j)}^n H_{(i+\frac{1}{2},j)}^n \right) \\ &\quad - \frac{\Delta t D_{x(i-\frac{1}{2},j)}^n}{2\Delta x^2} \left(S_{(i,j)}^n H_{(i+\frac{1}{2},j)}^n - S_{(i-1,j)}^n H_{(i-\frac{1}{2},j)}^n \right) \\ &\quad + \frac{\Delta t D_{y(i+\frac{1}{2},j)}^{n+1}}{2\Delta y^2} \left(S_{(i,j+1)}^{n+1} H_{(i,j+\frac{3}{2})}^{n+1} - S_{(i,j)}^{n+1} H_{(i,j+\frac{1}{2})}^{n+1} \right) \\ &\quad - \frac{\Delta t D_{y(i,j-\frac{1}{2})}^{n+1}}{2\Delta y^2} \left(S_{(i,j)}^{n+1} H_{(i,j+\frac{1}{2})}^{n+1} - S_{(i,j-1)}^{n+1} H_{(i,j-\frac{1}{2})}^{n+1} \right) \\ &\quad + \frac{\Delta t D_{y(i+\frac{1}{2},j)}^n}{2\Delta y^2} \left(S_{(i,j+1)}^n H_{(i,j+\frac{3}{2})}^n - S_{(i,j)}^n H_{(i,j+\frac{1}{2})}^n \right) \\ &\quad - \frac{\Delta t D_{y(i,j-\frac{1}{2})}^n}{2\Delta y^2} \left(S_{(i,j)}^n H_{(i,j+\frac{1}{2})}^n - S_{(i,j-1)}^n H_{(i,j-\frac{1}{2})}^n \right) \end{aligned}$$

sendo que $D_{y(i,j\pm\frac{1}{2})}^{n,n+1}$ e $D_{x(i\pm\frac{1}{2},j)}^{n,n+1}$ são os coeficientes de dispersão longitudinal e transversal discretos dados na seção 2.6. Porém, a componente advectiva da ETM é aproximada via esquema fluxo numérico tomando nesta abordagem os fluxos como

sendo advectados pelo esquema *upwind* de primeira ordem. A difusão horizontal é aproximada via Crank-Nicolson.

4.7 Modelo de turbulência na forma discreta

O modelo de parametrização de turbulência descrito na seção 2.7.2 é apresentado nessa seção na forma discreta. A sua discretização considerou os parâmetros empregados em um corpo de água bem misturado. Usando as notações e parâmetros introduzidos na seção 2.7.2, toma-se que $\alpha = 0,007$. Desse modo, e denotando o comprimento de mistura horizontal por $L = \Delta x = \Delta y$, onde Δx e Δy são os espaçamentos horizontais da malha da hidrodinâmica, os coeficientes da viscosidade cinemática turbulenta horizontal são constantes, e são tais que $\nu_h = 0,007(\Delta x)^{4/3}$ para as duas equações de conservação da quantidade de movimento do sistema de EDPs *shallow water*.

Os coeficientes de difusibilidade turbulenta horizontal para a equação do transporte escalar de substâncias é considerado, pelos motivos discutidos na seção 2.7.2, constante e igual a $\varepsilon_h \approx 0,2$. Essa escolha é aplicável às difusões nas direções X e Y .

Os coeficientes da viscosidade cinemática turbulenta vertical das ESW são obtidos como em [KIM94]. Considera-se que na expressão do comprimento de mistura vertical L_m , que o atrito interno γ é tal que $[\xi(h/H)\Psi(R_i)] \approx \gamma$, em que $\gamma^2 \approx 0,001$, e onde a altura é dada pela k -ésima camada, isto é, tomando $H \approx \Delta z_k^n$. Portanto, tem-se:

$$\nu_v \equiv \nu_v(x, y, z, t) = 0,001(\Delta z_k^n)^2 \sqrt{[(\partial u / \partial z)^2 + (\partial v / \partial z)^2]}$$

onde toma-se $\Delta z_{(i,j,k)}^n$ como sendo a média $\Delta z_{(i,j,k)}^n = 0,5(\Delta z_{(i+1/2,j,k)}^n + \Delta z_{(i,j+1/2,k)}^n)$.

Já os coeficientes de difusibilidade turbulenta vertical da ETM são obtidos como:

$$\varepsilon_v \equiv \varepsilon_v(x, y, z, t) = 1,4\nu_v \exp(-0,096)$$

onde o número de Richardson é tomado como $R_i = 0,08$, já que o corpo de água é considerado bem misturado, para efeitos de simulação neste trabalho.

As CC para a ETM são obtidas considerando-se que devem ser do tipo não escoamento na superfície livre e na interface água-sedimento, como detalhado na seção 4.5. Já as condições de contorno para as ECQM_x são obtidas considerando-se o transporte vertical que, conforme os casos analisados na seção 4.1, fornece, para $k = M$ (fronteira superior), que:

$$\varepsilon_{v(i+1/2,j,M+1/2)} \frac{u_{(i+1/2,j,M+1)}^{n+1} - u_{(i+1/2,j,M)}^{n+1}}{\Delta z_{(i+1/2,j,M+1/2)}^n} \stackrel{CC superior}{=} (\tau_x^v)_{(i+1/2,j,M+1/2)}^n$$

e, para $k = m$ (fronteira inferior), vale:

$$\varepsilon_{v(i+1/2,j,m-1/2)} \frac{u_{(i+1/2,j,m)}^{n+1} - u_{(i+1/2,j,m-1)}^{n+1}}{\Delta z_{(i+1/2,j,m-1/2)}^n} \stackrel{CC inferior}{=} (\gamma_x)_{(i+1/2,j,m)}^n u_{(i+1/2,j,m)}^{n+1}$$

Analogamente, tem-se na ECQM_y, para $k = M$ (fronteira superior), que:

$$\varepsilon_{v(i,j+\frac{1}{2},M-\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},M)}^{n+1} - v_{(i,j+\frac{1}{2},M-1)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},M-\frac{1}{2})}^n} \stackrel{CC superior}{=} \left(\tau_y^v \right)_{(i,j+\frac{1}{2},M+\frac{1}{2})}^n$$

e, para $k = m$ (fronteira inferior), que:

$$\varepsilon_{v(i,j+\frac{1}{2},m-\frac{1}{2})} \frac{v_{(i,j+\frac{1}{2},m)}^{n+1} - v_{(i,j+\frac{1}{2},m-1)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},m-\frac{1}{2})}^n} \stackrel{CC inferior}{=} \left(\gamma_y \right)_{(i,j+\frac{1}{2},m)}^n v_{(i,j+\frac{1}{2},m)}^{n+1}$$

Portanto a forma discreta de $\sqrt{\left[(\partial u / \partial z)^2 + (\partial v / \partial z)^2 \right]}$ é

$$\left(\frac{\partial u}{\partial z} \right)_{(i,j,k)} \approx \frac{u_{(i+\frac{1}{2},j,k+1)}^{n+1} - u_{(i+\frac{1}{2},j,k)}^{n+1}}{\Delta z_{(i+\frac{1}{2},j,k+\frac{1}{2})}^n} e \left(\frac{\partial v}{\partial z} \right)_{(i,j,k)} \approx \frac{v_{(i,j+\frac{1}{2},M)}^{n+1} - v_{(i,j+\frac{1}{2},M-1)}^{n+1}}{\Delta z_{(i,j+\frac{1}{2},M-\frac{1}{2})}^n}$$

válida para $m \leq k \leq M - 1$, onde $\Delta z_{(i+\frac{1}{2},j+\frac{1}{2},k)}^n = 0,5 \left(\Delta z_{(i+\frac{1}{2},j+\frac{1}{2},k+1)}^n + \Delta z_{(i+\frac{1}{2},j+\frac{1}{2},k)}^n \right)$.

Note-se que deve-se construir esquemas numéricos que tenham uma viscosidade e/ou difusão numérica tais que não obscureçam os coeficientes das viscosidades cinemática turbulenta e das difusibilidade turbulenta calculados pelo modelo de parametrização, pois esse fornece as viscosidades e as difusões físicas para o modelo computacional. Essas questões são discutidas no anexo 1.

5 Métodos numéricos

Muitos dos métodos clássicos de soluções para problemas sob o escopo da CFD, são discutidos em detalhes por Patankar [PTK81], Fletcher [FLE88] e [FLE88a], Anderson [AND95] e Ferziger e Peric [FER97], entre outros. Pode-se destacar, também, os textos de Fortuna “Técnicas Computacionais para Dinâmica dos Fluidos” [FRT2000], e de De Bortoli “Introdução à Dinâmica de Fluidos Computacional” [BRL2000], dois textos nacionais sobre técnicas numéricas e computacionais para solução de escoamentos. Porém, o enfoque nesse capítulo é discutir abordagens para a construção de soluções de alta qualidade numérica para os termos advectivos das EDPs hiperbólicas, já que esses são os termos “mais críticos” dessas EDPs.

Como discutido no anexo 1, não existem resultados teóricos gerais para a análise da convergência da solução numérica de EDPs não lineares, pois, nesse caso, o teorema de Lax não é aplicável. É necessário construir e mostrar que um método particular de solução é convergente. Na busca por soluções de alta qualidade numérica, desenvolveram-se muitas soluções. Por exemplo, Rood [ROO87] mostrou que, já na década de 80, existiam mais de 100 algoritmos advectivos. Outros trabalhos, como os de Baptista [BAP95] e Gross [GRO99,] mostraram outras coleções de algoritmos mais atuais. LeVeque [LEV96] e Sons [SON2000] discutem, também, vários algoritmos para soluções com alta qualidade numérica da advecção em escoamentos incompressíveis.

Uma vez que existem muitos métodos disponíveis na literatura técnica, objetivase, neste trabalho, construir algoritmos para a solução numérica dos modelos computacionais, tendo como base aqueles com sólida fundamentação matemática, não considerando os desenvolvidos sob abordagens puramente heurísticas. Uma referência para balisar essa escolha, que deve ter como compromisso o equilíbrio entre a qualidade numérica e a eficiência computacional, é o trabalho de Byun, Young, Pleim, Odman e Alapaty [BYU99], que analisaram e compararam uma gama de métodos para a solução da advecção 1D.

Tais métodos são os MPDATA (*Multidimensional Positive-Definite Advective Transport Algorithm*) de Smolarkiewicz [SMO82], [SMO83] e [SMO84], o ARF (*Area preserving Flux-Form Advection Algorithm*) de Bott [BOT89], [BOT89a]; o *Piecewise Parabolic Methods* (PPM) de Colella e Woodward [COL84]; o Yamartino (YAM); *Flux Corrected Transport* (FCT) de Boris e Book [BOR73], [BO075] e [BOR76]; o semi-Lagrangeano (SLT); o *Accurate Space Derivative* (ASD), e o *Chapeau Function Method* (HAT). Os resultados indicaram que as soluções obtidas por esses métodos, quanto à qualidade numérica, têm as seguintes características:

1. ARF: é acurado e massa conservativo, porém pode criar *ripples* (rampas ou oscilações) e *overshoots* (sobre-especificação), mas é definido-positivo;
2. ASD: é acurado, porém não é estritamente massa conservativo, não é monotônico mas, geralmente, fornece resultados positivos;
3. FCT: é acurado e massa conservativo, e não cria quaisquer *ripples* e *overshoots*, mas tem certa difusão numérica;
4. HAT: é acurado e massa conservativo, mas pode levar a *ripples*, que podem crescer e causar instabilidade;
5. MPDATA: é de acurácia relativamente baixa, é massa conservativo, mas pode levar a *ripples* em campos de alta concentração;
6. PPM: é acurado, massa conservativo, monotônico e tem pequena difusão numérica, mas o pico de *clipping* (pontos críticos) pode ser significativo;
7. SLT: tem menor acurácia e problemas de conservação de massa; a sua única característica positiva é que não tem *ripples*;

8. YAM: é muito acurado para comprimentos de ondas curtos e é massa conservativo, mas pode levar a *overshoots*.

O tempo de execução de tais métodos foram normalizados com respeito ao esquema de SMO. Nessa situação, os tempos de execução de CPU desses métodos aplicados à solução da EDP da advecção linear 1D mostraram que a única solução menos custosa que a MPDATA foi a ARF. Os outros esquemas foram mais custosos, sendo que o mais custoso foi o ASD, que exigiu quatro vezes mais tempo de CPU que o MPDATA. Não obstante a eficiência computacional dos métodos MPDATA e ARF, um exame na literatura técnica, particularmente aquela de solução numérica para escoamentos de superfície livre, como em [GOM95], [LAN97], [BYU99] e [GRO99], mostrou que os métodos de Smolarkiewicz e de Bott estão se tornando populares devido a seus atributos, já que Smolarkiewicz e de Bott desenvolveram versões monotônicas e mais acuradas de seus métodos.

A versão MARF (*area monotone preserving flux-form advection algorithm*) de Bott ([BOT92] e [BOT93]) emprega a abordagem FCT para limitar os fluxos; de modo a minimizar os erros que surgem quando do emprego do modo de divisão do tempo (*time-splitting*). Além disso, Chlond [CHL94] construiu uma versão localmente modificada do método de Bott, na qual são introduzidas funções monitores auto-ajustáveis para obter um esquema monotônico, conservativo e com pouca difusão numérica. Isso é obtido combinando o algoritmo de Bott com um esquema de interpolação exponencial. Já Smolarkiewicz desenvolveu uma versão não oscilatória do MPDATA, o chamado *Multidimensional Positive Definite Advection Transport Algorithm: Nonoscillatory Option* [SMO90].

No entanto neste trabalho a opção foi pelo emprego de métodos que se mostram robustos sob quaisquer situações. Um exemplo particularmente importante desse caso é o método da limitação dos fluxos (*Flux-Limiter Methods*) de Sweby [SWE84]. Tal método, como a exemplo do FCT e do PPM, pertence à classe TVD (*Total Variation Diminishing*), que é um dos conceitos fundamentais dos mais modernos e sofisticados métodos de solução numérica, que são discutidos nos textos de Colella [COL98], Trangenstein [TRA2000], Lomax, Pulliam e Zingg [LOM2000], Benden, Brouwer, Mirani e Schrijvers [HEM2000] e de Levy [LEV2001].

O suporte teórico oferecido pelo conceito TVD e as intrínsecas características numéricas e computacionais do método FLM, discutidas no anexo 1, tornam essa abordagem particularmente importante na solução de problemas hiperbólicos com altos gradientes de concentração, como neste trabalho. Por essas razões, foi essa a abordagem escolhida para a solução do transporte escalar de substâncias 3D.

5.1 Método com limitação de fluxo: formulação de Sweby

A construção de esquemas lineares em diferenças finitas com alta ordem pode não ser o melhor caminho, pois os esquemas numéricos resultantes podem reduzir a dispersão numérica, mas também podem fornecer soluções fisicamente incorretas quando da solução de escoamentos hiperbólicos complexos, dado que muitas vezes a condição de entropia não é satisfeita [LOM2000]. Uma alternativa, como discutido nas seções precedentes, é construir esquemas numéricos não lineares que empreguem o conceito de diminuição (ou limitação) das variações totais, de modo a produzir soluções únicas e com alta resolução. Tais esquemas são obtidos, neste trabalho, empregando o método de limitação de fluxo desenvolvido de Sweby.

Um modo de se obter um esquema com limitação de fluxo que seja TVD é adicionar um termo de correção, visando a eliminar as oscilações que aparecem no entorno das descontinuidades [CUM2000]. Essa correção parte da noção de que um método de primeira ordem não gera oscilações porque o termo difusivo ou de viscosidade artificial presente é suficientemente grande para suavizar a solução

numérica. Isso não acontece com um método de segunda ordem que é apropriado para regiões de solução suave. Porém, tais métodos introduzem oscilações no entorno de descontinuidades ou no entorno de altos gradientes.

Assim, a idéia é usar uma “média ponderada”, de um método conservativo de primeira ordem, com outro conservativo de segunda ordem, de modo que, próximo das descontinuidades, o peso maior é do método de primeira ordem e, portanto, para um fluxo de baixa ordem F_1 . Em regiões de solução suave, o peso é maior para o método de segunda ordem e, portanto, para um fluxo de alta ordem F_2 . O fluxo é calculado como $F = F_1 + \Psi(r)F_2$, onde Ψ é a função limitadora do fluxo. No caso geral, isso significa que um método não linear F^H pode ser visto como sendo um método linear F^L , mais uma correção. Nessa situação, a função fluxo F interpolante e limitada é escrita como $F = F^L + \Psi(F^H - F^L)$ ([LEV96] e [HEI99]), onde Ψ é a função de restrição (ou função limitante) de fluxo.

Um outro atributo dessa função é que ela deve reconhecer a localização das descontinuidades ou dos altos gradientes, para a limitação dos fluxos. Isso pode ser feito analisando a suavidade da solução através de funções monitores, que podem ser definidas através de gradientes da solução $\nabla\Phi$ em pontos consecutivos da malha. Assim, a função $\Psi \equiv \Psi(r)$ é escrita em função do quociente $r = \Phi_i - \Phi_{i-1} / \Phi_{i+1} - \Phi_i$, dependendo da direção do escoamento (veja a seção 5.1.2).

O método FLM considera a definição de fluxo numérico e usa o método fluxo dividido (*flux-splitting*) para integrar a EDP na forma conservativa. Se os fluxos numéricos não estão definidos nas faces das células, eles devem ser obtidos através de interpolação, e a escolha da função interpoladora fornece o tipo e a ordem da aproximação. O exemplo clássico de função interpolante é a função-K de van Leer [LER85]. A forma geral, para uma variável Φ , é como:

$$\Phi_{(i+\frac{1}{2})}^n = \Phi_{(i,j,k)}^n + \frac{1}{4} \left[(1+K) \left(\Phi_{(i+1,j,k)}^n - \Phi_{(i,j,k)}^n \right) + (1-K) \left(\Phi_{(i,j,k)}^n - \Phi_{(i-1,j,k)}^n \right) \right] \quad (5.1)$$

onde o parâmetro $-1 \leq K \leq 1$ em (5.1) é escolhido para definir um esquema em particular. Por exemplo, o esquema Fromm [FRO68] de segunda ordem é obtido com $K = 0$; o QUICK de Leonard [LEO79] é obtido com $K = 1/2$; o CUS (*Cubic Interpolation Scheme*), um esquema de terceira ordem, é obtido com $K = 1/3$; o *upwind* é obtido com $K = -1$; e o esquema em diferenças central com $K = 1$. Uma variedade de esquemas de interpolação de alta ordem pode ser construída usando como ponto de partida a função de interpolação de van Leer (5.1), como feito por [HUN95], [ZIJ95], [ZIJ97] e [HEI99], entre outros.

Porém, essas interpolações podem destruir a positividade devido ao surgimento de oscilações. Assim, deve-se estabelecer alguns critérios para manter a positividade das soluções. O critério, no método de Sweby [SWE84], é empregar funções limitadoras de fluxos, para as quais, mostram-se condições necessárias e suficientes para a limitação da variação total da solução. Note-se, porém, que a função limitante pode ser arbitrária, e a escolha desta ou daquela função não tem, ainda, uma firme base matemática. Elas são obtidas de modo empírico, por inspeção ou através de experiências numéricas, apesar de muitas dessas funções serem obtidas dentro do escopo da teoria.

5.1.1 Funções interpolantes

O método de limitação de fluxos empregado neste trabalho é o FLM (*flux-limiter methods*) formulado por Sweby [SWE84]. Nessa abordagem, considera-se que o valor do fluxo numérico Φ pode ser escrito como a soma de um termo difusivo *upwind* de primeira ordem com um termo antidifusivo. A parte antidifusiva de alta ordem é multiplicada por uma função limitadora de fluxo Ψ , que é função não linear, razão *upwind* de gradientes consecutivos. Ou seja, para escoamentos 1D na direção X , tem-se $\Psi \equiv \Psi(r_{i+\frac{1}{2}})$, onde:

$$r_{i+\frac{1}{2}} = \frac{\Phi_{i+1} - \Phi_i + \varepsilon}{\Phi_i - \Phi_{i-1} + \varepsilon}$$

sendo $\varepsilon \approx 10^{-9}$ um parâmetro para evitar a divisão por zero em escoamentos uniformes.

Com o emprego da formulação de Sweby, pode-se construir um esquema de solução tipo fluxo limitado, que emprega uma função Ψ limitadora de fluxo. Tal esquema, empregado em [ZIJ95], [HUN96] e [ZIJ97], é como:

$$\Phi_{(i+\frac{1}{2})} = \Phi_{(i)} + \frac{1}{2} \Psi(r_{i+\frac{1}{2}}) (\Phi_i - \Phi_{i-1}) \quad \left. \right\} (5.2)$$

Note-se que (5.2) não sofre nenhuma restrição para construir e incorporar esquemas tipo *upwind* parciais (*upwind biased*) para obter soluções acuradas dos esquemas numéricos, pois estes podem ser obtidos a partir da escolha de uma função limitante de fluxo.

Para obter uma solução TVD para a EDP da advecção, deve-se mostrar que o limite $\Psi(r)$ deve satisfazer as condições definidas pelo teorema de Harten. Nesse caso, mostra-se que, para velocidades constantes e positivas, (5.2) fornece uma solução TVD se a função de limitação de fluxo da EDP da advecção pura satisfaz a desigualdade $0 \leq \Psi(r) \leq 2r$ [ZIJ95]. Condições para outras funções de restrição podem ser vistas em [ZIJ97].

Além disso, esses resultados podem ser estendidos para englobar discretizações espaciais que consideram uma abordagem análoga ao método- θ , com $0 \leq \theta \leq 1$. Em tais discretizações o esquema resultante é explícito de primeira ordem quando $\theta = 0$. Esquemas completamente implícitos são obtidos quando $\theta = 1$, e quando $\theta = 0,5$ obtêm-se esquemas tipo Crank-Nicolson. No caso desse tipo de discretização, uma condição suficiente para (5.2) é dada por:

$$0 \leq \Psi(r) \leq \min(2r, M) \quad \left. \right\} (5.3)$$

onde $1 \leq M = -2 + (2/C_r(1-\theta))$ e, por acurácia, toma-se $M = 1$. No caso de $M = 2$, (5.3) fornece o critério desenvolvido originalmente por Sweby [SWE84].

Essas mesmas análises podem ser empregadas para obter funções limitadoras de fluxo específicas para a EDP da advecção-difusão. Se $C_r = u \Delta t / \Delta x$ denota o número de Courant do escoamento e se $P_e = |u| \Delta x / \varepsilon_h$ denota o número de Péclet da malha, o esquema-K não produz soluções com oscilações para a advecção-difusão, se cumpre a condição [ZIJ97]:

$$\left. \left\{ |u| \frac{\Delta x}{\varepsilon_h} = P_e \leq P_{e(critico)} = \frac{4}{(1+K)} \right\} \right\} (5.4)$$

O valor crítico $P_{e(critico)}$ é obtido exigindo-se que os elementos fora da diagonal principal, por envolverem contribuições dos termos advectivos e difusivos, sejam não negativos. Essa é uma condição suficiente para a supressão das oscilações numéricas [ZIJ97], que aparecem quando altos gradientes de concentração estão presentes em sub-regiões nas quais a advecção é dominante.

Note-se que o número de Péclet pode ser visto como uma métrica que estabelece o grau de como o transporte é dominado pela advecção [ZHE95], [CMO99]. Basta observar que, para problemas puramente advectivos, tem-se que $\varepsilon_h \rightarrow 0$ e, portanto, $P_e \rightarrow \infty$. Quando a difusão é significativa, o número de Péclet é pequeno. Mostra-se que, quando $P_e \leq 2$, as oscilações são eliminadas ([FLE88] e [FER97]). Quando $P_e > P_{e(critico)}$, deve-se adicionar alguma difusão artificial para que o esquema seja livre de oscilações.

Uma idéia é, portanto, obter funções limitadoras de fluxo para a equação da advecção-difusão, embora alguns trabalhos indiquem que essa abordagem não resulta em soluções com significativa melhora na acurácia [ZIJ95]. De qualquer modo, tais funções limitadoras podem ser construídas de modo análogo às condições dadas para a EDP da advecção pura, e que levem em conta a restrição (5.4). Nesse caso, para uma aproximação via diferenças centrais para a difusão, quando aplicada a condição (5.3), mostra-se que um esquema numérico para a EDP da advecção-difusão é TVD, se a função de limitação de fluxo Ψ cumpre a seguinte condição [ZIJ97]:

$$\frac{2r}{P_e} \leq \Psi(r) \leq \min \left[M + \frac{2r}{P_e}, \left(2 + \frac{2}{P_e} \right) r \right]$$

para $\forall M \geq 1$. Para $P_e \rightarrow \infty$, o critério (5.3) é recuperado.

Com essas considerações, é possível desenvolver para a equação da advecção-difusão uma classe de limites de fluxos, que são lineares por partes. O desenvolvimento é baseado no esquema-K e depende do número de Péclet. Essa classe é chamada por Zijlema [ZIJ97] de classe de K-limites de fluxos difusivos lineares por partes (DPL), e podem ser usados quando $P_e \geq 4(1+K)^{-1}$.

5.1.2 Funções interpolantes para velocidades arbitrárias

Dada a formulação de Sweby, nesta seção, são desenvolvidas expressões para interpolação dos valores nas faces das células, isto é, para a obtenção dos fluxos numéricos para a malha espaço-alternada Arakawa C ilustrada na fig. 3.1. Tais expressões são escritas, para velocidades positivas ou negativas, de modo a obter a solução da advecção horizontal da concentração $S_{(i,j)}$, e são como:

$$\Phi_{(i+\frac{1}{2},j)} = \begin{cases} \Phi_{(i,j)} + \frac{1}{2} \Psi(r_{(i+\frac{1}{2},j)}^+) (\Phi_{(i,j)} - \Phi_{(i-1,j)}) & , u_{(i+\frac{1}{2},j)} > 0 \\ \Phi_{(i+1,j)} - \frac{1}{2} \Psi(r_{(i+\frac{1}{2},j)}^-) (\Phi_{(i+2,j)} - \Phi_{(i+1,j)}) & , u_{(i+\frac{1}{2},j)} < 0 \end{cases} \quad (5.5)$$

em que:

$$r_{(i+\frac{1}{2},j)}^+ = \frac{\Phi_{(i+1,j)} - \Phi_{(i,j)} + \varepsilon}{\Phi_{(i,j)} - \Phi_{(i-1,j)} + \varepsilon} \quad e \quad r_{(i+\frac{1}{2},j)}^- = \frac{\Phi_{(i+1,j)} - \Phi_{(i,j)} + \varepsilon}{\Phi_{(i+2,j)} - \Phi_{(i+1,j)} + \varepsilon} \quad (5.6)$$

Note-se que a definição das funções monitores em (5.6) decorre do fato de que essas funções devem reconhecer a localização das descontinuidades e/ou dos oscilações. Essa condição é cumprida considerando o quociente de gradientes consecutivos da solução em pontos consecutivos da malha. Ou seja, o gradiente de concentração da face *upstream*, dividido pelo gradiente de concentração da segunda face *upstream* [AWA2000] (veja fig. 6.8). Assim, se $u_{(i+\frac{1}{2})} > 0$, pode-se definir, para uma variável dependente Φ qualquer que, em malhas equiespaçadas, tem-se:

$$\nabla\Phi_{(i)} \equiv \frac{\partial\Phi}{\partial x} = \frac{\Phi_{(i)} - \Phi_{(i-1)}}{\Delta x} \text{ e } \nabla\Phi_{(i+1)} \equiv \frac{\partial\Phi}{\partial x} = \frac{\Phi_{(i+1)} - \Phi_{(i)}}{\Delta x}$$

e, portanto, define-se $\nabla\Phi_{(i+1)}/\nabla\Phi_{(i)} = r_{(i+\frac{1}{2})}^+$. Analogamente, se $u_{(i+\frac{1}{2})} < 0$, então:

$$\nabla\Phi_{(i+1)} \equiv \frac{\partial\Phi}{\partial x} = \frac{\Phi_{(i+2)} - \Phi_{(i+1)}}{\Delta x} \text{ e } \nabla\Phi_{(i)} \equiv \frac{\partial\Phi}{\partial x} = \frac{\Phi_{(i+1)} - \Phi_{(i)}}{\Delta x}$$

e, então, define-se $\nabla\Phi_{(i)}/\nabla\Phi_{(i+1)} = r_{(i+\frac{1}{2})}^-$.

Note-se, antes de estender essas idéias ao caso multidimensional, que o estêncil (ou molécula) computacional que define a dependência de dados entre os valores nodais do esquema (5.5). Uma representação para o estêncil computacional da formulação FLM, caso 1D e direção X , é como dada na fig. 6.8. Seu emprego prático na obtenção de esquemas aproximativos pode ser visto em [AME77].

5.1.3 Esquemas de limitação de fluxo: caso multidimensional

Por razões de robustez e simplicidade algorítmica, que foram discutidas na anexo 1, os esquemas numéricos multidimensionais foram construídos através de decomposições unidimensionais na direção normal de cada uma das faces das células. Portanto, considerando as notações e definições anteriores, o esquema de interpolação 2D (advecção horizontal), na presença de um campo de velocidades arbitrário, é escrito como:

1. para a direção X e face S_E , tem-se:

$$\Phi_{(i+\frac{1}{2},j)} = \begin{cases} \Phi_{(i,j)} + \frac{1}{2} \Psi(r_{(i+\frac{1}{2},j)}^+) (\Phi_{(i,j)} - \Phi_{(i-1,j)}) & , u_{(i+\frac{1}{2},j)} > 0 \\ \Phi_{(i+1,j)} - \frac{1}{2} \Psi(r_{(i+\frac{1}{2},j)}^-) (\Phi_{(i+2,j)} - \Phi_{(i+1,j)}) & , u_{(i+\frac{1}{2},j)} < 0 \end{cases}$$

em que:

$$r_{(i+\frac{1}{2},j)}^+ = \frac{\Phi_{(i+1,j)} - \Phi_{(i,j)} + \varepsilon}{\Phi_{(i,j)} - \Phi_{(i-1,j)} + \varepsilon} \text{ e } r_{(i+\frac{1}{2},j)}^- = \frac{\Phi_{(i+1,j)} - \Phi_{(i,j)} + \varepsilon}{\Phi_{(i+2,j)} - \Phi_{(i+1,j)} + \varepsilon}$$

2. para a direção X e face S_W , tem-se:

$$\Phi_{(i-\frac{1}{2},j)} = \begin{cases} \Phi_{(i-1,j)} + \frac{1}{2} \Psi(r_{(i-\frac{1}{2},j)}^+) (\Phi_{(i-1,j)} - \Phi_{(i-2,j)}) & , u_{(i-\frac{1}{2},j)} > 0 \\ \Phi_{(i,j)} - \frac{1}{2} \Psi(r_{(i-\frac{1}{2},j)}^-) (\Phi_{(i+1,j)} - \Phi_{(i,j)}) & , u_{(i-\frac{1}{2},j)} < 0 \end{cases}$$

em que:

$$r_{(i-\frac{1}{2},j)}^+ = \frac{\Phi_{(i,j)} - \Phi_{(i-1,j)} + \varepsilon}{\Phi_{(i-1,j)} - \Phi_{(i-2,j)} + \varepsilon} \text{ e } r_{(i-\frac{1}{2},j)}^- = \frac{\Phi_{(i,j)} - \Phi_{(i-1,j)} + \varepsilon}{\Phi_{(i+1,j)} - \Phi_{(i,j)} + \varepsilon}$$

3. para a direção Y e face S_N , tem-se:

$$\Phi_{(i,j+\frac{1}{2})} = \begin{cases} \Phi_{(i,j)} + \frac{1}{2} \Psi \left(\lambda_{(i,j+\frac{1}{2})}^+ \right) \left(\Phi_{(i,j)} - \Phi_{(i,j-1)} \right) & , \nu_{(i,j+\frac{1}{2})} > 0 \\ \Phi_{(i,j+1)} - \frac{1}{2} \Psi \left(\lambda_{(i,j+\frac{1}{2})}^- \right) \left(\Phi_{(i,j+2)} - \Phi_{(i,j+1)} \right) & , \nu_{(i,j+\frac{1}{2})} < 0 \end{cases}$$

em que:

$$\lambda_{(i,j+\frac{1}{2})}^+ = \frac{\Phi_{(i,j+1)} - \Phi_{(i,j)} + \varepsilon}{\Phi_{(i,j)} - \Phi_{(i,j-1)} + \varepsilon} \text{ e } \lambda_{(i,j+\frac{1}{2})}^- = \frac{\Phi_{(i,j+1)} - \Phi_{(i,j)} + \varepsilon}{\Phi_{(i,j+2)} - \Phi_{(i,j+1)} + \varepsilon}$$

4. para a direção Y e face S_S , tem-se:

$$\Phi_{(i,j-\frac{1}{2})} = \begin{cases} \Phi_{(i,j-1)} + \frac{1}{2} \Psi \left(\lambda_{(i,j-\frac{1}{2})}^+ \right) \left(\Phi_{(i,j-1)} - \Phi_{(i,j-2)} \right) & , \nu_{(i,j-\frac{1}{2})} > 0 \\ \Phi_{(i,j)} - \frac{1}{2} \Psi \left(\lambda_{(i,j-\frac{1}{2})}^- \right) \left(\Phi_{(i,j+1)} - \Phi_{(i,j)} \right) & , \nu_{(i,j-\frac{1}{2})} < 0 \end{cases}$$

em que:

$$\lambda_{(i,j-\frac{1}{2})}^+ = \frac{\Phi_{(i,j)} - \Phi_{(i,j-1)} + \varepsilon}{\Phi_{(i,j-1)} - \Phi_{(i,j-2)} + \varepsilon} \text{ e } \lambda_{(i,j-\frac{1}{2})}^- = \frac{\Phi_{(i,j)} - \Phi_{(i,j-1)} + \varepsilon}{\Phi_{(i,j+1)} - \Phi_{(i,j)} + \varepsilon}$$

onde, como discutido na seção 5.1.2, as funções $r_{(i\pm\frac{1}{2},j)}^{+,-}$ e $\lambda_{(i,j\pm\frac{1}{2})}^{+,-}$ verificam se a monotonicidade do esquema numérico é violada. Por esse motivo, são chamadas de funções suavizantes ou monitores. Pode-se mostrar, denotando $\tau = r$ ou $\tau = \lambda$, que ocorrem as seguintes condições [TRA2000]:

1. se $\tau > 0$, então a solução numérica tem um comportamento monotônico, no ponto, crescente ou decrescente;
2. se $\tau < 0$, então a solução numérica tem um extremo local no ponto, devido às oscilações numéricas;
3. se $\tau \approx 1$, a solução numérica tem um comportamento suave no ponto;
4. se $\tau \approx 0$ ou $\tau \rightarrow \infty$, então a solução numérica tem uma descontinuidade no ponto.

Assim, se os gradientes têm o mesmo sinal ($\tau > 0$), a função limitadora de fluxo deve determinar uma média ponderada, como discutido na seção 5.1, gerando funções de interpolação de segunda ordem de acurácia para as faces das células. No caso de gradientes terem sinais diferentes ($\tau < 0$), ocorrem oscilações e, nesse caso, uma abordagem para tratar desse problema é tomar $\Psi(\tau) = 0$ de modo que o esquema é somente de primeira ordem no entorno de descontinuidades ou no entorno de altos gradientes. Isso significa especificar, quando $\tau > 0$, funções limitadoras que sejam não lineares e nos casos que isso não acontece, ou seja, quando $\tau \leq 0$, pode-se especificar funções limitadoras que sejam lineares de primeira ordem de acurácia. Além disso, essas funções limitadoras devem ser, em quaisquer casos, tais que os valores muito maiores do que a unidade sejam “naturalmente” limitados.

Note-s, porém, que Zijlema [ZIJ97] mostra que esquemas TVD não necessitam ser de primeira ordem de acurácia nos pontos extremos, mas reduzir localmente a solução para primeira ordem de acurácia em pontos extremos é a abordagem tradicionalmente encontrada na literatura técnica. Exemplos são os trabalhos “*Numerical Convection Schemes in Phoenix*” [SPA2001], que implementa diversas funções de interpolação, e o artigo original de Osher e Chakravarthy, “*High Resolution*

Schemes and the Entropy Condition” [OSH84]. Assim, essa foi a escolha implementada neste trabalho, pois foi a que se mostrou numericamente mais robusta e mais simples de implementar.

5.1.4 Funções limitadoras de fluxo

Vários limites para os fluxos são propostos na literatura técnica, como pode ser visto em [WAT95] [HUN95], [ZIJ95], [ZIJ97], [SPA2001], [SNO98] e [HEI99]. Essa variedade de funções limitadoras de fluxos decorre do fato de que o emprego do esquema $-K$ de van Leer não obedece à condição (5.3), pois é linear e causa oscilações em regiões onde existem altos gradientes.

Funções limitadoras de fluxos não diferenciáveis podem ser desenvolvidas notando que a função de van Leer pode ser escrita como:

$$\Psi_K(r) = \frac{1}{2}(1+K)r + \frac{1}{2}(1-K)$$

e, assim, para assegurar que esse esquema seja TVD, impõe-se a condição (5.3), de modo que se obtém uma classe de funções de restrição de fluxos, que é dada por:

$$\Psi_{K,M}(r) = \max\left[0, \min\left(\Psi_{K,M \geq 1}, 2r\right)\right]$$

Essa classe de funções de restrição de fluxo, e outras mais, tem a vantagem de ter alta flexibilidade, no sentido de definir esquemas lineares e não lineares. Uma desvantagem é que a função Ψ não é diferenciável, o que pode afetar, em alguns casos, o comportamento da solução [ZIJ95]. O parâmetro M permite controlar a solução de altos gradientes. Um grande valor de M conduz a uma melhor acurácia no entorno desses gradientes. Porém, para manter a monotonicidade da solução, essa escolha exige que o passo de tempo seja muito pequeno, o que é indesejável visto que se deve manter a eficiência computacional.

Com essas estratégias e idéias, pode-se introduzir uma abordagem unificada de modo a não modificar as funções interpolantes já desenvolvidas, sendo que o esquema numérico resultante depende apenas de qual função de restrição de fluxos é empregada para a solução de escoamentos e/ou de transporte escalar de substâncias multidimensionais dominados pela advecção, ou para solução daqueles eventos em que a difusão é importante.

Neste trabalho são implementadas e analisadas uma ampla coleção de funções, composta de 32 funções de restrição de fluxo. Muitas dessas funções foram geradas particularizando esta ou aquela função limitadora através da especificação de parâmetros constantes. Por exemplo, tomando $\Phi = 2$ na função Sweby, tem-se a função Roe Superbee; tomando $\Phi = 1$ na função Chakravanthy-Osher, gera-se a função Minmod; tomando $K = -1$ e $M = 1$ na função PL-Kappa, tem-se a função Davis; tomando $K = \frac{1}{3}$ e $M = 2$ na função PL-Kappa gera-se a função Koren, e assim por diante. Com algumas funções básicas, gera-se uma série de esquemas numéricos lineares e não lineares, como em [HUN95], [ZIJ95], [ZIJ97] e [HEI99], que fornecem soluções positivas ou TVD.

Como discutido anteriormente, quando a monotonicidade é violada, as funções monitores são tais que $\tau < 0$. Nesse caso, as funções de restrição de fluxo retornam zero, ou seja, $\Psi(\tau) \equiv 0$ gerando, quando aplicadas às funções de interpolação de fluxo dadas na seção 5.1.3, o esquema *upwind* de primeira ordem de acurácia. Além disso, quando determinada função de restrição de fluxo não estiver definida em algum ponto do domínio, escolhe-se, para efeito de implementação e para simplificar o tratamento computacional das exceções, que tal função seja, também, tal que $\Psi(\tau) \equiv 0$. Sob

essas considerações, e denotando, nessa seção, a expressão *caso contrário* simplesmente por CC, os limites analisados e implementados neste trabalho são:

a) Limites diferenciáveis

1. LUS (*Linear Upwind Scheme* - linear de primeira ordem):

$$\Psi(r) \equiv 0$$

2. Kappa, com $K=1/3$ (linear de alta ordem):

$$\Psi(K,r) = \begin{cases} \frac{1}{2}(1+K)r + \frac{1}{2}(1-K), & \text{se } r > 0 \text{ e } -1 \leq K \leq 1 \\ 0, & \text{CC} \end{cases}$$

3. Kappa simétrico, com $K=1/3$ (linear de alta ordem):

$$\Psi^s(K,r) = \begin{cases} \frac{1}{2}(1-K)r + \frac{1}{2}(1+K), & \text{se } r > 0 \text{ e } -1 \leq K \leq 1 \\ 0, & \text{CC} \end{cases}$$

4. QUICK (*Quadratic Upwind Scheme*) Kappa com $K=0,5$ (linear de alta ordem):

$$\Psi(r) = \begin{cases} \frac{3}{4}r + \frac{1}{4}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

5. CHARM (*Cubic-Parabolic High Accuracy Resolution Method*) (não linear):

$$\Psi(r) = \begin{cases} \frac{r(3r+1)}{(r+1)^2}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

6. HCUS (*Harmonic Cubic Upwind Scheme*) (não linear):

$$\Psi(r) = \begin{cases} \frac{3(r+|r|)}{2(r+2)}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

7. HQUICK (*Harmonic Quadratic Upwind Scheme*) (não linear):

$$\Psi(r) = \begin{cases} \frac{2(r+|r|)}{(r+3)}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

8. USR-Kappa (*Upstream Symmetric Limiter*), com $K=0$ (linear):

$$\Psi(K,r) = \begin{cases} (r+|r|), & \text{se } 0 < r \leq \frac{1}{3} \\ \frac{(8+9K)^2 + (2-12K)r + 2+3K}{3(1+r)^2}, & \text{se } r > \frac{1}{3} \\ 0, & \text{CC} \end{cases}$$

9. R-Kappa, com $K=1/3$ (não linear):

$$\Psi(K, r) = \begin{cases} (r+|r|) \frac{-r^2 + (3+K)r - K}{(1+r)^2}, & \text{se } 0 < r \leq 1 \text{ e } -1 \leq K < 0 \\ \frac{(2+K)r - K}{1+r}, & \text{se } r \geq 1 \text{ e } -1 \leq K < 0 \\ (r+|r|) \frac{(1+K)r + 1 - K}{(1+r)^2}, & \text{se } 0 \leq K \leq 1 \\ 0, & \text{CC} \end{cases}$$

10. van Leer harmônico (TVD) (não linear):

$$\Psi(r) = \begin{cases} \frac{(r+|r|)}{(1+r)}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

11. van Albada (não linear):

$$\Psi(r) = \begin{cases} \frac{(r^2 + r)}{(r^2 + 1)}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

12. Ospre (não linear):

$$\Psi(r) = \begin{cases} \frac{3(r^2 + r)}{2(r^2 + r + 1)}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

13. Fromm (linear):

$$\Psi(r) = \begin{cases} \frac{1}{2}r + \frac{1}{2}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

14. 23A (não linear):

$$\Psi(r) = \begin{cases} \frac{4r^2 + 2r}{3(r+1)^2}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

15. 23B (não linear):

$$\Psi(r) = \begin{cases} \frac{2r^2 + r}{2r^2 - r + 2}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

16. ISNAS (*Interpolation Scheme which is Nonoscillatory for Advected Scalars*) (não linear):

$$\Psi(r) = \begin{cases} \frac{(r+|r|)(3r+1)}{2(r+1)^2}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

b) Limites não diferenciáveis:

17. Sweby:

$$\Psi(r, \Phi) = \begin{cases} \max[0, \min(\Phi r, 1), \min(r, \Phi)], & \text{se } r > 0 \text{ e } -1 \leq \Phi \leq 1 \\ 0, & \text{CC} \end{cases}$$

18. Chakravanthy-Osher:

$$\Psi(r, \Phi) = \begin{cases} \max[0, \min(r, \Phi)], & \text{se } r > 0 \text{ e } -1 \leq \Phi \leq 1 \\ 0, & \text{CC} \end{cases}$$

19. MC (*Monotonized Central*):

$$\Psi(r) = \begin{cases} \max\left[0, \min\left(\frac{1+r}{2}, \min(2, 2r)\right)\right], & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

20. USPL-Kappa (*Upstream Symmetric Piecewise Linear*), com $K=0$:

$$\Psi(K, r, M) = \begin{cases} \max\left[0, \min\left(2r, \frac{2}{3}\right), \min(M, \Psi(K, r), 2r)\right], & \text{se} \\ & r > 0 \text{ e } -1 \leq K \leq 1 \text{ e } M \geq 1 \\ 0, & \text{CC} \end{cases}$$

21. DPL-Kappa (*Diffusive Piecewise Linear Kappa*), com $K=0,75$ e $M=3/2$:

$$\Psi(K, r, M, P_e) = \begin{cases} \max\left[\frac{r+|r|}{P_e}, \min\left(M + \frac{2r}{P_e}, \Psi(K, r), \left(2 + \frac{2}{P_e}\right)r\right)\right], & \text{se} \\ & r > 0 \text{ e } M \geq 1 \text{ e } P_e > 4/(1+K) \text{ e } 0 < K < 1 \\ 0, & \text{CC} \end{cases}$$

22. MPL1-Kappa (*Modified Piecewise Linear 1*), com $K=0$ e $\alpha=0$:

$$\Psi(K, r, M, \alpha) = \begin{cases} \max\left[0, \min\left(2r, \frac{2}{3}\right), \min(M, \Psi(K, r), (2+\alpha)2r)\right], & \text{se} \\ & r > 0 \text{ e } M \geq 1 \text{ e } -1 \leq \alpha \leq 0 \text{ e } -1 \leq K \leq 1 \\ 0, & \text{CC} \end{cases}$$

23. MPL2-Kappa (*Modified Piecewise Linear 2*), com $K=-1$ e $\alpha=0$:

$$\Psi(K, r, M, \alpha) = \begin{cases} \max\left[\min(M, \Psi(K, r), (2+\alpha)r)\right] + \min[0, \max(\alpha, -ar)], & \text{se} \\ & r > 0 \text{ e } M \geq 1 \text{ e } \alpha \in [-1, -K] \cap [-1, 0] \text{ e } -1 \leq K \leq 1 \\ 0, & \text{CC} \end{cases}$$

24. SMART (*Sharp and Monotonic Algorithm for Realistic Transport*) (QUICK limitado)(PL-Kappa($r, 1/2, 4$)):

$$\Psi(r) = \begin{cases} \max\left[0, \min\left(2r, \frac{3}{4}r + \frac{1}{2}, 4\right)\right], & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

25. UMIST (*Upstream Monotonic Interpolation for Scalar Transport*) (QUICK limitado, TVD) (SPL-Kappa(r,1/2,2)):

$$\Psi(r) = \begin{cases} \max\left[0, \min\left(2r, \frac{1}{4} + \frac{3}{4}r, \frac{3}{4} + \frac{1}{4}r, 2\right)\right], & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

26. MUSCL (*Monotonic Upstream Scheme for Conservation Laws*) (Fromm limitado, TVD) (SPL-Kappa(r,0,2)):

$$\Psi(r) = \begin{cases} \max\left[0, \min\left(2r, \frac{1}{2} + \frac{1}{2}r, 2\right)\right], & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

27. Roe Superbee (TVD) (Sweby(r,2)):

$$\Psi(r) = \begin{cases} \max\left[0, \min(2r, 1), \min(r, 2)\right], & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

28. Minmod (*Minimum Modulus* - Chakravanthy-Osher(r,1)):

$$\Psi(r) = \begin{cases} \max\left[0, \min(r, 1)\right], & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

29. Koren (CUS limitado, TVD)(PL-Kappa(r,1/3,2)):

$$\Psi(r) = \begin{cases} \max\left[0, \min\left(2r, \frac{2r}{3} + \frac{1}{3}, 2\right)\right], & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

30. Davis (PL-Kappa(r,-1,1)), com K=-1:

$$\Psi(K, r) = \begin{cases} \max\left[0, \min(1, \Psi(-1, r), 2r)\right], & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

31. Hundsdorfer, com K=1:

$$\Psi(r) = \begin{cases} \max\left\{\min(1, r), \min\left[\min(1, r), \frac{1}{3} + \frac{1}{6}r\right]\right\}, & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

32. Kappa-PSI com $\Phi=1,25$

$$\Psi(K, r) = \begin{cases} \max(0, \min(\Phi r, 1, 0)), & \text{se } r > 0 \\ 0, & \text{CC} \end{cases}$$

As funções de limitação de fluxo PL-Kappa (*Piecewise Linear*) e SPL-Kappa (*Symmetric Piecewise Linear*) foram empregadas para gerar, respectivamente, as funções UMIST e MUSCL e, SMART, Koren e Davis.

Na literatura encontram-se várias funções que servem de *benchmark* para verificar a qualidade de esquemas numéricos, como podem ser vistos em [BAP95], [GRO99], [ZOU99], [BYU99] e [AWA00]. Neste trabalho, as soluções numéricas foram

verificadas analisando-se o caso 1D quando da aplicação da função de restrição de fluxo às funções de interpolação de fluxo para a solução da advecção horizontal (6.8). Nos experimentos numéricos empregou-se a conhecida função teste, a “onda quadrada” (*square wave*), dada como sendo o perfil inicial de concentração S^0 . Todos os tempos de CPU apresentados foram normalizados com respeito ao esquema *upwind* fluxo numérico.

As diferentes soluções numéricas para esse problema são comparadas com a sua solução numérica discreta exata, que é expressa por $S = S^0 \Delta x \Delta y \Delta t$. A análise e comparação da solução numérica fornecida por um particular esquema de solução e a solução discreta exata são feitos, neste trabalho, empregando as métricas discutidas no anexo 2 e considerando que o avanço da solução exata é caracterizado pelo número de Courant $C_r = u \Delta t / \Delta x$. Ele é que determina o quanto a solução pode se deslocar durante um mesmo passo de tempo. Ou seja, o número de Courant fornece o número de células (ou fração de célula) que uma solução é advectada em um passo de tempo.

As tab. 5.1 a 5.3 mostram os resultados obtidos quando da aplicação da função de restrição de fluxo UMIST as essas funções de interpolação de fluxo, mas no anexo 4 encontram-se tabelados (tab. A4.1 a A4.45) os resultados, discutidos, para todas as funções de restrição, de 1. a 32. Os resultados das tab. 5.1 a 5.3 foram os obtidos com a onda quadrada de perfil $1 \times 10 \times 10$, representando a concentração inicial $S^0 = 1,0$ de comprimento de 10 espaçamentos de malha em cada direção coordenada. O pulso onda quadrada é advectado na direção X (os resultados para a direção Y são idênticos) com os parâmetros $\Delta x = \Delta y = 1,0m$, $\Delta t = 1,0s$, $u = 0,25m/s$ e $v = 0,0m/s$. A distância de advecção é de $50\Delta x$, que é obtida depois de 200 passos de tempo, pois o número de Courant é $C_r = u\Delta t/\Delta x = 0,25$. Nesse caso empregou-se uma malha de 100×100 pontos.

Outros experimentos mais sofisticados podem ser vistos na bibliografia mencionada, mas neste trabalho empregou-se a onda quadrada, e os experimentos numéricos são efetuados para diferentes números de Courant. Seus resultados, onde RMQ significa raiz média quadrada, são como:

TABELA 5.1 - Resultados para o teste com a onda quadrada e o esquema UMIST

métrica × esquema	$C_r=0,25$	$C_r=0,40$
massa total (exata)	100,000000	100,000000
massa total (numérica)	99,9999771	100,000007
razão do pico (RP)	0,9999982	1,0000065
conhecimento para a RP	-0,0000001	-0,0000002
razão de massa	0,9999994	0,9999996
1 - razão de massa	0,0000005	0,0000003
razão de distribuição	0,8898882	0,9236349
máximo (numérico)	0,9999982	1,0000065
mínimo (numérico)	-0,0000001	-0,0000002
erro absoluto médio	0,0022301	0,0049246
erro da RMQ	0,1048088	0,0912286
norma L_1	0,1930242	0,9999925
norma L_2	0,2370105	0,9610593
norma L_∞	0,4584135	1,0000065
tempo total (normalizado)	1,0523	1,0591

Uma discussão completa sobre todas as funções de restrição implementadas é como aquela apresentada no anexo 4, onde se analisam os resultados da aplicação das

métricas a todos os esquemas numéricos. Porém, nessa seção somente se discute qualitativamente a função aqui apresentada (UMIST).

Nesse caso, a tab. 5.1 mostra que o esquema UMIST é massa conservativo. Esse esquema preserva o pico, indicando que a concentração máxima não é aumentada ou diminuída. Já a métrica do *conhecimento para a razão do pico*, que mede as oscilações introduzidas por esquemas não monotônicos, indica que o esquema não introduz oscilações. A *razão de distribuição*, que analisa a manutenção do formato do perfil de concentração, mostra que o perfil é mantido com boa aproximação (cerca de 89%). Já as métricas relacionadas aos erros e às normas refletem os resultados anteriores, mostrando que esse esquema não linear fornece alta qualidade numérica.

A tab. 5.2 apresenta os resultados obtidos com as mesmas condições empregadas para aqueles da tab. 5.1, somente que nesse caso se verifica a influência do uso de precisão simples (*float*) e de precisão dupla (*double*) nos resultados numéricos do esquema UMIST.

TABELA 5.2 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise da precisão dos resultados do esquema UMIST com parâmetros 0,00025, 100x100, 200 ciclos, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=1$ e <i>float</i>	$S^0=1$ e <i>double</i>
massa total (exata)	100,000000000000	100,000000000000
massa total (numérica)	99,99997711182	99,99996948242
razão do pico (RP)	0,999998211861	0,999996244907
conhecimento para a RP	-0,000000198382	-0,000000198867
razão de massa	0,999999465942	0,999999313354
1 - razão de massa	0,000000534058	0,000000686646
razão de distribuição	0,889888297737	0,889886712409
máximo (numérico)	0,999998211861	0,999996244907
mínimo (numérico)	-0,000000198382	-0,000000198867
erro absoluto médio	0,002231752050	0,002231760209
erro da RMQ	0,104808839929	0,104808781014
norma L_1	0,193024234772	0,193024940491
norma L_2	0,237010555258	0,237010346232
norma L_∞	0,458413541317	0,458413422108

Os resultados apresentados na tab. 5.2 mostram que os arredondamentos de aritmética de ponto flutuante no esquema numérico são praticamente irrelevantes, sendo os resultados oferecidos com precisão simples praticamente como aqueles oferecidos com o uso de precisão dupla. No anexo 4 apresenta-se um quadro comparativo desse atributo numérico para todas as 32 funções implementadas.

Note-se que os parâmetros 0,00025, 100x100, 200 ciclos, $S^0 \times 10 \times 10$ referem-se, respectivamente, a imposição de restrição para as métricas que empregam o erro relativo (RMQ, L_1 e L_2) onde se considera, para efeito de avaliação, somente as células cuja concentração é maior que 0,00025 mpm/100ml; à malha de pontos empregada no teste; o ciclo (número de passos de tempo) onde os resultados foram verificados; e ao perfil da função de teste, como discutido anteriormente.

A tab. 5.3 apresenta os resultados obtidos com o uso de precisão simples e das mesmas condições usadas na obtenção dos resultados mostrados na tab. 5.1, para diferentes perfis de concentração. Ou seja, alterou-se o “pulso” da concentração inicial, de modo a verificar se o esquema numérico retém sua (ótima) qualidade numérica, mesmo sob fortes gradientes de concentração.

TABELA 5.3 - Resultados para o teste com onda quadrada para $C=0,25$ analisando a influência do perfil para esquema UMIST com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	100,000000	1000000,000000	1000000000,000
massa total (numérica)	99,9999771	1000000,000000	999999744,0000
razão do pico (RP)	0,9999982	0,9999966	0,9999947
conhecimento para a RP	-0,0000001	-0,0000000	-0,0000000
razão de massa	0,9999994	0,9999995	0,9999995
1 - razão de massa	0,0000005	0,0000004	0,0000004
razão de distribuição	0,8898882	0,8898885	0,8898856
máximo (numérico)	0,9999982	9999,9667968	9999947,000000
mínimo (numérico)	-0,0000001	-0,0000124	-0,0106831
erro absoluto médio	0,0022301	22,3184309	22318,5496589
erro da RMQ	0,1048088	0,1285871	0,1438643
Norma L_1	0,1930242	0,1930321	0,1930331
Norma L_2	0,2370105	0,2370108	0,2370101
Norma L_∞	0,4584135	0,4584104	0,4584104

Os resultados apresentados na tab. 5.3 mostram que a solução numérica do esquema UMIST mantém toda a qualidade numérica, como as apresentados nas tab. 5.1 e 5.2, mesmo sob situações onde ocorrem elevados “picos” de concentração. As diferenças dos resultados fornecidos pelas métricas relativas aos erros e às normas apenas refletem as diferenças das quantidades envolvidas nos testes das tabelas anteriores e no teste dessa última tabela. No anexo 4 pode-se ver, também, uma análise análoga para outras funções de restrição de fluxo.

5.2 Método beta

O método FLM foi empregado, pelos motivos discutidos nas seções 5.1 e no anexo 1, para a obtenção da solução da advecção horizontal da EDP do transporte escalar de substâncias 3D. Por outro lado, a solução da advecção horizontal é obtida com o emprego do método *beta* desenvolvido por Gross [GRO98]. Essa escolha é induzida por duas questões fundamentais na solução do transporte vertical. A primeira questão é a manutenção da estabilidade da solução, mesmo em regiões “muito rasas” em corpos de água. A segunda questão diz a respeito à difusão numérica, devido ao fato de que ela deve ser menor que a difusão física.

Essas questões também foram abordadas no anexo 1, porém, nessa seção destaca-se que o espaçamento vertical da malha é, geralmente, construído de modo que seja muito menor que o espaçamento horizontal da malha. Além disso, as alturas (magnitude do espaçamento vertical) das células das camadas podem ser muito pequenas, aproximando-se de zero em células que sofrem os processos de alagamento e de secamento. Para manter a estabilidade da solução numérica nesses casos, uma discretização explícita para a advecção vertical exige um passo de tempo muito pequeno, menor que o necessário, quando se considera somente o aspecto da acurácia, o que reduz a eficiência computacional. Uma solução para esse problema é construir um esquema completamente implícito. Porém, nesse caso, a dependência de dados torna complicada a obtenção de uma solução paralela eficiente.

A segunda questão considera o fato de que uma difusão numérica excessiva, produzida pela solução do transporte vertical, encobrirá a difusão física, calculada pelo modelo de turbulência, tornando inócua a modelagem turbulenta e o cálculo preciso da mistura vertical (ver anexo 1).

A alternativa é, portanto, desenvolver um esquema semi-implícito para o transporte vertical, de modo que a componente horizontal seja aproximada explicitamente como, por exemplo, pelo método FLM. A componente vertical deve ser aproximada implicitamente. Essa abordagem mantém a estabilidade da solução e maximiza sua eficiência computacional, características altamente desejáveis em modelos 3D.

Um exame detalhado da literatura técnica disponível mostrou poucas alternativas para obter um esquema semi-implícito com essas características. Uma abordagem é tratar os termos explícitos do transporte escalar com um esquema semi-Lagrangeano como feito em [CAS92], que não tem os atributos de ser altamente acurado e massa conservativo. Um esquema de solução que obedece tais exigências é o esquema $-\beta$, que foi desenvolvido por Gross [GRO98].

Esse esquema é semi-implícito e obtido através da combinação do método $-\theta$ com o esquema QUICK de Leonard [LEO79], e é empregado para resolver semi-implicitamente a advecção vertical. A difusão vertical é resolvida empregando o método semi-implícito Crank-Nicolson. A combinação desses dois métodos gera um esquema numérico semi-implícito para o transporte vertical.

A análise da difusão mostrou que o esquema $-\beta$ tem difusão numérica de pelo menos uma ordem de magnitude menor que a difusão física (ver anexo 1). Além disso, apesar de o método permitir o surgimento de alguma oscilação numérica, ele é de segunda ordem de acurácia, para todos os números de Courant. Assim, o método $-\beta$ de Gross tem os atributos necessários à solução de alta qualidade numérica. A difusão vertical é resolvida com o método Crank-Nicolson, dado que o objetivo é manter a estabilidade do esquema, mesmo quando o espaçamento vertical Δz é pequeno. Os termos de difusão são agregados às CC do modelo de transporte vertical, como mostrado na seção 4.7.

Assim, os esquemas numéricos resultantes para o transporte vertical podem ser empregados em regiões com grande variação no número de Courant, dado que essa característica é marcante em lagos e estuários, pois, embora predominem escoamentos com baixos números de Courant, escoamentos com altas velocidades verticais podem ocorrer, sobretudo em regiões com fortes gradientes batimétricos ou bruscos movimentos baroclínicos [GRO98].

Na construção do método $-\beta$ para a solução da advecção vertical considera-se, inicialmente, a advecção escrita na forma fluxo $S_k^{n+1} = S_k^n - C_z (S_U - S_L)$, onde $C_z = w(\Delta t/\Delta z)$ é o número de Courant, Δt é o passo de tempo, e S_L e S_U são os fluxos nas faces inferior e superior, respectivamente. Esse esquema é conservativo, por construção, se $(S_L)_k = (S_U)_{k-1}$ e, assim, basta especificar aproximações para S_U , pois a aproximação para S_L é obtida simplesmente empregando um deslocamento para a esquerda no índice k .

Considerando inicialmente que as velocidades são positivas, o fluxo superior interpolado com método QUICK fornece a expressão:

$$S_U = \frac{1}{2}(S_k^n + S_{k+1}^n) - \frac{1}{8}(S_{k-1}^n - 2S_k^n + S_{k+1}^n)$$

porém, o esquema QUICK é explícito e não é estável para a advecção pura [GRO98]. Um modo de melhorar sua estabilidade e acurácia é empregar a abordagem trapezoidal [HIR92] na sua forma particular, o método $-\theta$, onde $0 < \theta < 1$. Nessa situação, o fluxo S_U é aproximado como:

$$S_U = \frac{\theta}{2}(S_k^{n+1} + S_{k+1}^{n+1}) + \frac{1-\theta}{2}(S_k^n + S_{k+1}^n) - \frac{1}{8}(S_{k-1}^n - 2S_k^n + S_{k+1}^n) \quad (5.7)$$

onde em (5.7) o termo de correção do esquema QUICK é $0,125(S_{k-1}^n - 2S_k^n + S_{k+1}^n)$ e é explícito para que a matriz seja do tipo 3-diagonal, já visando a obter um esquema numérico que seja semi-implícito, ficando a “implicidade” restrita à componente vertical, sendo a componente horizontal completamente explícita.

Nessa situação a paralelização do modelo de transporte 3D considera somente uma decomposição 2D, onde a dependência de dados fica restrita apenas à coluna de água. Isso é particularmente atraente sob o ponto de vista da solução paralela, pois o particionamento do domínio pode ser apenas em duas dimensões. Essas questões são tratadas no capítulo 7.

Quando se toma $\theta = 0,5$ em (5.7), o esquema é estável e tem acurácia temporal de segunda ordem. Porém isso se verifica somente quando $|C_z| \leq 1$. Para manter a estabilidade para altos números de Courant, emprega-se o β – esquema, que depende do número de Courant. Com isso a interpolação do fluxo S_U , onde $0 < \beta_k \equiv \beta_k(C_z) < 1$ é como definido por (A1.2), e é dado por:

$$S_U = \beta_k(C_z) \left[\theta S_k^{n+1} + (1-\theta) S_k^n \right] + (1 - \beta_k(C_z)) \left[\frac{\theta}{2}(S_k^{n+1} + S_{k+1}^{n+1}) + \frac{1-\theta}{2}(S_k^n + S_{k+1}^n) - \frac{1}{8}(S_{k-1}^n - 2S_k^n + S_{k+1}^n) \right] \quad (5.8)$$

Sendo o esquema conservativo, isto é, vale que $(S_L)_k = (S_U)_{k-1}$, a expressão para o fluxo S_L é obtida diretamente de (5.8) como:

$$S_L = \beta_k(C_z) \left[\theta S_{k-1}^{n+1} + (1-\theta) S_{k-1}^n \right] + (1 - \beta_k(C_z)) \left[\frac{\theta}{2}(S_{k-1}^{n+1} + S_k^{n+1}) + \frac{1-\theta}{2}(S_{k-1}^n + S_k^n) - \frac{1}{8}(S_{k-2}^n - 2S_{k-1}^n + S_k^n) \right] \quad (5.9)$$

Desse modo, a expressão $S_k^{n+1} = S_k^n - C_z(S_u - S_L)$ é aproximada, usando (5.8) e (5.9) como:

$$\begin{aligned} & \left. \begin{aligned} & S_k^{n+1} + \theta C_z \left[\beta_k^{n+1}(S_k^{n+1} - S_{k-1}^{n+1}) + \frac{1}{2}(1 - \beta_k^{n+1})(S_{k-1}^{n+1} + S_{k-1}^n) \right] \\ & = S_k^n - (1-\theta) C_z \left[\beta_k^n(S_k^n + S_{k-1}^n) + \frac{1}{2}(1 - \beta_k^n)(S_{k+1}^n - S_{k-1}^n) \right] \\ & \quad - \frac{1}{8}(1 - \beta_k^n)(S_{k-2}^n - 3S_{k-1}^n + 3S_k^n - S_{k+1}^n) \end{aligned} \right\} \quad (5.10) \end{aligned}$$

A expressão (5.10) deve ser estabelecida para velocidades positivas e negativas, dado que o escoamento é não permanente e pode mudar de direção, mesmo localmente. Essa formulação pode ser obtida a partir da definição dos números de Courant locais e dos fluxos numéricos na direção vertical. As expressões obtidas são descritas e empregadas na seção 6.5, quando da obtenção dos esquemas numéricos para a ETM 3D. A tab. 5.4 mostra alguns resultados para o esquema beta.

TABELA 5.4 - Resultados para o teste para o esquema beta (adaptado de [GRO98])

Perfil × esquema numérico	<i>passo</i>	Seno ao quadrado	semi-elipse
	Resultados com a métrica do erro absoluto		
Esquema beta	2,21	0,78	1,90
Upwind trapezoidal	5,34	6,74	7,98
Diferenças centrais trapezoidal	6,99	2,95	3,77
QUICKEST	1,41	0,26	1,16

Os resultados apresentados na tab. 5.4 foram obtidos considerando-se a advecção de três perfis de teste em uma malha uniforme, onde cada perfil foi transportado a uma distância de 45 pontos da malha (espaçamentos verticais, Δz) empregando os parâmetros $w=1,0\text{m/s}$, $\Delta z=0,01\text{m}$ e $\Delta t=0,005\text{s}$, ou seja, o número de Courant $C_z = w(\Delta t/\Delta z) = 0,50$. O perfil *passo* (*step*) unitário consiste, nesse caso, em estabelecer o valor da concentração como $S=1,0$ no lado esquerdo e $S=0,0$ no lado direito. O segundo perfil consiste em uma onda senoide ao quadrado (*sine-squared*) de comprimento de $20\Delta z$. Esse perfil é expresso como $S = \text{sen}(\pi z/20\Delta z)^2$, onde $0 \leq z \leq 20\Delta z$. O terceiro perfil consiste em uma semi-elipse (*semi-ellipse*) de comprimento de $20\Delta z$, centrada em z_c , descrita por $S = \sqrt{1 - (z - z_c)^2 / (10\Delta z)^2}$, onde $0 \leq z \leq 20\Delta z$.

Os resultados mostram que os melhores esquemas numéricos são o método beta e o QUICKEST. O esquema *upwind* é altamente difusivo, enquanto o esquema em diferenças centrais é fortemente oscilatório. Note-se que o esquema QUICKEST, que é explícito, tem uma acurácia um pouco melhor que a do método beta mas isso somente se deve ao fato de que o número de Courant é $C_z = 0,50$ o que garante a estabilidade desse método. A grande vantagem do esquema beta é que ele é, dentro das características discutidas no anexo 1, estável para todos números de Courant, apesar de permitir o surgimento de alguma oscilação, o que não acontece com esquemas explícitos que são, necessariamente, limitados por pelo número de Courant $C_z \leq 1,0$. Resultados adicionais, e um ampla discussão sobre os atributos numéricos e computacionais do esquema, beta podem ser vistos em [GRO98].

6 Esquemas numéricos

Para obter soluções numéricas estáveis e acuradas, de modo que os sistemas de equações sejam lineares, deve-se construir funções interpolantes para os valores das variáveis dos termos constituintes das EDPs, além de decidir se as aproximações são explícitas ou implícitas, dado que cada qual tem suas vantagens e suas desvantagens. Em geral, um esquema numérico implícito é mais difusivo que o correspondente esquema explícito. Mas a abordagem implícita, por ter estabilidade incondicional, não restringe a magnitude do passo de tempo, que de qualquer modo não pode ser arbitrariamente grande para não degradar a acurácia da solução.

Portanto, deve-se ter um compromisso entre a eficiência computacional e a acurácia da solução. Uma estratégia nesse sentido é construir um esquema semi-implícito. É essa a abordagem que norteia este trabalho.

Para este trabalho, no caso das ESWs 2D e 3D, seguiu-se a abordagem de Casulli, na qual os gradientes da elevação da superfície nas ECQMs e o divergente da velocidade da EC são discretizados implicitamente. Por questões de estabilidade, os termos de atrito na interface água-sedimento nas ECQMs, são, também, discretizados implicitamente. Os coeficientes de atrito na interface água-superfície livre são discretizados explicitamente. Essa abordagem fornece esquemas numéricos semi-implícitos para a solução do escoamento.

No caso da ETM 3D, a abordagem foi a de discretizar implicitamente o transporte vertical, e explicitamente os demais termos. O esquema numérico resultante é semi-implícito. No caso da ETM 2D, os esquemas numéricos também foram construídos através da abordagem semi-implícita. O paralelismo é obtido via decomposição de dados e/ou de domínio, mas essas questões são tratadas no capítulo 7.

6.1 Esquemas semi-implícito: ESW

Considerando as escolhas mencionadas anteriormente, a solução para as ESW é obtida através da substituição formal das expressões (4.5) e (4.6), que são as equações da conservação da quantidade do movimento (ECQM_x e ECQM_y), na expressão (4.3), que é a equação da continuidade integrada na vertical (EC-IV). Com isso obtém-se um sistema de equações lineares e SDP *tipo 5-diagonal*, cujas incógnitas são somente os termos barotrópicos (gradientes dos níveis da água), e que é escrito como:

$$E_{(i-1,j)}\eta_{(i,j-1)}^{n+1} + C_{(i,j)}\eta_{(i,j)}^{n+1} + D_{(i+1,j)}\eta_{(i,j+1)}^{n+1} + F_{(i,j-1)}\eta_{(i-1,j)}^{n+1} + S_{(i,j+1)}\eta_{(i+1,j)}^{n+1} = B_{(i,j)}^n \quad (6.1)$$

onde os coeficientes são:

$$C_{(i,j)} = 1 + g \left(\frac{\Delta t}{\Delta x} \right)^2 \theta^2 \left\{ \left[(\Delta Z)^T A^{-1} \Delta Z \right]_{(i+\frac{1}{2},j)}^n + \left[(\Delta Z)^T A^{-1} \Delta Z \right]_{(i-\frac{1}{2},j)}^n \right\} \\ + g \left(\frac{\Delta t}{\Delta y} \right)^2 \theta^2 \left\{ \left[(\Delta Z)^T A^{-1} \Delta Z \right]_{(i,j+\frac{1}{2})}^n + \left[(\Delta Z)^T A^{-1} \Delta Z \right]_{(i,j-\frac{1}{2})}^n \right\}$$

$$E_{(i+1,j)} = -g \left(\frac{\Delta t}{\Delta x} \right)^2 \theta^2 \left[(\Delta Z)^T A^{-1} \Delta Z \right]_{(i-\frac{1}{2},j)}^n$$

$$D_{(i,j)} = -g \left(\frac{\Delta t}{\Delta x} \right)^2 \theta^2 \left[(\Delta Z)^T A^{-1} \Delta Z \right]_{(i+\frac{1}{2},j)}^n$$

$$F_{(i+1,j)} = -g \left(\frac{\Delta t}{\Delta y} \right)^2 \theta^2 \left[(\Delta Z)^T A^{-1} \Delta Z \right]_{(i,j-\frac{1}{2})}^n$$

$$\begin{aligned}
S_{(i,j)} &= -g \left(\frac{\Delta t}{\Delta y} \right)^2 \theta^2 [(\Delta Z)^T A^{-1} \Delta Z]_{(i,j+\frac{1}{2})}^n \\
B_{(i,j)}^n &= -\frac{\Delta t}{\Delta x} (1-\theta) \left\{ [(\Delta Z)^T U]_{(i+\frac{1}{2},j)}^n + [(\Delta Z)^T U]_{(i-\frac{1}{2},j)}^n \right\} - \frac{\Delta t}{\Delta y} (1-\theta) \left\{ [(\Delta Z)^T V]_{(i,j+\frac{1}{2})}^n \right. \\
&\quad \left. + [(\Delta Z)^T V]_{(i,j-\frac{1}{2})}^n \right\} - \frac{\Delta t}{\Delta x} \theta \left\{ [(\Delta Z)^T A^{-1} G]_{(i+\frac{1}{2},j)}^n + [(\Delta Z)^T A^{-1} G]_{(i-\frac{1}{2},j)}^n \right\} \\
&\quad - \frac{\Delta t}{\Delta y} \theta \left\{ [(\Delta Z)^T A^{-1} G]_{(i,j+\frac{1}{2})}^n + [(\Delta Z)^T A^{-1} G]_{(i,j-\frac{1}{2})}^n \right\} + \eta_{(i,j)}^n
\end{aligned}$$

sendo os vetores $U_{(i+\frac{1}{2},j)}^{n+1}$, $V_{(i,j+\frac{1}{2})}^{n+1}$, $\Delta Z_{(i+\frac{1}{2},j)}^n$, $\Delta Z_{(i,j+\frac{1}{2})}^n$, $G_{(i+\frac{1}{2},j)}^n$ e $G_{(i,j+\frac{1}{2})}^n$, e as matrizes $A_{(i+\frac{1}{2},j)}^n$ e $A_{(i,j+\frac{1}{2})}^n$ como descritos na seção 4.1.

Os valores das velocidades horizontais, que são responsáveis pela troca das quantidades de movimentos entre as camadas, são calculados resolvendo os $N_x N_y$ sistemas de equações 3-diagonal e SDP de dimensão N_z via algoritmo de Cholesky, após a substituição em (4.5) e (4.5) dos valores dos níveis recém calculados, via algoritmo do GC, em (6.1).

As velocidades verticais são obtidas a partir do conhecimento das velocidades horizontais nos níveis de tempo $n+1$ e n , empregando a expressão (4.4), que é a forma discreta da equação da continuidade, e que é resolvida explicitamente da primeira camada para a última camada (o implementação e os esquemas consideram que a ordem dos índices é crescente de baixo para cima na coluna de água, conforme ilustrado na fig. 6.9), ou seja, como:

$$\begin{aligned}
w_{(i,j,k+\frac{1}{2})}^{n+1} &= -(\theta^{-1} - 1) w_{(i,j,k+\frac{1}{2})}^n + w_{(i,j,k-\frac{1}{2})}^{n+1} + (\theta^{-1} - 1) w_{(i,j,k-\frac{1}{2})}^n \\
&\quad - \frac{\theta^{-1}}{\Delta x} \left(u_{(i+\frac{1}{2},j,k)}^{n+\theta} \Delta z_{(i+\frac{1}{2},j,k)}^n - u_{(i-\frac{1}{2},j,k)}^{n+\theta} \Delta z_{(i-\frac{1}{2},j,k)}^n \right) \\
&\quad - \frac{\theta^{-1}}{\Delta y} \left(v_{(i,j+\frac{1}{2},k)}^{n+\theta} \Delta z_{(i,j+\frac{1}{2},k)}^n - v_{(i,j-\frac{1}{2},k)}^{n+\theta} \Delta z_{(i,j-\frac{1}{2},k)}^n \right)
\end{aligned}$$

em que $u_{(i\pm\frac{1}{2},j,k)}^{n+\theta} = 0,5u_{(i\pm\frac{1}{2},j,k)}^{n+1} + 0,5u_{(i\pm\frac{1}{2},j,k)}^n$ e $v_{(i,j\pm\frac{1}{2},k)}^{n+\theta} = 0,5v_{(i,j\pm\frac{1}{2},k)}^{n+1} + 0,5v_{(i,j\pm\frac{1}{2},k)}^n$, tomando se $\theta = 0,5$. As velocidades verticais são responsáveis pela troca de massa.

A ESW como sendo um caso particular das ESW 3D com apenas uma camada, e denotada como ESW 2D, de modo a diferenciá-la das ESW 2D integrada na vertical (ESW 2D-IV). O sistema de equações para as ESW 2D e para as ESW 2D-IV é análogo ao (6.1). Mas, nesses dois casos, seus coeficientes são:

$$\begin{aligned}
E_{(i+1,j)} &= -g \left(\frac{\Delta t}{\Delta x} \right)^2 \theta^2 \left(\frac{h_{u(i-\frac{1}{2},j)} + \eta_{(i-1)}^n}{1 + \Delta t \tau_x^{s^n}} \right) \\
D_{(i+1,j)} &= -g \left(\frac{\Delta t}{\Delta x} \right)^2 \theta^2 \left(\frac{h_{u(i+\frac{1}{2},j)} + \eta_{(i)}^n}{1 + \Delta t \tau_x^{s^n}} \right) \\
F_{(i+1,j)} &= -g \left(\frac{\Delta t}{\Delta y} \right)^2 \theta^2 \left(\frac{h_{v(i,j-\frac{1}{2})} + \eta_{(j-1)}^n}{1 + \Delta t \tau_y^{s^n}} \right)
\end{aligned}$$

$$\begin{aligned}
S_{(i,j+1)} &= -g \left(\frac{\Delta t}{\Delta y} \right)^2 \theta^2 \left(\frac{h_{v(i,j+\frac{1}{2})} + \eta_{(j)}^n}{1 + \Delta t \tau_{y(i,j+\frac{1}{2})}^{s^n}} \right) \\
C_{(i,j)} &= 1 + g \left(\frac{\Delta t}{\Delta x} \right)^2 \theta^2 \left\{ \left(\frac{h_{u(i+\frac{1}{2},j)} + \eta_{(j)}^n}{1 + \Delta t \tau_{x(i+\frac{1}{2},j)}^{s^n}} \right) + \left(\frac{h_{u(i-\frac{1}{2},j)} + \eta_{(j-1)}^n}{1 + \Delta t \tau_{x(i-\frac{1}{2},j)}^{s^n}} \right) \right\} \\
&\quad + g \left(\frac{\Delta t}{\Delta x} \right)^2 \theta^2 \left\{ \left(\frac{h_{v(i,j+\frac{1}{2})} + \eta_{(j)}^n}{1 + \Delta t \tau_{y(i,j+\frac{1}{2})}^{s^n}} \right) + \left(\frac{h_{v(i,j-\frac{1}{2})} + \eta_{(j-1)}^n}{1 + \Delta t \tau_{y(i,j-\frac{1}{2})}^{s^n}} \right) \right\} \\
B_{(i,j)}^n &= -\frac{\Delta t}{\Delta x} (1-\theta) \left[\left(h_{u(i+\frac{1}{2},j)} + \eta_{(i)}^n \right) U_{(i+\frac{1}{2},j)}^n - \left(h_{u(i-\frac{1}{2},j)} + \eta_{(i-1)}^n \right) U_{(i-\frac{1}{2},j)}^n \right] \\
&\quad - \frac{t}{\Delta y} (1-\theta) \left[\left(h_{v(i,j+\frac{1}{2})} + \eta_{(j)}^n \right) V_{(i,j+\frac{1}{2})}^n - \left(h_{v(i,j-\frac{1}{2})} + \eta_{(j-1)}^n \right) V_{(i,j-\frac{1}{2})}^n \right] \\
&\quad - \frac{\Delta t}{\Delta x} \theta \left[\left(\frac{h_{u(i+\frac{1}{2},j)} + \eta_{(i)}^n}{1 + \Delta t \tau_{x(i+\frac{1}{2},j)}^{s^n}} \right) F U_{(i+\frac{1}{2},j)}^n - \left(\frac{h_{u(i-\frac{1}{2},j)} + \eta_{(i-1)}^n}{1 + \Delta t \tau_{x(i-\frac{1}{2},j)}^{s^n}} \right) F U_{(i-\frac{1}{2},j)}^n \right] \\
&\quad - \frac{\Delta t}{\Delta y} \theta \left[\left(\frac{h_{v(i,j+\frac{1}{2})} + \eta_{(j)}^n}{1 + \Delta t \tau_{y(i,j+\frac{1}{2})}^{s^n}} \right) F V_{(i,j+\frac{1}{2})}^n - \left(\frac{h_{v(i,j-\frac{1}{2})} + \eta_{(j-1)}^n}{1 + \Delta t \tau_{y(i,j-\frac{1}{2})}^{s^n}} \right) F V_{(i,j-\frac{1}{2})}^n \right] + \eta_{(i,j)}^n
\end{aligned}$$

onde $\eta_{(i)}^n \stackrel{def}{=} \eta_{(i,j)}^n$ ou $\eta_{(i)}^n \stackrel{def}{=} \eta_{(i-1,j)}^n$ e, $\eta_{(j)}^n \stackrel{def}{=} \eta_{(i,j)}^n$, ou $\eta_{(j)}^n \stackrel{def}{=} \eta_{(i,j-1)}^n$, dependendo da direção das velocidades horizontais.

Note-se, porém, que neste trabalho, o esquema numérico semi-implícito para as ESW integradas na vertical (2D-IV) foi desenvolvido independentemente do 3D e de seu caso particular, as ESW 2D. Em qualquer caso, os sistemas de equações resultantes são lineares, *tipo 5-diagonal*, e SDP quando satisfeitas as condições:

$$0,5 \left(\eta_{(i\pm 1,j)}^n + \eta_{(i,j)}^n \right) + h_{u(i\pm \frac{1}{2},j)} > 0 \text{ e } 0,5 \left(\eta_{(i,j\pm 1)}^n + \eta_{(i,j)}^n \right) + h_{v(i,j\pm \frac{1}{2})} > 0$$

onde $h_{u(i\pm \frac{1}{2},j)}$ e $h_{v(i,j\pm \frac{1}{2})}$ são as batimetrias nas direções X e Y . No caso das ESW 2D tomou-se $\theta = 0,5$ e no caso das ESW 2D-IV tomou-se $\theta = 1$ (acurácia temporal de primeira ordem). As alturas nas ESW 2D-IV foram obtidas através das expressões:

$$H u_{(i\pm \frac{1}{2},j)}^n = 0,5 \left(\eta_{(i\pm 1,j)}^n + \eta_{(i,j)}^n + 2h_{u(i\pm \frac{1}{2},j)} \right) \text{ e } H v_{(i,j\pm \frac{1}{2})}^n = 0,5 \left(\eta_{(i,j\pm 1)}^n + \eta_{(i,j)}^n + 2h_{v(i,j\pm \frac{1}{2})} \right)$$

enquanto as alturas das ESW 2D são obtidas como:

$$H u_{(i\pm \frac{1}{2},j)}^n \stackrel{def}{=} h_{u(i\pm \frac{1}{2},j)} + \eta_{(i\pm(I+1/2),j)}^n \text{ e } H v_{(i,j\pm \frac{1}{2})}^n \stackrel{def}{=} h_{v(i,j\pm \frac{1}{2})} + \eta_{(i,j\pm(J+1/2))}^n$$

em que I e J são as funções sinal definidas na seção 4.4. Desse modo os algoritmos numéricos para as ESW 2D e 3D fornecem expressões consistentes com os esquemas de solução para a ETM 3D, satisfazendo, portanto a condição CTC. Os esquemas de solução para as ESW 2D-IV não contemplam tal atributo.

Para verificar a qualidade das soluções numéricas produzidas pelos modelos de escoamento 2D e 3D, no tocante às velocidades e aos níveis, fizeram-se experimentos numéricos como apresentados e discutidos em [SIL86], [SCE91], [KIM94], [CAS94] e [KAP98]. Esses experimentos consistem na solução da ESW 1D e de um modelo

aproximado para as ESW 3D. Desse modo, a verificação entre os resultados teóricos e resultados numéricos são somente entre aqueles resultantes da solução de sistemas de equações simplificados, onde não existem os termos de atrito e de viscosidade. Mas esses modelos teóricos são importantes na fase de verificação, onde se busca investigar a qualidade numérica dos algoritmos desenvolvidos e implementados. As soluções analíticas para as ESW 1D (6.2), bem como para o experimento que trata do modelo 3D, são calculadas através do emprego do *software MAPLE V®*, que é um pacote de computação simbólica desenvolvido pela *Waterloo Maple Inc.*

O primeiro experimento compara a solução numérica e a solução analítica das equações *shallow water* 1D. Essas equações são escritas como:

$$\left. \begin{aligned} \frac{\partial \eta}{\partial t} + H \frac{\partial u}{\partial x} &= 0 \\ \frac{\partial u}{\partial t} + g \frac{\partial \eta}{\partial x} &= 0 \end{aligned} \right\} (6.2)$$

sendo que a solução analítica de (6.2), no caso dos níveis, é como:

$$\eta(x, t) = a \operatorname{sen}\left(\omega t - \frac{x}{\lambda}\right) + \eta^0(x, 0)$$

e a solução analítica, no caso da componente de velocidade na direção X, é como:

$$u(x, t) = \frac{a}{H} \sqrt{gH} \operatorname{sen}\left(\omega t - \frac{x}{\lambda}\right)$$

em que as quantidades físicas envolvidas são como:

1. $\eta = \eta(x, t)$: nível de água;
2. $u = u(x, t)$: componente (direção X) da velocidade;
3. $H(x, t)$: profundidade;
4. $c = \sqrt{gH}$: celeridade da onda;
5. a : amplitude;
6. T : período;
7. $\lambda = cT$: comprimento de onda;
8. $\omega = 2\pi/T$: frequência angular;
9. $k = 2\pi/\lambda$: número de onda;
10. L : espaçamento na direção da onda (comprimento da malha de teste);
11. $\eta^0(x, 0)$: nível de água de repouso, acima do plano de referência;
12. t : tempo.

Substituindo esses parâmetros físicos nos termos de soluções analíticas de (6.2) obtém-se a expressão para os níveis (elevação da superfície):

$$\eta(x, t) = a \operatorname{sen}\left(\frac{2\pi}{T} t - \frac{x}{\sqrt{gHT}}\right) + \eta^0(x, 0)$$

e, a expressão para a componente u da velocidade horizontal, que é dada por:

$$u(x, t) = \frac{a}{H} \sqrt{gH} \operatorname{sen}\left(\frac{2\pi}{T} t - \frac{x}{\sqrt{gHT}}\right)$$

Note-se que especificando um nível em um ponto x tem-se uma “onda senoidal” propagando-se sem deformação de uma extremidade a outra de um “canal” com

profundidade constante. Assim, o primeiro experimento consiste em verificar a qualidade numérica do modelo 2D integrado na vertical e, também, a qualidade numérica do modelo 3D com apenas uma camada. Para esses casos, foram empregados os seguintes parâmetros:

1. $u(x, 0) = U(x, 0) \equiv 0 \text{ m/s}$: velocidade inicial;
2. $H(x, t) = 4,0 \text{ m}$: profundidade uniforme;
3. $c = \sqrt{9,81 \times 4,0} \approx 6,2642 \text{ m/s}$: celeridade da onda;
4. $a = 0,1 \text{ m}$: amplitude;
5. $T = 720 \text{ s}$: período;
6. $L = 2800 \text{ m}$: comprimento do canal;
7. $\eta^0(x, 0) = 1,0 \text{ m}$: nível inicial;
8. $t = 2160 \text{ s}$: tempo de simulação.

Considerando-se uma malha de teste de 7×14 pontos (sete linhas na direção Y por 14 colunas na direção X), simula-se o escoamento na direção X e os níveis de água em um canal retangular com uma profundidade uniforme de 4,0 m. Nesse experimento o espaçamento é $\Delta x = 500 \text{ m}$ e passo de tempo é $\Delta t = 24 \text{ s}$ e, portanto, a simulação requer 90 passos de tempos. As fig. 6.1 e 6.2 mostram os resultados analíticos e numéricos obtidos com tais parâmetros.

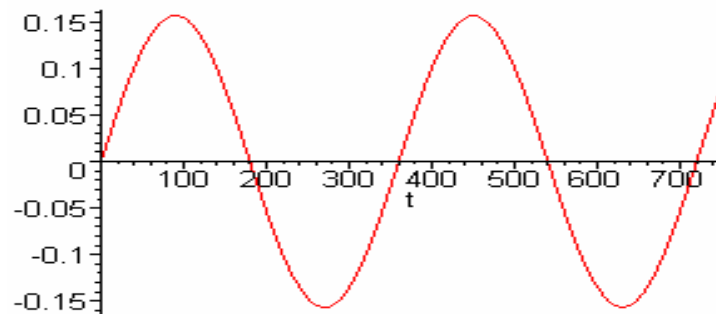


FIGURA 6.1 - Curva obtida com a solução analítica de (6.2) para os níveis de água depois de 90 passos de tempo (720s) e considerando-se o ponto onde se tem $L=2800 \text{ m}$

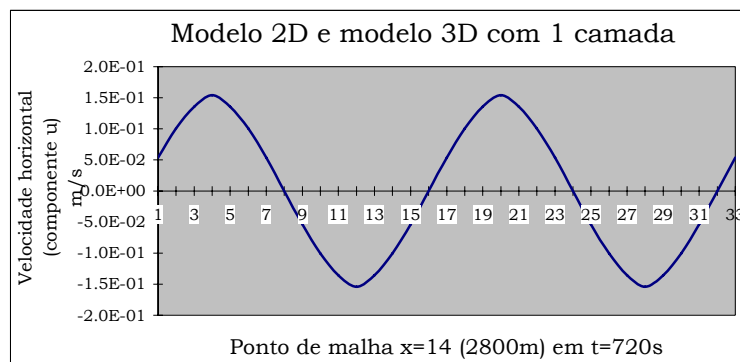


FIGURA 6.2 - Curva obtida com a solução numérica de (6.2) para os níveis de água depois de 90 passos de tempo (720s) e considerando-se o ponto onde se tem $L=2800 \text{ m}$

Na verificação da qualidade da componente horizontal da velocidade, empregou-se a métrica do erro relativo e, uma análise comparativa entre os resultados analíticos e os numéricos, como ilustradas nas fig. 6.1 e 6.2, mostra que o erro máximo é menor que 0,5%, o que indica que o modelo é suficientemente acurado.

Dado que os níveis de água são calculados somente a partir da camada superior, tanto no modelo 2D quanto no modelo 3D, e os algoritmos numéricos para o cálculo desses níveis são idênticos nos dois modelos, a análise comparativa dos resultados desses níveis é feita no experimento a seguir.

Note-se que todos os resultados analisados nas tabelas dessa seção foram obtidos empregando a malha de ponto com 7 linhas e 14 colunas. Porém, os resultados mostrados referem-se somente à linha central (número 4), dado que os resultados para as outras linhas são completamente coincidentes com este. Todas as velocidades verticais (w) e horizontais (u) são nulas no tempo inicial ($t=0$). Além disso, todos os níveis (elevação da superfície da água) são idênticos, tanto para os modelos 2D quanto para o modelo 3D.

O segundo experimento consiste na avaliação das soluções produzidas para o modelo hidrodinâmico 3D. Nesse caso pode-se mostrar, como em [KIM94], que as soluções analíticas são tais que, para os níveis, tem-se:

$$\begin{aligned}\eta(x,t) &= a \cos(kx) \cos(\omega t) \\ &= a \cos\left(\frac{2\pi}{\lambda} x\right) \cos\left(\frac{2\pi}{T} t\right)\end{aligned}$$

e a velocidade horizontal (componente na direção X) é obtida pela expressão:

$$\begin{aligned}u(x,t) &= \frac{a}{h} c \operatorname{sen}(kx) \operatorname{sen}(\omega t) \\ &= \frac{a}{h} \sqrt{gh} \operatorname{sen}\left(\frac{2\pi}{\lambda} x\right) \operatorname{sen}\left(\frac{2\pi}{T} t\right)\end{aligned}$$

e onde a velocidade vertical é calculada como:

$$\begin{aligned}w(x,t) &= \frac{a}{h} \omega (h + \eta) \cos(kx) \operatorname{sen}(\omega t) \\ &= \frac{a}{h} \frac{2\pi}{T} (h + \eta) \cos\left(\frac{2\pi}{\lambda} x\right) \operatorname{sen}\left(\frac{2\pi}{T} t\right)\end{aligned}$$

Nesse experimento o campo de velocidades e os níveis são simulados em um canal retangular. Tal canal é, também, formado por malha com 7×14 pontos (sete linhas na direção Y por 14 colunas na direção X), e com profundidade total uniforme, quando em repouso, de 4,0 m. O espaçamento horizontal adotado é $\Delta x = 200m$. O espaçamento vertical adotado é $\Delta z = 1,0m$ e considerou-se um modelo 3D formado com quatro camadas. O passo de tempo é $\Delta t = 5s$, e os parâmetros adotados foram:

1. $u(x,0) \equiv 0 m/s$: velocidade (horizontal) inicial;
2. $w(x,0) \equiv 0 m/s$: velocidade (vertical) inicial;
3. $h(x,t) = 4,0m$: altura total;
4. $c = \sqrt{9,81 \times 4,0} \approx 6,2642 m/s$ quando se considera a altura total $h(x,t) = 4,0m$ (ou apenas uma camada). Para a avaliação das componentes das velocidades horizontais e verticais no caso do modelo 3D com quatro camadas, toma-se $h(x,t) = \Delta z = 1,0m$;

5. $a = 0,4 m$: amplitude;
6. $T = 720 s$: período;
7. $L = 2800 m$: comprimento do canal;
8. $\eta^0(x, 0) = 1,0 m$: nível inicial;
9. $t = 2160 s$: tempo de simulação

As fig. 6.3 e 6.4 mostram, respectivamente, os resultados analíticos e numéricos obtidos para os níveis de água (elevação da superfície a partir de um plano de referência).

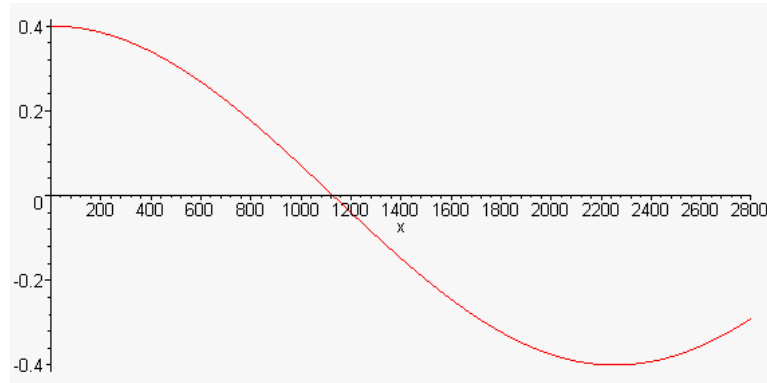


FIGURA 6.3 - Curva de níveis obtida com a solução analítica para a elevação do nível da água depois de 432 passos de tempo (2160s), e considerando-se o ponto de malha onde se tem $L=2800m$

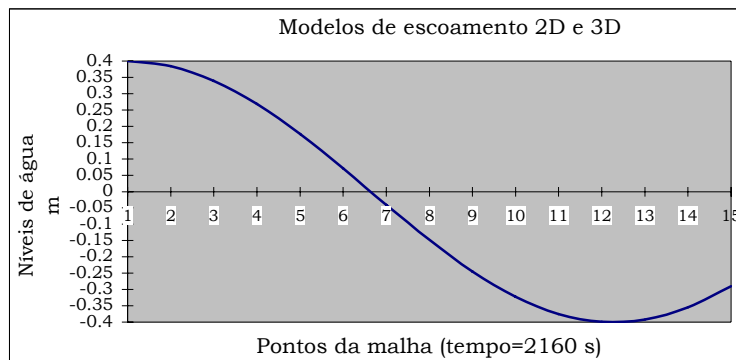


FIGURA 6.4 - Curva de níveis obtida com a solução numérica para a elevação do nível da água depois de 432 passos de tempo (2160s), e considerando-se o ponto de malha onde se tem $L=2800m$

Na verificação da qualidade dos níveis de água empregou-se a métrica do erro relativos e, uma análise comparativa entre os resultados analíticos e os numéricos, como nas fig. 6.3 e 6.4, mostrou que erro máximo é de aproximadamente 0,5%.

Uma outra análise refere-se às velocidades verticais e horizontais. As velocidade verticais são avaliadas considerando-se a acurácia temporal de primeira e a de segunda ordem. Ou seja, empregando esquemas numéricos com o parâmetro teta $\theta = 1,0$ e $\theta = 0,5$, respectivamente. As fig. 6.5 e 6.6 mostram os resultados obtidos (solução numérica) para as velocidades verticais do modelo 3D com quatro (4) camadas de altura uniforme e igual a $1,0m$. Os resultados analíticos não são apresentados em forma de tabelas, e os resultados numéricos são apresentados de modo que as seqüências —▲—, —●—, —◆—, e —■— indicam, respectivamente, os resultados para a primeira, segunda, terceira e quarta camadas do modelo 3D.

Uma análise comparativa entre os resultados analíticos e os numéricos, mostrados nas fig. 6.5 e 6.6, empregando o erro relativo, mostra que o erro máximo é menor que aproximadamente 0,5% com esquemas de segunda ordem de acurácia, e menor que aproximadamente 1,0% com os esquemas de primeira ordem de acurácia.

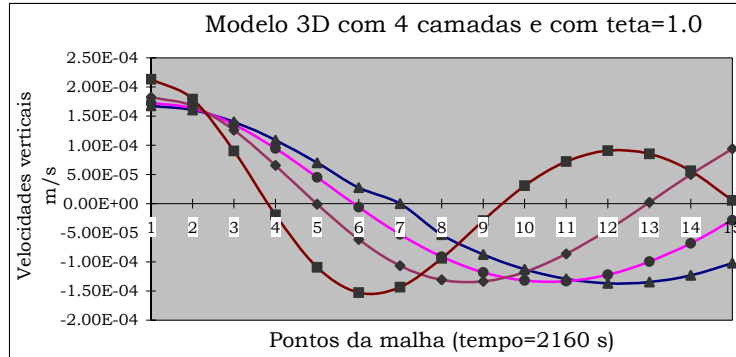


FIGURA 6.5 - Resultados obtidos com a solução numérica para as velocidades verticais depois de 432 passos de tempo (2160s), empregando o algoritmo de cálculo com acurácia de primeira ordem

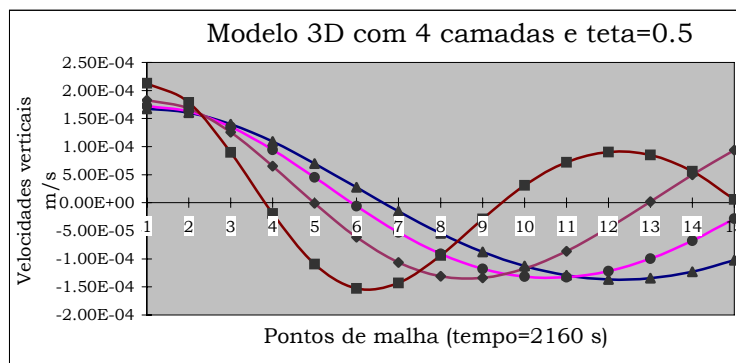


FIGURA 6.6 - Resultados obtidos com a solução numérica para as velocidades verticais depois de 432 passos de tempo (2160s), empregando o algoritmo de cálculo com acurácia de segunda ordem

A fig. 6.7 mostra os resultados obtidos para a componente horizontal (direção X, componente u) da velocidade.

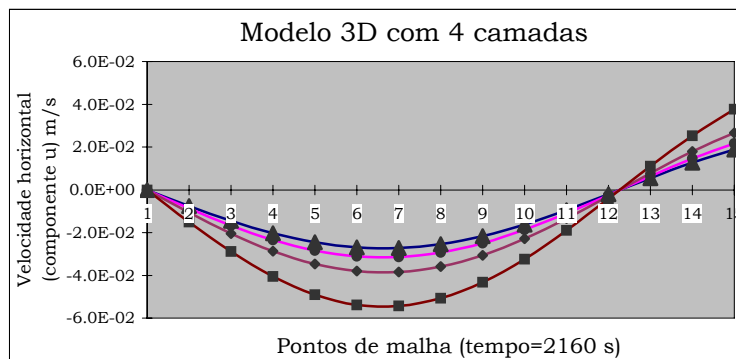


FIGURA 6.7 - Resultados obtidos com a solução numérica para as velocidades horizontais (modelo 3D com quatro camadas) depois de 432 passos de tempo (2160s)

Uma análise comparativa entre os resultados analíticos e os numéricos, mostrado na fig. 6.7, empregando o erro relativo, mostra que o erro máximo é de 0,1%, indicando que o esquema numérico é suficientemente acurado para simulações de longos períodos de tempo.

Note-se, porém, que todos esses experimentos consideram situações ideais para a solução numérica. Na prática a influência da topografia, etc. torna esses resultados insustentáveis impulsionando as pesquisas de modelagem, técnicas numéricas e computacionais, etc. que visam obter soluções condizentes com a realidade física.

6.1.1 Solução explícita para termos das ESW

Como discutido no anexo 1, onde se apresenta e discute, entre outras questões, a importância de se desenvolver esquemas semi-Lagrangianos para a solução do escoamento, neste trabalho optou-se por construir soluções Eulerianas para os operadores explícitos $FU_{(i+\frac{1}{2},j)}^n$ e $FV_{(i,j+\frac{1}{2})}^n$ nas ESW discretas.

Nesse caso, os operadores explícitos agregam as discretizações explícitas para o termos da advecção, dos atritos superior e inferior, da força de Coriolis e da difusão horizontal. Na solução dos termos advectivos para as ESW, emprega-se esquemas numéricos *upwind* de primeira ordem e, assim, a advecção da ECQM_x é aproximada considerando que:

$$\frac{\partial U}{\partial x} \approx \frac{U_{(i+\frac{1}{2},j)}^n - U_{(i-\frac{1}{2},j)}^n}{\Delta x} \text{ se } U_{(i+\frac{1}{2},j)}^n \geq 0 \text{ e } \frac{\partial U}{\partial x} \approx \frac{U_{(i+\frac{3}{2},j)}^n - U_{(i+\frac{1}{2},j)}^n}{\Delta x} \text{ se } U_{(i+\frac{1}{2},j)}^n < 0$$

$$\frac{\partial U}{\partial y} \approx \frac{U_{(i+\frac{1}{2},j)}^n - U_{(i+\frac{1}{2},j-1)}^n}{\Delta y} \text{ se } \bar{V}_{(i+\frac{1}{2},j)}^n \geq 0 \text{ e } \frac{\partial U}{\partial y} \approx \frac{U_{(i+\frac{1}{2},j+1)}^n - U_{(i+\frac{1}{2},j)}^n}{\Delta y} \text{ se } \bar{V}_{(i+\frac{1}{2},j)}^n < 0$$

em que a componente V da velocidade discreta não está definida na posição $(i+\frac{1}{2},j)$. Portanto $\bar{V}_{(i+\frac{1}{2},j)}^n$ é aproximado como a média dos quatro pontos circunvizinhos da malha numérica nos quais V^n é conhecido, ou seja, como:

$$\bar{V}_{(i+\frac{1}{2},j)}^n = 0,25 \left(V_{(i,j+\frac{1}{2})}^n + V_{(i,j-\frac{1}{2})}^n + V_{(i+1,j+\frac{1}{2})}^n + V_{(i+1,j-\frac{1}{2})}^n \right)$$

Os termos de atrito inferior e superior são escritos na forma discreta, respectivamente, como:

$$\left(\tau_x^s \right)_{(i+\frac{1}{2},j)}^n \approx \frac{g \sqrt{\left(U_{(i+\frac{1}{2},j)}^n \right)^2 + \left(\bar{V}_{(i+\frac{1}{2},j)}^n \right)^2}}{0,25 \left(C_{h(i+1,j)} + C_{h(i,j)} \right)^2 H u_{(i+\frac{1}{2},j)}^n}, \left(\tau_x^v \right)_{(i+\frac{1}{2},j)}^n \approx \frac{C_D \rho_{ar} W_x}{\rho_{\acute{a}gua} H u_{(i+\frac{1}{2},j)}^n}$$

enquanto os termos de Coriolis, e de difusão horizontal que é aproximado via diferenças centrais, são escritos na forma discreta, respectivamente, como:

$$\Phi \bar{V}_{(i+\frac{1}{2},j)}^n \approx \Phi 0,25 \left(V_{(i,j+\frac{1}{2})}^n + V_{(i,j-\frac{1}{2})}^n + V_{(i+1,j+\frac{1}{2})}^n + V_{(i+1,j-\frac{1}{2})}^n \right)$$

e

$$\varepsilon_h \nabla^2 U_{(i+\frac{1}{2},j)}^n \approx \varepsilon_h \left[\left(\frac{U_{(i+\frac{3}{2},j)}^n - 2U_{(i+\frac{1}{2},j)}^n + U_{(i-\frac{1}{2},j)}^n}{\Delta x^2} \right) + \left(\frac{U_{(i+\frac{1}{2},j+1)}^n - 2U_{(i+\frac{1}{2},j)}^n + U_{(i+\frac{1}{2},j-1)}^n}{\Delta y^2} \right) \right]$$

Analogamente, o operador explícito para a direção Y agrega as discretizações explícitas para os termos da advecção, dos atritos superior e inferior, da difusão

horizontal e do termo de Coriolis. As aproximações para os termos advectivos para a ECQM_y que são definidas nos pontos $(i, j + \frac{1}{2})$ são tais que:

$$\frac{\partial V}{\partial x} \approx \frac{V_{(i,j+\frac{1}{2})}^n - V_{(i-1,j+\frac{1}{2})}^n}{\Delta x} \text{ se } \bar{U}_{(i,j+\frac{1}{2})}^n \geq 0 \text{ e } \frac{\partial V}{\partial x} \approx \frac{V_{(i+1,j+\frac{1}{2})}^n - V_{(i,j+\frac{1}{2})}^n}{\Delta x} \text{ se } \bar{U}_{(i,j+\frac{1}{2})}^n < 0$$

$$\frac{\partial V}{\partial y} \approx \frac{V_{(i,j+\frac{1}{2})}^n - V_{(i,j-\frac{1}{2})}^n}{\Delta y} \text{ se } V_{(i,j+\frac{1}{2})}^n \geq 0 \text{ e } \frac{\partial V}{\partial y} \approx \frac{V_{(i,j+\frac{3}{2})}^n - V_{(i,j+\frac{1}{2})}^n}{\Delta y} \text{ se } V_{(i,j+\frac{1}{2})}^n < 0$$

onde a componente U da velocidade discreta não está definida na posição $(i, j + \frac{1}{2})$.

Logo $\bar{U}_{(i,j+\frac{1}{2})}^n$ é aproximado como a média dos quatro pontos circunvizinhos da malha numérica nos quais U^n é conhecido como:

$$\bar{U}_{(i,j+\frac{1}{2})}^n = 0,25 \left(U_{(i-\frac{1}{2},j)}^n + U_{(i-\frac{1}{2},j+1)}^n + U_{(i+\frac{1}{2},j)}^n + U_{(i+\frac{1}{2},j+1)}^n \right)$$

Analogamente à ECQM_x, os termos de atrito inferior e superior são escritos na forma discreta, respectivamente, como:

$$\left(\tau_y^s \right)_{(i+\frac{1}{2},j)}^n \approx \frac{g \sqrt{\left(\bar{U}_{(i,j+\frac{1}{2})}^n \right)^2 + \left(V_{(i,j+\frac{1}{2})}^n \right)^2}}{0,25 \left(C_{h(i,j+1)} + C_{h(i,j)} \right)^2 H v_{(i,j+\frac{1}{2})}^n} \text{ e } \left(\tau_y^v \right)_{(i,j+\frac{1}{2})}^n \approx \frac{C_D \rho_{ar} W_y}{\rho_{\text{água}} H v_{(i,j+\frac{1}{2})}^n}$$

enquanto os termos de Coriolis, e de difusão horizontal que é aproximado via diferenças centrais, são escritos na forma discreta, respectivamente, como:

$$\Phi \bar{U}_{(i,j+\frac{1}{2})}^n \approx \Phi 0,25 \left(U_{(i+\frac{1}{2},j)}^n + U_{(i-\frac{1}{2},j)}^n + U_{(i+\frac{1}{2},j+1)}^n + U_{(i-\frac{1}{2},j+1)}^n \right)$$

e

$$\varepsilon_h \nabla^2 V_{(i,j+\frac{1}{2})}^n \approx \varepsilon_h \left[\left(\frac{V_{(i,j+\frac{3}{2})}^n - 2V_{(i,j+\frac{1}{2})}^n + V_{(i,j-\frac{1}{2})}^n}{\Delta x^2} \right) + \left(\frac{V_{(i+1,j+\frac{3}{2})}^n - 2V_{(i,j+\frac{1}{2})}^n + V_{(i-1,j-\frac{1}{2})}^n}{\Delta y^2} \right) \right]$$

em que os detalhes das deduções para as ESW 3D e ESW 2D-IV podem ser encontrados em [RIZ2002a].

Os operadores explícitos empregados nos esquemas numéricos semi-implícitos para as ESW 2D e para as ESW 3D são como nas expressões abaixo. Especificamente no caso das ESW 2D, eles são escritos como:

$$\left. \begin{aligned} FU_{(i+\frac{1}{2},j)}^n &= U_{(i+\frac{1}{2},j)}^n - \frac{\Delta t}{2\Delta x} \left[\left(U_{(i+\frac{1}{2},j)}^n - \left| U_{(i+\frac{1}{2},j)}^n \right| \right) U_{(i+\frac{3}{2},j)}^n + 2 \left| U_{(i+\frac{1}{2},j)}^n \right| U_{(i+\frac{1}{2},j)}^n \right. \\ &\quad \left. - \left(U_{(i+\frac{1}{2},j)}^n + \left| U_{(i+\frac{1}{2},j)}^n \right| \right) U_{(i-\frac{1}{2},j)}^n \right] - \frac{\Delta t}{2\Delta y} \left[\left(\bar{V}_{(i+\frac{1}{2},j)}^n - \left| \bar{V}_{(i+\frac{1}{2},j)}^n \right| \right) U_{(i+\frac{1}{2},j+1)}^n \right. \\ &\quad \left. + 2 \left| \bar{V}_{(i+\frac{1}{2},j)}^n \right| U_{(i+\frac{1}{2},j)}^n - \left(\bar{V}_{(i+\frac{1}{2},j)}^n + \left| \bar{V}_{(i+\frac{1}{2},j)}^n \right| \right) U_{(i+\frac{1}{2},j-1)}^n \right] \\ &\quad + \Delta t \varepsilon_h \nabla^2 U_{(i+\frac{1}{2},j)}^n + \Delta t \left(\tau_x^v \right)_{(i+\frac{1}{2},j)}^n + \Delta t \Phi \bar{V}_{(i+\frac{1}{2},j)}^n \end{aligned} \right\} (6.3)$$

e como:

$$\left. \begin{aligned}
FV_{(i,j+\frac{1}{2})}^n &= V_{(i,j+\frac{1}{2})}^n - \frac{\Delta t}{2\Delta x} \left[\left(\bar{U}_{(i,j+\frac{1}{2})}^n - \left| \bar{U}_{(i,j+\frac{1}{2})}^n \right| \right) V_{(i+1,j+\frac{1}{2})}^n + 2 \left| \bar{U}_{(i,j+\frac{1}{2})}^n \right| V_{(i,j+\frac{1}{2})}^n \right. \\
&\quad \left. - \left(\bar{U}_{(i,j+\frac{1}{2})}^n + \left| \bar{U}_{(i,j+\frac{1}{2})}^n \right| \right) V_{(i-1,j+\frac{1}{2})}^n \right] - \frac{\Delta t}{2\Delta y} \left[\left(V_{(i,j+\frac{1}{2})}^n - \left| V_{(i,j+\frac{1}{2})}^n \right| \right) V_{(i,j+\frac{3}{2})}^n \right. \\
&\quad \left. + 2 \left| V_{(i,j+\frac{1}{2})}^n \right| V_{(i,j+\frac{1}{2})}^n - \left(V_{(i,j+\frac{1}{2})}^n + \left| V_{(i,j+\frac{1}{2})}^n \right| \right) V_{(i,j-\frac{1}{2})}^n \right] \\
&\quad + \Delta t \varepsilon_n \nabla^2 V_{(i,j+\frac{1}{2})}^n + \Delta t (\tau_y^v)_{(i,j+\frac{1}{2})}^n - \Delta t \Phi \bar{U}_{(i,j+\frac{1}{2})}^n
\end{aligned} \right\} (6.4)$$

de modo que são usados para calcular as velocidades horizontais, depois que os níveis de água são calculados, utilizando-se as expressões (4.5) e (4.6), ou suas análogas 2D (seção 4.3). No caso das ESW 3D, os termos advectivos das componentes horizontal e vertical são obtidos observando as seguintes notações:

$$\begin{aligned}
u_{(i+\frac{1}{2})}^\pm &= \left(u_{(i+\frac{1}{2},j,k)}^n \pm \left| u_{(i+\frac{1}{2},j,k)}^n \right| \right), \quad \bar{v}_{(i+\frac{1}{2})}^\pm = \left(\bar{v}_{(i+\frac{1}{2},j,k)}^n \pm \left| \bar{v}_{(i+\frac{1}{2},j,k)}^n \right| \right), \\
\bar{w}_{(i+\frac{1}{2})}^\pm &= \left(\bar{w}_{(i+\frac{1}{2},j,k)}^n \pm \left| \bar{w}_{(i+\frac{1}{2},j,k)}^n \right| \right), \quad \bar{u}_{(j+\frac{1}{2})}^\pm = \left(\bar{u}_{(i,j+\frac{1}{2},k)}^n \pm \left| \bar{u}_{(i,j+\frac{1}{2},k)}^n \right| \right), \\
v_{(j+\frac{1}{2})}^\pm &= \left(v_{(i,j+\frac{1}{2},k)}^n \pm \left| v_{(i,j+\frac{1}{2},k)}^n \right| \right), \quad \bar{w}_{(j+\frac{1}{2})}^\pm = \left(\bar{w}_{(i,j+\frac{1}{2},k)}^n \pm \left| \bar{w}_{(i,j+\frac{1}{2},k)}^n \right| \right).
\end{aligned}$$

e, então, tem-se:

$$\begin{aligned}
u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} &= \frac{1}{2\Delta x} \left[\bar{u}_{(i+\frac{1}{2})}^- u_{(i+\frac{3}{2},j,k)}^n + 2 \left| u_{(i+\frac{1}{2},j,k)}^n \right| u_{(i+\frac{1}{2},j,k)}^n - \bar{u}_{(i+\frac{1}{2})}^+ u_{(i-\frac{1}{2},j,k)}^n \right] \\
&\quad + \frac{1}{2\Delta y} \left[\bar{v}_{(i+\frac{1}{2})}^- u_{(i+\frac{1}{2},j+1,k)}^n + 2 \left| \bar{v}_{(i+\frac{1}{2},j,k)}^n \right| u_{(i+\frac{1}{2},j,k)}^n - \bar{v}_{(i+\frac{1}{2})}^+ u_{(i+\frac{1}{2},j-1,k)}^n \right] \\
&\quad + \frac{1}{2\Delta z} \left[\bar{w}_{(i+\frac{1}{2})}^- u_{(i+\frac{1}{2},j,k+1)}^n + 2 \left| \bar{w}_{(i+\frac{1}{2},j,k)}^n \right| u_{(i+\frac{1}{2},j,k)}^n - \bar{w}_{(i+\frac{1}{2})}^+ u_{(i+\frac{1}{2},j,k-1)}^n \right]
\end{aligned}$$

e

$$\begin{aligned}
\bar{u} \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} &= \frac{1}{2\Delta x} \left[\bar{u}_{(j+\frac{1}{2})}^- v_{(i+1,j+\frac{1}{2},k)}^n + 2 \left| \bar{u}_{(i,j+\frac{1}{2},k)}^n \right| v_{(i,j+\frac{1}{2},k)}^n - \bar{u}_{(j+\frac{1}{2})}^+ v_{(i-1,j+\frac{1}{2},k)}^n \right] \\
&\quad + \frac{1}{2\Delta y} \left[\bar{v}_{(j+\frac{1}{2})}^- v_{(i,j+\frac{3}{2},k)}^n + 2 \left| \bar{v}_{(i,j+\frac{1}{2},k)}^n \right| v_{(i,j+\frac{1}{2},k)}^n - \bar{v}_{(j+\frac{1}{2})}^+ v_{(i,j-\frac{1}{2},k)}^n \right] \\
&\quad + \frac{1}{2\Delta z} \left[\bar{w}_{(j+\frac{1}{2})}^- v_{(i,j+\frac{1}{2},k+1)}^n + 2 \left| \bar{w}_{(i,j+\frac{1}{2},k)}^n \right| v_{(i,j+\frac{1}{2},k)}^n - \bar{w}_{(j+\frac{1}{2})}^+ v_{(i,j+\frac{1}{2},k-1)}^n \right]
\end{aligned}$$

em que as componentes das velocidades horizontais e vertical \bar{u} , \bar{v} e \bar{w} são aproximadas como uma média dos quatro pontos circunvizinhos da malha numérica, como feito para as componentes \bar{V} e \bar{U} . Detalhes em [RIZ2002a].

6.1.2 Condições de contorno: supressão de termos

Note-se que as expressões obtidas na seção 6.1.1 são válidas apenas para células internas ao domínio. Deve-se obter expressões de modo a contemplar os casos das células adjacentes às fronteiras sul (S) e norte (N) da malha numérica e que não são esquinas da malha; das células adjacentes às fronteiras oeste (W) e leste (E) da malha numérica e que, também, não são esquinas da malha. Essas particularizações são necessárias e devem-se ao fato de que o estêncil empregado nas discretizações das ESW é o de 5-pontos. Especificamente, para as ESW tem-se os seguintes casos:

1. em células internas onde vale $2 \leq i \leq n-1$ e $2 \leq j \leq m-1$, as expressões empregadas para os cálculos são as completas, como as (6.3) e (6.4) e seus termos constituintes;
2. em células adjacentes às fronteiras sul, norte, oeste e leste, mas que não são esquinas da malha, é necessário considerar as seguintes aproximações para determinadas expressões: na fronteira inferior (sul), isto é, para $j=1$ tem-se $0,5(\eta_{(i,0)}^n + \eta_{(i,1)}^n) \approx \eta_{(i,1)}^n$ e $FV_{(i,1/2)}^n \approx V_{(i,1/2)}^{n+1}$; na fronteira superior (norte), isto é, para $j=m$ tem-se $0,5(\eta_{(i,m+1)}^n + \eta_{(i,m)}^n) \approx \eta_{(i,m)}^n$ e $FV_{(i,m+1/2)}^n \approx V_{(i,m+1/2)}^{n+1}$; na fronteira esquerda (oeste), isto é, para $i=1$ tem-se $0,5(\eta_{(0,j)}^n + \eta_{(1,j)}^n) \approx \eta_{(1,j)}^n$ e $FU_{(1/2,j)}^n \approx U_{(1/2,j)}^{n+1}$; na fronteira direita (leste), isto é, para $i=n$, tem-se $0,5(\eta_{(n+1,j)}^n + \eta_{(n,j)}^n) \approx \eta_{(n,j)}^n$ e $FU_{(n+1/2,j)}^n \approx U_{(n+1/2,j)}^{n+1}$;
3. em células que são esquinas da malha $(1,1)$, $(n,1)$, $(1,m)$ e (n,m) , as aproximações são semelhantes: na fronteira esquerda inferior, isto é, em $i=1$ e $j=1$, tem-se $0,5(\eta_{(1,0)}^n + \eta_{(1,1)}^n) \approx \eta_{(1,1)}^n$, $0,5(\eta_{(0,1)}^n + \eta_{(1,1)}^n) \approx \eta_{(1,1)}^n$, $FV_{(1,1/2)}^n \approx V_{(1,1/2)}^{n+1}$ e $FU_{(1/2,1)}^n \approx U_{(1/2,1)}^{n+1}$; na fronteira esquerda superior, isto é, em $i=1$ e $j=m$, $0,5(\eta_{(1,m+1)}^n + \eta_{(1,m)}^n) \approx \eta_{(1,m)}^n$, $0,5(\eta_{(0,m)}^n + \eta_{(1,m)}^n) \approx \eta_{(1,m)}^n$, $FV_{(1,m+1/2)}^n \approx V_{(1,m+1/2)}^{n+1}$ e $FU_{(1/2,m)}^n \approx U_{(1/2,m)}^{n+1}$; na fronteira direita inferior, isto é, em $i=n$ e $j=1$, tem-se $0,5(\eta_{(n+1,1)}^n + \eta_{(n,1)}^n) \approx \eta_{(n,1)}^n$, $0,5(\eta_{(n,0)}^n + \eta_{(n,1)}^n) \approx \eta_{(n,1)}^n$, $FU_{(n+1/2,1)}^n \approx U_{(n+1/2,1)}^{n+1}$ e $FV_{(n,1/2)}^n \approx V_{(n,1/2)}^{n+1}$; e na fronteira direita superior, ou seja, em $i=n$ e $j=m$, tem-se $0,5(\eta_{(n+1,m)}^n + \eta_{(n,m)}^n) \approx \eta_{(n,m)}^n$, $FU_{(n+1/2,m)}^n \approx U_{(n+1/2,m)}^{n+1}$, $0,5(\eta_{(n,m+1)}^n + \eta_{(n,m)}^n) \approx \eta_{(n,m)}^n$, e $FV_{(n,m+1/2)}^n \approx V_{(n,m+1/2)}^{n+1}$.

Outras expressões são aproximadas através da supressão de termos, de modo que estejam definidas no domínio computacional. Uma abordagem mais sofisticada é desenvolvida neste trabalho para a ETM e usa a técnica de extrapolação, como pode ser visto na seção 6.5.1.

6.2 Esquema semi-implícito fluxo numérico: ETM 2D-IV

Após determinadas manipulações algébricas, que podem ser vistas em [RIZ2002a], o esquema semi-implícito gera sistemas de equações lineares cujas matrizes são do *tipo 5-diagonal* não simétricas. Se a componente advectiva da ETM 2D é aproximada via esquema *upwind* fluxo numérico e a difusão horizontal é via Crank-Nicolson, os sistemas de equações resultantes são lineares, e suas únicas incógnitas são as concentrações S . Esse sistema de equações é escrito na forma matricial como:

$$E_{(i-1,j)} S_{(i-1,j)}^{n+1} + C_{(i,j)} S_{(i,j)}^{n+1} + D_{(i+1,j)} S_{(i+1,j)}^{n+1} + F_{(i,j-1)} S_{(i,j-1)}^{n+1} + S_{(i,j+1)} S_{(i,j+1)}^{n+1} = B_{(i,j)}^n \quad (6.5)$$

em que os coeficientes de (6.5) são como:

$$E_{(i-1,j)} = \left[-\frac{\Delta t}{2\Delta x} \left(U_{(i-1/2,j)}^{n+1} + \left| U_{(i-1/2,j)}^{n+1} \right| \right) H_{(i-1/2,j)}^{n+1} \right] - \frac{\Delta t}{2\Delta x^2} D_{x(i-1/2,j)}^{n+1} H_{(i-1/2,j)}^{n+1}$$

$$\begin{aligned}
D_{(i+1,j)} &= \left[\frac{\Delta t}{2\Delta x} \left(U_{(i+\frac{1}{2},j)}^{n+1} - \left| U_{(i+\frac{1}{2},j)}^{n+1} \right| \right) H_{(i+\frac{1}{2},j)}^{n+1} \right] - \frac{\Delta t}{2\Delta x^2} D_{x(i+\frac{1}{2},j)}^{n+1} H_{(i+\frac{3}{2},j)}^{n+1} \\
F_{(i,j-1)} &= \left[-\frac{\Delta t}{2\Delta y} \left(V_{(i,j-\frac{1}{2})}^{n+1} + \left| V_{(i,j-\frac{1}{2})}^{n+1} \right| \right) H_{(i-\frac{1}{2},j)}^{n+1} \right] - \frac{\Delta t}{2\Delta y^2} D_{y(i-\frac{1}{2},j)}^{n+1} H_{(i-\frac{1}{2},j)}^{n+1} \\
S_{(i,j+1)} &= \left[\frac{\Delta t}{2\Delta y} \left(V_{(i,j+\frac{1}{2})}^{n+1} - \left| V_{(i,j+\frac{1}{2})}^{n+1} \right| \right) H_{(i+\frac{1}{2},j)}^{n+1} \right] - \frac{\Delta t}{2\Delta y^2} D_{y(i,j+\frac{1}{2})}^{n+1} H_{(i,j+\frac{3}{2})}^{n+1} \\
C_{(i,j)} &= \frac{\Delta t}{2\Delta x} \left[\left(U_{(i+\frac{1}{2},j)}^{n+1} + \left| U_{(i+\frac{1}{2},j)}^{n+1} \right| \right) H_{(i+\frac{1}{2},j)}^{n+1} \right] - \frac{\Delta t}{2\Delta x} \left[\left(U_{(i-\frac{1}{2},j)}^{n+1} - \left| U_{(i-\frac{1}{2},j)}^{n+1} \right| \right) H_{(i-\frac{1}{2},j)}^{n+1} \right] \\
&+ \frac{\Delta t}{2\Delta y} \left[\left(V_{(i,j+\frac{1}{2})}^{n+1} + \left| V_{(i,j+\frac{1}{2})}^{n+1} \right| \right) H_{(i,j+\frac{1}{2})}^{n+1} \right] - \frac{\Delta t}{2\Delta y} \left[\left(V_{(i,j-\frac{1}{2})}^{n+1} - \left| V_{(i,j-\frac{1}{2})}^{n+1} \right| \right) H_{(i,j-\frac{1}{2})}^{n+1} \right] \\
&+ \frac{\Delta t}{2\Delta x^2} D_{x(i+\frac{1}{2},j)}^{n+1} H_{(i+\frac{1}{2},j)}^{n+1} - \frac{\Delta t}{2\Delta x^2} D_{x(i-\frac{1}{2},j)}^{n+1} H_{(i+\frac{1}{2},j)}^{n+1} + \frac{\Delta t}{2\Delta y^2} D_{y(i,j+\frac{1}{2})}^{n+1} H_{(i,j+\frac{1}{2})}^{n+1} \\
&- \frac{\Delta t}{2\Delta y^2} D_{y(i,j-\frac{1}{2})}^{n+1} H_{(i,j+\frac{1}{2})}^{n+1} + H_{(i,j)}^{n+1} \\
B_{(i,j)}^n &= \frac{\Delta t}{2\Delta x^2} D_{x(i+\frac{1}{2},j)}^n \left(S_{(i+1,j)}^n H_{(i+\frac{3}{2},j)}^n - S_{(i,j)}^n H_{(i+\frac{1}{2},j)}^n \right) - \frac{\Delta t}{2\Delta x^2} D_{x(i-\frac{1}{2},j)}^n \left(S_{(i,j)}^n H_{(i+\frac{1}{2},j)}^n - \right. \\
&- S_{(i-1,j)}^n H_{(i-\frac{1}{2},j)}^n \left. + \frac{\Delta t}{2\Delta y^2} D_{y(i,j+\frac{1}{2})}^n \left(S_{(i,j+1)}^n H_{(i,j+\frac{3}{2})}^n - S_{(i,j)}^n H_{(i,j+\frac{1}{2})}^n \right) - \frac{\Delta t}{2\Delta y^2} D_{y(i,j-\frac{1}{2})}^n \right. \\
&\left. \left(S_{(i,j)}^n H_{(i,j+\frac{1}{2})}^n - S_{(i,j-1)}^n H_{(i,j-\frac{1}{2})}^n \right) + S_{(i,j)}^n H_{(i,j)}^n (1 + K\Delta t) + \Delta t H_{(i,j)}^n S_{fonte}^n \right.
\end{aligned}$$

onde, para I e J , são usadas as funções sinal definidas na seção 4.4 e:

$$\begin{aligned}
H_{(i+\frac{1}{2},j)}^{n+1} &= h_{u(i+\frac{1}{2},j)}^{n+1} + \eta_{(i+(I+1/2),j)}^{n+1}, \quad H_{(i,j+\frac{1}{2})}^{n+1} = h_{v(i,j+\frac{1}{2})}^{n+1} + \eta_{(i,j+(J+1/2))}^{n+1} \\
H_{(i,j)}^{n+1} &= 0,5 \left(h_{u(i+\frac{1}{2},j)}^{n+1} + h_{v(i,j+\frac{1}{2})}^{n+1} \right) + \eta_{(i,j)}^{n+1}
\end{aligned}$$

onde $K = 2,3/T_{90}$, e os coeficientes longitudinais e transversais, como na seção 2.6, são $D_L = 0,23 U_* H_u$ e $D_T = 5,93 U_* H_v$, onde $U_* = g(C_z)^{-1} \left(\sqrt{U^2 + V^2} \right)$.

O sistema de equações (6.5) originou-se quando do emprego da abordagem tipo fluxo numérico para obter uma solução *upwind* para a advecção. Nesse caso, os valores nas faces de uma célula 2D podem ser obtidos, para a componente $U = U(x, y, t)$, considerando a seguinte definição:

$$\text{sign} \left(U_{(i+\frac{1}{2},j)}^{n+1} \right) \stackrel{\text{def}}{=} \frac{\left| U_{(i+\frac{1}{2},j)}^{n+1} \right|}{U_{(i+\frac{1}{2},j)}^{n+1}}, \text{ isto é, } \text{sign} \left(U_{(i+\frac{1}{2},j)}^{n+1} \right) = \begin{cases} 1, & \text{se } U_{(i+\frac{1}{2},j)}^{n+1} > 0 \\ -1, & \text{se } U_{(i+\frac{1}{2},j)}^{n+1} < 0 \end{cases}$$

e que:

$$1 \pm \text{sign} \left(U_{(i+\frac{1}{2},j)}^{n+1} \right) = 1 \pm \frac{\left| U_{(i+\frac{1}{2},j)}^{n+1} \right|}{U_{(i+\frac{1}{2},j)}^{n+1}} = \frac{U_{(i+\frac{1}{2},j)}^{n+1} \pm \left| U_{(i+\frac{1}{2},j)}^{n+1} \right|}{U_{(i+\frac{1}{2},j)}^{n+1}}$$

Assim, uma aproximação via fluxo numérico fornece, para a face direita da célula, a seguinte expressão:

$$\begin{aligned}
S_E &= \frac{1}{2} \left[1 + \text{sign} \left(u_{(i+\frac{1}{2},j)}^{n+1} \right) \right] S_{(i,j)}^{n+1} + \frac{1}{2} \left[1 - \text{sign} \left(u_{(i+\frac{1}{2},j)}^{n+1} \right) \right] S_{(i+1,j)}^{n+1} \\
&= \frac{1}{2} \left[\frac{u_{(i+\frac{1}{2},j)}^{n+1} + |u_{(i+\frac{1}{2},j)}^{n+1}|}{u_{(i+\frac{1}{2},j)}^{n+1}} \right] S_{(i,j)}^{n+1} + \frac{1}{2} \left[\frac{u_{(i+\frac{1}{2},j)}^{n+1} - |u_{(i+\frac{1}{2},j)}^{n+1}|}{u_{(i+\frac{1}{2},j)}^{n+1}} \right] S_{(i+1,j)}^{n+1}
\end{aligned}$$

e, portanto:

$$\begin{aligned}
S_{(i+\frac{1}{2},j)}^{n+1} &\stackrel{\text{not}}{\equiv} u_{(i+\frac{1}{2},j)}^{n+1} S_E \\
&= \frac{1}{2} \left(u_{(i+\frac{1}{2},j)}^{n+1} + |u_{(i+\frac{1}{2},j)}^{n+1}| \right) S_{(i,j)}^{n+1} + \frac{1}{2} \left(u_{(i+\frac{1}{2},j)}^{n+1} - |u_{(i+\frac{1}{2},j)}^{n+1}| \right) S_{(i+1,j)}^{n+1}
\end{aligned}$$

De modo análogo, tem-se a seguinte expressão para a face esquerda:

$$\begin{aligned}
S_W &= \frac{1}{2} \left[1 + \text{sign} \left(u_{(i-\frac{1}{2},j)}^{n+1} \right) \right] S_{(i-1,j)}^{n+1} + \frac{1}{2} \left[1 - \text{sign} \left(u_{(i-\frac{1}{2},j)}^{n+1} \right) \right] S_{(i,j)}^{n+1} \\
&= \frac{1}{2} \left[\frac{u_{(i-\frac{1}{2},j)}^{n+1} + |u_{(i-\frac{1}{2},j)}^{n+1}|}{u_{(i-\frac{1}{2},j)}^{n+1}} \right] S_{(i-1,j)}^{n+1} + \frac{1}{2} \left[\frac{u_{(i-\frac{1}{2},j)}^{n+1} - |u_{(i-\frac{1}{2},j)}^{n+1}|}{u_{(i-\frac{1}{2},j)}^{n+1}} \right] S_{(i,j)}^{n+1}
\end{aligned}$$

e portanto:

$$\begin{aligned}
S_{(i-\frac{1}{2},j)}^{n+1} &\stackrel{\text{not}}{\equiv} S_W u_{(i-\frac{1}{2},j)}^{n+1} \\
&= \frac{1}{2} \left(u_{(i-\frac{1}{2},j)}^{n+1} + |u_{(i-\frac{1}{2},j)}^{n+1}| \right) S_{(i-1,j)}^{n+1} + \frac{1}{2} \left(u_{(i-\frac{1}{2},j)}^{n+1} - |u_{(i-\frac{1}{2},j)}^{n+1}| \right) S_{(i,j)}^{n+1}
\end{aligned}$$

Substituindo nas expressões anteriores a componente u pela componente v da velocidade e os índices i por j , obtêm-se expressões como nas faces de cima e de baixo. Especificamente essas expressões são como:

$$\begin{aligned}
S_{(i,j+\frac{1}{2})}^{n+1} &\stackrel{\text{not}}{\equiv} v_{(i,j+\frac{1}{2})}^{n+1} S_N \\
&= \frac{1}{2} \left(v_{(i,j+\frac{1}{2})}^{n+1} + |v_{(i,j+\frac{1}{2})}^{n+1}| \right) S_{(i,j)}^{n+1} + \frac{1}{2} \left(v_{(i,j+\frac{1}{2})}^{n+1} - |v_{(i,j+\frac{1}{2})}^{n+1}| \right) S_{(i,j+1)}^{n+1}
\end{aligned}$$

e

$$\begin{aligned}
S_{(i,j-\frac{1}{2})}^{n+1} &\stackrel{\text{not}}{\equiv} S_W v_{(i,j-\frac{1}{2})}^{n+1} \\
&= \frac{1}{2} \left(v_{(i,j-\frac{1}{2})}^{n+1} + |v_{(i,j-\frac{1}{2})}^{n+1}| \right) S_{(i,j-1)}^{n+1} + \frac{1}{2} \left(v_{(i,j-\frac{1}{2})}^{n+1} - |v_{(i,j-\frac{1}{2})}^{n+1}| \right) S_{(i,j)}^{n+1}
\end{aligned}$$

Os resultados numéricos obtidos pelo emprego dessa abordagem na solução da ETM 2D-IV são semelhantes àqueles apresentados nas tab. da seção 5.1.4. A diferença resume-se ao fato de que a discretização para a equação de transporte escalar 2D-IV considerou uma abordagem semi-implícita gerando um sistema de equações como descrito pela expressão (6.5). Nesse caso, o passo de tempo não é limitado pelo número de Courant.

6.3 Esquemas semi-implícitos: método ADI

Alguns dos esquemas numéricos para soluções de EDPs 2D implementados neste trabalho empregaram o método ADI como estratégia para melhorar a eficiência computacional, dado que os sistemas de equações gerados com essa abordagem são 3-diagonal. Após certa manipulação algébrica [RIZ2002a], e considerando a abordagem

de Casulli, o esquema numérico via método ADI decompõe o passo de tempo em dois. No primeiro meio passo de tempo, os termos na direção Y são considerados explícitos, e no segundo meio passo de tempo os termos na direção X são considerados explícitos.

Com isso, geram-se dois conjuntos de sistemas de equações lineares com matrizes SDP, um para cada direção coordenada. Para o primeiro meio passo de tempo, o sistema de equações lineares é escrito como:

$$E_{(i-1,j)}\eta_{(i-1,j)}^{n+\frac{1}{2}} + C_{(i,j)}\eta_{(i,j)}^{n+\frac{1}{2}} + D_{(i+1,j)}\eta_{(i+1,j)}^{n+\frac{1}{2}} = B_{(i,j)}^n \quad (6.6)$$

e os coeficientes em (6.6) para o nível de tempo $[n, n + \frac{1}{2}]$ são:

$$C_{(i,j)} = 1 + \frac{g}{2} \left(\frac{\Delta t}{\Delta x} \right)^2 \left(\frac{Hu_{(i+\frac{1}{2},j)}^n}{1 + \gamma_{x(i+\frac{1}{2},j)}^n \Delta t} + \frac{Hu_{(i-\frac{1}{2},j)}^n}{1 + \gamma_{x(i-\frac{1}{2},j)}^n \Delta t} \right)$$

$$E_{(i-1,j)} = -\frac{g}{2} \left(\frac{\Delta t}{\Delta x} \right)^2 \left(\frac{Hu_{(i-\frac{1}{2},j)}^n}{1 + \gamma_{x(i-\frac{1}{2},j)}^n \Delta t} \right)$$

$$D_{(i+1,j)} = -\frac{g}{2} \left(\frac{\Delta t}{\Delta x} \right)^2 \left(\frac{Hu_{(i+\frac{1}{2},j)}^n}{1 + \gamma_{x(i+\frac{1}{2},j)}^n \Delta t} \right)$$

$$B_{(i,j)}^n = \eta_{(i,j)}^n - \left(\frac{\Delta t}{2\Delta y} \right) \left(Hv_{(i,j+\frac{1}{2})}^n V_{(i,j+\frac{1}{2})}^n - Hv_{(i,j-\frac{1}{2})}^n V_{(i,j-\frac{1}{2})}^n \right)$$

$$- \left(\frac{\Delta t}{2\Delta x} \right) \left[\left(\frac{Hu_{(i+\frac{1}{2},j)}^n}{1 + \gamma_{x(i+\frac{1}{2},j)}^n \Delta t} \right) \left(FU_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} \right) - \left(\frac{Hu_{(i-\frac{1}{2},j)}^n}{1 + \gamma_{x(i-\frac{1}{2},j)}^n \Delta t} \right) \left(FU_{(i-\frac{1}{2},j)}^{n-\frac{1}{2}} \right) \right]$$

Após os níveis de água $\eta_{(i-1,j)}^{n+\frac{1}{2}}, \eta_{(i,j)}^{n+\frac{1}{2}}, \eta_{(i+1,j)}^{n+\frac{1}{2}}$ serem calculados, as velocidades $U_{(i+\frac{1}{2},j)}^{n+\frac{1}{2}}$ são recuperadas através de uma expressão semelhante a (4.5). Mais especificamente, através da expressão:

$$U_{(i+\frac{1}{2},j)}^{n+\frac{1}{2}} = \left(1 + \Delta t (\tau_x^s)_{(i+\frac{1}{2},j)}^n \right)^{-1} \left[FU_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} - g \frac{\Delta t}{\Delta x} (\eta_{(i+1,j)}^{n+\frac{1}{2}} - \eta_{(i,j)}^{n+\frac{1}{2}}) \right]$$

em que o operador explícito $FU_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}}$ é dado como:

$$FU_{(i+\frac{1}{2},j)}^n = U_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} - \frac{\Delta t}{2\Delta x} \left[\left(U_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} - \left| U_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} \right| \right) U_{(i+\frac{3}{2},j)}^{n-\frac{1}{2}} + 2 \left| U_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} \right| U_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} \right.$$

$$\left. - \left(U_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} + \left| U_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} \right| \right) U_{(i-\frac{1}{2},j)}^{n-\frac{1}{2}} \right] - \frac{\Delta t}{2\Delta y} \left[\left(\bar{V}_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} - \left| \bar{V}_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} \right| \right) U_{(i+\frac{1}{2},j+1)}^{n-\frac{1}{2}} \right.$$

$$\left. + 2 \left| \bar{V}_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} \right| U_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} - \left(\bar{V}_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} + \left| \bar{V}_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} \right| \right) U_{(i+\frac{1}{2},j-1)}^{n-\frac{1}{2}} \right]$$

$$+ \Delta t \varepsilon_h \nabla^2 U_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} + \Delta t (\tau_x^v)_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}} + \Delta t \Phi \bar{V}_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}}$$

Para o segundo meio passo de tempo $[n + \frac{1}{2}, n + 1]$, o sistema de equações lineares é escrito como:

$$E_{(i,j-1)}\eta_{(i,j-1)}^{n+1} + C_{(i,j)}\eta_{(i,j)}^{n+1} + D_{(i,j+1)}\eta_{(i,j+1)}^{n+1} = B_{(i,j)}^{n+\frac{1}{2}} \quad (6.7)$$

em que os coeficientes em (6.7) são escritos como:

$$\begin{aligned}
 E_{(i,j-1)} &= -\frac{g}{2} \left(\frac{\Delta t}{\Delta y} \right)^2 \left(\frac{Hv_{(i,j-\frac{1}{2})}^{n+\frac{1}{2}}}{1 + \gamma_{y(i,j-\frac{1}{2})}^{n+\frac{1}{2}} \Delta t} \right) \\
 D_{(i,j+1)} &= -\frac{g}{2} \left(\frac{\Delta t}{\Delta y} \right)^2 \left(\frac{Hv_{(i,j+\frac{1}{2})}^{n+\frac{1}{2}}}{1 + \gamma_{y(i,j+\frac{1}{2})}^{n+\frac{1}{2}} \Delta t} \right) \\
 C_{(i,j)} &= 1 + \frac{g}{2} \left(\frac{\Delta t}{\Delta y} \right)^2 \left(\frac{Hv_{(i,j+\frac{1}{2})}^{n+\frac{1}{2}}}{1 + \gamma_{y(i,j+\frac{1}{2})}^{n+\frac{1}{2}} \Delta t} + \frac{Hv_{(i,j-\frac{1}{2})}^{n+\frac{1}{2}}}{1 + \gamma_{y(i,j-\frac{1}{2})}^{n+\frac{1}{2}} \Delta t} \right) \\
 B_{(i,j)}^{n+\frac{1}{2}} &= \eta_{(i,j)}^{n+\frac{1}{2}} - \left(\frac{\Delta t}{2\Delta x} \right) \left(Hu_{(i+\frac{1}{2},j)}^{n+\frac{1}{2}} U_{(i+\frac{1}{2},j)}^{n+\frac{1}{2}} - Hu_{(i-\frac{1}{2},j)}^{n+\frac{1}{2}} U_{(i-\frac{1}{2},j)}^{n+\frac{1}{2}} \right) \\
 &\quad - \left(\frac{\Delta t}{2\Delta y} \right) \left[\left(\frac{Hv_{(i,j+\frac{1}{2})}^{n+\frac{1}{2}}}{1 + \gamma_{y(i,j+\frac{1}{2})}^{n+\frac{1}{2}} \Delta t} \right) \left(FV_{(i,j+\frac{1}{2})}^n \right) - \left(\frac{Hv_{(i,j-\frac{1}{2})}^{n+\frac{1}{2}}}{1 + \gamma_{y(i,j-\frac{1}{2})}^{n+\frac{1}{2}} \Delta t} \right) \left(FV_{(i,j-\frac{1}{2})}^n \right) \right]
 \end{aligned}$$

e, depois que os níveis de água $\eta_{(i,j-1)}^{n+1}, \eta_{(i,j)}^{n+1}, \eta_{(i,j+1)}^{n+1}$ são calculados, as velocidades $V_{(i,j+\frac{1}{2})}^{n+1}$ são recuperadas através da expressão:

$$V_{(i,j+\frac{1}{2})}^{n+1} = \left(1 + \Delta t \left(\tau_y^s \right)_{(i,j+\frac{1}{2})}^{n+\frac{1}{2}} \right)^{-1} \left[FV_{(i,j+\frac{1}{2})}^n - g \frac{\Delta t}{\Delta y} \left(\eta_{(i,j+1)}^{n+1} - \eta_{(i,j)}^{n+1} \right) \right]$$

em que o operador explícito $FV_{(i,j+\frac{1}{2})}^n$ é dado como:

$$\begin{aligned}
 FV_{(i,j+\frac{1}{2})}^n &= V_{(i,j+\frac{1}{2})}^n - \frac{\Delta t}{2\Delta x} \left[\left(\bar{U}_{(i,j+\frac{1}{2})}^n - \left| \bar{U}_{(i,j+\frac{1}{2})}^n \right| \right) V_{(i+1,j+\frac{1}{2})}^n + 2 \left| \bar{U}_{(i,j+\frac{1}{2})}^n \right| V_{(i,j+\frac{1}{2})}^n \right. \\
 &\quad \left. - \left(\bar{U}_{(i,j+\frac{1}{2})}^n + \left| \bar{U}_{(i,j+\frac{1}{2})}^n \right| \right) V_{(i-1,j+\frac{1}{2})}^n \right] - \frac{\Delta t}{2\Delta y} \left[\left(V_{(i,j+\frac{1}{2})}^n - \left| V_{(i,j+\frac{1}{2})}^n \right| \right) V_{(i,j+\frac{3}{2})}^n \right. \\
 &\quad \left. + 2 \left| V_{(i,j+\frac{1}{2})}^n \right| V_{(i,j+\frac{1}{2})}^n - \left(V_{(i,j+\frac{1}{2})}^n + \left| V_{(i,j+\frac{1}{2})}^n \right| \right) V_{(i,j-\frac{1}{2})}^n \right] \\
 &\quad + \Delta t \varepsilon_h \nabla^2 V_{(i,j+\frac{1}{2})}^n + \Delta t \left(\tau_y^v \right)_{(i,j+\frac{1}{2})}^n - \Delta t \Phi \bar{U}_{(i,j+\frac{1}{2})}^n
 \end{aligned}$$

As aproximações para as expressões $\left(\tau_x^s \right)_{(i+\frac{1}{2},j)}^n$, $\nabla^2 U_{(i+\frac{1}{2},j)}^n$, $\left(\tau_x^v \right)_{(i+\frac{1}{2},j)}^n$, $Hu_{(i+\frac{1}{2},j)}^n$, $FU_{(i+\frac{1}{2},j)}^{n-\frac{1}{2}}$, $\bar{V}_{(i+\frac{1}{2},j)}^n$, $\left(\tau_y^s \right)_{(i,j+\frac{1}{2})}^{n+\frac{1}{2}}$, $\nabla^2 V_{(i,j+\frac{1}{2})}^{n+\frac{1}{2}}$, $\left(\tau_y^v \right)_{(i,j+\frac{1}{2})}^{n+\frac{1}{2}}$, $Hv_{(i,j+\frac{1}{2})}^{n+\frac{1}{2}}$, $FV_{(i,j+\frac{1}{2})}^n$ e $\bar{U}_{(i,j+\frac{1}{2})}^{n+\frac{1}{2}}$ são como as da seção 6.1.1, exceto o nível de tempo, que deve considerar os meios passos de tempo do esquema ADI. Os sistemas de equações lineares 3-diagonal para as linhas i e colunas j , tais $1 \leq i \leq N_x$ e $1 \leq j \leq N_y$ são como tratados na seção 3.5.3. Detalhes podem ser vistos em [RIZ2002a].

Outros esquemas semi-implícitos foram desenvolvidos e implementados para a solução das ESW 2D-IV e da ETM 2D-IV. Nesse esquemas empregou-se a abordagem de Leendertse, mas esses esquemas numéricos não são detalhados neste trabalho e podem ser vistos em [RIZ2002a]. Os sistemas de equações resultantes do emprego da abordagem de Leendertse (o uso do método ADI e solução via algoritmo de Thomas) são não simétricas, dado que são gerados pelo acoplamento das equações ECQM_x, EC_x, ECQM_y e EC_y.

6.4 Esquema numérico para advecção horizontal: método FLM

A construção de solução para a advecção horizontal da ETM 3D emprega o método de restrição de fluxos. Os fluxos horizontais são obtidos com o método de fluxos de Sweby [SWE84], como em [BLO94], [HUN95], [ZIJ95], [SPE96], [HUN96] e [HEI99]. Além disso, as interpolações podem ser obtidas empregando o método $-\theta$ de modo que os esquemas resultantes tenham uma acurácia temporal de segunda ordem. A advecção na forma discreta e conservativa é escrita como:

$$\varphi_{(i,j,k)}^{n+1} = \varphi_{(i,j,k)}^n - \frac{\Delta t}{\Delta x} \left(F_{(i+\frac{1}{2},j,k)} - F_{(i-\frac{1}{2},j,k)} \right) - \frac{\Delta t}{\Delta y} \left(F_{(i,j+\frac{1}{2},k)} - F_{(i,j-\frac{1}{2},k)} \right)$$

e os fluxos numéricos nas faces das células são como em [SPE96]:

$$F_{(i\pm\frac{1}{2},j,k)}^n = \max\left(u_{(i\pm\frac{1}{2},j,k)}^n, 0\right) \left(F_{(i\pm\frac{1}{2},j,k)}^n\right)^+ + \min\left(u_{(i\pm\frac{1}{2},j,k)}^n, 0\right) \left(F_{(i\pm\frac{1}{2},j,k)}^n\right)^- \\ F_{(i,j\pm\frac{1}{2},k)}^n = \max\left(v_{(i,j\pm\frac{1}{2},k)}^n, 0\right) \left(F_{(i,j\pm\frac{1}{2},k)}^n\right)^+ + \min\left(v_{(i,j\pm\frac{1}{2},k)}^n, 0\right) \left(F_{(i,j\pm\frac{1}{2},k)}^n\right)^-$$

em que esses fluxos numéricos, como discutido no anexo 1, são obtidas empregando funções interpolantes construídas com o método FLM, e são como:

$$\left(F_{(i\pm\frac{1}{2},j,k)}^n\right)^+, \left(F_{(i\pm\frac{1}{2},j,k)}^n\right)^-, \left(F_{(i,j\pm\frac{1}{2},k)}^n\right)^+ \text{ e } \left(F_{(i,j\pm\frac{1}{2},k)}^n\right)^-$$

As soluções FLM são escritas a partir das definições dos números de Courant locais horizontais, isto é, aqueles das direções X e Y , que são como:

$$C_{(i\pm\frac{1}{2},j,k)}^{n+\theta} = \frac{\Delta t}{\Delta x} u_{(i\pm\frac{1}{2},j,k)}^{n+\theta} \text{ e } C_{(i,j\pm\frac{1}{2},k)}^{n+\theta} = \frac{\Delta t}{\Delta y} v_{(i,j\pm\frac{1}{2},k)}^{n+\theta}$$

sendo $u_{(i\pm\frac{1}{2},j,k)}^{n+\theta} = \theta u_{(i\pm\frac{1}{2},j,k)}^{n+1} + (1-\theta) u_{(i\pm\frac{1}{2},j,k)}^n$ e $v_{(i,j\pm\frac{1}{2},k)}^{n+\theta} = \theta v_{(i,j\pm\frac{1}{2},k)}^{n+1} + (1-\theta) v_{(i,j\pm\frac{1}{2},k)}^n$.

Usando a definição dos números de Courant locais e das velocidades como anteriormente introduzidos, os fluxos nas faces são obtidos com uma abordagem FLM trapezoidal, isto é, do emprego do método $-\theta$, para obter um esquema numérico geral, em dois níveis de tempo, como:

1. considerando a notação $S_{(i+\frac{1}{2})}^{n+\theta} \equiv (S_E)^{n+\theta}$, tem-se:

$$(S_E)^{n+\theta} = \begin{cases} C_{(i+\frac{1}{2},j,k)}^{n+\theta} > 0 \Rightarrow (S_E^+)^{n+\theta} = \theta \left[S_{(i,j,k)}^{n+1} + \frac{1}{2} \Psi \left(r_{(i+\frac{1}{2},j,k)}^+ \right)^{n+1} \left(S_{(i,j,k)}^{n+1} - S_{(i-1,j,k)}^{n+1} \right) \right] \\ \quad + (1-\theta) \left[S_{(i,j,k)}^n + \frac{1}{2} \Psi \left(r_{(i+\frac{1}{2},j,k)}^+ \right)^n \left(S_{(i,j,k)}^n - S_{(i-1,j,k)}^n \right) \right] \\ C_{(i+\frac{1}{2},j,k)}^{n+\theta} < 0 \Rightarrow (S_E^-)^{n+\theta} = \theta \left[S_{(i+1,j,k)}^{n+1} - \frac{1}{2} \Psi \left(r_{(i+\frac{1}{2},j,k)}^- \right)^{n+1} \left(S_{(i+2,j,k)}^{n+1} - S_{(i+1,j,k)}^{n+1} \right) \right] \\ \quad + (1-\theta) \left[S_{(i+1,j,k)}^n - \frac{1}{2} \Psi \left(r_{(i+\frac{1}{2},j,k)}^- \right)^n \left(S_{(i+2,j,k)}^n - S_{(i+1,j,k)}^n \right) \right] \end{cases}$$

onde as funções monitoras são, para $\varepsilon \approx 10^{-9}$, definidas por:

$$r_{(i+\frac{1}{2},j,k)}^+ = \frac{S_{(i+1,j,k)}^n - S_{(i,j,k)}^n + \varepsilon}{S_{(i,j,k)}^n - S_{(i-1,j,k)}^n + \varepsilon} \text{ e } r_{(i+\frac{1}{2},j,k)}^- = \frac{S_{(i+1,j,k)}^n - S_{(i,j,k)}^n + \varepsilon}{S_{(i+2,j,k)}^n - S_{(i+1,j,k)}^n + \varepsilon}$$

e, portanto:

$$\begin{aligned}
(S_E)^{n+\theta} &= \max\left(C_{(i+\frac{1}{2},j,k)}^{n+\theta}, 0\right)(S_E^+)^{n+\theta} + \min\left(C_{(i+\frac{1}{2},j,k)}^{n+\theta}, 0\right)(S_E^-)^{n+\theta} \\
&= \frac{1}{2}\left(C_{(i+\frac{1}{2},j,k)}^{n+\theta} + \left|C_{(i+\frac{1}{2},j,k)}^{n+\theta}\right|\right)(S_E^+)^{n+\theta} + \frac{1}{2}\left(C_{(i+\frac{1}{2},j,k)}^{n+\theta} - \left|C_{(i+\frac{1}{2},j,k)}^{n+\theta}\right|\right)(S_E^-)^{n+\theta}
\end{aligned}$$

2. analogamente, com a notação $(S_{(i-\frac{1}{2})})^{n+\theta} \equiv (S_W)^{n+\theta}$, tem-se:

$$(S_W)^{n+\theta} = \begin{cases} C_{(i-\frac{1}{2},j,k)}^{n+\theta} > 0 \Rightarrow (S_W^+)^{n+\theta} = \theta \left[S_{(i-1,j,k)}^{n+1} + \frac{1}{2} \Psi(r_{(i-\frac{1}{2},j,k)}^+)^{n+1} (S_{(i-1,j,k)}^{n+1} - S_{(i-2,j,k)}^{n+1}) \right] \\ \quad + (1-\theta) \left[S_{(i-1,j,k)}^n + \frac{1}{2} \Psi(r_{(i-\frac{1}{2},j,k)}^+)^n (S_{(i-1,j,k)}^n - S_{(i-2,j,k)}^n) \right] \\ C_{(i-\frac{1}{2},j,k)}^{n+\theta} < 0 \Rightarrow (S_W^-)^{n+\theta} = \theta \left[S_{(i,j,k)}^{n+1} - \frac{1}{2} \Psi(r_{(i-\frac{1}{2},j,k)}^-)^{n+1} (S_{(i+1,j,k)}^{n+1} - S_{(i,j,k)}^{n+1}) \right] \\ \quad + (1-\theta) \left[S_{(i,j,k)}^n - \frac{1}{2} \Psi(r_{(i-\frac{1}{2},j,k)}^-)^n (S_{(i+1,j,k)}^n - S_{(i,j,k)}^n) \right] \end{cases}$$

onde as funções monitoras são definidas por:

$$r_{(i-\frac{1}{2},j,k)}^+ = \frac{S_{(i,j,k)}^n - S_{(i-1,j,k)}^n + \varepsilon}{S_{(i-1,j,k)}^n - S_{(i-2,j,k)}^n + \varepsilon} \text{ e } r_{(i-\frac{1}{2},j,k)}^- = \frac{S_{(i,j,k)}^n - S_{(i-1,j,k)}^n + \varepsilon}{S_{(i+1,j,k)}^n - S_{(i,j,k)}^n + \varepsilon}$$

e, portanto:

$$\begin{aligned}
(S_W)^{n+\theta} &= \max\left(C_{(i-\frac{1}{2},j,k)}^{n+\theta}, 0\right)(S_W^+)^{n+\theta} + \min\left(C_{(i-\frac{1}{2},j,k)}^{n+\theta}, 0\right)(S_W^-)^{n+\theta} \\
&= \frac{1}{2}\left(C_{(i-\frac{1}{2},j,k)}^{n+\theta} + \left|C_{(i-\frac{1}{2},j,k)}^{n+\theta}\right|\right)(S_W^+)^{n+\theta} + \frac{1}{2}\left(C_{(i-\frac{1}{2},j,k)}^{n+\theta} - \left|C_{(i-\frac{1}{2},j,k)}^{n+\theta}\right|\right)(S_W^-)^{n+\theta}
\end{aligned}$$

3. considerando a notação $(S_{(j+\frac{1}{2})})^{n+\theta} \equiv (S_N)^{n+\theta}$, tem-se:

$$(S_N)^{n+\theta} = \begin{cases} C_{(i,j+\frac{1}{2},k)}^{n+\theta} > 0 \Rightarrow (S_N^+)^{n+\theta} = \theta \left[S_{(i,j,k)}^{n+1} + \frac{1}{2} \Psi(\lambda_{(i,j+\frac{1}{2},k)}^+)^{n+1} (S_{(i,j,k)}^{n+1} - S_{(i,j-1,k)}^{n+1}) \right] \\ \quad + (1-\theta) \left[S_{(i,j,k)}^n + \frac{1}{2} \Psi(\lambda_{(i,j+\frac{1}{2},k)}^+)^n (S_{(i,j,k)}^n - S_{(i,j-1,k)}^n) \right] \\ C_{(i,j+\frac{1}{2},k)}^{n+\theta} < 0 \Rightarrow (S_N^-)^{n+\theta} = \theta \left[S_{(i,j+1,k)}^{n+1} - \frac{1}{2} \Psi(\lambda_{(i,j+\frac{1}{2},k)}^-)^{n+1} (S_{(i,j+2,k)}^{n+1} - S_{(i,j+1,k)}^{n+1}) \right] \\ \quad + (1-\theta) \left[S_{(i,j+1,k)}^n - \frac{1}{2} \Psi(\lambda_{(i,j+\frac{1}{2},k)}^-)^n (S_{(i,j+2,k)}^n - S_{(i,j+1,k)}^n) \right] \end{cases}$$

onde as funções monitoras são definidas por:

$$\lambda_{(i,j+\frac{1}{2},k)}^+ = \frac{S_{(i,j+1,k)}^n - S_{(i,j,k)}^n + \varepsilon}{S_{(i,j,k)}^n - S_{(i,j-1,k)}^n + \varepsilon} \text{ e } \lambda_{(i,j+\frac{1}{2},k)}^- = \frac{S_{(i,j+1,k)}^n - S_{(i,j,k)}^n + \varepsilon}{S_{(i,j+2,k)}^n - S_{(i,j+1,k)}^n + \varepsilon}$$

portanto, tem-se:

$$\begin{aligned}
(S_N)^{n+\theta} &= \max\left(C_{(i,j+\frac{1}{2},k)}^{n+\theta}, 0\right)(S_N^+)^{n+\theta} + \min\left(C_{(i,j+\frac{1}{2},k)}^{n+\theta}, 0\right)(S_N^-)^{n+\theta} \\
&= \frac{1}{2}\left(C_{(i,j+\frac{1}{2},k)}^{n+\theta} + \left|C_{(i,j+\frac{1}{2},k)}^{n+\theta}\right|\right)(S_N^+)^{n+\theta} + \frac{1}{2}\left(C_{(i,j+\frac{1}{2},k)}^{n+\theta} - \left|C_{(i,j+\frac{1}{2},k)}^{n+\theta}\right|\right)(S_N^-)^{n+\theta}
\end{aligned}$$

4. e, considerando a notação $(S_{(j-\frac{1}{2})})^{n+\theta} \equiv (S_S)^{n+\theta}$, tem-se:

$$(S_S)^{n+\theta} = \begin{cases} C_{(i,j-\frac{1}{2},k)}^{n+\theta} > 0 \Rightarrow (S_S^+)^{n+\theta} = \theta \left[S_{(i,j-1,k)}^{n+1} + \frac{1}{2} \Psi \left(\lambda_{(i,j-\frac{1}{2},k)}^+ \right)^{n+1} \left(S_{(i,j-1,k)}^{n+1} - S_{(i,j-2,k)}^{n+1} \right) \right] \\ \quad + (1-\theta) \left[S_{(i,j-1,k)}^n + \frac{1}{2} \Psi \left(\lambda_{(i,j-\frac{1}{2},k)}^+ \right)^n \left(S_{(i,j-1,k)}^n - S_{(i,j-2,k)}^n \right) \right] \\ C_{(i,j-\frac{1}{2},k)}^{n+\theta} < 0 \Rightarrow (S_S^-)^{n+\theta} = \theta \left[S_{(i,j,k)}^{n+1} - \frac{1}{2} \Psi \left(\lambda_{(i,j-\frac{1}{2},k)}^- \right)^{n+1} \left(S_{(i,j+1,k)}^{n+1} - S_{(i,j,k)}^{n+1} \right) \right] \\ \quad + (1-\theta) \left[S_{(i,j,k)}^n - \frac{1}{2} \Psi \left(\lambda_{(i,j-\frac{1}{2},k)}^- \right)^n \left(S_{(i,j+1,k)}^n - S_{(i,j,k)}^n \right) \right] \end{cases}$$

onde as funções monitoras são definidas por:

$$\lambda_{(i,j-\frac{1}{2},k)}^+ = \frac{S_{(i,j,k)}^n - S_{(i,j-1,k)}^n + \varepsilon}{S_{(i,j-1,k)}^n - S_{(i,j-2,k)}^n + \varepsilon} \quad \text{e} \quad \lambda_{(i,j-\frac{1}{2},k)}^- = \frac{S_{(i,j,k)}^n - S_{(i,j-1,k)}^n + \varepsilon}{S_{(i,j+1,k)}^n - S_{(i,j,k)}^n + \varepsilon}$$

e portanto:

$$\begin{aligned} (S_S)^{n+\theta} &= \max \left(C_{(i,j-\frac{1}{2},k)}^{n+\theta}, 0 \right) (S_S^+)^{n+\theta} + \min \left(C_{(i,j-\frac{1}{2},k)}^{n+\theta}, 0 \right) (S_S^-)^{n+\theta} \\ &= \frac{1}{2} \left(C_{(i,j-\frac{1}{2},k)}^{n+\theta} + \left| C_{(i,j-\frac{1}{2},k)}^{n+\theta} \right| \right) (S_S^+)^{n+\theta} + \frac{1}{2} \left(C_{(i,j-\frac{1}{2},k)}^{n+\theta} - \left| C_{(i,j-\frac{1}{2},k)}^{n+\theta} \right| \right) (S_S^-)^{n+\theta} \end{aligned}$$

Note-se que essas aproximações são formuladas a partir de expressões com dependência temporal em dois níveis de tempo, n e $n+1$ de modo que a acurácia temporal é de segunda ordem. Essa abordagem gera expressões para interpolação dos fluxos com uma parte implícita e outra parte explícita. Os termos implícitos são aqueles em θ . Os termos explícitos são aqueles em $1-\theta$. Porém, nessa abordagem gera-se sistemas de equações cujas matrizes dos coeficientes têm estrutura *tipo 9-diagonal*. Porém, objetivando manter a estrutura da matriz como 3-diagonal, onde as dependências implícitas estão apenas restritas a componente vertical, toma-se nessas expressões FLM trapezoidal (FLM-T) $\theta = 0,0$, de modo que apenas os índices k , $k-1$ e $k+1$ entram na geração da matriz, que é 3-diagonal. Conseqüentemente, as aproximações, específicas para a advecção horizontal, são como:

$$\begin{aligned} S_E &\equiv \begin{cases} S_E^+ = S_{(i,j,k)}^n + \frac{1}{2} \Psi \left(r_{(i+\frac{1}{2},j,k)}^+ \right) \left(S_{(i,j,k)}^n - S_{(i-1,j,k)}^n \right) & , se u_{(i+\frac{1}{2},j,k)}^{n+\theta} > 0 \\ S_E^- = S_{(i+1,j,k)}^n - \frac{1}{2} \Psi \left(r_{(i+\frac{1}{2},j,k)}^- \right) \left(S_{(i+2,j,k)}^n - S_{(i+1,j,k)}^n \right) & , se u_{(i+\frac{1}{2},j,k)}^{n+\theta} < 0 \end{cases} \\ S_W &\equiv \begin{cases} S_W^+ = S_{(i-1,j,k)}^n + \frac{1}{2} \Psi \left(r_{(i-\frac{1}{2},j,k)}^+ \right) \left(S_{(i-1,j,k)}^n - S_{(i-2,j,k)}^n \right) & , se u_{(i-\frac{1}{2},j,k)}^{n+\theta} > 0 \\ S_W^- = S_{(i,j,k)}^n - \frac{1}{2} \Psi \left(r_{(i-\frac{1}{2},j,k)}^- \right) \left(S_{(i+1,j,k)}^n - S_{(i,j,k)}^n \right) & , se u_{(i-\frac{1}{2},j,k)}^{n+\theta} < 0 \end{cases} \\ S_N &\equiv \begin{cases} S_N^+ = S_{(i,j,k)}^n + \frac{1}{2} \Psi \left(\lambda_{(i,j+\frac{1}{2},k)}^+ \right) \left(S_{(i,j,k)}^n - S_{(i,j-1,k)}^n \right) & , se v_{(i,j+\frac{1}{2},k)}^{n+\theta} > 0 \\ S_N^- = S_{(i,j+1,k)}^n - \frac{1}{2} \Psi \left(\lambda_{(i,j+\frac{1}{2},k)}^- \right) \left(S_{(i,j+2,k)}^n - S_{(i,j+1,k)}^n \right) & , se v_{(i,j+\frac{1}{2},k)}^{n+\theta} < 0 \end{cases} \end{aligned}$$

$$S_S \equiv \begin{cases} S_S^+ = S_{(i,j-1,k)}^n + \frac{1}{2} \Psi \left(\lambda_{(i,j-\frac{1}{2},k)}^+ \right) \left(S_{(i,j-1,k)}^n - S_{(i,j-2,k)}^n \right), & \text{se } v_{(i,j-\frac{1}{2},k)}^{\overline{n+\theta}} > 0 \\ S_S^- = S_{(i,j,k)}^n - \frac{1}{2} \Psi \left(\lambda_{(i,j-\frac{1}{2},k)}^- \right) \left(S_{(i,j+1,k)}^n - S_{(i,j,k)}^n \right), & \text{se } v_{(i,j-\frac{1}{2},k)}^{\overline{n+\theta}} < 0 \end{cases}$$

Para escrever essas funções aproximantes, algoritmicamente concisa e conveniente de modo a englobar velocidades positivas ou velocidades negativas, definem-se expressões em termos dos números de Courant como:

$$\begin{aligned} (C_{ry}^+)^{n+\theta}_{(i,j\pm\frac{1}{2},k)} &\equiv \left(C_{(i,j\pm\frac{1}{2},k)}^{n+\theta} + \left| C_{(i,j\pm\frac{1}{2},k)}^{n+\theta} \right| \right), & (C_{ry}^-)^{n+\theta}_{(i,j\pm\frac{1}{2},k)} &\equiv \left(C_{(i,j\pm\frac{1}{2},k)}^{n+\theta} - \left| C_{(i,j\pm\frac{1}{2},k)}^{n+\theta} \right| \right) \\ (C_{rx}^+)^{n+\theta}_{(i\pm\frac{1}{2},j,k)} &\equiv \left(C_{(i\pm\frac{1}{2},j,k)}^{n+\theta} + \left| C_{(i\pm\frac{1}{2},j,k)}^{n+\theta} \right| \right) & \text{e } (C_{rx}^-)^{n+\theta}_{(i\pm\frac{1}{2},j,k)} &\equiv \left(C_{(i\pm\frac{1}{2},j,k)}^{n+\theta} - \left| C_{(i\pm\frac{1}{2},j,k)}^{n+\theta} \right| \right) \end{aligned}$$

e, assim, os fluxos numéricos nas faces para a componente horizontal da célula computacional 3D (veja a fig. A1.2) são dados por (6.8):

$$\left. \begin{aligned} S_{(i+\frac{1}{2},j,k)}^n &\equiv S_E = \frac{1}{2} (C_{rx}^+)^{n+\theta}_{(i+\frac{1}{2},j,k)} S_E^+ + \frac{1}{2} (C_{rx}^-)^{n+\theta}_{(i+\frac{1}{2},j,k)} S_E^- \\ S_{(i-\frac{1}{2},j,k)}^n &\equiv S_W = \frac{1}{2} (C_{rx}^+)^{n+\theta}_{(i-\frac{1}{2},j,k)} S_W^+ + \frac{1}{2} (C_{rx}^-)^{n+\theta}_{(i-\frac{1}{2},j,k)} S_W^- \\ S_{(i,j+\frac{1}{2},k)}^n &\equiv S_N = \frac{1}{2} (C_{ry}^+)^{n+\theta}_{(i,j+\frac{1}{2},k)} S_N^+ + \frac{1}{2} (C_{ry}^-)^{n+\theta}_{(i,j+\frac{1}{2},k)} S_N^- \\ S_{(i,j-\frac{1}{2},k)}^n &\equiv S_S = \frac{1}{2} (C_{ry}^+)^{n+\theta}_{(i,j-\frac{1}{2},k)} S_S^+ + \frac{1}{2} (C_{ry}^-)^{n+\theta}_{(i,j-\frac{1}{2},k)} S_S^- \end{aligned} \right\} (6.8)$$

ou seja, nas faces E, W, N, S, L e U os fluxos numéricos são escritos na forma conservativa, respectivamente, como:

$$\begin{aligned} S_{(i+\frac{1}{2},j,k)}^n &\equiv u_{(i+\frac{1}{2},j,k)}^{\text{not } n} S_E, & S_{(i-\frac{1}{2},j,k)}^n &\equiv u_{(i-\frac{1}{2},j,k)}^{\text{not } n} S_W, & S_{(i,j+\frac{1}{2},k)}^n &\equiv v_{(i,j+\frac{1}{2},k)}^{\text{not } n} S_N, \\ S_{(i,j-\frac{1}{2},k)}^n &\equiv v_{(i,j-\frac{1}{2},k)}^{\text{not } n} S_S, & S_{(i,j,k-\frac{1}{2})}^n &\equiv w_{(i,j,k-\frac{1}{2})}^{\text{not } n} S_L & \text{e } S_{(i,j,k+\frac{1}{2})}^n &\equiv w_{(i,j,k+\frac{1}{2})}^{\text{not } n} S_U. \end{aligned}$$

São essas as expressões que efetivamente são usadas para interpolar a concentração das substâncias nas faces de uma célula computacional 3D, e são posicionadas conforme ilustrado pela fig. A1.2.

6.5 Esquema numérico semi-implícito: método beta

Os fluxos numéricos nas faces na direção vertical da célula computacional são calculados empregando o método $-\beta$ descrito na seção 5.1. Analogamente, para obter uma versão algorítmica concisa e conveniente, que considera velocidades positivas e velocidades negativas, definem-se expressões em termos dos números de Courant locais, como:

$$(C_{rz}^+)^{n+\theta}_{(i,j,k\pm\frac{1}{2})} \equiv \left(C_{(i,j,k\pm\frac{1}{2})}^{n+\theta} + \left| C_{(i,j,k\pm\frac{1}{2})}^{n+\theta} \right| \right) \text{ e } (C_{rz}^-)^{n+\theta}_{(i,j,k\pm\frac{1}{2})} \equiv \left(C_{(i,j,k\pm\frac{1}{2})}^{n+\theta} - \left| C_{(i,j,k\pm\frac{1}{2})}^{n+\theta} \right| \right)$$

A expressão para o fluxo da face superior pode ser escrita, considerando o caso 3D e a possibilidade de velocidades positivas e negativas, respectivamente, como [RIZ2002a]:

$$S_U^+ = \beta_{(i,j,k+\frac{1}{2})}^n \left(\theta S_{(i,j,k)}^{n+1} + (1-\theta) S_{(i,j,k)}^n \right) - \left(1 - \beta_{(i,j,k+\frac{1}{2})}^n \right) \frac{1}{8} \left(S_{(i,j,k-1)}^n - 2S_{(i,j,k)}^n + S_{(i,j,k+1)}^n \right)$$

e

$$S_U^- = \beta_{(i,j,k+\frac{1}{2})}^n \left(\theta S_{(i,j,k+1)}^{n+1} + (1-\theta) S_{(i,j,k+1)}^n \right) - \left(1 - \beta_{(i,j,k+\frac{1}{2})}^n \right) \frac{1}{8} \left(S_{(i,j,k)}^n - 2S_{(i,j,k+1)}^n + S_{(i,j,k+2)}^n \right)$$

tem-se, portanto, que o fluxo numérico para velocidades arbitrárias na face superior é interpolado como:

$$S_{(i,j,k+\frac{1}{2})}^n \equiv S_U = \frac{1}{2} \left(C_{rz}^+ \right)_{(i,j,k+\frac{1}{2})}^{n+\theta} S_U^+ + \frac{1}{2} \left(C_{rz}^- \right)_{(i,j,k+\frac{1}{2})}^{n+\theta} S_U^- \\ + C_{(i,j,k+\frac{1}{2})}^{n+\theta} \left(1 - \beta_{(i,j,k+\frac{1}{2})}^{n+\theta} \right) \left[\frac{\theta}{2} \left(S_{(i,j,k)}^{n+1} + S_{(i,j,k+1)}^{n+1} \right) + \frac{1-\theta}{2} \left(S_{(i,j,k)}^n + S_{(i,j,k+1)}^n \right) \right]$$

O caso para a expressão para o fluxo da face inferior é análogo e pode ser escrito para velocidades positivas e negativas, considerando as expressões:

$$S_L^+ = \beta_{(i,j,k-\frac{1}{2})}^n \left(\theta S_{(i,j,k-1)}^{n+1} + (1-\theta) S_{(i,j,k-1)}^n \right) - \left(1 - \beta_{(i,j,k-\frac{1}{2})}^n \right) \frac{1}{8} \left(S_{(i,j,k-2)}^n - 2S_{(i,j,k-1)}^n + S_{(i,j,k)}^n \right)$$

e

$$S_L^- = \beta_{(i,j,k-\frac{1}{2})}^n \left(\theta S_{(i,j,k)}^{n+1} + (1-\theta) S_{(i,j,k)}^n \right) - \left(1 - \beta_{(i,j,k-\frac{1}{2})}^n \right) \frac{1}{8} \left(S_{(i,j,k-1)}^n - 2S_{(i,j,k)}^n + S_{(i,j,k+1)}^n \right)$$

tem-se, portanto, que:

$$S_{(i,j,k-\frac{1}{2})}^n \equiv S_L = \frac{1}{2} \left(C_{rz}^+ \right)_{(i,j,k-\frac{1}{2})}^{n+\theta} S_L^+ + \frac{1}{2} \left(C_{rz}^- \right)_{(i,j,k-\frac{1}{2})}^{n+\theta} S_L^- \\ + C_{(i,j,k-\frac{1}{2})}^{n+\theta} \left(1 - \beta_{(i,j,k-\frac{1}{2})}^{n+\theta} \right) \left[\frac{\theta}{2} \left(S_{(i,j,k-1)}^{n+1} + S_{(i,j,k)}^{n+1} \right) + \frac{1-\theta}{2} \left(S_{(i,j,k-1)}^n + S_{(i,j,k)}^n \right) \right]$$

Ao contrário dos esquemas completamente explícitos construídos para a advecção horizontal (seção 5.1.3 ou seção 6.4), o esquema para a advecção vertical é, necessariamente, semi-implícito por construção.

Considerando as definições de $w_{(i,j,k\pm\frac{1}{2})}^{n+\theta}$, $u_{(i\pm\frac{1}{2},j,k)}^{n+\theta}$ e de $v_{(i,j\pm\frac{1}{2},k)}^{n+\theta}$ e dos números de Courant locais para as velocidades horizontais, dados na seção 6.4, e os números de Courant locais para a componente vertical, que para a face superior e inferior são, respectivamente, como:

$$C_{(i,j,k+\frac{1}{2})}^{n+\theta} = \frac{\Delta t w_{(i,j,k+\frac{1}{2})}^{n+\theta}}{\Delta z_{(i,j,k+\frac{1}{2})}^n} \text{ e } C_{(i,j,k-\frac{1}{2})}^{n+\theta} = \frac{\Delta t w_{(i,j,k-\frac{1}{2})}^{n+\theta}}{\Delta z_{(i,j,k-\frac{1}{2})}^n}$$

e considerando a definição dos números de Péclet locais, tem-se que:

$$\frac{C_{(i\pm\frac{1}{2},j,k)}^{n+\theta}}{P_{e(i\pm\frac{1}{2},j,k)}^{n+\theta}} = \frac{\varepsilon_h \Delta t}{(\Delta x)^2}, \quad \frac{C_{(i,j\pm\frac{1}{2},k)}^{n+\theta}}{P_{e(i,j\pm\frac{1}{2},k)}^{n+\theta}} = \frac{\varepsilon_h \Delta t}{(\Delta y)^2}, \quad \frac{C_{(i,j,k\pm\frac{1}{2})}^{n+\theta}}{P_{e(i,j,k\pm\frac{1}{2})}^{n+\theta}} = \frac{\varepsilon_v(x,y,z,t) \Delta t}{\left(\Delta z_{(i,j,k\pm\frac{1}{2})}^n \right)^2}$$

pois $\varepsilon_h = cte$ e, portanto, $P_{e(i\pm\frac{1}{2},j\pm\frac{1}{2},k)}^{n+\theta} \equiv P_e = cte$

Assim, em decorrência dessas definições e das aproximações estabelecidas, nas quais o transporte vertical (advecção e difusão) é discretizado implicitamente, e o transporte horizontal, os termos de decaimento e de fonte são discretizados explicitamente, constrói-se um esquema numérico semi-implícito gerando sistemas de equações lineares 3-diagonal para a ETM 3D. Esses sistemas de equações são escritos, tomando as variáveis dependentes no método $-\theta$ em três níveis de tempo de modo que se tem $\varphi^{n+\theta} = \theta\varphi^{n+1} + (1-\theta)\varphi^n$ e $\varphi^{n+\theta} = \theta\varphi^n + (1-\theta)\varphi^{n-1}$ com $0 < \theta \leq 1$, como:

$$E_{(k-1)} S_{(i,j,k-1)}^{n+1} + C_{(k)} S_{(i,j,k)}^{n+1} + D_{(k+1)} S_{(i,j,k+1)}^{n+1} = B_{(k)}^n \} \quad (6.9)$$

em que os coeficientes em (6.9) são tais que:

$$\begin{aligned}
E_{(k-1)} &= \frac{\theta}{2} \Delta z_{(i,j,k-1)}^{n+1} \left[-\beta_{(i,j,k-\frac{1}{2})}^{n+\theta} (C_{rz}^-)_{(i,j,k-\frac{1}{2})}^{n+\theta} - \left(1 - \beta_{(i,j,k-\frac{1}{2})}^{n+\theta}\right) C_{(i,j,k-\frac{1}{2})}^{n+\theta} \right] - \frac{C_{(i,j,k-\frac{1}{2})}^{n+\theta}}{2P_{e(i,j,k-\frac{1}{2})}^n} \\
D_{(k+1)} &= \frac{\theta}{2} \Delta z_{(i,j,k)}^{n+1} \left[\beta_{(i,j,k+\frac{1}{2})}^{n+\theta} (C_{rz}^-)_{(i,j,k+\frac{1}{2})}^{n+\theta} + \left(1 - \beta_{(i,j,k+\frac{1}{2})}^{n+\theta}\right) C_{(i,j,k+\frac{1}{2})}^{n+\theta} \right] - \frac{C_{(i,j,k+\frac{1}{2})}^{n+\theta}}{2P_{e(i,j,k+\frac{1}{2})}^n} \\
C_{(k)} &= \Delta z_{(i,j,k)}^{n+1} + \frac{\theta}{2} \Delta z_{(i,j,k)}^{n+1} \left[\beta_{(i,j,k+\frac{1}{2})}^{n+\theta} (C_{rz}^+)_{(i,j,k+\frac{1}{2})}^{n+\theta} + \left(1 - \beta_{(i,j,k+\frac{1}{2})}^{n+\theta}\right) C_{(i,j,k+\frac{1}{2})}^{n+\theta} \right] \\
&\quad - \frac{\theta}{2} \Delta z_{(i,j,k-1)}^{n+1} \left[\beta_{(i,j,k-\frac{1}{2})}^{n+\theta} (C_{rz}^-)_{(i,j,k-\frac{1}{2})}^{n+\theta} + \left(1 - \beta_{(i,j,k-\frac{1}{2})}^{n+\theta}\right) C_{(i,j,k-\frac{1}{2})}^{n+\theta} \right] \\
&\quad + \left[\frac{C_{(i,j,k-\frac{1}{2})}^{n+\theta}}{2P_{e(i,j,k-\frac{1}{2})}^n} + \frac{C_{(i,j,k+\frac{1}{2})}^{n+\theta}}{2P_{e(i,j,k+\frac{1}{2})}^n} \right]
\end{aligned}$$

e o vetor dos termos independentes é como:

$$\begin{aligned}
B_{(k)}^n &= S_{(i,j,k)}^n \Delta z_{(i,j,k)}^n \\
&\quad - \frac{(1-\theta)}{2} \Delta z_{(i,j,k)}^n \left[\beta_{(i,j,k+\frac{1}{2})}^{\overline{n+\theta}} (C_{rz}^+)_{(i,j,k+\frac{1}{2})}^{\overline{n+\theta}} S_{(i,j,k)}^n + \beta_{(i,j,k+\frac{1}{2})}^{\overline{n+\theta}} (C_{rz}^-)_{(i,j,k+\frac{1}{2})}^{\overline{n+\theta}} S_{(i,j,k+1)}^n \right] \\
&\quad + \frac{(1-\theta)}{2} \Delta z_{(i,j,k-1)}^n \left[\beta_{(i,j,k-\frac{1}{2})}^{\overline{n+\theta}} (C_{rz}^+)_{(i,j,k-\frac{1}{2})}^{\overline{n+\theta}} S_{(i,j,k-1)}^n + \beta_{(i,j,k-\frac{1}{2})}^{\overline{n+\theta}} (C_{rz}^-)_{(i,j,k-\frac{1}{2})}^{\overline{n+\theta}} S_{(i,j,k)}^n \right] \\
&\quad - \frac{(1-\theta)}{2} \Delta z_{(i,j,k)}^n \left[\left(1 - \beta_{(i,j,k+\frac{1}{2})}^{\overline{n+\theta}}\right) C_{(i,j,k+\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,k+1)}^n + S_{(i,j,k)}^n\right) \right] \\
&\quad + \frac{(1-\theta)}{2} \Delta z_{(i,j,k-1)}^n \left[\left(1 - \beta_{(i,j,k-\frac{1}{2})}^{\overline{n+\theta}}\right) C_{(i,j,k-\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,k)}^n + S_{(i,j,k-1)}^n\right) \right] \\
&\quad + \frac{\Delta z_{(i,j,k)}^n}{16} \left(1 - \beta_{(i,j,k+\frac{1}{2})}^{\overline{n+\theta}}\right) \left[\left(C_{rz}^+\right)_{(i,j,k+\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,k-1)}^n + S_{(i,j,k+1)}^n - 2S_{(i,j,k)}^n\right) \right. \\
&\quad \quad \left. + \left(C_{rz}^-\right)_{(i,j,k+\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,k)}^n + S_{(i,j,k+2)}^n - 2S_{(i,j,k+1)}^n\right) \right] \\
&\quad - \frac{\Delta z_{(i,j,k-1)}^n}{16} \left(1 - \beta_{(i,j,k-\frac{1}{2})}^{\overline{n+\theta}}\right) \left[\left(C_{rz}^+\right)_{(i,j,k-\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,k-2)}^n + S_{(i,j,k)}^n - 2S_{(i,j,k-1)}^n\right) \right. \\
&\quad \quad \left. + \left(C_{rz}^-\right)_{(i,j,k-\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,k-1)}^n + S_{(i,j,k+1)}^n - 2S_{(i,j,k)}^n\right) \right] \\
&\quad - \frac{1}{2} \left(C_{rx}^+\right)_{(i+\frac{1}{2},j,k)}^{\overline{n+\theta}} S_E^+ \Delta z_{(i+\frac{1}{2},j,k)}^n + \frac{1}{2} \left(C_{rx}^-\right)_{(i+\frac{1}{2},j,k)}^{\overline{n+\theta}} S_E^- \Delta z_{(i+\frac{1}{2},j,k)}^n \\
&\quad + \frac{1}{2} \left(C_{rx}^+\right)_{(i-\frac{1}{2},j,k)}^{\overline{n+\theta}} S_W^+ \Delta z_{(i-\frac{1}{2},j,k)}^n + \frac{1}{2} \left(C_{rx}^-\right)_{(i-\frac{1}{2},j,k)}^{\overline{n+\theta}} S_W^- \Delta z_{(i-\frac{1}{2},j,k)}^n \\
&\quad - \frac{1}{2} \left(C_{ry}^+\right)_{(i,j+\frac{1}{2},k)}^{\overline{n+\theta}} S_N^+ \Delta z_{(i,j+\frac{1}{2},k)}^n + \frac{1}{2} \left(C_{ry}^-\right)_{(i,j+\frac{1}{2},k)}^{\overline{n+\theta}} S_N^- \Delta z_{(i,j+\frac{1}{2},k)}^n \\
&\quad + \frac{1}{2} \left(C_{ry}^+\right)_{(i,j-\frac{1}{2},k)}^{\overline{n+\theta}} S_S^+ \Delta z_{(i,j-\frac{1}{2},k)}^n + \frac{1}{2} \left(C_{ry}^-\right)_{(i,j-\frac{1}{2},k)}^{\overline{n+\theta}} S_S^- \Delta z_{(i,j-\frac{1}{2},k)}^n
\end{aligned}$$

$$\begin{aligned}
& + \left[\frac{C_{r(i,j,k+\frac{1}{2})}^{\bar{n}+\theta}}{2P_{e(i,j,k+\frac{1}{2})}^n} (S_{(i,j,k+1)}^n - S_{(i,j,k)}^n) - \frac{C_{r(i,j,k-\frac{1}{2})}^{\bar{n}+\theta}}{2P_{e(i,j,k-\frac{1}{2})}^n} (S_{(i,j,k)}^n - S_{(i,j,k-1)}^n) \right] \\
& + \left[\frac{C_{r(i+\frac{1}{2},j,k)}^{\bar{n}+\theta}}{2P_{e(i+\frac{1}{2},j,k)}^n} (S_{(i+1,j,k)}^n - 2S_{(i,j,k)}^n + S_{(i-1,j,k)}^n) \right] \Delta z_{(i,j,k)}^n \\
& + \left[\frac{C_{r(i,j+\frac{1}{2},k)}^{\bar{n}+\theta}}{2P_{e(i,j+\frac{1}{2},k)}^n} (S_{(i,j+1,k)}^n - 2S_{(i,j,k)}^n + S_{(i,j-1,k)}^n) \right] \Delta z_{(i,j,k)}^n \\
& + \Delta t S_{(i,j,k)}^n \Delta z_{(i,j,k)}^n K + \Delta t \Delta z_{(i,j,k)}^n F_{fonte}^n
\end{aligned}$$

em que os espaçamentos para a célula 3D são expressos na forma discreta como:

$$\Delta z_{(i,j,k)}^n = 0,5 \left(\Delta z_{(i+\frac{1}{2},j,k)}^n + \Delta z_{(i-\frac{1}{2},j,k)}^n \right) \text{ e } \Delta z_{(i,j,k)}^n = 0,5 \left(\Delta z_{(i,j+\frac{1}{2},k)}^n + \Delta z_{(i,j-\frac{1}{2},k)}^n \right)$$

e são atualizados como:

$$\Delta z_{(M+\frac{1}{2},j,k)}^n = \Delta z_{(M+\frac{1}{2},j,k)}^n + \eta_{\left(i+\frac{I+1}{2},j\right)}^n \text{ e } \Delta z_{(i,M+\frac{1}{2},k)}^n = \Delta z_{(i,M+\frac{1}{2},k)}^n + \eta_{\left(i,\frac{J+1}{2}+j\right)}^n$$

em que $I = -\text{sign}\left(U_{(M+\frac{1}{2},j,k)}^n\right)$ $J = -\text{sign}\left(V_{(i,M+\frac{1}{2},k)}^n\right)$. Nas funções interpolantes para (6.9) toma-se $\theta = 0,5$ e β variando, como discutido no anexo 1.

6.5.1 Condições de contorno: extrapolação

Para implementar as condições de contorno para o transporte horizontal, considerou-se a dependência de dados do estêncil de 9-pontos do método FLM conforme as fig. 6.8(a) e 6.8(b), além do fato como que não há fluxo atravessando contornos sólidos. Além disso, especificaram-se condições de Dirichlet e de Neumann, sendo a primeira como CC *inflow*, e a segunda como CC *outflow*.

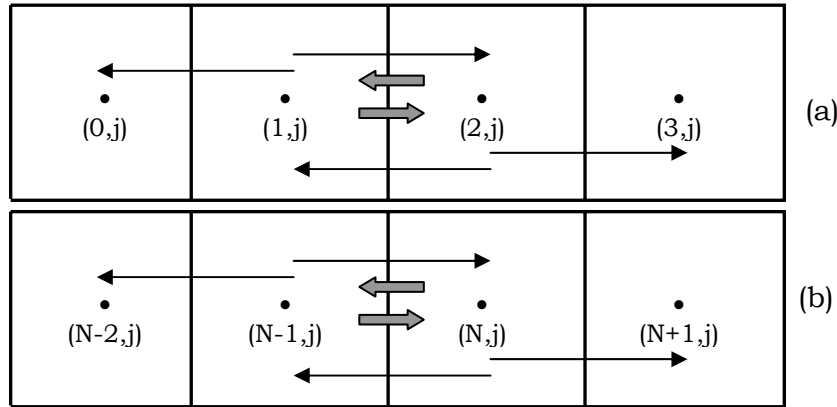


FIGURA 6.8 - Representação para molécula computacional para o transporte horizontal, para as CC esquerda (a) e para a CC direita (b)

As condições de contorno para o transporte vertical foram obtidas a partir do fato de que não há fluxo escalar na superfície livre e no leito (interface sedimento-água), e considerando a malha vertical (coluna vertical) dada na fig. 6.9.

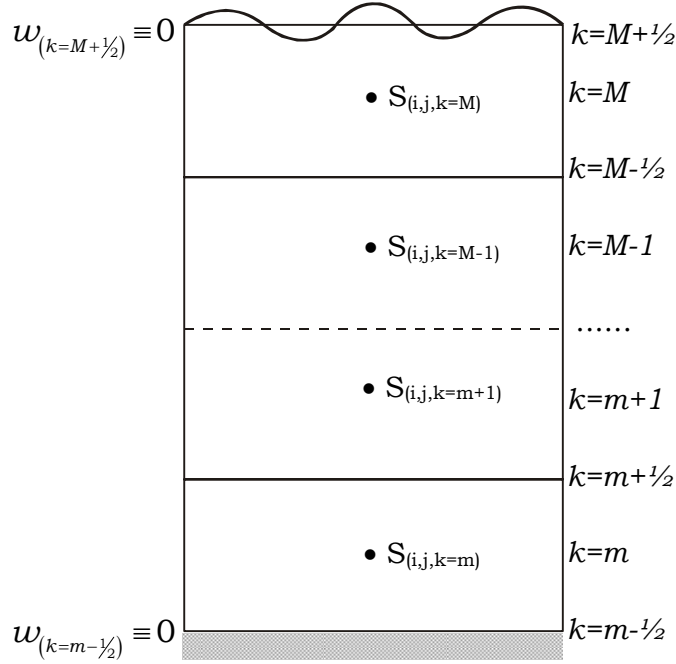


FIGURA 6.9 - Exemplo para uma coluna vertical representando as CC tipo não escoamento no topo e no fundo de uma coluna de água

Note-se que as CC podem ser implementadas através de extrapolação, como feito por [HUN95], ou através da supressão dos termos, como feito por [BLO94], [GRO98] e [HEI99]. Para o transporte vertical, empregou-se a supressão dos termos, e para o transporte horizontal, empregou-se a extrapolação. Isso porque, no transporte horizontal, há uma grande variedade de funções limitadoras de fluxo (seção 5.1.4) e a abordagem extrapolação é mais geral já que não depende de uma particular função, mas apenas dos pontos que não pertencem ao domínio computacional. Assim sendo, tem-se os seguintes casos:

1. transporte vertical: advecção

As condições de contorno para os fluxos verticais $(S_U) \equiv S_{U(i,j,k+1/2)}^n$ e $(S_L) \equiv S_{L(i,j,k-1/2)}^n$ são implementadas notando que esses fluxos estão definidos em $m+1 \leq k \leq M$ em S_L , e em $m \leq k \leq M-1$ em S_U . Assim:

1.1. não há fluxo no topo e no fundo da coluna de água. Logo para $k=M$ e para $k=m$, os fluxos $(S_U) \equiv S_{U(i,j,M+1/2)}^n$ e $(S_L) \equiv S_{L(i,j,m-1/2)}^n$ são, respectivamente, dados por:

$$w_{(i,j,M+1/2)}^{n+\theta} S_{U(i,j,M+1/2)} \equiv 0, \quad w_{(i,j,m-1/2)}^{n+\theta} S_{L(i,j,m-1/2)} \equiv 0$$

1.2. considerando as funções interpolantes, os fluxos devem ser tratados dependendo das velocidades, das faces *superior* do topo e do fundo.

1.2.1. para o fluxo $(S_U) \equiv (S_U)_{(i,j,m+1/2)}^n$, em $k=m$, quando $w_{(i,j,m+1/2)}^{n+\theta} > 0$, tem-se $(S_U^+)_{(i,j,m+1/2)}^n \approx \beta^{n+\theta} \left(\theta S_{(i,j,m)}^{n+1} + (1-\theta) S_{(i,j,m)}^n \right)$, pois o termo de correção $\frac{1}{8} (1-\beta) \left(S_{(i,j,m-1)}^n - 2S_{(i,j,m)}^n + S_{(i,j,m+1)}^n \right)$ inclui $S_{(i,j,m-1)}^n \notin CV|_{k=m}$, onde

CV é o conjunto das células válidas. Deve-se, portanto, retirar esse termo do vetor

$$B_{(k)}^n \cdot (S_U^-)_{(i,j,m+\frac{1}{2})}^n = 0, \text{ pois } (C_{rz}^-)_{(i,j,m+\frac{1}{2})}^{n+\theta} \equiv \left(C_{(i,j,m+\frac{1}{2})}^{n+\theta} - \left| C_{(i,j,m+\frac{1}{2})}^{n+\theta} \right| \right) = 0$$

e quando $w_{(i,j,m+\frac{1}{2})}^{n+\theta} < 0$, tem-se que:

$$(S_U^+)_{(i,j,m+\frac{1}{2})}^n = 0, \text{ pois } (C_{rz}^+)_{(i,j,m+\frac{1}{2})}^{n+\theta} \equiv \left(C_{(i,j,m+\frac{1}{2})}^{n+\theta} + \left| C_{(i,j,m+\frac{1}{2})}^{n+\theta} \right| \right) = 0$$

e onde o termo

$$\begin{aligned} (S_U^-)_{(i,j,m+\frac{1}{2})}^n &= \beta_{(i,j,m+\frac{1}{2})}^n \left(\theta S_{(i,j,m+1)}^{n+1} + (1-\theta) S_{(i,j,m+1)}^n \right) \\ &\quad - \left(1 - \beta_{(i,j,m+\frac{1}{2})}^{n+\theta} \right) \frac{1}{8} \left(S_{(i,j,m)}^n - 2S_{(i,j,m+1)}^n + S_{(i,j,m+2)}^n \right) \end{aligned}$$

pois está completamente definido. Similarmente, o termo

$$C_{(i,j,m+\frac{1}{2})}^{n+\theta} \left(1 - \beta_{(i,j,m+\frac{1}{2})}^{n+\theta} \right) \left[\frac{\theta}{2} \left(S_{(i,j,m)}^{n+1} + S_{(i,j,m+1)}^{n+1} \right) + \frac{1-\theta}{2} \left(S_{(i,j,m)}^n + S_{(i,j,m+1)}^n \right) \right]$$

está definido para qualquer velocidade ($w_{(i,j,m+\frac{1}{2})}^{n+\theta} < 0$ e $w_{(i,j,m+\frac{1}{2})}^{n+\theta} > 0$)

1.2.2. para o fluxo $(S_U) \equiv (S_U)_{(i,j,M-\frac{1}{2})}^n$, em $k = M - 1$, é tal que:

1.2.2.1 para $w_{(i,j,M-\frac{1}{2})}^{n+\theta} < 0$ tal que:

$$(S_U^-)_{(i,j,M-\frac{1}{2})}^n \approx \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(\theta S_{(i,j,M)}^{n+1} + (1-\theta) S_{(i,j,M)}^n \right),$$

pois o termo de correção $\frac{1}{8}(1-\beta) \left(S_{(i,j,M-1)}^n - 2S_{(i,j,M)}^n + S_{(i,j,M+1)}^n \right)$ inclui

$S_{(i,j,M+1)}^n \notin CV|_{k=M-1}$. Deve-se, portanto, retirar esse termo do vetor $B_{(k)}^n$;

$$(S_U^+)_{(i,j,M-\frac{1}{2})}^n = 0, \text{ pois } (C_{rz}^+)_{(i,j,M-\frac{1}{2})}^{n+\theta} \equiv \left(C_{(i,j,M-\frac{1}{2})}^{n+\theta} + \left| C_{(i,j,M-\frac{1}{2})}^{n+\theta} \right| \right) = 0$$

1.2.2.2 para $w_{(i,j,M-\frac{1}{2})}^{n+\theta} > 0$ tal que:

$$(S_U^-)_{(i,j,M-\frac{1}{2})}^n = 0, \text{ pois } (C_{rz}^-)_{(i,j,M-\frac{1}{2})}^{n+\theta} \equiv \left(C_{(i,j,M-\frac{1}{2})}^{n+\theta} - \left| C_{(i,j,M-\frac{1}{2})}^{n+\theta} \right| \right) = 0$$

e onde o termo

$$\begin{aligned} (S_U^+)_{(i,j,M-\frac{1}{2})}^n &= \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(\theta S_{(i,j,M-1)}^{n+1} + (1-\theta) S_{(i,j,M-1)}^n \right) \\ &\quad - \left(1 - \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \right) \frac{1}{8} \left(S_{(i,j,M-2)}^n - 2S_{(i,j,M-1)}^n + S_{(i,j,M)}^n \right) \end{aligned}$$

pois está completamente definido.

1.2.2.3 o termo

$$C_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(1 - \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \right) \left[\frac{\theta}{2} \left(S_{(i,j,M-1)}^{n+1} + S_{(i,j,M)}^{n+1} \right) + \frac{1-\theta}{2} \left(S_{(i,j,M-1)}^n + S_{(i,j,M)}^n \right) \right]$$

está definido para qualquer velocidade ($w_{(i,j,M-\frac{1}{2})}^{n+\theta} < 0$ e $w_{(i,j,M-\frac{1}{2})}^{n+\theta} > 0$)

1.3. considerando as funções interpolantes, os fluxos são aproximados dependendo das velocidades, das faces *inferiores* do topo e do fundo.

1.3.1. para o fluxo $(S_L) \equiv (S_L)_{(i,j,m+\frac{1}{2})}^n$, em $k = m + 1$ é tal que:

1.3.1.1 para $w_{(i,j,m+\frac{1}{2})}^{n+\theta} > 0$ tal que:

$$\begin{aligned} (\mathbf{S}_L^-)_{(i,j,m+\frac{1}{2})}^n &= 0, \text{ pois } (\mathbf{C}_{rz}^+)_{(i,j,m+\frac{1}{2})}^{n+\theta} \equiv \left(\mathbf{C}_{(i,j,m+\frac{1}{2})}^{n+\theta} - \left| \mathbf{C}_{(i,j,m+\frac{1}{2})}^{n+\theta} \right| \right) = 0 \\ (\mathbf{S}_L^+)_{(i,j,m+\frac{1}{2})}^n &\approx \beta_{(i,j,m+\frac{1}{2})}^{n+\theta} \left(\theta \mathbf{S}_{(i,j,m)}^{n+1} + (1-\theta) \mathbf{S}_{(i,j,m)}^n \right), \end{aligned}$$

pois o termo de correção $\frac{1}{8}(1-\beta)(\mathbf{S}_{(i,j,m-2)}^n - 2\mathbf{S}_{(i,j,m-1)}^n + \mathbf{S}_{(i,j,m)}^n)$ inclui $\mathbf{S}_{(i,j,m-2)}^n \notin CV|_{k=m+1}$. Deve-se, portanto, retirar esse termo do vetor $B_{(k)}^n$;

1.3.1.2 para $w_{(i,j,m+\frac{1}{2})}^{n+\theta} < 0$ tal que:

$$(\mathbf{S}_L^+)_{(i,j,m+\frac{1}{2})}^n = 0, \text{ pois } (\mathbf{C}_{rz}^+)_{(i,j,m+\frac{1}{2})}^{n+\theta} \equiv \left(\mathbf{C}_{(i,j,m+\frac{1}{2})}^{n+\theta} + \left| \mathbf{C}_{(i,j,m+\frac{1}{2})}^{n+\theta} \right| \right) = 0$$

e onde o termo

$$\begin{aligned} (\mathbf{S}_L^-)_{(i,j,m+\frac{1}{2})}^n &= \beta_{(i,j,m+\frac{1}{2})}^{n+\theta} \left(\theta \mathbf{S}_{(i,j,m+1)}^{n+1} + (1-\theta) \mathbf{S}_{(i,j,m+1)}^n \right) \\ &\quad - \left(1 - \beta_{(i,j,m+\frac{1}{2})}^{n+\theta} \right) \frac{1}{8} \left(\mathbf{S}_{(i,j,m)}^n - 2\mathbf{S}_{(i,j,m+1)}^n + \mathbf{S}_{(i,j,m+2)}^n \right) \end{aligned}$$

pois está completamente definido.

1.3.1.3 o termo

$$\mathbf{C}_{(i,j,m+\frac{1}{2})}^{n+\theta} \left(1 - \beta_{(i,j,m+\frac{1}{2})}^{n+\theta} \right) \left[\frac{\theta}{2} \left(\mathbf{S}_{(i,j,m)}^{n+1} + \mathbf{S}_{(i,j,m+1)}^{n+1} \right) + \frac{1-\theta}{2} \left(\mathbf{S}_{(i,j,m)}^n + \mathbf{S}_{(i,j,m+1)}^n \right) \right]$$

está definido para qualquer velocidade ($w_{(i,j,m+\frac{1}{2})}^{n+\theta} < 0$ e $w_{(i,j,m+\frac{1}{2})}^{n+\theta} > 0$)

1.3.2. para o fluxo $(\mathbf{S}_L) \equiv (\mathbf{S}_L)_{(i,j,M-\frac{1}{2})}^n$, em $k = M$, é tal que:

1.3.2.1 para $w_{(i,j,M-\frac{1}{2})}^{n+\theta} < 0$ tal que:

$$\begin{aligned} (\mathbf{S}_L^+)_{(i,j,M-\frac{1}{2})}^n &= 0, \text{ pois } (\mathbf{C}_{rz}^+)_{(i,j,M-\frac{1}{2})}^{n+\theta} \equiv \left(\mathbf{C}_{(i,j,M-\frac{1}{2})}^{n+\theta} + \left| \mathbf{C}_{(i,j,M-\frac{1}{2})}^{n+\theta} \right| \right) = 0 \\ (\mathbf{S}_L^-)_{(i,j,M-\frac{1}{2})}^n &\approx \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(\theta \mathbf{S}_{(i,j,M)}^{n+1} + (1-\theta) \mathbf{S}_{(i,j,M)}^n \right), \end{aligned}$$

pois o termo de correção $\frac{1}{8}(1-\beta)(\mathbf{S}_{(i,j,M-1)}^n - 2\mathbf{S}_{(i,j,M)}^n + \mathbf{S}_{(i,j,M+1)}^n)$ inclui $\mathbf{S}_{(i,j,M+1)}^n \notin CV|_{k=M}$ e deve-se, portanto, retirar esse termo do vetor $B_{(k)}^n$;

1.3.2.2 para $w_{(i,j,M-\frac{1}{2})}^{n+\theta} > 0$ tal que:

$$(\mathbf{S}_L^-)_{(i,j,M-\frac{1}{2})}^n = 0, \text{ pois } (\mathbf{C}_{rz}^-)_{(i,j,M-\frac{1}{2})}^{n+\theta} \equiv \left(\mathbf{C}_{(i,j,M-\frac{1}{2})}^{n+\theta} - \left| \mathbf{C}_{(i,j,M-\frac{1}{2})}^{n+\theta} \right| \right) = 0$$

e onde o termo

$$\begin{aligned} (\mathbf{S}_L^+)_{(i,j,M-\frac{1}{2})}^n &= \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(\theta \mathbf{S}_{(i,j,M-1)}^{n+1} + (1-\theta) \mathbf{S}_{(i,j,M-1)}^n \right) \\ &\quad - \left(1 - \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \right) \frac{1}{8} \left(\mathbf{S}_{(i,j,M-2)}^n - 2\mathbf{S}_{(i,j,M-1)}^n + \mathbf{S}_{(i,j,M)}^n \right) \end{aligned}$$

pois está completamente definido.

1.3.2.3 o termo

$$C_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(1 - \beta_{(i,j,M-\frac{1}{2})}^{n+\theta}\right) \left[\frac{\theta}{2} \left(S_{(i,j,M-1)}^{n+1} + S_{(i,j,M)}^{n+1} \right) + \frac{1-\theta}{2} \left(S_{(i,j,M-1)}^n + S_{(i,j,M)}^n \right) \right]$$

está definido para qualquer velocidade ($w_{(i,j,M-\frac{1}{2})}^{n+\theta} < 0$ e $w_{(i,j,M-\frac{1}{2})}^{n+\theta} > 0$)

Assim, em resumo, considerando essas aproximações para a advecção vertical, deve-se alterar o sistema de equações (6.9), tal que em $B_{(k)}^n$ tem-se:

- a. em $k = M$ (S_U) $\equiv S_{U(i,j,M+\frac{1}{2})}^n = 0$ e em $k = m$ (S_L) $\equiv S_{L(i,j,m-\frac{1}{2})}^n = 0$;
b. para os fluxos (S_U) $\equiv (S_U)_{(i,j,m+\frac{1}{2})}^n$ e (S_L) $\equiv (S_L)_{(i,j,m+\frac{1}{2})}^n$, tem-se no caso de $w_{(i,j,m+\frac{1}{2})}^{n+\theta} > 0$, que :

$$(S_U^+)_{(i,j,m+\frac{1}{2})}^n \approx \beta_{(i,j,m+\frac{1}{2})}^{n+\theta} \left(\theta S_{(i,j,m)}^{n+1} + (1-\theta) S_{(i,j,m)}^n \right) \text{ e } (S_U^-)_{(i,j,m+\frac{1}{2})}^n = 0$$

$$(S_L^-)_{(i,j,m+\frac{1}{2})}^n = 0 \text{ e } (S_L^+)_{(i,j,m+\frac{1}{2})}^n \approx \beta_{(i,j,m+\frac{1}{2})}^{n+\theta} \left(\theta S_{(i,j,m)}^{n+1} + (1-\theta) S_{(i,j,m)}^n \right)$$

e portanto:

$$+ \frac{\Delta z_{(i,j,k)}^n}{16} \left(1 - \beta_{(i,j,k+\frac{1}{2})}^{n+\theta}\right) \left[(C_{rz}^+)_{(i,j,k+\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k-1)}^n + S_{(i,j,k+1)}^n - 2S_{(i,j,k)}^n \right) \right. \\ \left. + (C_{rz}^-)_{(i,j,k+\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k)}^n + S_{(i,j,k+2)}^n - 2S_{(i,j,k+1)}^n \right) \right] \approx 0$$

$$- \frac{\Delta z_{(i,j,k-1)}^n}{16} \left(1 - \beta_{(i,j,k-\frac{1}{2})}^{n+\theta}\right) \left[(C_{rz}^+)_{(i,j,k-\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k-2)}^n + S_{(i,j,k)}^n - 2S_{(i,j,k-1)}^n \right) \right. \\ \left. + (C_{rz}^-)_{(i,j,k-\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k-1)}^n + S_{(i,j,k+1)}^n - 2S_{(i,j,k)}^n \right) \right] \approx 0$$

significando que S_U cumpre essas condições para $k = m$ e $w_{(i,j,m+\frac{1}{2})}^{n+\theta} > 0$.

Analogamente, S_L cumpre essas condições para $k = m + 1$ e $w_{(i,j,m+\frac{1}{2})}^{n+\theta} > 0$.

- c. para (S_U) $\equiv (S_U)_{(i,j,m+\frac{1}{2})}^n$ e (S_L) $\equiv (S_L)_{(i,j,m+\frac{1}{2})}^n$ com $w_{(i,j,m+\frac{1}{2})}^{n+\theta} < 0$ que:

$$(S_U^+)_{(i,j,m+\frac{1}{2})}^n = 0 \text{ e } (S_U^-)_{(i,j,m+\frac{1}{2})}^n = \beta_{(i,j,m+\frac{1}{2})}^n \left(\theta S_{(i,j,m+1)}^{n+1} + (1-\theta) S_{(i,j,m+1)}^n \right) \\ - \left(1 - \beta_{(i,j,m+\frac{1}{2})}^{n+\theta}\right) \frac{1}{8} \left(S_{(i,j,m)}^n - 2S_{(i,j,m+1)}^n + S_{(i,j,m+2)}^n \right)$$

$$(S_L^+)_{(i,j,m+\frac{1}{2})}^n = 0 \text{ e } (S_L^-)_{(i,j,m+\frac{1}{2})}^n = \beta_{(i,j,m+\frac{1}{2})}^{n+\theta} \left(\theta S_{(i,j,m+1)}^{n+1} + (1-\theta) S_{(i,j,m+1)}^n \right) \\ - \left(1 - \beta_{(i,j,m+\frac{1}{2})}^{n+\theta}\right) \frac{1}{8} \left(S_{(i,j,m)}^n - 2S_{(i,j,m+1)}^n + S_{(i,j,m+2)}^n \right)$$

e portanto:

$$+ \frac{\Delta z_{(i,j,k)}^n}{16} \left(1 - \beta_{(i,j,k+\frac{1}{2})}^{n+\theta}\right) \left[(C_{rz}^+)_{(i,j,k+\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k-1)}^n + S_{(i,j,k+1)}^n - 2S_{(i,j,k)}^n \right) \right. \\ \left. + (C_{rz}^-)_{(i,j,k+\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k)}^n + S_{(i,j,k+2)}^n - 2S_{(i,j,k+1)}^n \right) \right] \text{ e} \\ \approx \frac{\Delta z_{(i,j,k)}^n}{16} \left(1 - \beta_{(i,j,k+\frac{1}{2})}^{n+\theta}\right) (C_{rz}^-)_{(i,j,k+\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k)}^n + S_{(i,j,k+2)}^n - 2S_{(i,j,k+1)}^n \right)$$

$$\begin{aligned}
& -\frac{\Delta z_{(i,j,k-1)}^n}{16} \left(1 - \beta_{(i,j,k-\frac{1}{2})}^{n+\theta}\right) \left[\left(C_{rz}^+\right)_{(i,j,k-\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k-2)}^n + S_{(i,j,k)}^n - 2S_{(i,j,k-1)}^n\right) \right. \\
& \quad \left. + \left(C_{rz}^-\right)_{(i,j,k-\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k-1)}^n + S_{(i,j,k+1)}^n - 2S_{(i,j,k)}^n\right) \right] \\
& \approx -\frac{\Delta z_{(i,j,k-1)}^n}{16} \left(1 - \beta_{(i,j,k-\frac{1}{2})}^{n+\theta}\right) \left(C_{rz}^-\right)_{(i,j,k-\frac{1}{2})}^{n+\theta} \left(S_{(i,j,k-1)}^n + S_{(i,j,k+1)}^n - 2S_{(i,j,k)}^n\right)
\end{aligned}$$

significando que S_U cumpre essas condições para $k = m$ e $w_{(i,j,m+\frac{1}{2})}^{n+\theta} < 0$.

Analogamente, S_L cumpre essas condições para $k = m + 1$ e $w_{(i,j,m+\frac{1}{2})}^{n+\theta} < 0$.

d. no caso dos fluxos $(S_U) \equiv (S_U)_{(i,j,M-\frac{1}{2})}^n$ e $(S_L) \equiv (S_L)_{(i,j,M-\frac{1}{2})}^n$, tem-se quando $w_{(i,j,M-\frac{1}{2})}^{n+\theta} < 0$, que :

$$\begin{aligned}
(S_U^-)_{(i,j,M-\frac{1}{2})}^n & \approx \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(\theta S_{(i,j,M)}^{n+1} + (1-\theta) S_{(i,j,M)}^n\right) \text{ e } (S_U^+)_{(i,j,M-\frac{1}{2})}^n = 0 \\
(S_L^+)_{(i,j,M-\frac{1}{2})}^n & = 0 \text{ e } (S_L^-)_{(i,j,M-\frac{1}{2})}^n \approx \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(\theta S_{(i,j,M)}^{n+1} + (1-\theta) S_{(i,j,M)}^n\right),
\end{aligned}$$

e portanto:

$$\begin{aligned}
& +\frac{\Delta z_{(i,j,M)}^n}{16} \left(1 - \beta_{(i,j,M+\frac{1}{2})}^{n+\theta}\right) \left[\left(C_{rz}^+\right)_{(i,j,M+\frac{1}{2})}^{n+\theta} \left(S_{(i,j,M-1)}^n + S_{(i,j,M+1)}^n - 2S_{(i,j,M)}^n\right) \right. \\
& \quad \left. + \left(C_{rz}^-\right)_{(i,j,M+\frac{1}{2})}^{n+\theta} \left(S_{(i,j,M)}^n + S_{(i,j,M+2)}^n - 2S_{(i,j,M+1)}^n\right) \right] \approx 0 \\
& -\frac{\Delta z_{(i,j,M-1)}^n}{16} \left(1 - \beta_{(i,j,M-\frac{1}{2})}^{n+\theta}\right) \left[\left(C_{rz}^+\right)_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(S_{(i,j,M-2)}^n + S_{(i,j,M)}^n - 2S_{(i,j,M-1)}^n\right) \right. \\
& \quad \left. + \left(C_{rz}^-\right)_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(S_{(i,j,M-1)}^n + S_{(i,j,M+1)}^n - 2S_{(i,j,M)}^n\right) \right] \approx 0
\end{aligned}$$

significando que S_U cumpre essas condições para $k = M - 1$ e $w_{(i,j,M-\frac{1}{2})}^{n+\theta} < 0$.

Analogamente, S_L cumpre essas condições para $k = M$ e $w_{(i,j,M-\frac{1}{2})}^{n+\theta} < 0$.

e. para $(S_U) \equiv (S_U)_{(i,j,M-\frac{1}{2})}^n$ e $(S_L) \equiv (S_L)_{(i,j,M-\frac{1}{2})}^n$ com $w_{(i,j,M-\frac{1}{2})}^{n+\theta} > 0$, tem-se que:

$$\begin{aligned}
(S_U^-)_{(i,j,M-\frac{1}{2})}^n & = 0 \text{ e } (S_U^+)_{(i,j,M-\frac{1}{2})}^n = \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(\theta S_{(i,j,M-1)}^{n+1} + (1-\theta) S_{(i,j,M-1)}^n\right) \\
& \quad - \left(1 - \beta_{(i,j,M-\frac{1}{2})}^{n+\theta}\right) \frac{1}{8} \left(S_{(i,j,M-2)}^n - 2S_{(i,j,M-1)}^n + S_{(i,j,M)}^n\right) \\
(S_L^-)_{(i,j,M-\frac{1}{2})}^n & = 0 \text{ e } (S_L^+)_{(i,j,M-\frac{1}{2})}^n = \beta_{(i,j,M-\frac{1}{2})}^{n+\theta} \left(\theta S_{(i,j,M-1)}^{n+1} + (1-\theta) S_{(i,j,M-1)}^n\right) \\
& \quad - \left(1 - \beta_{(i,j,M-\frac{1}{2})}^{n+\theta}\right) \frac{1}{8} \left(S_{(i,j,M-2)}^n - 2S_{(i,j,M-1)}^n + S_{(i,j,M)}^n\right)
\end{aligned}$$

e portanto:

$$\begin{aligned}
& + \frac{\Delta z_{(i,j,M)}^n}{16} \left(1 - \beta_{(i,j,M+\frac{1}{2})}^{\overline{n+\theta}}\right) \left[\left(C_{rz}^+\right)_{(i,j,M+\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,M-1)}^n + S_{(i,j,M+1)}^n - 2S_{(i,j,M)}^n\right) \right. \\
& \quad \left. + \left(C_{rz}^-\right)_{(i,j,M+\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,M)}^n + S_{(i,j,M+2)}^n - 2S_{(i,j,M+1)}^n\right) \right] \quad e \\
& \approx \frac{\Delta z_{(i,j,M)}^n}{16} \left(1 - \beta_{(i,j,M+\frac{1}{2})}^{\overline{n+\theta}}\right) \left(C_{rz}^+\right)_{(i,j,M+\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,M-1)}^n + S_{(i,j,M+1)}^n - 2S_{(i,j,M)}^n\right) \\
& - \frac{\Delta z_{(i,j,M-1)}^n}{16} \left(1 - \beta_{(i,j,M-\frac{1}{2})}^{\overline{n+\theta}}\right) \left[\left(C_{rz}^+\right)_{(i,j,M-\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,M-2)}^n + S_{(i,j,M)}^n - 2S_{(i,j,M-1)}^n\right) \right. \\
& \quad \left. + \left(C_{rz}^-\right)_{(i,j,M-\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,M-1)}^n + S_{(i,j,M+1)}^n - 2S_{(i,j,M)}^n\right) \right] \\
& \approx - \frac{\Delta z_{(i,j,M-1)}^n}{16} \left(1 - \beta_{(i,j,M-\frac{1}{2})}^{\overline{n+\theta}}\right) \left(C_{rz}^+\right)_{(i,j,M-\frac{1}{2})}^{\overline{n+\theta}} \left(S_{(i,j,M-2)}^n + S_{(i,j,M)}^n - 2S_{(i,j,M-1)}^n\right)
\end{aligned}$$

significando que S_U cumpre essas condições para $k = M - 1$ e $w_{(i,j,M-\frac{1}{2})}^{n+\theta} > 0$.

Analogamente, S_L cumpre essas condições para $k = M$ e $w_{(i,j,M-\frac{1}{2})}^{n+\theta} > 0$.

2. transporte horizontal: difusão

2.1 para células *internas* $2 \leq i \leq N - 1$ e $2 \leq j \leq M - 1$ as expressões são inalteradas, ou seja:

$$\begin{aligned}
& \frac{C_{r(i+\frac{1}{2},j,k)}^{\overline{n+\theta}}}{2P_{e(i+\frac{1}{2},j,k)}^n} \left(S_{(i+1,j,k)}^n - 2S_{(i,j,k)}^n + S_{(i-1,j,k)}^n\right) \Delta z_{(i,j,k)}^x \\
& \frac{C_{r(i,j+\frac{1}{2},k)}^{\overline{n+\theta}}}{2P_{e(i,j+\frac{1}{2},k)}^n} \left(S_{(i,j+1,k)}^n - 2S_{(i,j,k)}^n + S_{(i,j-1,k)}^n\right) \Delta z_{(i,j,k)}^y
\end{aligned}$$

2.2 para células adjacentes às *fronteiras inferior* e que não correspondem às esquinas da malha, ou seja, para $j = 1$ e $2 \leq i \leq N - 1$, tem-se:

$$\begin{aligned}
& \frac{C_{r(i+\frac{1}{2},1,k)}^{\overline{n+\theta}}}{2P_{e(i+\frac{1}{2},1,k)}^n} \left(S_{(i+1,1,k)}^n - 2S_{(i,1,k)}^n + S_{(i-1,1,k)}^n\right) \Delta z_{(i,1,k)}^x \\
& \frac{C_{r(i,1+\frac{1}{2},k)}^{\overline{n+\theta}}}{2P_{e(i,1+\frac{1}{2},k)}^n} \left(S_{(i,2,k)}^n - 2S_{(i,1,k)}^n + S_{(i,0,k)}^n\right) \Delta z_{(i,j,k)}^y
\end{aligned}$$

onde $S_{(i,0,k)}^n = \frac{3}{2}S_{(i,1,k)}^n - \frac{1}{2}S_{(i,2,k)}^n$

2.3 para células adjacentes às *fronteiras superiores* e que não correspondem às esquinas da malha, ou seja, para $j = M$ e $2 \leq i \leq N - 1$ tem-se:

$$\begin{aligned}
& \frac{C_{r(i+\frac{1}{2},M,k)}^{\overline{n+\theta}}}{2P_{e(i+\frac{1}{2},M,k)}^n} \left(S_{(i+1,M,k)}^n - 2S_{(i,M,k)}^n + S_{(i-1,M,k)}^n\right) \Delta z_{(i,M,k)}^x \\
& \frac{C_{r(i,M+\frac{1}{2},k)}^{\overline{n+\theta}}}{2P_{e(i,M+\frac{1}{2},k)}^n} \left(S_{(i,M+1,k)}^n - 2S_{(i,M,k)}^n + S_{(i,M-1,k)}^n\right) \Delta z_{(i,M,k)}^y
\end{aligned}$$

onde $S_{(i,M+1,k)}^n = \frac{3}{2}S_{(i,M,k)}^n - \frac{1}{2}S_{(i,M-1,k)}^n$

2.4 para células adjacentes às *fronteiras esquerda* e que não correspondem às esquinas da malha, onde $i = 1$ e $2 \leq j \leq M - 1$, tem-se:

$$\frac{C_{r(1+\frac{1}{2},M,k)}^{n+\theta}}{2P_{e(1+\frac{1}{2},M,k)}^n} \left(S_{(0,j,k)}^n - 2S_{(1,j,k)}^n + S_{(2,j,k)}^n \right) \Delta^x z_{(1,j,k)}^n$$

onde $S_{(0,j,k)}^n = \frac{3}{2}S_{(1,j,k)}^n - \frac{1}{2}S_{(2,j,k)}^n$

$$\frac{C_{r(1,j+\frac{1}{2},k)}^{n+\theta}}{2P_{e(1,j+\frac{1}{2},k)}^n} \left(S_{(1,j+1,k)}^n - 2S_{(1,j,k)}^n + S_{(1,j-1,k)}^n \right) \Delta^y z_{(1,j,k)}^n$$

2.5 para células adjacentes às *fronteiras direitas* e que não correspondem às esquinas da malha, onde $i = N$ e $2 \leq j \leq M - 1$, tem-se:

$$\frac{C_{r(N+\frac{1}{2},M,k)}^{n+\theta}}{2P_{e(N+\frac{1}{2},M,k)}^n} \left(S_{(N+1,j,k)}^n - 2S_{(N,j,k)}^n + S_{(N-1,j,k)}^n \right) \Delta^x z_{(N,j,k)}^n$$

onde $S_{(N+1,j,k)}^n = \frac{3}{2}S_{(N,j,k)}^n - \frac{1}{2}S_{(N-1,j,k)}^n$

$$\frac{C_{r(N,j+\frac{1}{2},k)}^{n+\theta}}{2P_{e(N,j+\frac{1}{2},k)}^n} \left(S_{(N,j+1,k)}^n - 2S_{(N,j,k)}^n + S_{(N,j-1,k)}^n \right) \Delta^y z_{(N,j,k)}^n$$

Note-se que não é necessário, como feito para as ESW 3D, determinar as CC para células correspondentes às *esquinas* da malha onde $(1, 1, k)$, $(N, 1, k)$, $(1, M, k)$ e (N, M, k) , já que nas ESW os termos de difusão horizontal são compostos por termos em u e v para a ECQM_x e para a ECQM_y. No caso da ETM 3D, a difusão horizontal considera somente S para as duas direções coordenadas.

3. transporte vertical: difusão

Como as difusões não são calculadas, respectivamente, em $k = m - 1$ e $k = M + 1$, já que não há difusão no topo e no fundo da coluna de água, as expressões válidas são as que estão agregadas ao vetor $B_{(k)}^n$, válidas para $m \leq k \leq M$, e são tais que:

$$\frac{C_{r(i,j,k+\frac{1}{2})}^{n+\theta}}{2P_{e(i,j,k+\frac{1}{2})}^n} \left(S_{(i,j,k+1)}^n - S_{(i,j,k)}^n \right) e - \frac{C_{r(i,j,k-\frac{1}{2})}^{n+\theta}}{2P_{e(i,j,k-\frac{1}{2})}^n} \left(S_{(i,j,k)}^n - S_{(i,j,k-1)}^n \right)$$

4. transporte horizontal: advecção

As condições de contorno para a advecção horizontal considerando-se as diversas funções limitantes, são as seguintes:

4.1 para células interiores $3 \leq i \leq N - 2$ e $3 \leq j \leq M - 2$.

Todas as expressões para as funções monitoras e os termos de concentração estão completamente definidos;

4.2 para células das *fronteiras inferiores* dadas por $3 \leq i \leq N - 2$, $j = 1, 2$, os termos S_E^+ , S_E^- , S_W^+ , S_W^- , S_N^- e S_S^- permanecem inalterados; devendo-se aproximar os fluxos S_S^+ e S_N^+ , e as funções monitoras $\lambda_{(i,j-\frac{1}{2},k)}^-$, $\lambda_{(i,j-\frac{1}{2},k)}^+$ e

$\lambda_{(i,j+\frac{1}{2},k)}^+$;

4.2.1 Para que $S_N^+|_{ND;j=1}$, tem-se $\Delta S = (S_{(i,j+1,k)}^n - S_{(i,j-1,k)}^n)$. Então, se $v_{(i,1+\frac{1}{2},k)}^{n+\theta} < 0$ n tem-se $S_N^+|_{ND;j=1} \rightarrow 0$ e se define $\Delta S = 0$ e $\lambda_{(i,j+\frac{1}{2},k)}^+|_{S_N^+} \approx 1$. Se

$$v_{(i,1+\frac{1}{2},k)}^{n+\theta} > 0, \text{ tem-se } S_{(i,0,k)}^n|_{j=1} = \max\left(\frac{5}{6}S_{(i,1,k)}^n - \frac{1}{6}S_{(i,2,k)}^n + S_{(i,3,k)}^n, 0\right);$$

4.2.2 aproxima-se $\lambda_{(i,j-\frac{1}{2},k)}^-|_{S_S^-} \approx -1$, $\lambda_{(i,j-\frac{1}{2},k)}^+|_{S_S^+} \approx 1$ e $\lambda_{(i,j+\frac{1}{2},k)}^+|_{S_S^+} \approx 1$;

4.2.3 para $S_S^+|_{ND;j=1,2}$, tem-se $\Delta S = (S_{(i,j-1,k)}^n - S_{(i,j-2,k)}^n)$. Então, se

$v_{(i,1+\frac{1}{2},k)}^{n+\theta} < 0$, tem-se $S_S^+|_{ND;j=1,2} \rightarrow 0$ e se define $\Delta S = 0$ e $\lambda_{(i,j+\frac{1}{2},k)}^+|_{S_S^+} \approx 1$. Se

$v_{(i,1+\frac{1}{2},k)}^{n+\theta} > 0$, tem-se:

$$S_{(i,0,k)}^n|_{j=2} = \max\left(\frac{5}{6}S_{(i,1,k)}^n - \frac{1}{6}S_{(i,2,k)}^n + S_{(i,3,k)}^n, 0\right)$$

$$S_{(i,-1,k)}^n|_{j=1} = \max\left(\frac{5}{6}S_{(i,0,k)}^n - \frac{1}{6}S_{(i,1,k)}^n + S_{(i,2,k)}^n, 0\right);$$

4.3 para células das *fronteiras superiores* dadas por $3 \leq i \leq N-2$, $j = M, M-1$, os termos S_E^+ , S_E^- , S_W^+ , S_W^- , S_N^+ e S_S^+ permanecem inalterados; deve-se aproximar os fluxos S_S^- e S_N^- , e as funções $\lambda_{(i,j+\frac{1}{2},k)}^+$, $\lambda_{(i,j-\frac{1}{2},k)}^-$ e $\lambda_{(i,j-\frac{1}{2},k)}^+$.

4.3.1 Para que $S_S^-|_{ND;j=M-1}$, tem-se $\Delta S = (S_{(i,j+1,k)}^n - S_{(i,j,k)}^n)$. E, se

$v_{(i,M-\frac{1}{2},k)}^{n+\theta} > 0$, tem-se $S_S^-|_{ND;j=M-1} \rightarrow 0$ e se define $\Delta S = 0$ e $\lambda_{(i,j-\frac{1}{2},k)}^- \approx 1$. Se

$$v_{(i,M-\frac{1}{2},k)}^{n+\theta} < 0, \text{ então } S_{(i,M,k)}^n|_{j=M-1} = \max\left(\frac{5}{6}S_{(i,M-1,k)}^n - \frac{1}{6}S_{(i,M-2,k)}^n + S_{(i,M-3,k)}^n, 0\right)$$

4.3.2 para essas funções tem-se $\lambda_{(i,j+\frac{1}{2},k)}^+|_{S_N^+} \approx 1$, $\lambda_{(i,j+\frac{1}{2},k)}^-|_{S_N^-} \approx 1$,

$$\lambda_{(i,j-\frac{1}{2},k)}^-|_{S_S^-} \approx 1 \text{ e};$$

4.3.3 para $S_N^-|_{ND;j=M,M-1}$ tem-se que $\Delta S = (S_{(i,j+2,k)}^n - S_{(i,j+1,k)}^n)$. Então se

$v_{(i,M-\frac{1}{2},k)}^{n+\theta} > 0$ tem-se $S_N^-|_{ND;j=M,M-1} \rightarrow 0$. Se $v_{(i,M-\frac{1}{2},k)}^{n+\theta} < 0$ tem-se:

$$S_{(i,M,k)}^n|_{j=M-1} = \max\left(\frac{5}{6}S_{(i,M-1,k)}^n - \frac{1}{6}S_{(i,M-2,k)}^n + S_{(i,M-3,k)}^n, 0\right)$$

$$S_{(i,M+1,k)}^n|_{j=M} = \max\left(\frac{5}{6}S_{(i,M,k)}^n - \frac{1}{6}S_{(i,M-1,k)}^n + S_{(i,M-2,k)}^n, 0\right)$$

4.4 para células das *fronteiras esquerdas* dadas por $3 \leq j \leq M-2$, $i = 1, 2$, os termos S_E^- , S_W^- , S_N^+ , S_N^- , S_S^+ e S_S^- permanecem inalterados; devendo-se aproximar os fluxos S_E^+ e S_W^+ , e as funções $r_{(i+\frac{1}{2},j,k)}^+$, $r_{(i-\frac{1}{2},j,k)}^+$ e $r_{(i-\frac{1}{2},j,k)}^-$.

4.4.1 Para que $S_E^+|_{ND;i=1}$, tem-se $\Delta S = (S_{(i,j,k)}^n - S_{(i-1,j,k)}^n)$. Então, se

$u_{(1,j+\frac{1}{2},k)}^{n+\theta} < 0$, tem-se $S_E^+|_{ND;i=1} \rightarrow 0$ e se define $\Delta S = 0$ e $r_{(i+\frac{1}{2},j,k)}^+ \approx 1$. Se

$$u_{(1,j+\frac{1}{2},k)}^{n+\theta} > 0, \text{ tem-se } S_{(i,0,k)}^n|_{i=1} = \max\left(\frac{5}{6}S_{(i,1,k)}^n - \frac{1}{6}S_{(i,2,k)}^n + S_{(i,3,k)}^n, 0\right);$$

4.4.2 para as funções tem-se $r_{(i+\frac{1}{2},j,k)}^+ \approx 1$, $r_{(i-\frac{1}{2},j,k)}^+ \approx 1$ e $r_{(i-\frac{1}{2},j,k)}^- \approx 1$;

4.4.3 para $S_W^+|_{ND;i=1,2}$, tem-se que $\Delta S = (S_{(i-1,j,k)}^n - S_{(i-2,j,k)}^n)$. E, se $u_{(1,j+\frac{1}{2},k)}^{n+\theta} < 0$, tem-se $S_W^+|_{ND;i=1,2} \rightarrow 0$ e se define $\Delta S = 0$ e $r_{(i-\frac{1}{2},j,k)}^+ \approx 1$. Se $u_{(1,j+\frac{1}{2},k)}^{n+\theta} > 0$, tem-se:

$$S_{(0,j,k)}^n|_{i=2} = \max\left(\frac{5}{6}S_{(1,j,k)}^n - \frac{1}{6}S_{(2,j,k)}^n + S_{(3,j,k)}^n, 0\right)$$

$$S_{(-1,j,k)}^n|_{j=1} = \max\left(\frac{5}{6}S_{(0,j,k)}^n - \frac{1}{6}S_{(1,j,k)}^n + S_{(2,j,k)}^n, 0\right)$$

4.5 para células das *fronteiras direitas* dadas por $3 \leq j \leq M - 2$, $i = N, N - 1$, os termos S_E^+ , S_W^+ , S_N^+ , S_N^- , S_S^+ e S_S^- permanecem inalterados; deve-se aproximar os fluxos S_E^- e S_W^- , e as funções $r_{(i+\frac{1}{2},j,k)}^+$, $r_{(i-\frac{1}{2},j,k)}^-$ e $r_{(i-\frac{1}{2},j,k)}^-$.

4.5.1 Para que $S_W^-|_{ND;j=N-1}$, tem-se $\Delta S = (S_{(i+1,j,k)}^n - S_{(i,j,k)}^n)$. Se $u_{(M-\frac{1}{2},j,k)}^{n+\theta} > 0$, tem-se $S_W^-|_{ND;j=N-1} \rightarrow 0$ e se define $\Delta S = 0$ e $r_{(i-\frac{1}{2},j,k)}^- \approx 1$. Se $u_{(M-\frac{1}{2},j,k)}^{n+\theta} < 0$,

$$S_{(N,j,k)}^n|_{j=M-1} = \max\left(\frac{5}{6}S_{(N-1,j,k)}^n - \frac{1}{6}S_{(M-2,j,k)}^n + S_{(M-3,j,k)}^n, 0\right)$$

4.5.2 para as funções monitoras dessas funções tem-se que $r_{(i+\frac{1}{2},j,k)}^+|_{S_E^+} \approx 1$, $r_{(i-\frac{1}{2},j,k)}^-|_{S_E^-} \approx 1$ e $r_{(i-\frac{1}{2},j,k)}^-|_{S_W^-} \approx 1$;

4.5.3 para $S_E^-|_{ND;I=N,N-1}$, tem-se que $\Delta S = (S_{(i+2,j,k)}^n - S_{(i+1,j,k)}^n)$. Então, se $u_{(N-\frac{1}{2},j,k)}^{n+\theta} > 0$, tem-se $S_E^-|_{ND;I=N,N-1} \rightarrow 0$. Se $u_{(N-\frac{1}{2},j,k)}^{n+\theta} < 0$, tem-se:

$$S_{(N,j,k)}^n|_{j=N-1} = \max\left(\frac{5}{6}S_{(N-1,j,k)}^n - \frac{1}{6}S_{(N-2,j,k)}^n + S_{(N-3,j,k)}^n, 0\right)$$

$$S_{(N+1,j,k)}^n|_{j=N} = \max\left(\frac{5}{6}S_{(N,j,k)}^n - \frac{1}{6}S_{(N-1,j,k)}^n + S_{(N-2,j,k)}^n, 0\right)$$

Não é necessário determinar as CC para células correspondentes às *esquinas* da malha onde $(1, 1, k)$, $(2, 1, k)$, $(N, 1, k)$, $(N - 1, 1, k)$, $(1, M, k)$, $(1, M - 1, k)$, (N, M, k) , $(N - 1, M - 1, k)$, $(1, 2, k)$, $(2, 2, k)$, $(N, 2, k)$, $(N - 1, 2, k)$, $(2, M, k)$, $(2, M - 1, k)$, $(N - 1, M, k)$, $(N, M - 1, k)$, pois, no caso da ETM 3D, a advecção horizontal considera somente S para as duas direções coordenadas.

Nas fronteiras abertas pode-se implementar, se necessário, CC tipo gradiente nulo, como discutido nas seções 2.5.1, 3.4 e 3.5.1. Porém esse tipo de CC é somente especificado para esquemas implícitos que empregam a condição de gradiente nulo de modo a incorporar as CC no sistema de equações. Quando o transporte horizontal é explícito, como no caso 3D neste trabalho, não é necessário implementar essas CC. Detalhes em [RIZ2002a].

7 Solução Paralela: ambiente computacional e métodos

A programação paralela pode ser *explícita* ou *implícita*. Na implícita, o compilador gera o código de acordo com o grau de paralelismo determinado pelo programador, dentro das características do compilador. A explícita requer o total controle sobre as estratégias de implementação e sobre a própria implementação.

Não obstante o avanço tecnológico do *hardware* e do *software*, os avanços nos algoritmos nas décadas de 80 e 90, sobretudo aqueles de particionamento de malha, de decomposição de domínio, de pré-condicionamento e dos métodos iterativos, foram os principais elementos motivadores para o emprego de arquiteturas tipo *Massively Parallel Processors* (MPP) e/ou *clusters*. Sob essa argumentação, discutem-se nesse capítulo os métodos de estratégias de paralelização empregadas neste trabalho.

Este trabalho pode ser classificado como pertencente à classe de *estudos ambientais* (*Environmental Studies*) e só é factível através do emprego de modelagem computacional combinada com métodos experimentais, que são empregados para obter os dados utilizados na validação e na calibração do modelo computacional. Exemplos de aplicações que são próximas a esse trabalho, sob o ponto de vista de estratégias de solução paralela e de esquemas numéricos, são Modelagem de Reservatórios (*Reservoir Modeling*), Ciências Atmosféricas (*Atmospheric Science*) e (Modelagem Oceânica Global) *Global Ocean Modeling*, entre outras.

Os domínios práticos de tais problemas são de grande porte e exigem, para uma solução com alta resolução e qualidade numérica, uma capacidade de processamento e de armazenamento tal que torna imprescindível o uso de ambientes computacionais de alto desempenho. Sob tais ambientes, as aplicações podem ser realizadas com um nível de detalhe que não seria viável em abordagens computacionais sequenciais. Nesse sentido, os *clusters* de PCs têm se mostrado uma opção acessível e eficiente, dado o crescente desenvolvimento de redes de alta velocidade, e permitem obter desempenho equivalente ao de processadores vetoriais por um custo muito menor.

Porém, nesse tipo de arquitetura, dado a falta de ferramentas para auxiliar a paralelização, a programação paralela é *explícita*, exigindo o completo controle sobre as estratégias de implementação e sobre a própria implementação. Além disso, o tipo de paralelismo a ser explorado em um *cluster* depende da arquitetura disponível. Em *clusters* multiprocessados, deve-se explorar o paralelismo intra-nodal e o inter-nodal. No inter-nodal, a memória é distribuída e os processos comunicam-se por troca de mensagens. No paralelismo intra-nodal, a memória é compartilhada e o paralelismo é explorado usando *threads* ou processos escalonados pelo sistema operacional (SO).

Sob o ponto de vista da programação, o paradigma empregado neste trabalho é o SPMD (*Single Program Multiple Data*), pelo qual todos os processadores executam o mesmo código com sua porção local de dados. E, assim, questões cruciais em aplicações paralelas que são discutidas a seguir são os problemas da localidade dos dados, o balanceamento de carga e a comunicação, entre outras.

7.1 Plataforma computacional e ambiente de programação

Um *cluster* de PCs é um agregado de PCs, chamados de nodos, interconectados por redes de alta velocidade. Esses nodos trabalham juntos como se fossem um único e integrado sistema de computação. Os nodos que compõem um *cluster* podem ter um ou mais processadores. Quando existe mais de um processador por nodo, a arquitetura é dita multiprocessada. Nesse caso, além do paralelismo inter-nodal (memória distribuída), pode-se explorar o paralelismo intra-nodal (memória compartilhada). Os *clusters* podem, ainda, ser homogêneos se todos os nodos têm mesma arquitetura e mesmo sistema operacional, ou heterogêneos se os nodos possuem processadores, memórias ou sistemas operacionais diferentes.

Os algoritmos implementados foram executados em um *cluster* tipo heterogêneo multiprocessado do Instituto de Informática da UFRGS. Esse *cluster* é constituído por quatro nodos *Dual Pentium Pro* 200 MHz com 128 MB de memória RAM e por quatro nodos *Dual Pentium III* 500 MHz com 512 MB de memória RAM, conectados pela rede *Fast-Ethernet*. Outro *cluster* utilizado foi o disponível no LNCC, que é um *cluster* homogêneo, formado por 32 nodos com processadores *Pentium III* 800 MHz monoprocessados com 256 MB de memória RAM. Seus nodos são conectados pela rede *Fast-Ethernet*. Um terceiro *cluster* empregado foi o disponível no CPAD PUCRS/HP, que é composto por nodos *Pentium III* 550MHz *duais* com 512 MB de memória RAM conectados pela rede *Fast Ethernet*.

A disponibilização de arquiteturas heterogêneas multiprocessadas em Centros de Pesquisas é uma tendência, já que a necessidade de mais capacidade computacional pode ser suprida, adicionando-se nodos extras ao *cluster*. Devido ao rápido avanço tecnológico, cada atualização no *cluster* agrega máquinas com diferentes capacidades, resultando em *clusters* heterogêneos. Essas são questões relevantes para o *problema de mapeamento*, que é discutido nas próximas seções

As arquiteturas de memória compartilhada e distribuída são, sob o ponto de vista da programação, bem caracterizadas. Sob a memória compartilhada, os dados e as operações são globais, isto é, têm o mesmo endereçamento. Sob a memória distribuída, a comunicação entre processos é obtida através de trocas de mensagens. A biblioteca utilizada a MPICH 1.2.1 do ANL [GRP2000], uma implementação do padrão MPI (*Message Passing Interface*), que contempla um conjunto de 125 primitivas [PAC97]. O MPICH segue o paradigma SPMD, onde cada processador executa uma instância do mesmo algoritmo. O SO usado é o LINUX 2.2.1, e o compilador, o *gcc* versão 2.9.1.6.

Em *clusters* multiprocessados, deve-se explorar o paralelismo intra-nodal, um bom recurso para esse fim é o uso de múltiplas *threads*, pois compartilham código, dados e recursos. Uma *thread* pode ser vista como um fluxo de execução e, caso seja desejado, é possível aumentar explicitamente o número de *threads* existentes em um programa [PIC2001]. Existem alguns modelos de programação de *threads*, mas, nas plataformas originárias de UNIX, predomina o IEEE POSIX *threads* (*Pthreads*) que é o modelo padrão ISSO/IEC 99-45-1.

O trabalho que vem sendo desenvolvido por Picinin (dissertação de mestrado no GMCPAD do PPGC-UFRGS) enfoca a exploração dos dois níveis de paralelismo, o inter e o intra-nodal, em *clusters* multiprocessados, bem como o desenvolvimento de algoritmos de ordenação de mensagens entre os processos visando evitar *deadlock* e/ou melhorar o desempenho computacional, quando da falta de *buffer* pela biblioteca de troca de mensagens Os resultados obtidos ([PIC2001] e [PIC2002]) indicam que o ganho de desempenho usando *threads* no segundo nível de paralelismo, em relação ao uso somente de processos é, em média, 11%, o que é significativo.

Além disso, também se pode usar *threads* para sobrepor comunicação com processamento, como em [VOR93]. Nesse contexto, um resultado preliminar de André Luis Martinotto [MAR2002], que desenvolve sua dissertação de mestrado no GMCPAD do PPGC-UFRGS, enfocando decomposição de domínio e sobreposição de comunicação e computação, mostrou que o uso do MPI com *threads* na paralelização é efetivo, sobretudo quando o volume de comunicação global entre os processos é elevado. Nesse caso, o uso de múltiplas *threads* reduz o volume de comunicação entre processos, tornando essa abordagem vantajosa à medida que o número de processos cresce.

Porém, não obstante o desempenho oferecido pelo uso de *threads*, como o SO escalona processos nos nodos multiprocessados sem o custo de implementação do segundo nível de paralelismo (*threads*), os resultados apresentados neste trabalho consideram essa última abordagem. Em futuras otimizações o modelo computacional aqui desenvolvido poderá contemplar o uso de *threads* não só para explorar o segundo

nível de paralelismo (intra-nodal), mas também para sobrepor comunicação e computação, como discutido no final da seção 3.5.2.

7.2 Decomposição de dados

O processo de paralelizar um método de solução, distribuindo os dados e as instruções entre os processadores é chamado, neste trabalho, de decomposição de dados. Nessa abordagem o método de solução do problema (global) é distribuído e resolvido em paralelo. Essa é uma das abordagens que foi empregada neste trabalho para obter a solução paralela do algoritmo do gradiente conjugado e do algoritmo de Thomas. As comunicações coletivas empregadas nessas paralelizações são feitas através de operações que estão disponíveis no MPI.

No MPI, a comunicação dos processos é baseada no envio e no recebimento de mensagens. A transferência dos dados entre esses processos requer operações entre eles de modo que uma operação de envio deve ter uma respectiva operação de recebimento e vice-versa. Entre as primitivas do MPI destaca-se [PAC97]:

1. *primitivas de gerência*: responsáveis por determinar o número de processos, obter a identificação de cada um deles, inicializar e finalizar processos;
2. *primitivas de comunicação ponto a ponto*: responsáveis pelo envio e recebimento de mensagens entre dois processos;
3. *primitivas de comunicação em grupo*: responsáveis pela comunicação entre vários processos em um grupo.

Na criação dos processos, cada um recebe uma identificação que é única, e é através dessa identificação que os processos comunicam-se. A comunicação ponto a ponto é feita entre dois pontos, sendo um o receptor, e o outro, o emissor. As principais primitivas desse tipo são, no MPI, o *MPI_Send* e o *MPI_Receive*.

A comunicação coletiva é feita entre vários processos, e pode ser feita com um processo se comunicando com todos, todos comunicando-se com um e/ou com todos. Um exemplo para uma comunicação coletiva é a operação *broadcast* um para todos (1:N) que é responsável pelo envio do mesmo conjunto de dados de um nodo para todos os outros nodos. O *MPI_Broadcast* é uma primitiva de comunicação coletiva do MPI, pela qual um processo envia o mesmo dado para todos os outros processos.

A operação *gather* é utilizada para coletar as estruturas de dados distribuídas entre os nodos. É particularmente útil na multiplicação de uma matriz densa por um vetor, no sentido de obter o vetor completo em cada nodo para realizar a multiplicação paralela. Assim, um *gather* é uma comunicação que consiste de um único processo coletar uma *estrutura de dados* distribuída entre os outros processos. A operação *gather* é chamada, no MPI, de *MPI_Gather*. Outra operação é a *scatter*, que distribui os dados armazenados em um único nodo para os outros. É utilizada no particionamento inicial dos dados da matriz e do vetor entre os nodos. Na comunicação *scatter*, a *estrutura de dados* armazenada em um único nodo é distribuída para os outros nodos e, no MPI, é chamada de *MPI_Scatter*.

Outra operação frequentemente utilizada é o *reduce*, pela qual todos os processadores enviam seus dados a um nodo. Essa operação é utilizada nas operações de produto escalar entre dois vetores. Então a operação *reduce* coleta esses dados parciais e centraliza em um dos nodos. O MPI implementa essa operação através da primitiva *MPI_Allreduce*.

7.3 Particionamento, mapeamento e balanceamento de carga

Ambientes computacionais dotados de múltiplos processadores, como os *clusters* de PCs, podem resolver problemas de grande porte concorrentemente, quando da situação em que cada processador recebe um ou mais subdomínios, partes do domínio

global. Tais subdomínios devem ser gerados por algoritmos eficientes, que identifiquem as células internas e das fronteiras e que tenham as características de [SMI96]:

1. balancear a carga computacional entre os processadores, de modo que cada um tenha carga proporcional a sua capacidade de processamento;
2. diminuir o número de células nas fronteiras, de modo a minimizar o tempo de sincronização entre os processadores, dado que as aplicações paralelas são, geralmente, síncronas, e as comunicações são restritas às fronteiras;
3. assegurar a localidade dos dados, de modo a minimizar o número de dados que devem ser trocados a cada passo no tempo;
4. considerar discretizações arbitrárias e geometrias irregulares, permitindo um tratamento mais realístico para problemas definidos em domínios complexos.

Os problemas de particionamento de domínio (ou dados) em subdomínios (ou porções de dados), considerando-se a arquitetura disponível e exigindo o balanceamento de carga e a minimização da comunicação dos processos, durante tempo de execução, são questões que remetem a quatro das mais importantes áreas de estudos da Computação Científica Paralela (CCP). A CCP é definida como sendo o projeto e a análise de algoritmos para a solução de problemas em Ciência e Engenharia, adotando técnicas de paralelismo, de modo que esses algoritmos possam ser executados em máquinas paralelas, objetivando soluções computacionalmente eficientes e numericamente acuradas [HEA2001]. Essas quatro áreas estão sob o escopo dos métodos de particionamento de malha; dos algoritmos de mapeamento, dos métodos e algoritmos de balanceamento de carga, e dos métodos de decomposição de domínio.

Pode-se dizer, de modo aproximado, que métodos de particionamento de malha são aqueles que decompõem os dados entre os processos buscando maximizar a relação computação/comunicação. Algoritmos de mapeamento visam mapear o grafo da aplicação sobre o grafo da arquitetura. Métodos de balanceamento dinâmico de carga buscam manter, em tempo de execução, aquela relação computação/comunicação calculada no particionamento estático. Métodos de decomposição de domínio são um conjunto de técnicas matemáticas e computacionais pelas quais as soluções locais geram a solução do problema global.

Um dos enfoques deste trabalho é métodos de decomposição de domínio, enquanto métodos de solução e enquanto pré-condicionadores. Os problemas de particionamento de malha e balanceamento dinâmico de carga são tratados por Ricardo Vargas Dorneles [DOR2002]. Carvalho [CAR2002] estudou o mapeamento em arquiteturas heterogêneas, além do problema de particionamento de malha.

7.3.1 Heurísticas para o particionamento

O particionamento de malha numérica pode ser visto como um problema de particionamento de grafos, quando se considera o grafo $G \equiv (V, A, p_a, p_v)$ composto pelo conjunto $V = \{0, \dots, n-1\}$ com n nodos; pelo conjunto $A \subseteq V \times V$ de arestas; pelos pesos p_a dos nodos; e pelos pesos p_v dos vértices. Então, pode-se associar a cada nodo processos ou dados, e às arestas as comunicações ou dependência de dados, aos pesos p_a a carga computacional, e aos pesos p_v a carga de comunicação.

O problema de particionar o grafo em tantas partes quantos forem os processos, visando a minimizar o número de arestas entre eles, é conhecido como k – particionamento. A idéia básica do particionamento, no caso simples de a malha numérica ser gerada na discretização por diferenças finitas, é percorrer todas as células efetivas de cálculo do domínio, identificando-as e associando a cada uma delas a sua carga computacional (vértices) e a sua dependência de dados (arestas). A partir daí, pode-se gerar uma lista que contém a quantidade de vértices e de arestas, mais

uma enumeração das relações entre cada vértice. Deve-se obter um k – particionamento para essa lista, mas esse é um problema NP-difícil ([GAR79] e [TOS97]) e, portanto, abordagens heurísticas são as únicas viáveis.

A literatura a esse respeito é vasta, e algumas referências pertinentes são [WAL95], [BAK96], [PEL96], [DIE97], [KAR98], [SCH98], [SCH99], [DEM99], [WAL2000] e [KUM2002]. Em Schloegel [SCH2000a] e Kumar [KUM2002] tem-se uma síntese de vários métodos e algoritmos comparando suas principais características como, a qualidade, tempo de execução e o grau de paralelismo.

As heurísticas para o particionamento podem ser classificadas em globais e locais. As globais são conhecidas como heurísticas de construção, já que utilizam a descrição do grafo como entrada e geram seu particionamento. As locais são conhecidas como heurísticas de melhoramento, pois utilizam o grafo e o particionamento como entrada e tentam melhorar o particionamento através do rearranjo dos vértices (células computacionais). Alguns métodos da classe das heurísticas globais empregados para o particionamento de malhas cartesianas são aqueles que particionam a malha em qualquer uma das direções coordenadas. Eles são discutidos em [DOR2001], [VOL97], [VOL97a] e [ROE97], e exemplos são:

1. SSTRIP (*straight stripwise partitioning*): cada submalha recebe um determinado número de linhas ou colunas inteiras;
2. STRIP (*stripwise partitioning*): análogo ao STRIP, exceto que, ou a linha ou a coluna da fronteira pode ser particionada em dois processadores adjacentes;
3. RCB (*recursive coordinate bisection*): utiliza as informações das coordenadas da malha para fazer a bissecção, que é aplicada recursivamente para cortar o grafo em mais de duas partes;
4. ORB (*orthogonal recursive bisection*): a bissecção no passo atual é feita em uma direção ortogonal à bissecção do passo anterior. A bissecção pode ser feita particionando a linha ou coluna da fronteira entre dois processadores, garantindo submalhas de mesmo tamanho;
5. SORB (*straight orthogonal recursive bisection*): é análogo ao método ORB, porém cada processador recebe apenas linhas ou colunas inteiras, o que pode gerar subdomínios de diferentes tamanhos.

Outros exemplos de métodos globais sofisticados, que servem para qualquer tipo de malha, são os espectrais como o RSB (*recursive spectral bisection*) [LUC2000]; os gulosos (*greedy*) ([HEN93] e [SZW99]); o HSFC (*Hilbert space-filling curve*) ([PIL96] e [ALU97]), entre outros, que podem ser vistos em [DOR2001], bem como as estratégias tipo multinível (*multilevel algorithm*) ([DEM96] e [KUM2002]).

Heurísticas globais geralmente necessitam de refinamentos nos seus cortes. Esses refinamentos são obtidos através de heurísticas locais. Algumas das principais heurísticas dessa classe são ([KER70] e [DOR2001]):

1. K-L (*Kernighan-Lin*): usa a estratégia de trocar pares de nodos, como no método guloso, considerando um aumento no tempo do corte, evitando mínimos locais;
2. F-M (*Fiduccia-Mattheyses*): é o resultado da modificação do algoritmo K-L, gerando uma versão mais complexa, porém mais eficiente daquele algoritmo.

A escolha desta ou daquela heurística deve levar em conta as características da aplicação, considerando o custo do particionamento \times qualidade da partição, o grau de paralelismo de tais heurísticas, entre outras questões.

Os algoritmos acima mencionados estão implementados em bibliotecas ou pacotes de particionamento de grafos. Quatro dos pacotes mais conhecidos são: METIS [KAR98], JOSTLE [WAL2000], CHACO [HEN95] e SCOTCH [PEL96].

A fig. 7.1 mostra um exemplo para o particionamento para o domínio do Lago Guaíba usando o algoritmo RCB. O particionamento foi gerado considerando-se como

espaçamento horizontal $\Delta x = \Delta y = 50m$. As espessuras das camadas verticais variam conforme o número de camadas. Com essas escolhas tem-se, para o caso da fig. 7.1, 185.124 vértices (células efetivas de cálculo) e 366.032 arestas quando com somente 1 camada, e 390.781 vértices e 776.525 arestas quando com 8 camadas.

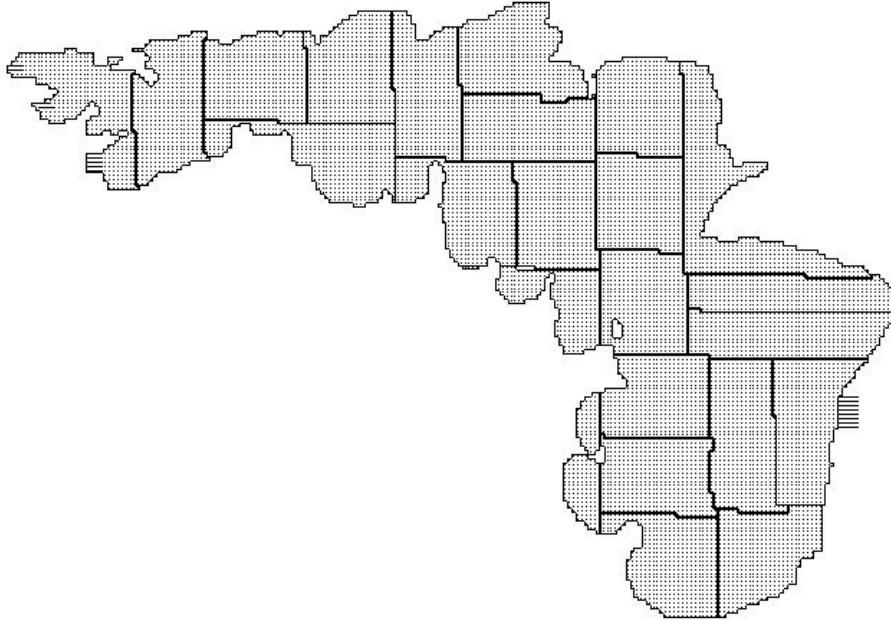


FIGURA 7.1 - Particionamento, em 23 processos, para o Lago Guaíba usando o algoritmo RCB

A fig. 7.2 mostra um particionamento para o Lago Guaíba em 16 processos usando a biblioteca METIS. Esse exemplo ilustra um campo de velocidades superficial para o modelo com 8 camadas, e onde a configuração geométrica utilizada foi aquela obtida com o arquivo Guaiba_50.

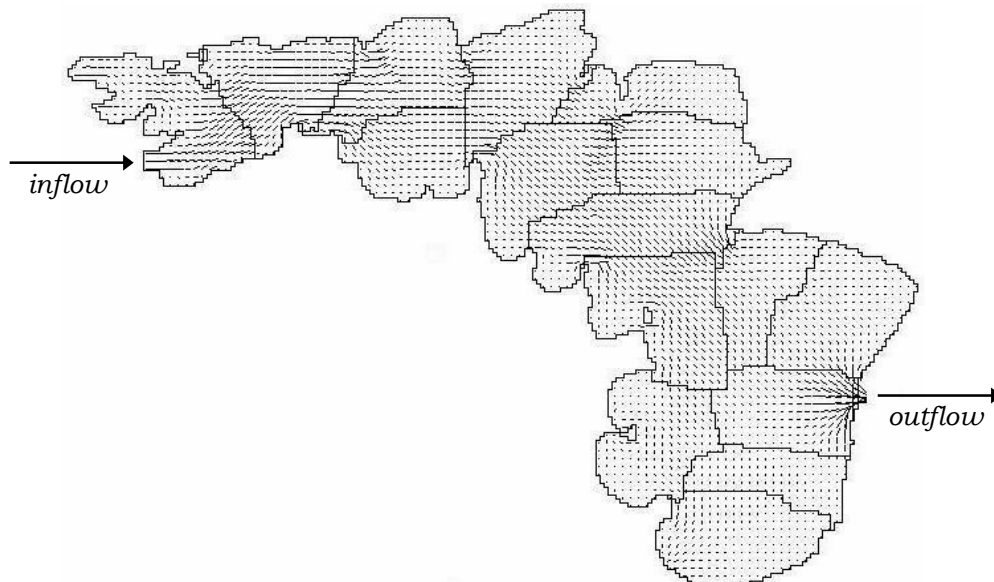


FIGURA 7.2 - Particionamento, em 16 processos, para o Lago Guaíba usando a biblioteca METIS, onde se destaca o campo de velocidades

Note-se que em soluções paralelas cada subdomínio deve trocar dados com os subdomínios vizinhos para atualizar as condições de contorno e/ou a sobreposição. Um aspecto importante para que essas trocas sejam realizadas com o menor custo é o

ordenamento das mensagens. Os problemas decorrentes de um ordenamento ineficiente das mensagens vão desde a queda de desempenho, que pode ocorrer se diversos processos tentam enviar mensagens para o mesmo processo simultaneamente, até a ocorrência de *deadlocks*.

Neste trabalho o particionamento do domínio é centralizado no processo 0, que cria a estrutura em memória descrevendo a configuração geométrica, e sobre essa estrutura é executado o particionamento, gerando-se para cada subdomínio, estruturas que o descrevem, bem como a estrutura dos subdomínios vizinhos. De posse dessas estruturas, cada processo aloca memória para todo o seu subdomínio e inicializa as variáveis de cada célula do domínio.

A ordenação de mensagens considera que a escolha do ordenamento das mensagens é efetuada de modo centralizado pelo mesmo processo que efetua o particionamento, e a seqüência de trocas de cada processo é enviada no início da simulação juntamente com as informações de cada partição, para o processo responsável. Essa implementação utiliza a numeração dos processos para definir a ordem de troca de mensagem.

7.3.2 Métodos e algoritmos de mapeamento

Uma distribuição não equilibrada da carga computacional pode acarretar ociosidade em alguns processadores, pelo tempo de espera inserido por sincronismos. Isso ocorre devido ao fato de que simulações de eventos reais tem dependência física nas fronteiras artificialmente criadas pelo algoritmo de particionamento. Em geral, os processos não podem resolver suas partes em paralelo sem troca de informações nessas fronteiras. Portanto, a distribuição de computação deve levar em conta que os processadores podem ter diferentes capacidades e/ou receber tarefas mais pesadas.

Se os processadores possuírem diferentes capacidades de processamento e/ou comunicação como, por exemplo, velocidade do processador, memória RAM, redes de interconexão, etc., deve-se considerar essas diferenças de modo que as cargas sejam proporcionais à capacidade de processamento e/ou comunicação de cada nodo da plataforma heterogênea na qual se resolve a simulação computacional [CAR2002].

Esse problema pode ser formalizado de modo análogo ao problema de particionamento de grafos, notando-se que a arquitetura também pode ser vista como um grafo. Seja o grafo $H \equiv (N, R, p_n, p_r)$, em que $N = \{0, \dots, p-1\}$ é o conjunto dos processadores (ou processos); $R \subseteq H \times H$ é o conjunto das arestas; aos pesos p_n é associada a capacidade de processamento e/ou memória de cada nodo; e aos pesos p_r , a largura de banda e latência da rede ([WAL2000] e [DOR2001]).

Assim, a composição particionamento de malha e balanceamento de carga pode ser vista como um problema de mapeamento de grafos. Deve-se encontrar um mapeamento $\Pi: G \rightarrow H$, do grafo da aplicação para o grafo da arquitetura, de modo a maximizar a relação computação/comunicação considerando-se a arquitetura disponível. Nesse caso, o problema do mapeamento é dependente da rede de interconexão. Se os nodos são interligados através de um *switch* na rede de conexão (onde todos os nodos se comunicam simultaneamente), como o caso do *cluster* do II-UFRGS, então, sob o ponto de vista de grafos, essa rede pode ser vista como um grafo fortemente conectado. O problema do mapeamento reduz-se, portanto, a maximizar a relação computação/comunicação através de um particionamento estático que leve em conta a capacidade de processamento de cada nodo.

A literatura técnica disponível sobre o problema de mapeamento é restrita, e um dos poucos algoritmos encontrados é o da Bissecção Recursiva Dual (BRD), desenvolvido por Pellegrini [PEL96] para ser utilizado na biblioteca SCOTCH, que se distingue por não ser apenas uma biblioteca de particionamento de grafos; mas

também de mapeamento. O algoritmo BRD utiliza um grafo valorado, para representar a plataforma computacional, onde se definem pesos para os vértices, que representam a capacidade de processamento dos nodos, e se definem pesos para as arestas, que representam a comunicação entre os nodos. O algoritmo minimiza a comunicação mantendo o balanceamento de carga, com certo erro.

Outra abordagem é aquela onde se faz o mapeamento após um agrupamento (*clusterização*) dos nodos do grafo da aplicação, gerando um *cluster* multinível, e um agrupamento dos vértices do grafo dos processadores, gerando um *cluster* multinível. A seguir emprega-se um algoritmo para mapear um grafo sobre o outro. Outro algoritmo é aquele baseado no paradigma da divisão-e-conquista, e utiliza uma variação do algoritmo Kernighan-Lin. Nessa abordagem, as tarefas são alocadas, considerando-se redes de interconexão tipo hipercubo, através de repetidas bissecções sobre o grafo representativo das tarefas. Nesse caso, considera-se que as cargas sejam balanceadas com certa tolerância, que é especificada. Discussões adicionais sobre essas abordagens podem ser vistas em [CAR2002]. Neste trabalho considerou-se, no entanto, que a rede de interconexão é fortemente conectada e os nodos são homogêneos.

7.3.3 Heurísticas e métodos de balanceamento de carga

Algoritmos de particionamento de grafos dividem a computação estática. Para problemas nos quais a carga é conhecida no tempo inicial e permanece ao longo do tempo, o balanceamento de carga é estático. Se a carga computacional alocada a cada processador é alterada em tempo de execução, esse processo é chamado de balanceamento dinâmico [QUI94]. Na definição dos algoritmos de balanceamento dinâmico, deve-se considerar ([WAT95] e [DOR2001]):

1. *a avaliação da carga*: necessária para detectar o grau do desbalanceamento e identificar os processos nos quais devem ocorrer as transferências de tarefas;
2. *a avaliação do custo/benefício*: deve ser considerada no cálculo do novo particionamento e da nova migração de dados, pois é necessário levar em conta que esses processos têm um custo que pode ser significativo.

A detecção da carga é obtida usando métricas que indiquem quais processadores estão desbalanceados. Para redistribuir a carga deve-se parar o processamento, recalcular a malha, reparticioná-la e mapeá-la novamente. Assim, deve-se verificar se a relação custo/benefício é compensatória.

Se for compensatória, uma estratégia é usar algoritmos de difusão que, geralmente, garantem um novo balanceamento de carga com pouca movimentação de dados ([DIE97a] e [SCH2000]). Com essa estratégia, avalia-se a carga de cada um dos processadores e compara-se sua carga com a dos processadores vizinhos. Se a diferença estiver acima de um limite, transfere-se parte da carga de um processador a outro, tal como ocorre na difusão do calor [GHO96]. Detalhes e outras estratégias podem ser vistos em [DOR2001] e [DOR2002].

Um exemplo concreto da necessidade do balanceamento dinâmico de carga é a aplicação feita neste trabalho. O modelo de transporte de substâncias é resolvido com a malha refinada encaixada na malha da hidrodinâmica. Entretanto, como a pluma de substâncias pode tomar outras direções no decorrer do tempo, ocorrem freqüentes desbalanceamentos da carga que, geralmente, são significativos. Nesse caso, o problema multifísica pode ser abordado como:

1. particionar a malha global, considerando que os nodos da malha refinada sejam pesos para os nodos das sub-regiões nas quais ocorre o transporte. Nesse caso, os subproblemas da hidrodinâmica e os do transporte de substâncias são alocados no mesmo processador. A comunicação é mínima, mas o balanceamento de carga pode não ser o mais adequado;

2. particionar a malha global e a refinada independentemente, e mapeá-las aos processadores. Nesse caso, o balanceamento da carga pode ser feito empregando-se algoritmos de difusão, como definidos por [DOR2002]. O problema dessa abordagem é que pode ser necessário movimentar todos os dados locais da malha da hidrodinâmica para o processador que tenha a respectiva malha do transporte, pois os valores da hidrodinâmica são dados de entrada para a solução do transporte.

Essas e outras estratégias são objetos de pesquisa de Dorneles [DOR2002], e estão sendo feitas análises e experiências numéricas para verificar qual abordagem é a mais conveniente, sob o ponto de vista numérico-computacional, considerando-se as questões do balanceamento dinâmico de carga para a solução combinada para o escoamento e para o transporte de substâncias. Essa questão é imprescindível, pois a hidrodinâmica e o transporte escalar de substâncias formam um modelo multifísica, onde os resultados produzidos pelo modelo hidrodinâmico (velocidades e níveis de água) são dados de entrada para o modelo de transporte, que calcula a concentração das substâncias (bactérias coliformes)

Porém, o escoamento e o transporte têm diferentes escalas espaço-temporais de resolução, como destacado no anexo 1, tornando necessário a construção de uma malha mais refinada, para o transporte, que encaixada na malha mais grosseira, do escoamento. Essa malha é construída através de refinamento local empregando-se interpolações (h-método), e foi discutida na seção 3.6.1.

Note-se que os resultados apresentados neste trabalho referem-se somente à execução separada da hidrodinâmica e do transporte. Não se considerou aqui a questão da solução multifísica com balanceamento de carga, onde se executa a hidrodinâmica e o transporte escalar simultaneamente. Essa questão é tratada com a aplicação de algoritmos e estratégias de balanceamento dinâmico de carga, e pode ser visto na tese de Dorneles [DOR2002]

7.4 Métodos de decomposição de domínio

Uma classe de métodos matemáticos e computacionais que são empregados para resolver problemas pertencentes à CCP é aquela dos métodos de decomposição de domínio (MDD). Não obstante a vasta literatura, a expressão *decomposição de domínio* tem diferentes interpretações para diferentes especialistas. Não existe unanimidade na literatura técnica sobre a terminologia. Uma classificação proposta é a de Christara [CHR90], na qual a decomposição de domínio é a classe à qual pertencem os MDD sem sobreposição. O particionamento de domínio é a classe na qual o domínio é decomposto em subdomínios com sobreposição. O particionamento de dados é a classe dos métodos que usam o paralelismo no método de solução. Outra classificação, adotada neste trabalho é aquela de Smith, Bjorstad e Gropp [SMI96], que consiste em:

1. *decomposição de dados*: nessa abordagem, gera-se um único sistema de equações que é distribuído entre os processadores de modo que possa ser resolvido independentemente e em paralelo. Algumas vezes essa abordagem é, também, chamada de particionamento de dados;
2. *decomposição de domínio*: nessa abordagem, empregam-se métodos e técnicas para acoplar as soluções dos subproblemas gerados a partir da decomposição do domínio original. Desse modo, a solução do sistema de equações global é obtida pela combinação das soluções desses subproblemas.

Então, o particionamento (ou divisão) do domínio em subdomínios pertence ao escopo dos métodos de particionamento de malha, questão tratada na seção 7.3, enquanto a decomposição de domínio é um conjunto de técnicas e métodos para acoplar os diferentes subdomínios.

Esse é o mesmo ponto de vista de Chan e Mathew [CHA94], que consideram que métodos de decomposição de domínio são técnicas, para resolver EDPs, baseadas na decomposição do domínio espacial do problema em vários subdomínios. Essas técnicas devem assegurar que as subsoluções convirjam, dentro de uma dada acurácia, para a solução verdadeira do problema. É essa a caracterização dada neste trabalho aos MDD, particularmente aqueles de Schwarz que são MDD iterativos.

Um panorama completo sobre as diferentes abordagens em MDD pode ser visto em trabalhos como os de Chan e Mathew [CHA94], Smith, Bjorstad e Gropp [SMI96] e de Quarteroni e Valli [QUA99]. Uma síntese de alguns dos principais conceitos e aplicações de MDD é como feita em [MOU2001]. Nos trabalhos de Jinchao Xu [XUJ92], de Dryja e Widlund [DRY89], e de Tallec [TAL94], discutem-se o tratamento matematicamente rigoroso e abstrato dos MDD, sob o ponto de vista de métodos iterativos por decomposição de espaços e correção de subespaços (*iterative methods by space decomposition and subspace correction*). Porém, neste trabalho, a ênfase dada aos MDD é somente aos seus aspectos computacionais e numéricos.

Assim, na computação científica paralela, os MDD são particularmente atraentes, pois acoplam as diferentes partes de modo que, em geral, as comunicações ficam restritas à fronteira. Algumas das maiores motivações para o uso dos MDD são o fato de que seu emprego possibilita ([SMI92] e [GRP94]):

1. o uso de dados locais, necessitando de pouca comunicação global, a qual geralmente fica restrita à sincronização da solução;
2. uma completa flexibilidade no tratamento das soluções de EDPs definidas em regiões com geometria complexa e/ou que exibem comportamentos diferentes em diferentes partes do subdomínio;
3. a construção de pré-condicionadores que são empregados para acelerar métodos iterativos de solução do subespaço de Krylov.

Métodos de decomposição de domínio sem sobreposição não são abordados neste trabalho e podem ser vistos em [SMI96] e [BOL98], entre outros. MDD com sobreposição são empregados devido a sua simplicidade e ao fato de que podem ser usados como método de solução e como pré-condicionador.

No primeiro caso, os MDD são utilizados como um método iterativo de solução que resolve o problema de valor de contorno global a partir da combinação das soluções dos subproblemas de valor de contorno. É essa a abordagem do MDD de Schwarz discutido a seguir. No segundo caso, gera-se um pré-condicionador, que é distribuído por construção, de modo que pode ser empregado no passo de pré-condicionamento de um método iterativo.

Há toda uma teoria consolidada a respeito de métodos de decomposição de domínio aplicado à solução paralela de EDPs elípticas. Os resultados fornecem condições necessárias e suficientes nas quais os algoritmos são escaláveis. Ou seja, a solução não depende do número de subdomínios utilizados, nem do grau de sobreposição ou das características da malha original [SMI96], mas somente do estêncil computacional. Problemas hiperbólicos ou incompletamente parabólicos não têm a mesma base matemática, apesar dos bons resultados numéricos quando do emprego de estratégias semelhantes às das EDPs elípticas na obtenção de suas soluções. É sob esse ponto de vista que os métodos de decomposição de domínio são empregados neste trabalho para obter a solução paralela do modelo computacional aqui desenvolvido.

7.4.1 Métodos de Schwarz

O método alternante de Schwarz (MAS) foi desenvolvido em 1870 por H. A. Schwarz. É caracterizado pela decomposição do domínio limitado $\bar{\Omega} \subset \mathbb{R}^{n \leq 3}$ em

subdomínios sobrepostos $\bar{\Omega}_i$, tal que $\bar{\Omega} \subseteq \bigcup_{i=1}^N \bar{\Omega}_i$, onde $\bar{\Omega}_i = \Omega_i \cup \partial\Omega_i$ denota o fecho do domínio sendo $\partial\Omega_i$ a fronteira de Ω_i . As fronteiras artificialmente criadas pelas sobreposições são denotadas por $\Gamma_i = \partial\Omega_i \cap \Omega$, e a fronteira real é denotada por $\partial\Omega_i \setminus \Gamma_i$, que são os pontos de $\partial\Omega_i$ que não estão em Γ_i .

Para esclarecer essas notações, no caso de dois subdomínios, a região aberta é denotada por $\Omega = \Omega_1 \cup \Omega_2$, onde Ω_1 e Ω_2 são abertos, e não incluem as fronteiras. Uma representação desse caso é como mostrado pela fig. 7.3.

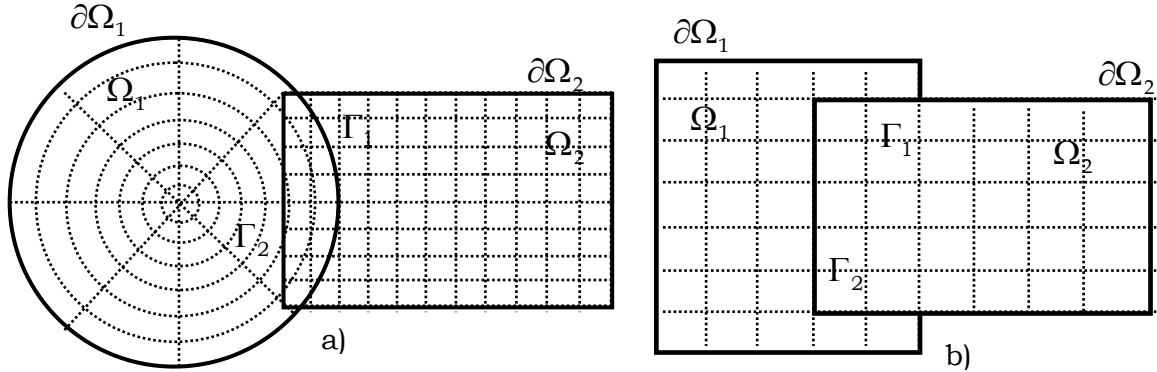


FIGURA 7.3 - Subdomínios Ω_1 e Ω_2 gerados pelo MAS. a) representa esses dois subdomínios não possuindo malhas emparelhadas; e b) representa esses dois subdomínios possuindo malhas emparelhadas.

Desse modo, o método alternante de Schwarz resolve iterativamente o problema elíptico definido em $\Omega = \Omega_1 \cup \Omega_2$

$$\begin{cases} L\varphi = f; \varphi \in \Omega \\ \varphi = g; \varphi \in \partial\Omega \end{cases}$$

encontrando a solução aproximada $\varphi_i^{(k)}$ em $\bar{\Omega}_i = \Omega_i \cup \partial\Omega_i$ depois de k iterações, resolvendo os dois subproblemas. Empregando as notações anteriores, e denotando $\varphi_2^{(k)}|_{\Gamma_1}$ e $\varphi_1^{(k)}|_{\Gamma_2}$ como sendo as restrições de $\varphi_2^{(k)}$ e de $\varphi_1^{(k)}$ às fronteiras Γ_1 e Γ_2 , respectivamente, um algoritmo para o MAS é como na fig. 7.4.

inicialização: escolher uma solução inicial $\varphi_2^{(1)}$ em Ω_2

iteração: Para $k=1,2,\dots,n$ com $\varphi_2^{(k)}$ conhecido, resolver $\varphi_1^{(k+1)}$ em Ω_1

$$\begin{cases} L\varphi_1^{(k+1)} = f; \varphi_1 \in \Omega_1 \\ \varphi_1^{(k+1)} = g_1; \varphi_1 \in \partial\Omega_1 \setminus \Gamma_1 \text{ e depois } u_2^{(k+1)} \text{ em } \Omega_2 \\ \varphi_1^{(k+1)} = \varphi_2^{(k)}|_{\Gamma_1}; \varphi_1 \in \Gamma_1 \end{cases} \begin{cases} L\varphi_2^{(k+1)} = f; \varphi_2 \in \Omega_2 \\ \varphi_2^{(k+1)} = g_2; \varphi_2 \in \partial\Omega_2 \setminus \Gamma_2 \\ \varphi_2^{(k+1)} = \varphi_1^{(k+1)}|_{\Gamma_2}; \varphi_2 \in \Gamma_2 \end{cases}$$

término: até que a solução convirja com um determinado critério de parada como, por

exemplo, $\|\varphi_2^{(k)}|_{\Gamma_1} - \varphi_2^{(k-1)}|_{\Gamma_1}\| \leq \varepsilon$ e $\|\varphi_1^{(k)}|_{\Gamma_2} - \varphi_1^{(k-1)}|_{\Gamma_2}\| \leq \varepsilon$

FIGURA 7.4 - Algoritmo alternante de Schwarz

Assim, o MAS resolve o problema global, estabelecendo determinadas condições de contorno nas fronteiras artificiais e usando as condições de contorno do problema

original nas fronteiras que coincidem com as fronteiras do domínio original. Note-se que podem surgir dificuldades adicionais para obter as sobreposições nas interfaces entre os subproblemas, como o caso em que não há emparelhamento nas submalhas (*nonmatching grids*), como exemplificado pela fig. 7.3. b), caso em que se necessita de procedimentos de interpolação.

Outras situações são como aquelas que consideram a solução de diferentes EDPs nos subdomínios ou que as condições de contorno estabelecidas entre os subdomínios não são as de Dirichlet ([SMI92] e [GOO99]). Nesses casos, deve-se estabelecer quais as condições de transmissão nessas interfaces [TAL94]. Algumas dessas CC podem ser obtidas a partir da CC de Robin, que é definida por:

$$\alpha\varphi_1^{(k+1)} + \beta \frac{\partial\varphi_1^{(k+1)}}{\partial n} \equiv \alpha\varphi_2^{(k)} + \beta \frac{\partial\varphi_2^{(k)}}{\partial n}$$

e, assim, quando $\alpha \neq 0$ e $\beta \equiv 0$, tem-se a CC de Dirichlet $\alpha\varphi_1^{(k+1)} = \alpha\varphi_2^{(k)}$, e quando $\alpha \equiv 0$ e $\beta \neq 0$, tem-se a CC de Neumann $\beta\left(\varphi_1^{(k+1)}\right)_n = \beta\left(\varphi_2^{(k)}\right)_n$.

Essas dificuldades não se apresentam neste trabalho, pois as malhas são emparelhadas e as EDPs são as mesmas em cada subdomínio. Nessa situação, nas fronteiras artificiais as condições de contorno são do tipo Dirichlet e não trazem dificuldades adicionais para o emprego de MDD de Schwarz.

Visando a estabelecer uma notação similar e concisa para o MAS, e para os demais MDD com sobreposição que podem ser dele derivados, o método alternante pode ser escrito empregando uma formulação matricial como em [BOU2000], [LIE2001] e [ALO2002]. Especificamente, denota-se, respectivamente, $\varphi_{\Omega_i}^{(k)}$ a solução aproximada em Ω_i ; $\varphi_{\partial\Omega_i \setminus \Gamma_i}^{(k)}$ a solução nas fronteiras reais; $\varphi_{\Gamma_i}^{(k)}$ a solução nas fronteiras artificiais, depois de k iterações, e A_i a forma discreta do operador diferencial L restrita a Ω_i , onde A_{Ω_i} , $A_{\Omega \setminus \Gamma_i}$ e A_{Γ_i} denotam, respectivamente, os pontos interiores, os pontos das fronteiras verdadeiras, e os pontos da fronteira artificial. Então, o problema contínuo $L\varphi = f$, denotado na forma discreta (matricial) por $A\varphi = b$, é escrito, para malhas emparelhadas, como:

$$\left(A_{\Omega_i} \ A_{\partial\Omega_i \setminus \Gamma_i} \ A_{\Gamma_i} \right) \varphi_i^{(k+1)} \equiv \begin{pmatrix} \varphi_{\Omega_i}^{(k+1)} \\ \varphi_{\partial\Omega_i \setminus \Gamma_i}^{(k+1)} \\ \varphi_{\Gamma_i}^{(k+1)} \end{pmatrix} = b_i$$

e, assim, o MAS é expresso, para i subdomínios, na forma de:

$$A_{\Omega_i} \varphi_{\Omega_i}^{(k+1)} = b_i - \underbrace{A_{\partial\Omega_i \setminus \Gamma_i} \varphi_{\partial\Omega_i \setminus \Gamma_i}^{(k+1)}}_{CC \text{ real}} - \underbrace{A_{\Gamma_i} \varphi_{\Gamma_i}^{(k)}}_{CC \text{ artificial}} = b_i - \underbrace{A_{\partial\Omega_i \setminus \Gamma_i} g_i}_{CC \text{ Dirichlet}} - \underbrace{A_{\Gamma_i} \varphi_{\Gamma_j}^{(k)}}_{j \neq i}$$

ou seja:

$$\varphi_{\Omega_i}^{(k+1)} = \varphi_{\Omega_i}^{(k)} + A_{\Omega_i}^{-1} \left(b_i - A_{\Omega_i} \varphi_{\Omega_i}^{(k)} - A_{\partial\Omega_i \setminus \Gamma_i} g_i - A_{\Gamma_i} \varphi_{\Omega_j}^{(k)} \right)$$

assim, quando $i=1,2$, tem-se:

$$\varphi_{\Omega_1}^{(k+1)} = \varphi_{\Omega_1}^{(k)} + A_{\Omega_1}^{-1} \left(b_1 - A_{\Omega_1} \varphi_{\Omega_1}^{(k)} - A_{\partial\Omega_1 \setminus \Gamma_1} g_1 - A_{\Gamma_1} \varphi_{\Omega_2}^{(k)} \right)$$

e

$$\varphi_{\Omega_2}^{(k+1)} = \varphi_{\Omega_2}^{(k)} + A_{\Omega_2}^{-1} \left(b_2 - A_{\Omega_2} \varphi_{\Omega_2}^{(k)} - A_{\partial\Omega_2 \setminus \Gamma_2} g_2 - A_{\Gamma_2} \varphi_{\Omega_1}^{(k+1)} \right)$$

Note-se que, quando as malhas não são emparelhadas, como aquelas mostradas na fig. 7.3 b), é necessário interpolar os valores nodais no interior de Ω_i para os nodos na curva Γ_j através do operador discreto $I_{\Omega_i \rightarrow \Gamma_j}$, e nesse caso, $A_{\Gamma_j} \varphi_{\Omega_i}^{(k)}$ é substituído por $A_{\Gamma_j} I_{\Omega_i \rightarrow \Gamma_j} \varphi_{\Omega_i}^{(k)}$.

Usando essas notações pode-se rescrever o MAS (e os outros MDD) de modo matricial, observando-se ainda as seguintes definições

$$\varphi_{\Omega_1}^{(k)} \equiv R_1 \varphi_i^{(k)} \Leftrightarrow (I \ 0) \begin{pmatrix} \varphi_{\Omega_1}^{(k)} \\ \varphi_{\Omega \setminus \Omega_1}^{(k)} \end{pmatrix} = \varphi_{\Omega_1}^{(k)}, \quad \varphi_{\Omega_2}^{(k)} \equiv R_2 \varphi_i^{(k)} \Leftrightarrow (0 \ I) \begin{pmatrix} \varphi_{\Omega \setminus \Omega_2}^{(k)} \\ \varphi_{\Omega_2}^{(k)} \end{pmatrix} = \varphi_{\Omega_2}^{(k)}$$

$$A_{\Omega_i} \equiv R_i A R_i^T \Leftrightarrow (I \ 0) \begin{pmatrix} A_{\Omega_i} & A_{\Gamma_i} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} I \\ 0 \end{pmatrix} \Leftrightarrow (I \ 0) \begin{pmatrix} A_{\Omega_i} \\ 0 \end{pmatrix} = A_{\Omega_i}$$

$$M^{-1} \equiv R_i^T (R_i A R_i^T)^{-1} R_i \Leftrightarrow R_i^T (R_i^{-T} A^{-1} R_i^{-1}) R_i = A^{-1}$$

$$M_i \equiv R_i A R_i^T$$

notando que, em implementações práticas, R_i^T e R_i não são formados explicitamente e fazem apenas papel de operadores (matrizes) de restrição R_i e de extensão R_i^T , de forma a expressar os algoritmos de modo formal, similar e conciso como em [SMI92], [CHA94] e [SMI96].

Assim, R_i denota a matriz, que, aplicada ao vetor solução, retorna os valores associados com determinados nodos (restrição), e R_i^T mapeia os valores à matriz global A (extensão). Matematicamente, sob o ponto de vista de subespaços de correção, M_i restringe o resíduo a um determinado subdomínio, no qual o problema é resolvido localmente através de uma correção, e estende a solução para todo o subdomínio. Portanto, os valores de uma submatriz A_i , associada a uma sub-região Ω_i , são obtidos pela operação $A_i = R_i A R_i^T$ com $1 \leq i \leq N$ subproblemas. Especificamente, pode-se escrever que:

$$\begin{cases} A_{\Omega_i} \varphi_{\Omega_i}^{(k+1)} = b_i; \quad \varphi_i^{(k+1)} \in \Omega_i \\ \varphi_{\partial\Omega_i \setminus \Gamma_i}^{(k+1)} = g_i; \quad \varphi_{\partial\Omega_i \setminus \Gamma_i}^{(k+1)} \in \partial\Omega_i \setminus \Gamma_i \\ \varphi_{\Gamma_i}^{(k+1)} = \varphi_{\Omega_{j \neq i}}^{(k)}; \quad \varphi_{\Gamma_i}^{(k+1)} \in \Gamma_i \end{cases} \Leftrightarrow A_i \varphi_i^{(k+1)} = b_i \equiv (A_{\Omega_i} \ A_{\partial\Omega_i \setminus \Gamma_i} \ A_{\Gamma_i}) \begin{pmatrix} \varphi_{\Omega_i}^{(k+1)} \\ \varphi_{\partial\Omega_i \setminus \Gamma_i}^{(k+1)} \\ \varphi_{\Gamma_i}^{(k+1)} \end{pmatrix} = b_i$$

e, então, considerando que nas fronteiras reais as CC são conhecidas, pode-se eliminá-las agregando-as ao vetor dos termos independentes, ou seja, fazendo:

$$A_{\Omega_i} \varphi_{\Omega_i}^{(k+1)} + \underbrace{A_{\partial\Omega_i \setminus \Gamma_i} \varphi_{\partial\Omega_i \setminus \Gamma_i}^{(k+1)}}_{CC \text{ Dirichlet}} + \underbrace{A_{\Gamma_i} \varphi_{\Gamma_{j \neq i}}^{(k+1)}}_{\varphi_{\Gamma_{j \neq i}}^{(k)} = \varphi_{\Omega_{j \neq i}}^{(k)}} = b_i \Rightarrow \begin{cases} A_{\Omega_i} \varphi_{\Omega_i}^{(k+1)} = b_i - \underbrace{A_{\partial\Omega_i \setminus \Gamma_i} g_i}_{\tilde{b}_i} - \underbrace{A_{\Gamma_i} \varphi_{\Gamma_{j \neq i}}^{(k+1)}}_{CC \text{ artificiais}} \\ = \tilde{b}_i - A_{\Gamma_i} \varphi_{\Gamma_{j \neq i}}^{(k+1)} \end{cases}$$

Com essa expressão escreve-se o método alternante de Schwarz, no caso de dois subdomínios e considerando que nas fronteiras artificiais g_i é conhecido, como:

$$A_{\Omega_1} \varphi_{\Omega_1}^{(k+1)} + \underbrace{A_{\partial\Omega_1 \setminus \Gamma_1} \varphi_{\partial\Omega_1 \setminus \Gamma_1}^{(k+1)}}_{CC \text{ Dirichlet} = g_1} + \underbrace{A_{\Gamma_1} \varphi_{\Gamma_1}^{(k+1)}}_{\varphi_{\Gamma_1}^{(k+1)} = \varphi_{\Omega_2|_{\Gamma_1}}^{(k)}} = b_1 \Rightarrow \begin{cases} A_{\Omega_1} \varphi_{\Omega_1}^{(k+1)} = b_1 - \underbrace{A_{\partial\Omega_1 \setminus \Gamma_1} g_1}_{\tilde{b}_1} - A_{\Gamma_1} \varphi_{\Omega_2}^{(k)} \\ = \tilde{b}_1 - A_{\Gamma_1} \varphi_{\Omega_2}^{(k)} \end{cases}$$

$$A_{\Omega_2} \varphi_{\Omega_2}^{(k+1)} + \underbrace{A_{\partial\Omega_2 \setminus \Gamma_2} \varphi_{\partial\Omega_2 \setminus \Gamma_2}^{(k+1)}}_{CC \text{ Dirichlet} = g_2} + \underbrace{A_{\Gamma_2} \varphi_{\Gamma_2}^{(k+1)}}_{\varphi_{\Gamma_2}^{(k+1)} = \varphi_{\Omega_1|_2}^{(k+1)}} = b_2 \Rightarrow \begin{cases} A_{\Omega_2} \varphi_{\Omega_2}^{(k+1)} = \underbrace{b_2 - A_{\partial\Omega_2 \setminus \Gamma_2} g_2}_{\tilde{b}_2} - A_{\Gamma_2} \varphi_{\Omega_1}^{(k+1)} \\ = \tilde{b}_2 - A_{\Gamma_2} \varphi_{\Omega_1}^{(k+1)} \end{cases}$$

que pode ser visto como sendo o método de Gauss-Seidel em blocos aplicado ao sistema de equações, pois cada subdomínio usa como condição de contorno as soluções anteriormente obtidas. Esse sistema, cuja dimensão é maior que N , pois os valores de φ em $\Omega_1 \cap \Omega_2 \neq \emptyset$ são duplicados em φ_{Ω_1} e φ_{Ω_2} , é escrito como:

$$\begin{pmatrix} A_{\Omega_1} & A_{\Gamma_1} \\ A_{\Gamma_2} & A_{\Omega_2} \end{pmatrix} \begin{pmatrix} \varphi_{\Omega_1} \\ \varphi_{\Omega_2} \end{pmatrix} = \begin{pmatrix} \tilde{b}_1 \\ \tilde{b}_2 \end{pmatrix}$$

Usando essas terminologias, pode-se obter uma variante do método alternante de Schwarz escrevendo-o em dois meios-passos. Isso é obtido através da introdução de um passo intermediário $k+\frac{1}{2}$, e essa abordagem fornece a expressão:

$$\left. \begin{aligned} \varphi^{(k+\frac{1}{2})} &= \varphi^{(k)} + \begin{pmatrix} A_{\Omega_1}^{-1} & 0 \\ 0 & 0 \end{pmatrix} (b - A\varphi^{(k)}) \\ \varphi^{(k+1)} &= \varphi^{(k+\frac{1}{2})} + \begin{pmatrix} 0 & 0 \\ 0 & A_{\Omega_2}^{-1} \end{pmatrix} (b - A\varphi^{(k+\frac{1}{2})}) \end{aligned} \right\}$$

que pode ser escrita, substituindo a solução $\varphi^{(k+\frac{1}{2})}$ do primeiro meio-passo na solução $\varphi^{(k+\frac{1}{2})}$ e no resíduo $b - A\varphi^{(k+\frac{1}{2})}$ do segundo meio-passo, como:

$$\varphi^{(k+1)} = \varphi^{(k)} + \begin{pmatrix} A_{\Omega_1}^{-1} & 0 \\ 0 & 0 \end{pmatrix} (b - A\varphi^{(k)}) + \begin{pmatrix} 0 & 0 \\ 0 & A_{\Omega_2}^{-1} \end{pmatrix} \left[b - A \begin{pmatrix} A_{\Omega_1}^{-1} & 0 \\ 0 & 0 \end{pmatrix} (b - A\varphi^{(k)}) \right]$$

que é escrito, usando as definições anteriores, como

$$\varphi^{(k+1)} = \varphi^{(k)} + M_1^{-1} (b - A\varphi^{(k)}) + M_2^{-1} (b - A\varphi^{(k)}) - M_2^{-1} A M_1^{-1} (b - A\varphi^{(k)})$$

isto é:

$$\varphi^{(k+1)} = \varphi^{(k)} + \left(M_1^{-1} + M_2^{-1} - M_2^{-1} A M_1^{-1} \right) (b - A\varphi^{(k)}) \quad (7.1)$$

A expressão (7.1) é uma das variantes do método alternante de Schwarz e é chamada de MDD multiplicativo de Schwarz (MMS), no qual pré-condicionador dado pelo método de Schwarz é $M^{-1} = M_1^{-1} + M_2^{-1} - M_2^{-1} A M_1^{-1}$. Definindo o erro de $\varphi^{(k)}$ por $e^{(k)} = \varphi^e - \varphi^{(k)}$, onde φ^e é a solução exata, tem-se:

$$\begin{aligned} e^{(k+1)} &= e^{(k)} - \left(M_1^{-1} + M_2^{-1} - M_2^{-1} A M_1^{-1} \right) A e^{(k)} \\ &= \left[I - \left(M_1^{-1} + M_2^{-1} - M_2^{-1} A M_1^{-1} \right) A \right] e^{(k)} = \left(I - M_2^{-1} A \right) \left(I - M_1^{-1} A \right) e^{(k)} \end{aligned}$$

e, portanto, a matriz de iteração é um produto de duas matrizes, de onde vem a designação do método *multiplicativo* de Schwarz.

Note-se que o MMS é inerentemente seqüencial e, sendo assim, o paralelismo pode ser obtido no MMS introduzindo métodos multicor. Além disso, a matriz pré-condicionadora não é simétrica e, portanto, não pode ser empregada como um pré-condicionador para o algoritmo do gradiente conjugado, mesmo se A é SDP. Porém, tal matriz pré-condicionadora pode tornar-se SDP quando se adiciona mais um passo intermediário, como pode ser visto em [ALO2002].

Uma outra versão do método alternante de Schwarz, que considera atualizações simultâneas nas fronteiras é o chamado método aditivo de Schwarz (MAAS). Esse método considera que o passo intermediário $k+\frac{1}{2}$ seja substituído pelo passo anterior k na formação do resíduo no segundo meio-passo. Desse modo, o MAAS é escrito como:

$$\left. \begin{aligned} \varphi^{(k+\frac{1}{2})} &= \varphi^{(k)} + M_1^{-1} (b - A\varphi^{(k)}) \\ \varphi^{(k+1)} &= \varphi^{(k+\frac{1}{2})} + M_2^{-1} (b - A\varphi^{(k)}) \end{aligned} \right\}$$

ou seja, substituindo a solução $\varphi^{(k+\frac{1}{2})}$ do primeiro meio-passo na solução $\varphi^{(k+\frac{1}{2})}$ do segundo meio-passo, tem-se:

$$\varphi^{(k+1)} = \varphi^{(k)} + (M_1^{-1} + M_2^{-1})(b - A\varphi^{(k)})$$

que é a forma do MAAS escrito para dois subdomínios, na qual o pré-condicionador dado pelo método de Schwarz é $M^{-1} = M_1^{-1} + M_2^{-1}$, que é simétrico e pode ser empregado diretamente no algoritmo do GC. Note-se que esse método pode ser visto como o método iterativo de Jacobi com sobreposição em blocos, pois todos os subdomínios usam a solução da última iteração como condição de contorno.

A matriz pré-condicionadora pode ser generalizada para N subdomínios, que podem ser resolvidos de forma independente e, portanto, gerando a expressão $M^{-1} = M_1^{-1} + \dots + M_N^{-1}$. Com isso, o método aditivo de Schwarz para N subdomínios pode ser escrito, para $M_i^{-1} = R_i^T A_i^{-1} R_i$, como em (7.2):

$$\varphi^{(k+1)} = \varphi^{(k)} + M_{MAAS}^{-1} \equiv \varphi^{(k)} + \sum_{i=1}^N M_i^{-1} (b - A\varphi^{(k)}) \quad (7.2)$$

Assim, os algoritmos seqüenciais do MDD multiplicativo de Schwarz (MMS) e do método aditivo de Schwarz (MAAS) resolvem o problema global (iterativamente), decompondo-o em subdomínios. O algoritmo para o MMS não é aqui apresentado, mas o algoritmo do MAAS é como apresentado na fig. 7.5.

Inicialização: $k \leftarrow 1, \varphi^{(1)} \leftarrow 0, \varepsilon$

Enquanto $\|b - A\varphi^{(k)}\| \geq \varepsilon$ *faça*

1. $\varphi^{(k+1)} \leftarrow \varphi^{(k)} + (M_1^{-1} + M_2^{-1})(b - A\varphi^{(k)})$
2. $k \leftarrow k + 1$

FIGURA 7.5 - Algoritmo do MDD aditivo de Schwarz

Neste trabalho implementou-se o MAAS no qual todos os subdomínios usam a solução da última iteração como CC, de modo que cada um dos subdomínios pode ser resolvido independentemente, ficando as comunicações restritas às fronteiras ([CHA94], [SMI96] e [PAG97]). Outra solução paralela desenvolvida e implementada é aquela na qual o MAAS é visto como um pré-condicionador.

7.4.2 Método aditivo de Schwarz: solução paralela

A primeira consequência da formulação do método aditivo de Schwarz (MAAS) é que a expressão (7.2) tem um paralelismo ótimo, no sentido de que a troca de dados e o sincronismo são restritos às fronteiras dos subdomínios adjacentes. Assim, o MAAS resolve o problema global em paralelo obtendo sua solução de modo iterativo e de modo independente nos k subdomínios. A troca de dados ocorre entre as fronteiras a

cada ciclo do método de solução entre esses subdomínios, até que se cumpra o critério de convergência global, de modo que a combinação das soluções locais gere a solução global, dentro de uma acurácia.

Para o método aditivo de Schwarz com sobreposição pode-se mostrar, sob a hipótese de que $\Omega_i \cap \Omega_j \cap \Omega_k \neq \emptyset$, $\forall i \neq j \neq k$, que o algoritmo converge. Além disso, a presença de regiões sobrepostas assegura a continuidade da solução $\varphi_i^{(k+1)}$ e de suas derivadas entre os subdomínios [DEB98]. O algoritmo do MAAS aplicado à solução paralela é como mostrado na fig. 7.6, onde não se destaca os passos de comunicação, necessários em uma implementação paralela.

Inicialização: escolha uma solução inicial $\varphi_i^{(1)}$ em cada subdomínio Ω_i ;

Iteração: para $k \geq 1$, com $\varphi_i^{(k)}$ conhecido, calcular $\varphi_i^{(k+1)}$ tal que:

1. $A\varphi_i^{(k+1)} = b$ em Ω_i ;
2. $\varphi_i^{(k+1)} = g_i$ em $\partial\Omega_i \setminus \Gamma_i$;
3. $\varphi_i^{(k+1)} = \varphi_j^{(k)}$ em $\Gamma_j = \partial\Omega \cap \Omega_j$, para $1 \leq j \leq N$, tal que $\Omega_i \cap \Omega_j \neq \emptyset$

Término: até que $\sup\left(\left\|\varphi_j^{(k+1)} - \varphi_j^{(k)}\right\|_2 / \left\|\varphi_j^{(k)}\right\|_2\right) \leq \varepsilon$, onde ε é a acurácia desejada.

FIGURA 7.6 - Algoritmo aditivo de Schwarz (em paralelo)

Para exemplificar esses conceitos, notações e algoritmos, toma-se como exemplo um domínio discreto particionado em quatro subdomínios. Nesse caso, pode-se tomar uma sobreposição de uma ou de duas células adjacentes as fronteiras, dado o estêncil de 5-pontos utilizado na discretização, como em [CHF94] e [GOO96]. Esses subproblemas podem ser escritos, usando as notações anteriormente introduzidas, como:

$$\left\{ \begin{array}{l} A\varphi_1^{(k+1)} = f; \varphi_1 \in \Omega_1 \\ \varphi_1^{(k+1)} = g_1; \varphi_1 \in \partial\Omega_1 \setminus \Gamma_1 \\ \varphi_1^{(k+1)} = \varphi_2^{(k)}|_{\Gamma_2}; \varphi_2 \in \Gamma_2 \\ \varphi_1^{(k+1)} = \varphi_4^{(k)}|_{\Gamma_4}; \varphi_4 \in \Gamma_4 \end{array} \right\}, \left\{ \begin{array}{l} A\varphi_2^{(k+1)} = f; \varphi_2 \in \Omega_2 \\ \varphi_2^{(k+1)} = g_2; \varphi_2 \in \partial\Omega_2 \setminus \Gamma_2 \\ \varphi_2^{(k+1)} = \varphi_1^{(k)}|_{\Gamma_1}; \varphi_1 \in \Gamma_1 \\ \varphi_2^{(k+1)} = \varphi_3^{(k)}|_{\Gamma_3}; \varphi_3 \in \Gamma_3 \end{array} \right\},$$

$$\left\{ \begin{array}{l} A\varphi_4^{(k+1)} = f; \varphi_4 \in \Omega_4 \\ \varphi_4^{(k+1)} = g_4; \varphi_4 \in \partial\Omega_4 \setminus \Gamma_4 \\ \varphi_4^{(k+1)} = \varphi_1^{(k)}|_{\Gamma_1}; \varphi_1 \in \Gamma_1 \\ \varphi_4^{(k+1)} = \varphi_3^{(k)}|_{\Gamma_3}; \varphi_3 \in \Gamma_3 \end{array} \right\} \text{ e } \left\{ \begin{array}{l} A\varphi_3^{(k+1)} = f; \varphi_3 \in \Omega_3 \\ \varphi_3^{(k+1)} = g_3; \varphi_3 \in \partial\Omega_3 \setminus \Gamma_3 \\ \varphi_3^{(k+1)} = \varphi_2^{(k)}|_{\Gamma_2}; \varphi_2 \in \Gamma_2 \\ \varphi_3^{(k+1)} = \varphi_4^{(k)}|_{\Gamma_4}; \varphi_4 \in \Gamma_4 \end{array} \right\}$$

onde cada submatriz A_i é a matriz da região Ω_i , e é gerada quando se impõem CCs de Dirichlet homogêneas nas fronteiras artificiais (sobreposições) Γ_i , ou condições de contorno corretas quando a fronteira do subdomínio coincidir com uma fronteira do domínio (fronteira real) ([SLV97] e [PAG97]). Assim, os sub-problemas são problemas de valor de contorno bem postos, no sentido matemático, pois as CCs assim estabelecidas os definem como tal. A fig. 7.7 mostra uma representação para um domínio particionado em quatro subdomínios, com 8×8 pontos com ordenação natural, na qual se vêem as fronteiras artificiais geradas pelas sobreposições.

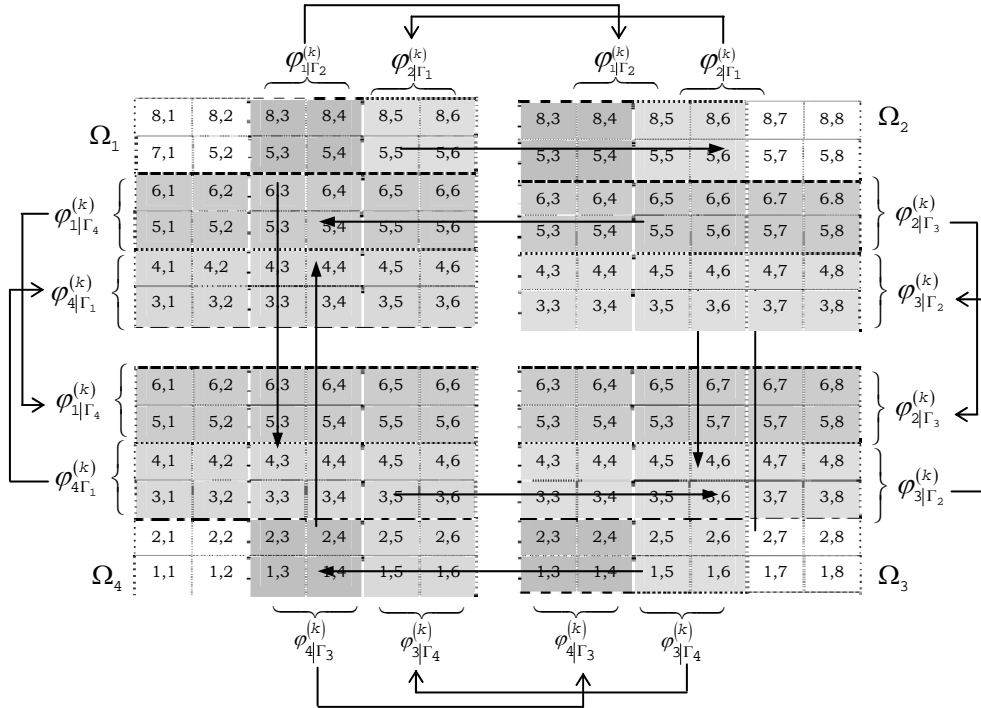


FIGURA 7.7 - Domínio particionado em quatro subdomínios, com CC tipo Dirichlet homogêneas. A quantidade de células na sobreposição depende do estêncil computacional utilizado na construção do esquema numérico

As fig. 7.8 e 7.9 mostram, respectivamente, o *speedup* e a eficiência para o domínio computacional particionado em 23 processos usando a biblioteca METIS, e resolvido empregando o algoritmo do método aditivo de Schwarz mostrado na fig. 7.6. Os resultados indicam que a solução paralela implementada é escalável, pelo menos até 23 nodos, sendo que as anomalias registradas nos processadores 11 e 13 referem-se a má convergência do método aditivo de Schwarz especificamente para esses processos, e/ou ao particionamento não apropriadamente efetivo oferecido pela biblioteca METIS. Esses resultados foram obtidos usando o arquivo Guaiba_100, que é aquele arquivo onde se toma os espaçamentos horizontais como $\Delta x = \Delta y = 100m$, e com apenas uma camada.

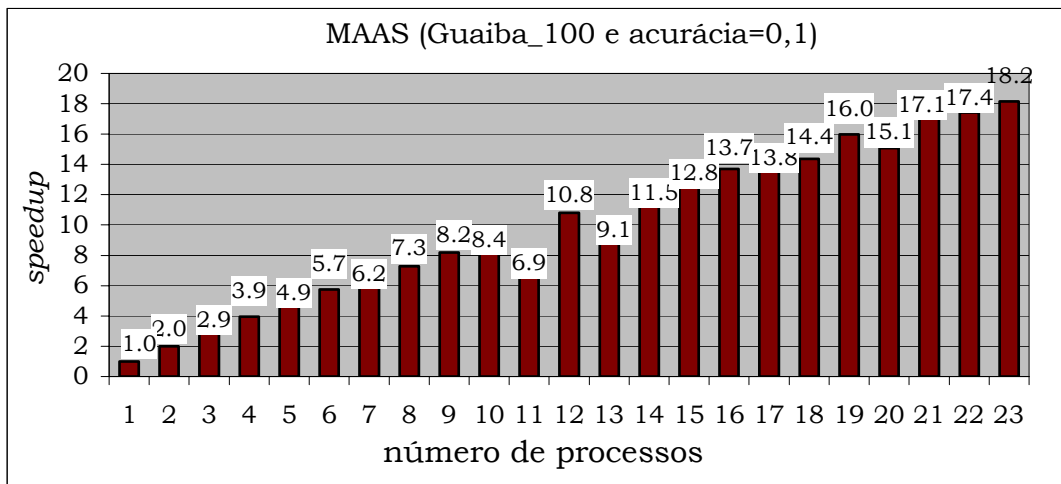


FIGURA 7.8 - *Speedup* \times número de processos empregando o algoritmo do MAAS

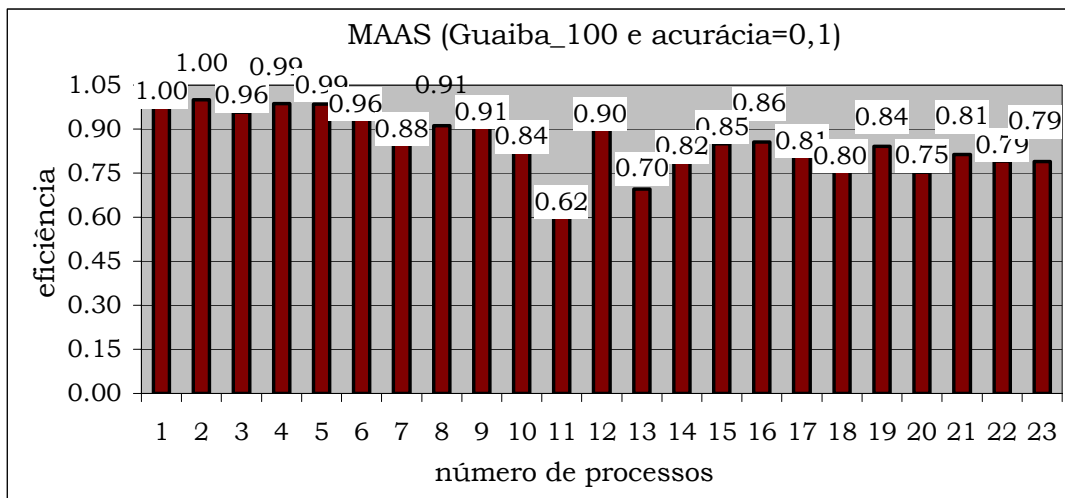


FIGURA 7.9 - Eficiência × número de processos empregando o algoritmo do MAAS

Note-se, finalmente, que no subdomínio ilustrado na fig. 3.12, destacaram-se as enumerações e identificações das células internas; das células que são adjacentes internas às fronteiras do subdomínio (e que formam o chamado “estêncil interno”); e das células que são adjacentes internas às fronteiras dos outros subdomínios (e que formam o chamado “estêncil externo”). Essa identificação e caracterização das células é indispensável para o emprego do método aditivo de Schwarz sob duas abordagens.

Na primeira abordagem emprega-se o algoritmo mostrado na fig. 7.6 para obter a solução paralela via decomposição de domínio. Nesse caso as células que fazem o papel dos estêncis interno e externo compõem as sobreposições que, por sua vez, necessitam de estêncis para que valores válidos estejam definidos. A fig. 7.7 ilustra essa primeira abordagem, resultado da aplicação do algoritmo detalhado na fig. 7.6. Na segunda abordagem, usa-se o método de decomposição de domínio como pré-condicionador, que já é gerado de modo distribuído. Essa segunda opção gera o método Krylov-Schwarz de solução, que é um método de Schwarz acelerado por um método de Krylov, como discutido nas seções 7.6.2 e 7.8.

7.5 Métodos de solução para sistemas de equações lineares

Os sistemas de equações são resolvidos empregando-se métodos diretos ou métodos iterativos. Os métodos diretos resolvem-nos em um número finito e conhecido de passos, e os erros que ocorrem devem-se à aritmética de ponto flutuante. Os métodos iterativos resolvem os sistemas de equações através de aproximações sucessivas da solução, e os erros são devidos às iterações e aos arredondamentos da aritmética de ponto flutuante. Ao contrário dos métodos diretos, que resolvem o sistema de equações com um número pré-definido de passos, os métodos iterativos calculam a solução até que um critério de parada seja alcançado. Esse critério pode ser o número máximo de iterações ou a tolerância máxima de erro.

A escolha por esse ou aquele tipo de método é induzida pelo porte do problema a ser resolvido. Muitos dos sistemas de equações resolvidos atualmente são da ordem de milhares ou milhões de incógnitas nodais e a solução via métodos diretos equivale a calcular a inversa das matrizes dos coeficientes, em forma exata, tornando essa abordagem impraticável, por problemas de armazenamento e pela dependência de operações e/ou de dados que dificulta a sua paralelização.

A eliminação Gaussiana, exemplo canônico dos métodos diretos, resolve os sistemas matriciais através de seqüências de operações elementares entre linhas e/ou colunas. Esse procedimento leva ao processo de *fill-in*, que é o surgimento de elementos

não nulos em posições onde não existiam originalmente, destruindo a esparsidade da matriz, o que dificulta o desenvolvimento e implementação dos algoritmos de armazenamento e de solução.

Note-se que muitas aplicações realísticas, particularmente aquelas pertencentes ao escopo da CFD, resultam, quando discretizadas, em sistemas de equações altamente esparsos. A densidade dos elementos não nulos das matrizes decrescente com o porte do problema. Por exemplo, matrizes de problemas 2D de ordem 40×40 têm 256×10^4 elementos na matriz, mas apenas 8×10^3 (0,312%) são não nulos. Essa esparsidade deve ser aproveitada na otimização do armazenamento [STR94].

A abordagem mais apropriada é, portanto, aquela que emprega métodos iterativos que sejam robustos e computacionalmente eficientes. E a classe mais efetiva é a dos subespaços de Krylov, sendo que a eficiência decorre do fato de que os métodos pertencentes a essa classe têm boas propriedades de convergência, sobretudo quando as matrizes são bem condicionadas [ARA97a]. Além disso, são construídos sobre operações básicas em álgebra linear e, portanto, são altamente paralelizáveis. Dois dos métodos mais populares dessa classe são o gradiente conjugado (GC) e o resíduo mínimo generalizado (GMRES).

7.5.1 Métodos iterativos do subespaço de Krylov

Dada a potencialidade dos métodos iterativos quanto a otimização, armazenamento, eficiência computacional e grau de paralelismo, eles são os mais populares para a solução de sistemas de equações esparsos e de grande porte. A classe dos métodos do subespaço de Krylov é também chamada de métodos não estacionários, devido ao fato de que passos da iteração atual dependem da iteração anterior. Van der Vorst [VOR94a] traça um panorama do desenvolvimento, sob uma perspectiva histórica, dessa classe, e referências são [BAR94], [KEL95], [GOL96], [AXE96], [SAA96] e [VAR99], entre outras.

Porém, a despeito de muitos nomes e acrônimos que proliferam em subespaços de Krylov, os algoritmos dessa classe são formulados sobre poucas idéias fundamentais [ARA97]. Na solução de $A\varphi = b$, tais métodos encontram uma solução aproximada φ^m para um subespaço m – dimensional, tal que:

$$\varphi^0 + K^m$$

em que φ^0 é a aproximação inicial, impondo a condição Petrov-Galerkin:

$$b - A\varphi^m \perp L^m$$

em que L^m é um subespaço de dimensão m , e $K = K(A, r^0)$ é o subespaço de Krylov definido, para $r^0 = b - A\varphi^0$, por:

$$K^m(A, r^0) = \text{span}(r^0, Ar^0, A^2r^0, \dots, A^{m-1}r^0)$$

Os métodos que decorrem dessas idéias podem ser agrupados em três classes, dependendo de como o subespaço L^m é escolhido ([SAA96] e [CHO97]):

1. $L^m = K^m$: é a classe do método de projeção ortogonal ou de Galerkin. Quando A é SDP, essa escolha de L^m minimiza a A – norma do erro. Exemplos são o GC, no caso simétrico e o FOM (e equivalentes como o ORTHORES e o método de Projeção de Axelsson) no caso não simétrico;
2. $L^m = AK^m$: é a classe do método dos resíduos mínimos. Essa escolha de L^m minimiza a norma residual $\|b - A\varphi^m\|_2$. Exemplos são o GMRES e seus equivalentes (por exemplo, o GCR);

3. $L^m = K^m(A^T, r_0^*)$ onde r_0^* é um vetor não paralelo a r_0 . Esta escolha de L^m é projetada para A não simétrica e fornece relações de recorrência para bases do subespaço de Krylov. Métodos dessa classe podem necessitar de menos armazenagem que os algoritmos do GMRES ou do FOM, mas são mais propensos a parar (*breakdown*). Exemplos são o BCG, CGS.

A escolha por este ou aquele método iterativo é decorrente da natureza das matrizes. Neste trabalho, as matrizes dos sistemas de equações lineares para as ESW são simétricas definidas e positivas (SDP). As matrizes dos sistemas de equações para a ETM são não simétricas. A escolha recaiu no emprego dos algoritmos do gradiente conjugado e do resíduo mínimo generalizado, uma vez que a literatura técnica, e nossas próprias experiências numéricas, mostram que esses algoritmos são eficientes, robustos e possuem alto grau de paralelismo.

Outra classe que está se popularizando é aquela dos métodos Newton-Krylov-Schwarz (NKS), que decorrem da combinação de métodos não-lineares tipo Newton-Krylov e de métodos lineares tipo Krylov-Schwarz. O primeiro faz uma aproximação, via método iterativo do subespaço de Krylov, para uma das etapas do algoritmo de Newton. O segundo fornece um pré-condicionador tipo Schwarz para acelerar o método iterativo ([KEY95], [KEY97], [CRU97], [GRP98] e [DUF99]). Essa classe não foi empregada dado que os sistemas de equações são lineares.

7.6 Pré-condicionamento

Para acelerar a convergência de um método iterativo, pode-se pré-condicionar a matriz, já que a taxa de convergência de um método iterativo depende da distribuição dos autovalores da matriz dos coeficientes do sistema de equações lineares. A taxa pode ser melhorada, transformando o sistema original $A\varphi = b$ em um outro na forma $MA\varphi = Mb$, onde M é a matriz pré-condicionadora. O sistema de equações resultante é equivalente ao original, no sentido de que tem mesma solução, com melhor taxa de convergência por controlar os autovalores pequenos ou grandes [CHO97], resultando em melhor *número de condição*. Para matrizes SDP, isso significa que os seus autovalores devem estar agrupados próximos à unidade. O caso de matrizes não simétricas é mais complicado, e não há resultados gerais sobre a convergência a partir da análise dos autovalores.

O método de pré-condicionamento que explora somente as informações da matriz dos coeficientes para acelerar a convergência da solução do sistema de equações é dito pré-condicionador algébrico. Aquele método que considera as características físicas ou geométricas é dito pré-condicionador físico ou geométrico. Neste trabalho, utilizou-se somente a primeira categoria, visto que são de propósito geral e podem ser aplicados a uma grande classe de problemas. São robustos, mas não ótimos como os obtidos por técnicas específicas, os quais, porém, só podem ser aplicados aos problemas que os originaram [BEN99].

Uma classificação para os pré-condicionadores é aquela que considera o lado do pré-condicionamento, de modo que se tem os tipos: à esquerda $M^{-1}A\varphi = M^{-1}b$; à direita $AMM^{-1}\varphi = b$; e à esquerda e à direita $M^{-1}AN^{-1}N\varphi = M^{-1}b$, onde M e N representam transformações simples ou complexas da matriz original A . O tipo do pré-condicionador a ser usado depende da escolha do método iterativo. Em métodos de minimização dos resíduos, como o GMRES, o pré-condicionamento à direita pode ser mais efetivo pois, em aritmética exata, os resíduos para sistemas de equações pré-condicionados à direita são idênticos aos resíduos verdadeiros. Mendelsohn mostrou que o lado do pré-condicionamento é importante para problemas não simétricos mal condicionados [BEN99].

O pré-condicionamento pode, também, ser considerado ser implícito ou explícito. No implícito obtém-se uma matriz M de modo a aproximar a matriz A sem a necessidade de calcular M^{-1} diretamente. Na abordagem explícita calcula-se a matriz M^{-1} de modo que se possa aproximar A^{-1} diretamente.

Em qualquer caso o pré-condicionador deve ser obtido rapidamente e de modo que $M^{-1}A \approx I$, com mínimo de custo computacional. Ou seja, deve ser uma boa aproximação de A , ser fácil de construir e de aplicar. A primeira propriedade assegura boa taxa de convergência, enquanto a segunda assegura que cada iteração não seja custosa. Essas duas exigências são mutuamente contrastantes, sendo necessário um equilíbrio entre elas. Como a eficiência do pré-condicionador é, geralmente, inversamente proporcional ao tempo se sua construção [CAT98], deve-se conciliar o ganho da convergência com o tempo de pré-condicionamento, já que o objetivo é diminuir o tempo total de execução do algoritmo.

A seguir, são apresentadas algumas classes de pré-condicionadores. Outras importantes, como as geradas por soluções iterativas e o reordenamento dos sistemas, podem ser vistas em [BEN97] e [BEN99].

1. *particionamento da matriz $A = M - N$* : pertencem a essa classe, onde M é não singular, os pré-condicionadores Jacobi, Gauss-Seidel, SOR, entre outros, e as suas versões blocos;
2. *fatorações incompletas de A* : essa classe é a mais importante classe de pré-condicionadores algébricos, dada sua flexibilidade e importância. Exemplos são o ILU para matrizes não simétricas e o IC para matrizes simétricas. Uma discussão das várias versões, considerando os diferentes níveis de preenchimento e redução de tolerância, pode ser vista em [BEN99]. Os pré-condicionadores dessa classe são explícitos;
3. *aproximações para a inversa*: nessa classe, os pré-condicionamentos são construídos explicitamente através da definição de matrizes M^{-1} que aproximam diretamente A^{-1} . Essa abordagem difere da obtida pelo pré-condicionamento polinomial, pois a matriz pré-condicionadora não é implicitamente expressa como um polinômio em A , porém é construída e armazenada explicitamente [BEN99]. Os pré-condicionadores dessa classe são implícitos;
4. *aproximações polinomiais*: é um pré-condicionador que aproxima diretamente a inversa de uma matriz. É particularmente importante em matrizes SDP, pois não altera sua estrutura definida-positiva [CUN92]. Os pré-condicionadores dessa classe são implícitos;

Os pré-condicionadores utilizados neste trabalho para acelerar as soluções dos sistemas de equações lineares são o polinomial e o de Schwarz, pois os MDD de Schwarz podem ser vistos, também, como pré-condicionadores, pois a composição das soluções (aproximadas) locais gera um pré-condicionador global e distribuído. Nesse caso, utilizaram-se as fatorações incompletas de Cholesky (IC) e a diagonal Cholesky (DIC) para obter a aproximação para as inversas locais. As discussões a respeito dessas abordagens é como segue.

7.6.1 Pré-condicionador polinomial

Um modo de obter uma aproximação para a inversa A^{-1} no algoritmo do GC é empregar um polinômio que não destrua a simetria e a positividade da matriz dos coeficientes quando a ela aplicado. Um modo de se obter esse tipo de pré-condicionador é baseado na série truncada de Neumann, que é como segue.

Sejam I a matriz identidade e $\rho(J) = \max\{|\lambda|; \lambda \in \Gamma_{\square}(J)\}$ raio espectral de uma matriz $J_{n \times n}$, onde Γ_{\square} é o conjunto dos autovalores reais de $J_{n \times n}$. Então, se $\rho(J) < 1$, a inversa de $I - J$ pode ser expressa como uma série de potências em J , ou seja, $(I - J)^{-1} = \sum_{k=0}^m J^k$. Reciprocamente, para a decomposição $A = L + D + U$ e a matriz de iteração $J = -D^{-1}(L + U)$, pode-se mostrar que $\rho(J) < 1$ [VAR99]. Portanto, se A é escrita como $A = D(I - J)$, onde $(I - J)^{-1} = \sum_{k=0}^m J^k$, a inversa A^{-1} é obtida através da expressão:

$$\begin{aligned} A^{-1} &= (D(I - J))^{-1} = (I - J)^{-1} D^{-1} \\ &= \left\{ \sum_{k=0}^m J^k \right\} D^{-1} = \sum_{k=0}^m (-1)^k (D^{-1}(L + U))^k D^{-1} \end{aligned}$$

Assim, a estratégia do pré-condicionamento polinomial consiste em truncar a série depois de k termos e obter uma aproximação M_k^{-1} para A^{-1} . Por exemplo, para $k = 0$ e $k = 1$ tem-se, respectivamente:

$$M_0^{-1} = D^{-1} \text{ e } M_1^{-1} = D^{-1} - D^{-1}(L + U)D^{-1}$$

onde essas expressões geram matrizes de pré-condicionamento cujo número máximo de elementos por linha é ou um ou cinco, respectivamente. Neste trabalho empregou-se o pré-condicionador para $k = 0$ (pré-condicionador diagonal ou de Jacobi) e $k = 1$, pois nesses casos não ocorrem preenchimento de A .

Uma desvantagem desse pré-condicionador, como percebido nas experiências numéricas, e como apontado por Benzi [BEN99], é que ele não reduz o número de iterações no mesmo nível daqueles métodos de fatoração incompleta. Mesmo quando há a redução do número de iterações, não existe redução no tempo total do ciclo, pois o número adicional de operações no pré-condicionamento tem um tempo igual ou maior. As experiências numéricas realizadas para este trabalho mostram que o tempo total de execução permanece aproximadamente igual. A solução do GC não pré-condicionado convergiu com 11 ou 8 iterações, dependendo do erro tomado, enquanto a do GC com pré-condicionador polinomial convergiu, nesses casos, com 6 ou 4 iterações. Essa redução não melhorou o desempenho, pois o tempo gasto com o pré-condicionamento foi semelhante ao tempo ganho com a aceleração da convergência. Na tab. 7.1 e nas fig. 7.13 a 7.18, estão sintetizados alguns desses resultados.

7.6.2 Pré-condicionador MDD aditivo de Schwarz

O MDD aditivo de Schwarz decompõe a solução do domínio em soluções de N subproblemas sobrepostos. Nessa abordagem os subdomínios usam a solução da última iteração como condição de contorno, de modo que cada um deles é resolvido independentemente, e as comunicações ficam restritas às fronteiras. Assim, pode-se usar essa abordagem gerando um pré-condicionador já distribuído, onde o grau de paralelismo só depende do número de processadores disponíveis. Nesse caso, os passos $d := M^{-1}r$ e $s := M^{-1}r$ do algoritmo do GC da fig. 7.12 pode ser obtido de modo distribuído para os N processos gerados pelo algoritmo de particionamento de malha.

Esse pré-condicionador é obtido através das soluções dos subproblemas $d_i := M_i^{-1}r_i$, com $0 \leq i \leq N$. Especificamente, empregando as notações da seção 7.4.1, se R_i denota a matriz, que aplicada ao vetor solução, retorna os valores associados com determinados nodos, e R_i^T mapeia os valores à matriz global A , então os valores de

uma submatriz A_i , associada com uma sub-região Ω_i , são obtidos pela operação $A_i = R_i A R_i^T$. Usando essas notações, a matriz pré-condicionadora, gerada via MAAS de Schwarz, é como a dada em (7.2), ou seja, $M^{-1} = \sum_i R_i^T A_i^{-1} R_i$. Essa abordagem gera uma solução chamada de Krylov-Schwarz (KS), que é um método de Krylov acelerado por um método de Schwarz. Veja a seção 7.8.

Para aproximar o problema localmente de modo computacionalmente eficiente, ou seja obter M_i^{-1} tal que $M_i^{-1} A_i \approx I_i$, foram empregadas duas fatorações incompletas, a Cholesky incompleta IC(0), e a fatoração de Cholesky diagonal incompleta DIC(0). A notação "(0)" nessas fatorações incompletas indica que o nível de preenchimento da matriz é nulo, mantendo a esparsidade da matriz original. Ver detalhes e discussões adicionais em [SAA99] ou em [BEN99].

Esses pré-condicionadores para matrizes SDP são construídos em dois passos. No primeiro passo determina-se um fator incompleto, e no segundo passo, resolvem-se, a cada iteração do método do subespaço de Krylov, dois sistemas triangulares.

No caso específico da fatoração IC, procura-se determinar uma decomposição aproximada à decomposição $A = LDL^T$ da matriz A SDP, de modo que em cada iteração k , apenas os elementos a_{ij} de A , que correspondem a um conjunto $X^{(k)}$, sejam modificados. Esse conjunto satisfaz as seguintes condições [JÚD96]:

$$\left. \begin{aligned} X^{(k)} &\subseteq \{k, \dots, n\} \times \{k, \dots, n\} \\ (i, i) &\in X^{(k)}, \forall i \geq k \text{ e } (i, j) \in X^{(k)} \Rightarrow (j, i) \in X^{(k)} \end{aligned} \right\}$$

lembrando que o objetivo é determinar $M_{IC(0)}^{-1}$ tal que seja uma boa aproximação para A^{-1} . Ou seja, deve-se determinar apenas o fator incompleto $L_{IC(0)}$ de $M_{IC(0)} = L_{IC(0)} D L_{IC(0)}^T$, pois a decomposição completa $A = LDL^T$ já resolve o problema, mas esse é um método direto de custo da $O(N^3)$. Um algoritmo é como ilustrado na fig. 7.10 ([JÚD96] e [PIC2002]).

```

para k=1,2,...,n-1
  para i=k+1,...,n
    se (i,k) ∈ X(k) faça
      aux = aik
      aik = aik / akk
      para j=k+1,...,i
        se (i,j) ∈ X(k) e (j,k) ∈ X(k) faça
          aij = aij - aux . ajk

```

FIGURA 7.10 - Algoritmo para o cálculo do fator incompleto de Cholesky, IC(0)

Se os elementos a_{kk} , com $k=1, \dots, n$, são todos positivos, o processo termina com os fatores aproximados $L_{IC(0)}$ e D da matriz SDP, no qual se tem:

$$d_{kk} = a_{kk} \text{ e } l_{ik} = \begin{cases} a_{ik}, & \text{se } i \in X^{(k)} \\ 0, & \text{se } i \notin X^{(k)} \end{cases}$$

e, assim, se $L_{IC(0)}$ é uma aproximação incompleta para L , o pré-condicionador para o sistema SDP resolvido pelo algoritmo do gradiente conjugado, que é montado nos passos $d := M_{IC(0)}^{-1}r$ e $s := M_{IC(0)}^{-1}r$, é tal como:

$$\begin{aligned} p = M_{IC(0)}^{-1}r &\Leftrightarrow p = (L_{IC(0)}DL_{IC(0)}^T)^{-1}r \Leftrightarrow (L_{IC(0)}DL_{IC(0)}^T)p = r \\ &\Leftrightarrow (L_{IC(0)}D)\underbrace{L_{IC(0)}^T p}_z = r \Leftrightarrow L_{IC(0)}\underbrace{Dz}_y = r \Leftrightarrow L_{IC(0)}y = r \end{aligned}$$

Resolvendo o sistema triangular de equações $L_{IC(0)}y = r$, obtém-se y . Sendo $Dz = y$ tem-se $z = D^{-1}y$ e, resolvendo-se, o sistema triangular de equações $(L_{IC(0)})^T p = z$, tem-se a solução p , onde p ou é d ou é s . Sendo o sistema SDP vale que $L_{IC(0)} = L_{IC(0)}^T$ e os dois sistemas triangulares são resolvidos, respectivamente, com varreduras, *backward* e *forward*.

O segundo caso, o DIC(0), é uma versão simplificada do ILC(0), onde se toma $A = L + D + L^T$ e o pré-condicionador como $M = (D_{DIC(0)} + L)D_{DIC(0)}^{-1}(D_{DIC(0)} + L^T)$, sendo $D_{DIC(0)}$ o fator diagonal incompleto que deve ser calculado. Um algoritmo é como apresentado na fig. 7.11 ([BAR94] e [PIC2002]):

para $i=1,2,\dots$
 $d_{ii}=a_{ii}$
 para $j=1,2,\dots$
 $d_{ii}=1/d_{ii}$
 para $j=i+1,i+2,\dots$
 se $(i,j) \in X^{(k)}$ e $(j,i) \in X^{(k)}$ faça
 $d_{jj}=d_{jj}-a_{ji}d_{ii}a_{ij}$

FIGURA 7.11 - Algoritmo para o cálculo do fator incompleto diagonal, DIC(0)

Dispondo da aproximação $D_{DIC(0)}$, o pré-condicionador para o algoritmo do GC é montado nos passos $d := M_{DIC(0)}^{-1}r$ e $s := M_{DIC(0)}^{-1}r$, como acima. E, nesse caso também não há necessidade de calcular $M_{DIC(0)}^{-1}$ diretamente, pois pode-se fazer:

$$\begin{aligned} d = M_{DIC(0)}^{-1}r &\Leftrightarrow \left[(D_{DIC(0)} + L)D_{DIC(0)}^{-1} \underbrace{(D_{IC(0)} + L^T)d}_z \right] = r \\ &\Leftrightarrow \left[(D_{DIC(0)} + L) \underbrace{D_{DIC(0)}^{-1}z}_y \right] = r \Leftrightarrow (D_{DIC(0)} + L)y = r \end{aligned}$$

Resolvendo o sistema triangular $(D_{DIC(0)} + L)y = r$ obtém-se y . Sendo $D_{DIC(0)}^{-1}z = y$ tem-se z , e resolvendo-o sistema triangular $(D_{IC(0)} + L^T)d = z$ tem-se a solução d . Sendo a matriz SDP vale que $L = L^T$ e os dois sistemas triangulares são resolvidos com varreduras, *backward* e *forward*.

7.7 Algoritmo do gradiente conjugado

Para sistemas de equações cujas matrizes são SDP, o método do subespaço de Krylov mais eficiente é o gradiente conjugado (GC) [SHW94]. O algoritmo do GC é, por construção, altamente eficiente na obtenção da solução de sistemas de equações com matrizes esparsas e SDP. O princípio de obtenção da solução pelo GC é o da minimização do funcional f . O GC parte do fato de que o gradiente de f aponta na direção mais crescente da função (ou funcional) quadrática.

A estratégia do método do GC constitui-se, em resumo, em dar passos em cada iteração, na direção oposta à do gradiente, exigindo-se que a direção já pesquisada não seja repetida, até encontrar o mínimo estrito e global. A função atinge seu mínimo quando $\nabla f(x) = A\varphi - b = 0$, significando que ao encontrar a solução do GC, encontra-se a solução do sistema de equações lineares. A garantia de que esse mínimo é estrito e global e ocorre na solução $\varphi = A^{-1}b$, decorre de A ser SDP em relação ao produto interno $\langle u, v \rangle = u^T Av$ em \mathbb{R}^n [SLA97].

Em implementações do GC deve-se considerar a inicialização, os critérios de parada e as propriedades de convergência. Quando não se tem uma boa estimativa para φ^0 , toma-se $\varphi^0 = 0$ como valor inicial. Pode-se também utilizar o vetor φ calculado na iteração anterior como aproximação para a próxima iteração, sem reinicializar novamente o GC a partir da estimativa φ^0 . O critério de parada empregado neste trabalho utiliza a norma do resíduo r^k , tal que $\|r^k\| \leq \varepsilon \|r_0\|$, onde $\varepsilon \leq 10^{-6}$ e r^0 é o resíduo inicial. Outro critério de parada é o número máximo de iterações k_{\max} . Mas neste trabalho ele não é usado, já que a acurácia desejada é obtida com poucas iterações, no máximo 11 iterações.

No tocante à convergência do GC, ela pode ser definida pela norma da energia $\|e\|_A = \sqrt{e^T Ae}$. A cota superior para os erros é obtida em termos da A -norma e é dada, para k iterações, como:

$$\|x - x^k\|_A \leq 2 \|x - x^k\|_A = \left(\sqrt{K(A)} - 1 / \sqrt{K(A)} + 1 \right)_k$$

onde $K(A) = \lambda_{\max} / \lambda_{\min}$, com λ_{\max} e λ_{\min} sendo, respectivamente, o maior e o menor autovalor [GOL96]. Porém, esse limite é pessimista, e a acurácia de $\{\varphi^k\}_{k=0}^m$ é melhor que a desigualdade acima, cuja consequência é que o GC converge rapidamente na A -norma se $K(A) \leq 1$ ([POL92] e [GOL96]). Se A tem autovalores agrupados, a convergência é mais rápida e nisso reside a importância de pré-condicionar o sistema de equações. Deve-se melhorar o número de condição de A para acelerar a convergência do método iterativo.

7.7.1 GC paralelo: particionamento de dados

Decomposição de dados é a forma de distribuir os dados entre os processadores de uma arquitetura de memória distribuída de modo que possam ser resolvidos independentemente. Essa abordagem foi utilizada para paralelizar o algoritmo do GC descrito em [SHW94]. A paralelização do GC foi baseada na paralelização das operações vetoriais que compõem o algoritmo do GC. Para esse fim, algumas estruturas de dados devem ser distribuídas entre os processadores e são, algumas vezes, replicadas.

A estratégia de particionamento de dados é centrada na distribuição da matriz e dos vetores que compõem o sistema matricial $A\varphi = b$, onde φ é a solução do sistema de equações, entre os processadores do *cluster*. As comunicações coletivas empregadas na paralelização do algoritmo do GC são as operações *broadcast*, *scatter*, *gather* e *reduce*, que são disponíveis no MPI.

O primeiro passo na paralelização de um algoritmo é definir quais estruturas devem ser distribuídas entre os processos. No caso da paralelização do GC mostrado na fig. 7.12, cada processador recebe uma parte do sistema $A\varphi = b$. O vetor das incógnitas x é inicializado com elementos nulos e, portanto, o vetor resíduo r é inicializado com o conteúdo do vetor b (passo 3), como também o vetor d (passo 4). Durante todo o tempo, cada processador armazena uma parte do vetor b . No passo 5, calcula-se δ_{nov} através de produtos internos do vetor r . Observe-se que, no cálculo de produtos internos como $r^T r$, cada processador gera sua porção de dados, e esses resultados parciais são somados e distribuídos aos processos. A primitiva *MPI_Allreduce* realiza as operações sobre os resultados parciais e distribui o resultado final entre os processos.

A operação mais crítica é o produto matriz Ad . Como o vetor q é gerado e distribuído entre os processos, para executar a operação $q \leftarrow Ad$, (passo 8) no caso de uma matriz densa, é necessário que cada processador tenha a sua parte do vetor d . Contudo, como a matriz é esparsa cada processador necessita somente de uma parte do vetor d , pois muitas entradas de A são nulas.

No caso das matrizes *tipo 5-diagonal*, cada processador necessita de, no máximo 5 elementos do vetor d para calcular cada elemento em q , e isso reduz significativamente a quantidade de comunicação necessária para replicar o vetor d entre os processos, já que apenas uma parte do vetor é trocada. Esse procedimento também é feito no produto $A\varphi$ (passo 12). Como a matriz dos coeficientes é, nesse caso, *tipo 5-diagonal*, a estrutura de armazenamento da matriz foi otimizada, e ela é armazenada no formato CSR, como discutida na seção 3.5.2. Depois do passo 8, cada processador executa o produto $d^T q$ e, na sua parte do vetor, é executada a redução de tal modo a permitir o cálculo da soma dos produtos parciais. O resultado é enviado a todos os processos. Da mesma forma, os passos 10 e 12 executam suas partes em cada processador.

Existem cinco pontos de comunicação entre os processos: o primeiro distribui o resultado do produto $A\varphi$ (passo 3); o segundo distribui os valores de $r^T r$ para os processos (passo 5); no terceiro, trocam-se partes do vetor d (passo 8); o quarto consiste na redução para distribuir o resultado do produto $d^T q$ (passo 9); e o quinto distribui os valores de $r^T r$ para cada processador (passo 15).

N : dim.; p : nr. processadores; $\varphi^0=0$, $\varepsilon=10^{-6}$: tolerância.; k_{max} : nr. máx

1. para $i = 1$ até N/p faça
 - envie N/p linhas de A e b para processo i
2. $k \leftarrow 1$
3. $r \leftarrow b - A\varphi$
 - faça $A\varphi$ em paralelo
 - faça $b - A\varphi$ em paralelo
4. $d \leftarrow M^{-1}r$
 - faça $M^{-1}r$ em paralelo
5. $\delta_{\text{nov}} \leftarrow r^T d$
 - faça $r^T d$ em paralelo
 - faça *AllReduce* de δ_{nov}

6. $\delta_0 \leftarrow \delta_{\text{nov}}o$
7. enquanto $k < k_{\text{max}}$ e $\delta_{\text{nov}}o > \varepsilon^2 \delta_0$ faça
 8. $q \leftarrow Ad$
 - faça atualize d em paralelo
 - faça Ad em paralelo
 9. $\alpha \leftarrow \delta_{\text{nov}}o / d^T q$
 - faça $d^T q$ em paralelo
 - faça AllReduce de $d^T q$
 - faça $\delta_{\text{nov}}o / d^T q$ em paralelo
 10. $\varphi \leftarrow \varphi + \alpha d$
 - faça αd em paralelo
 - faça $\varphi + \alpha d$ em paralelo
 11. $r \leftarrow r - \alpha q$
 - faça αq em paralelo
 - faça $r - \alpha q$ em paralelo
 12. $s \leftarrow M^{-1} r$
 - faça $M^{-1} r$ em paralelo
 13. $\delta_{\text{vel}}ho \leftarrow \delta_{\text{nov}}o$
 14. $\delta_{\text{nov}}o \leftarrow r^T s$
 - faça $r^T s$ em paralelo
 - faça AllReduce de $\delta_{\text{nov}}o$
 15. $\beta \leftarrow \delta_{\text{nov}}o / \delta_{\text{vel}}ho$
 - faça AllReduce em $\delta_{\text{nov}}o$
 16. $d \leftarrow s + \beta d$
 - faça βd em paralelo
 - faça $s + \beta d$ em paralelo
 17. $k \leftarrow k + 1$
 18. fim enquanto
 19. faça gather em φ

FIGURA 7.12 - Algoritmo do gradiente conjugado pré-condicionado paralelizado

7.8 Método Krylov-Schwarz

A primeira consequência da formulação do método aditivo de Schwarz (MAAS) é que a matriz de pré-condicionamento da expressão (7.2) pode ser empregada para obter o chamado método Krylov-Schwarz de solução para sistemas de equações lineares, que é um método de Schwarz, na versão aditiva, acelerado por um método iterativo do subespaço de Krylov.

Essa abordagem decorre da necessidade de acelerar a solução iterativa. Especificamente, os MDD multiplicativo e aditivo são métodos iterativos e, enquanto métodos iterativos, eles convergem lentamente (linearmente), pois podem ser vistos como um caso particular da iteração de Richardson para o sistema de equações $A\varphi = b$, e cuja taxa de convergência é linear. Para mostrar esse fato, basta observar que: $A\varphi = b \Leftrightarrow (\varphi - \varphi) = -A\varphi + b \Leftrightarrow \varphi = (I - A)\varphi + b$, e a iteração de Richardson é definida como:

$$\varphi^{(k+1)} = (I - A)\varphi^{(k)} + b$$

sendo a iteração de Richardson pré-condicionada escrita como [KEL95]:

$$\varphi^{(k+1)} = (I - BA)\varphi^{(k)} + Bb \Leftrightarrow \varphi^{(k+1)} = \varphi^{(k)} - B(A\varphi^{(k)} - b)$$

ou seja, para N subdomínios, tem-se, denotando $B \equiv M^{-1}$ em (7.2), que a iteração de Richardson pode ser escrita como [TAL94]:

$$\varphi^{(k+1)} = \varphi^{(k)} - \left(\sum_{i=1}^N R_i^T A_i^{-1} R_i \right) (b - A\varphi^{(k)})$$

Apesar de o MMS (que é análogo ao método Gauss-Seidel em blocos) convergir aproximadamente duas vezes mais rápido que o MAAS (que é análogo ao método de Jacobi em blocos), ele não é particularmente atraente do ponto de vista do paralelismo. Mas como a matriz de iteração do MAAS é distribuída, por construção, ela pode ser usada para obter o passo de pré-condicionamento de um método iterativo de Krylov. Uma idéia é, portanto, empregar o método aditivo de Schwarz como um pré-condicionador de um método iterativo do subespaço de Krylov. Nessa abordagem a taxa de convergência do método de Krylov pode ser, então, significativamente melhorada, de modo que alcance resultados melhores daqueles obtidos pela convergência linear dos métodos de decomposição de domínio.

Essa abordagem é empregada para resolver sistemas de equações lineares, e o método de solução assim gerado é chamado de método Krylov-Schwarz. Esse método é caracterizado pelo uso de um método iterativo de solução distribuído, no qual um único sistema de equações é formado e resolvido em paralelo (*data parallel*). Esse método iterativo é pré-condicionado usando o método aditivo de Schwarz, e o pré-condicionador global é formado através da combinação das soluções locais.

Especificamente, o pré-condicionamento do algoritmo do GC requer a construção de duas etapas adicionais, os passos $d := M^{-1}r$ e $s := M^{-1}r$. Nessa abordagem, deve-se obter a matriz M tal que $M^{-1}A \square I$ e, portanto, se $M_{aprox} = L_{aprox} D L_{aprox}^T$ é uma aproximação para M , tem-se que o pré-condicionador distribuído e aproximado é escrito como:

$$M_{MAAS}^{-1} = \sum_{i=1}^b M_{i,aprox}^{-1}$$

onde $M_{i,aprox}^{-1} = R_i^T A_{i,aprox}^{-1} R_i$, sendo $A_{i,aprox} = R_i A_{aprox} R_i^T$ a aproximação local para $A_i = R_i A R_i^T$. Assim, o método de Krylov-Schwarz consiste em resolver as submatrizes (de modo aproximado) localmente; calcular o pré-condicionador global a partir dessas submatrizes; e resolver o algoritmo do gradiente conjugado global até que a convergência seja alcançada. Discussões adicionais podem ser vistas em [BJO95], [PAG97] e [SLV97].

Como foi usado o estêncil de 5-pontos nas construções dos algoritmos numéricos que usam o algoritmo do GC para a obtenção da solução paralela, é necessário um halo de pelo menos uma célula nas fronteiras dos sub-domínios. Assim, após a obtenção das soluções locais, ou seja, após o cálculo dos subproblemas $d_i, s_i := M_i^{-1}r_i$, deve-se corrigir os valores das células adjacentes as fronteiras artificiais para estabelecer condições de contorno do tipo Dirichlet homogêneas entre os sub-domínios.

Neste trabalho a opção foi trocar, a cada iteração do GC, os valores do estêncil entre as células adjacentes e pegar a média de seus valores de modo a estabelecer essas CC homogêneas de Dirichlet nos passos pré-condicionados do algoritmo do GC. Porém uma outra abordagem é como aquela implementada em [BJO95], que consiste em usar a mesma estrutura de dados da solução via decomposição de dados, como

discutida na seção 3.5.2, onde se separa as células internas e as células externas de modo a identificar os “estêncil ou sobreposição interna e externa” permitindo não só a especificação das CCs *outflow* para a *decomposição de dados*, mas também usar a mesma estrutura para a especificação das *sobreposições* (CCs homogêneas) para a *decomposição de domínio* (pré-condicionamento), sem a necessidade de tomar a média dos valores das células das fronteiras interna e externa.

As fig. 7.13 a 7.18, e a tab. 7.1, mostram alguns dos resultados obtidos com o uso dos pré-condicionadores polinomial, e as fatorações incompletas de Cholesky e Cholesky diagonal, como apresentados nas fig. 7.10 e 7.11. A tab. 7.1 mostra o número de iterações necessárias para a convergência do algoritmo do gradiente conjugado, onde o erro utilizado como critério de convergência do GC foi 10^{-8} . Nessa tabela GC, GC-Polinomial, GC-KS-DIC e GC-KS-IC significam, respectivamente, os algoritmos do GC sem pré-condicionamento, do GC pré-condicionado com a aproximação polinomial, com o DIC(0), e com o IC(0). Esses dois últimos algoritmos formam o método Krylov-Schwarz com soluções locais fornecidas pelos algoritmos DIC(0) e IC(0). Esses resultados foram obtidos usando o arquivo Guaiba_100, que é aquele arquivo onde se toma os espaçamentos horizontais como $\Delta x = \Delta y = 100m$.

Note-se que neste trabalho somente se analisou numericamente a convergência dos métodos iterativos do subespaço de Krylov e dos métodos de decomposição de domínio de Schwarz. Uma análise sistemática sobre a qualidade da convergência e estimativas de erros em métodos do subespaço de Krylov, teórica e numérica, pode ser vista na tese de Êzio Araújo [ARA97a].

TABELA 7.1 - Número de iterações do GC com e sem pré-condicionamento

Versões do Algoritmo	GC	GC-Polinomial	GC-KS-DIC	GC-KS-IC
Número de Iterações	8	4	3	3

Note-se que na tab. 7.1 mostra-se o número de iterações para formar um ciclo, que é definido como sendo o número total de iterações necessárias para obter a precisão desejada, ou cumprir um critério de parada, que no caso do algoritmo do GC, é a norma do resíduo $\|r^k\| \leq \varepsilon \|r_0\|$, onde $\varepsilon = 10^{-8}$ e r^0 é o resíduo inicial.

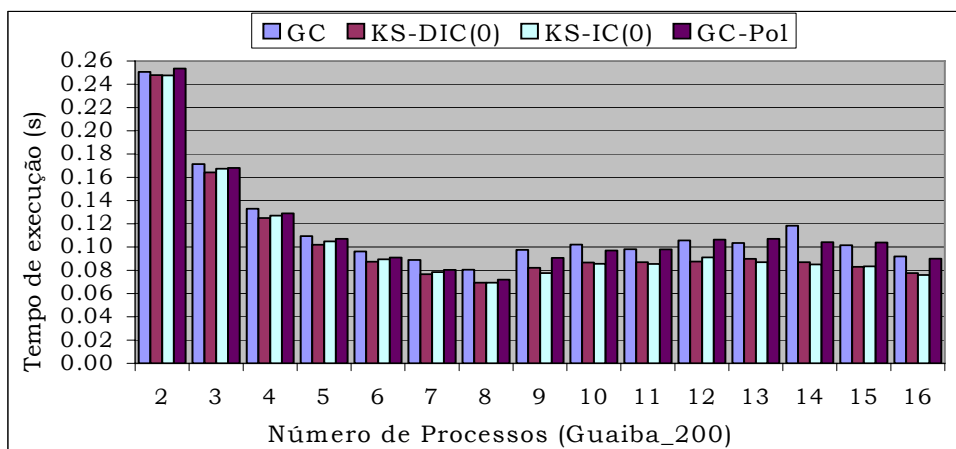


FIGURA 7.13 - Tempos de execução, em segundos, para o GC com e sem pré-condicionamento (GC, GC-Polinomial, GC-KS-DIC e GC-KS-IC). Resultados para Guaiba_200 ($\Delta x = \Delta y = 200m$)

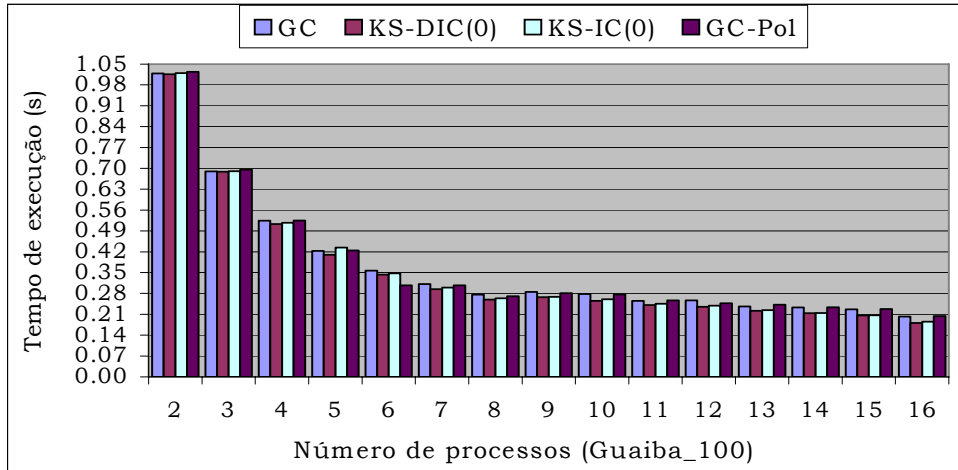


FIGURA 7.14 - Tempos de execução, em segundos, para o GC com e sem pré-condicionamento (GC, GC-Polinomial, GC-KS-DIC e GC-KS-IC). Resultados para Guaiba_100 ($\Delta x = \Delta y = 100m$)

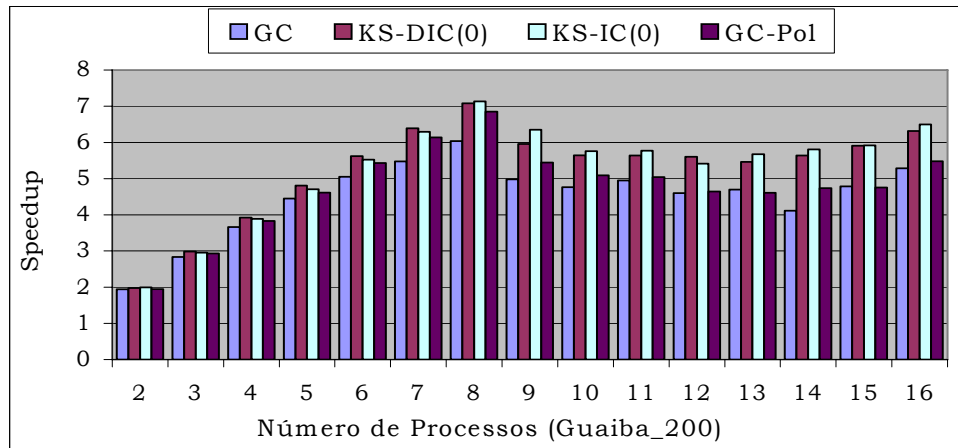


FIGURA 7.15 - *Speedup* \times número de processos, comparando o desempenho dos algoritmos dos GC, GC-Polinomial, GC-KS-DIC e GC-KS-IC. Resultados para Guaiba_200 ($\Delta x = \Delta y = 200m$)

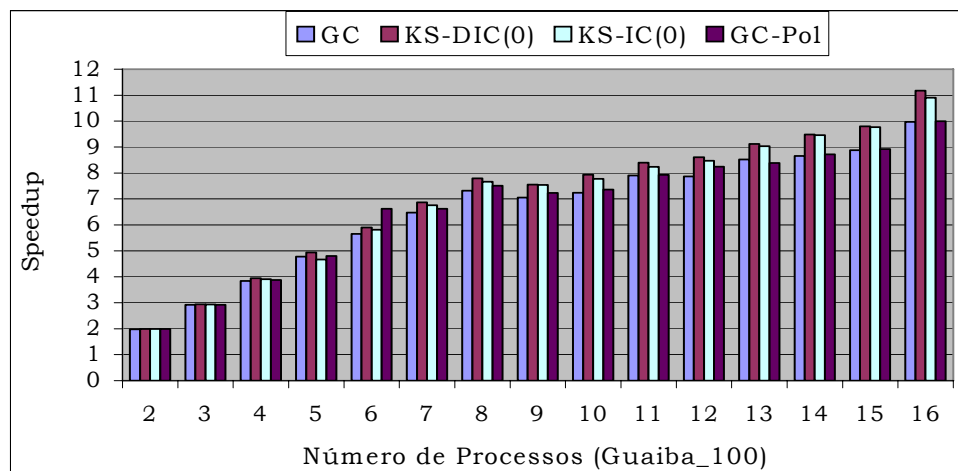


FIGURA 7.16 - *Speedup* \times número de processos, comparando o desempenho dos algoritmos dos GC, GC-Polinomial, GC-KS-DIC e GC-KS-IC. Resultados para Guaiba_100 ($\Delta x = \Delta y = 100m$)

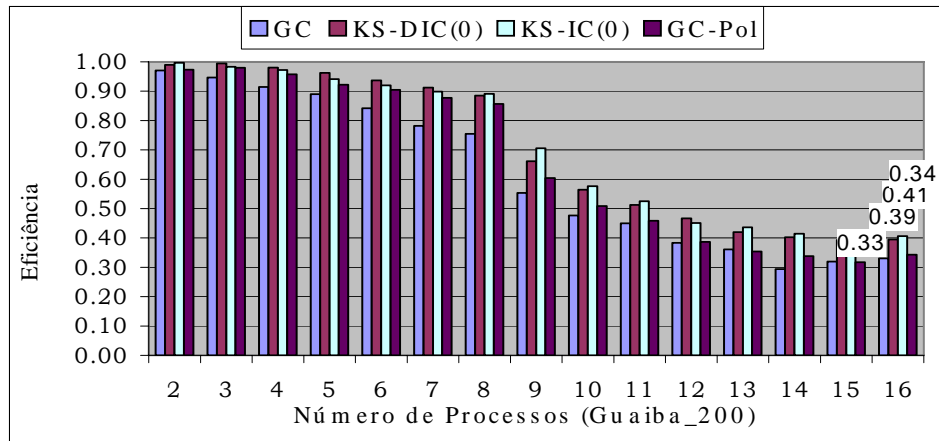


FIGURA 7.17 - Eficiência \times número de processos, comparando o desempenho dos algoritmos dos GC, GC-Polinomial, GC-KS-DIC e GC-KS-IC. Resultados para Guaiba_200 ($\Delta x = \Delta y = 200m$)

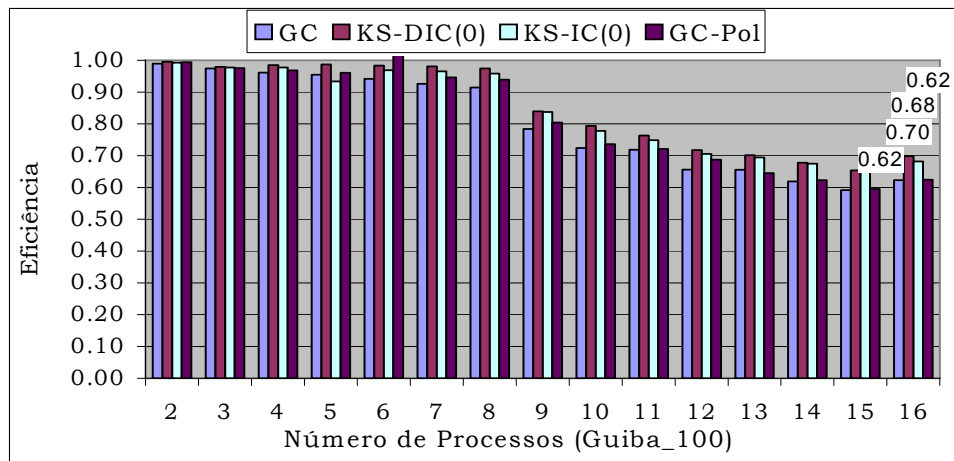


FIGURA 7.18 - Eficiência \times número de processos, comparando o desempenho dos algoritmos dos GC, GC-Polinomial, GC-KS-DIC e GC-KS-IC. Resultados para Guaiba_100 ($\Delta x = \Delta y = 100m$)

Note-se que o GC com pré-condicionador polinomial reduziu o número de iterações de 8 para 4, mas apresentou tempos de execução similares aos obtidos com o GC sem pré-condicionamento. Esse resultado deve-se ao fato de que a aceleração da convergência do GC com o uso do pré-condicionador polinomial não é suficiente para compensar o tempo gasto com a construção do pré-condicionador polinomial. Os pré-condicionadores tipo fatoração incompleta (do método Krylov-Schwarz) reduziram o número de iterações de 8 para 3 e apresentaram um menor tempo de execução em relação ao GC sem pré-condicionador, à medida que o número de subdomínios cresceu e conseqüentemente as matrizes de coeficientes dos mesmos ficaram menores. Os resultados foram obtidos com o arquivo de entrada da geometria Guaiba_100 e Guaiba_200, que são aqueles arquivos onde se toma $\Delta x = \Delta y = 100m$ e $\Delta x = \Delta y = 200m$ para representar a geometria discreta (componente horizontal). O modelo tem, nesses testes, apenas uma camada.

As fig. 7.19 e 7.20 mostram alguns resultados referentes aos tempos de execução, comparando o desempenho do algoritmo MAAS, dado na fig. 7.6, com o método Krylov-Schwarz, aproximado localmente pelo algoritmo DIC(0), e com o algoritmo do gradiente conjugado, para a solução paralela do sistema (6.1).

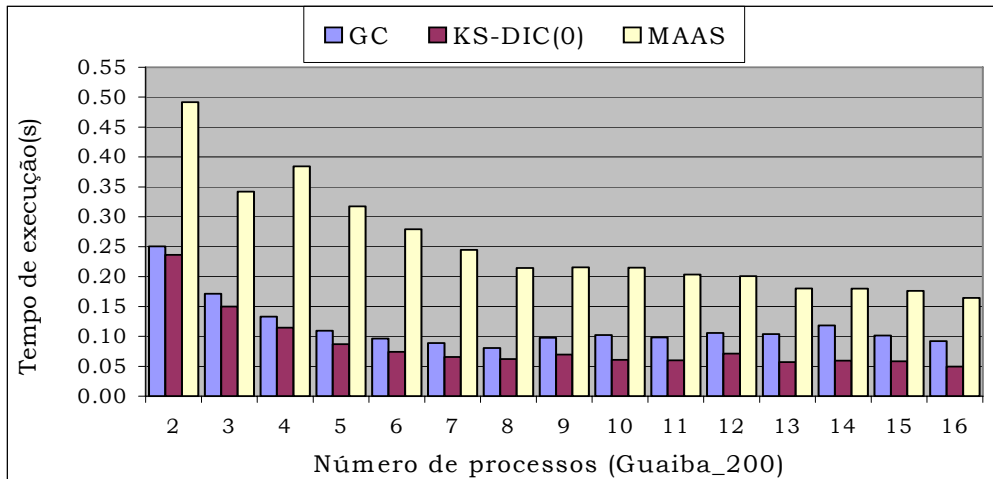


FIGURA 7.19 - Tempos de execução, em segundos, comparando o desempenho do MAAS \times método Krylov-Schwarz-DIC(0) e o algoritmo do GC. Resultados para Guaiba_200 ($\Delta x = \Delta y = 200m$)

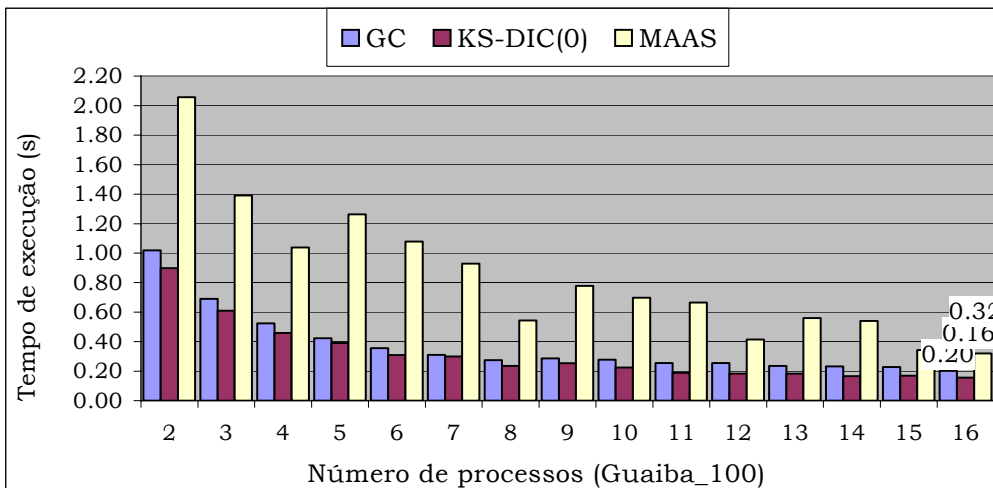


FIGURA 7.20 - Tempos de execução, em segundos, comparando o desempenho do MAAS \times método Krylov-Schwarz-DIC(0) e o algoritmo do GC. Resultados para Guaiba_100 ($\Delta x = \Delta y = 100m$)

Os resultados mostrados nas fig. 7.19 e 7.20 indicam que o tempo de execução, em segundos, do método aditivo de Schwarz (MAAS) é, para esse teste e para nossa implementação, no mínimo duas vezes maior que aquele tempo obtido com o método Krylov-Schwarz, em que a aproximação local é obtida com o algoritmo do DIC (GC-KS-DIC). O mesmo resultado ocorre com o algoritmo do GC. Esses resultados foram obtidos com o emprego de precisão dupla (*double*), e do uso da configuração geométrica dada pelo arquivo Guaiba_100 ($\Delta x = \Delta y = 100m$).

O erro utilizado como critério de convergência do GC e do Krylov-Schwarz-DIC(0) foi 10^{-6} . Para a acurácia do MAAS em paralelo tomou-se como critério de convergência que $\sup \left(\left\| \varphi_i^{(k+1)} - \varphi_i^{(k)} \right\|_2 / \left\| \varphi_i^{(k)} \right\|_2 \right) \leq \varepsilon = 0,1$. Nesse caso, o número de iterações do MAAS varia, dependendo se o problema é mais ou menos *permanente* (veja esse conceito na seção 2.1), de 2 a 3, até atingir esse critério de convergência. Resultados obtidos executando os algoritmos no *cluster* disponível no CPAD PUCRS/HP, que é composto por nodos *Pentium III 550MHz duais* conectados pela rede *Fast Ethernet*.

Outra questão indispensável na avaliação de soluções paralelas é verificar sua qualidade numérica, isto é sua acurácia, em relação a solução seqüencial, ou em relação àquela solução obtida com um só processo. Considerando-se que a solução numérica “monoprocessada” é, numericamente, a correta, um modo de avaliar a solução paralela é empregar a métrica do erro relativo. Nesse caso, se φ_i^m e φ_i^p são, respectivamente, as soluções monoprocessada e paralela, essa medida de qualidade é obtida pela expressão $E_R = |\varphi_i^p - \varphi_i^m| / \varphi_i^m$.

As fig. 7.21 a 7.24 mostram, respectivamente, a acurácia numérica obtida comparando-se a solução com um só processador com aquela solução obtida com dois processadores (processos), executando-se o algoritmo de Krylov-Schwarz (gradiente conjugado + DIC local) para obter a solução dos sistemas de equações lineares e SDP tipo 5-diagonal (6.1), cujas incógnitas a serem calculadas são os níveis da água. Esses resultados foram obtidos executando-se 500 ciclos e um erro máximo da $O(10^{-8})$ no algoritmo do GC. As fig. 7.21 e 7.22 referem-se aos resultados obtidos com precisão simples (*float*), enquanto as fig. 7.23 e 7.24 referem-se aos resultados obtidos com precisão dupla (*double*).

Nesses experimentos numéricos considerou-se como domínio computacional uma malha de 30×30 pontos, que é particionada na direção Y em dois subdomínios, onde cada um deles possui 30×15 células efetivas de cálculo (30 “linhas” por 15 “colunas”). As indicações “lado esquerdo (LE)” e “lado direito (LD)” referem-se, portanto, aos subdomínios “esquerdo” e “direito” gerados por esse particionamento, cuja sobreposição é de uma célula.

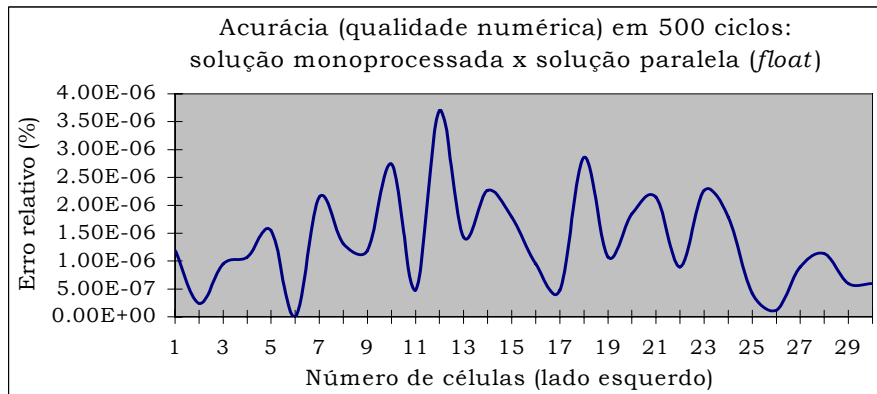


FIGURA 7.21 - Acurácia obtida comparando-se a solução em um só processo com a solução obtida com dois processos (LE e *float*)

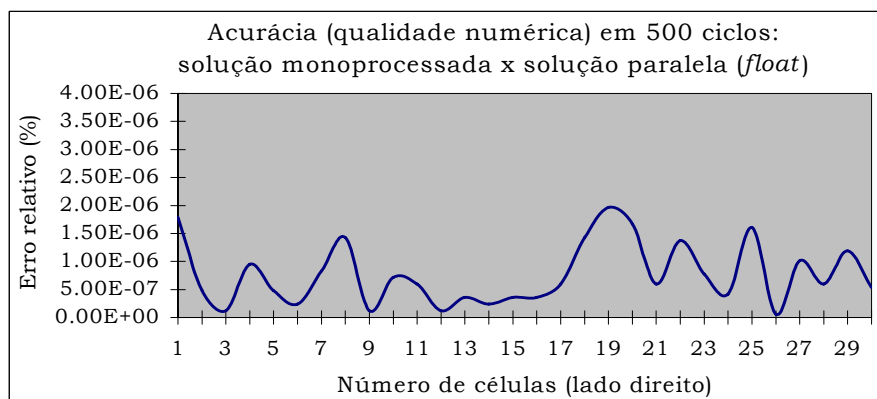


FIGURA 7.22 - Acurácia obtida comparando-se a solução em um só processo com a solução obtida com dois processos (LD e *float*)

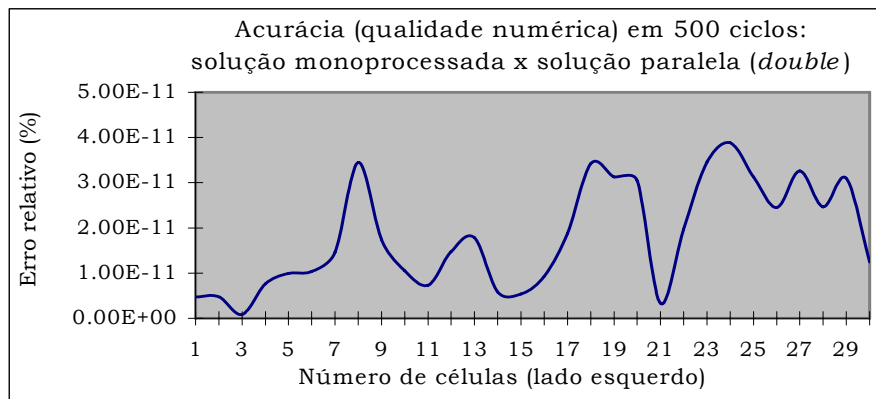


FIGURA 7.23 - Acurácia obtida comparando-se a solução em um só processo com a solução obtida com dois processos (LE e *double*)

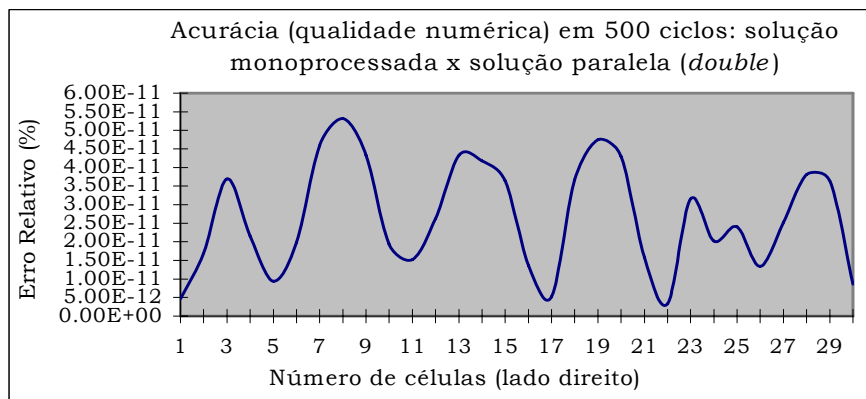


FIGURA 7.24 - Acurácia obtida comparando-se a solução em um só processo com aquela solução obtida com dois processos (LD e *double*)

Note-se que nas execuções paralelas o produto matriz por vetor não é executado na mesma ordem como aquela que é feita quando se executa em um só processador. Como destacado nas seções 3.5.1 e 3.5.2, as enumerações das células são feitas localmente e de modo que o produto matriz \times vetor é feito em duas etapas. Na primeira etapa faz-se esse produto para as células internas, e na segunda etapa faz-se esse produto para as células dos estêncies. Desse modo, o arredondamento em ponto flutuante na solução paralela é diferente do arredondamento em ponto flutuante na solução monoprocessada. De qualquer modo, as fig. 7.21 a 7.24 mostram que a acurácia entre a solução monoprocessada e a solução paralela (caso de dois subdomínios, mas válida em geral) só depende da aritmética de ponto flutuante.

7.9 Algoritmo do resíduo mínimo generalizado

O *Generalized Minimal Residual Method* (GMRES) é um método iterativo para a solução de matrizes não simétricas. Ele constrói a solução, gerando uma seqüência (base do subespaço) de vetores ortogonais, usando, por exemplo, o processo de Arnoldi. Mas esse processo, quando usa a ortogonalização de Gram-Schmidt, define vetores ortogonais somente em aritmética real. Na aritmética de ponto flutuante os vetores não são ortogonais [NOB86]. Para resolver esse problema substitui-se o algoritmo de Gram-Schmidt clássico por outros algoritmos que são matematicamente equivalentes, mas computacionalmente superiores, como o algoritmo de Gram-Schmidt modificado.

Os vetores dessa base são armazenados na matriz de Hessenberg, que é resolvida por um procedimento de retro-substituição após a eliminação da diagonal principal

através de métodos como a rotação de Givens, e um novo resíduo é calculado com a aproximação encontrada. Se a convergência não for atingida dentro de um ciclo (formado pelo número de vetores na base de Krylov), a solução mais recente é usada para iniciar a próxima iteração. Esse algoritmo é mostrado na fig. 7.25.

O GMRES é um método robusto e obtém a solução aproximada com norma residual mínima. Sua desvantagem é que os produtos matriz-vetor aumentam linearmente com as iterações, e todos os vetores da base do subespaço de Krylov têm que ser armazenados, o que é um problema quando a dimensão m do subespaço cresce. A solução mais empregada é reinicializar o algoritmo, fixando-se a dimensão m do subespaço. Essa estratégia gera o conhecido GMRES(m) que tem a desvantagem de não ter a robustez do GMRES, uma vez que a convergência não está mais garantida. Existem situações nas quais o algoritmo pode estagnar ou ter uma convergência muito lenta [VOR94]. Mas, se a matriz é real e positiva, o GMRES(m) converge [SAA96].

Como o algoritmo pode convergir mais ou menos rapidamente para um dado valor de m , a questão é saber como escolher esse valor adequadamente, porém esse é ainda um problema em aberto. Silva [SLV97] sugere $5 \leq m \leq 30$ para problemas de CFD. Outra estratégia é corrigir o armazenamento e a computação excessiva truncando a ortogonalização. Essa abordagem, conhecida como ortogonalização incompleta (*incomplete orthogonalization method*), pode ser vista em [SAA99].

1. $\varepsilon_{tol} = tol$; $k = \text{base de } K$; $p_{max} = \text{max. iter.}$

2. $\varepsilon \leftarrow \varepsilon_{tol} b$; $\varphi = 0$

ciclo GMRES

3. Para $p=1, \dots, p_{max}$ faça

4. $v_1 = b - A\varphi$; $\bar{e} = \|v_1\|$; $v_1 = v_1 / \|v_1\|$

iterações do GMRES

5. Para $i=1, \dots, m$ faça /Arnoldi/

6. $v_{i+1} = Av_i$

7. para $j=1, \dots, i$ faça

8. $\beta_{i+1,j} = \langle v_{i+1}, v_j \rangle$ e $v_{i+1} \leftarrow v_{i+1} - \beta_{i+1,j} v_j$

9. fim do j

10. $\bar{h}^{(i)} = \{\beta_{i+1,1}, \dots, \beta_{i+1,i}, \|v_{i+1}\|\}^T$; $v_{i+1} = v_{i+1} / \|v_{i+1}\|$

11. para $j=1, \dots, i-1$ faça /Givens/

12.
$$\begin{Bmatrix} \bar{h}_j^{(i)} \\ \bar{h}_{j+1}^{(i)} \end{Bmatrix} \leftarrow \begin{bmatrix} c_j & s_j \\ -s_j & c_j \end{bmatrix} \begin{Bmatrix} \bar{h}_j^{(i)} \\ \bar{h}_{j+1}^{(i)} \end{Bmatrix}$$

13. fim do j

14. $r = \sqrt{\left(\bar{h}_i^{(i)}\right)^2 + \left(\bar{h}_{i+1}^{(i)}\right)^2}$;

15. $c_i = \bar{h}_i^{(i)} / r$ $s_i = \bar{h}_{i+1}^{(i)} / r$

16. $\bar{h}_i^{(i)} \leftarrow r$

17. $\bar{h}_{i+1}^{(i)} \leftarrow 0$

18. $\bar{e}_{i+1} = -s_i \bar{e}_i$; $\bar{e}_i = c_i \bar{e}_i$ /fim Givens/

19. se $|\bar{e}_{i+1}| \leq \varepsilon_{tol}$ /convergência

20 *fim do i /fim Arnoldi*
fim das iterações do GMRES

21. resolver

$$\begin{bmatrix} \bar{h}_1^{(1)} & \dots & \bar{h}_1^{(i-1)} & \bar{h}_1^{(i)} \\ 0 & \ddots & \vdots & \vdots \\ \vdots & \ddots & \bar{h}_{i-1}^{(i-1)} & \bar{h}_{i-1}^{(i)} \\ 0 & \dots & 0 & \bar{h}_i^{(i)} \end{bmatrix} \begin{Bmatrix} y_1 \\ \vdots \\ y_{i-1} \\ y_i \end{Bmatrix} = \begin{Bmatrix} \bar{e}_1 \\ \vdots \\ \bar{e}_{i-1} \\ \bar{e}_i \end{Bmatrix}$$

22. cálculo da solução $\varphi \leftarrow \varphi + \sum_{j=1}^i y_j v_j$

23. /convergência/ se $|\bar{e}_{i+1}| \leq \varepsilon$ *fim do p*

fim do ciclo GMRES

24. x é a solução aproximada

FIGURA 7.25 - Algoritmo GMRES(m)

O algoritmo do GMRES(m) foi empregado neste trabalho para obter a solução do sistema de equações lineares (6.5) para a equação do transporte escalar 2D, cujas matrizes são não simétricas. Nos nossos experimentos numéricos, o valor de m que melhor conciliou a exigência de memória e a rapidez de convergência foi $m=5$. O erro utilizado como critério de convergência do GMRES(m) foi de $\varepsilon=10^{-7}$. Como as matrizes são bem condicionadas não foi necessário pré-condicionar o sistema, pois a convergência do GMRES(m) é obtida com apenas 3 iterações.

A versão apresentada na fig 7.25 é a seqüencial. Neste trabalho o GMRES(m) não foi paralelizado, sendo que a solução paralela para a ETM 2D-IV (veja o esquema numérico e o sistema de equações resultante na seção 6.2) foi obtida somente com o uso do algoritmo de decomposição de domínio aditiva de Schwarz, como mostrado na fig 7.7, com um erro relativo de $\varepsilon=10^{-3}$. Um estudo comparativo do desempenho do algoritmo do GMRES paralelo, do método Krylov-Schwarz e do método aditivo de Schwarz, com e sem o uso de *threads*, na solução paralela de sistemas não simétricos, é aquele que está sendo conduzido por Delcino Picinin Jr. e André Luis Martinotto.

7.10 Método Direto: solução local e pipeline

Quando as matrizes são 3-diagonal, pode-se resolvê-las empregando o algoritmo de Thomas. O método de Thomas transforma a matriz, eliminando a diagonal inferior e normalizando a diagonal principal. Na segunda varredura (*backward*), obtém a solução do sistema de equações através de retro-substituição. Devido às essas características o algoritmo não preenche a matriz, e é de $O(N)$. O algoritmo é como na fig. 7.26.

1. $d'_{(1,j)} = d_{(1,j)} / c_{(1,j)} ; i = 1$
2. $d'_{(i,j)} = d_{(i,j)} / (c_{(i,j)} - e_{(i,j)} d'_{(i-1,j)}) ; i = 2, \dots, n$
3. $b'_{(1,j)} = b_{(1,j)} / c_{(1,j)} ; i = 1$
4. $b'_{(i,j)} = b_{(i,j)} - e_{(i,j)} b'_{(i-1,j)} / (c_{(i,j)} - e_{(i,j)} d'_{(i-1,j)}) ; i = 2, \dots, n$
5. $\varphi_{(n,j)} = b'_{(n,j)} ; i = n$
6. $\varphi_{(i,j)} = b'_{(i,j)} - d'_{(i,j)} \varphi_{(i+1,j)} ; i = n-1, \dots, 1$

FIGURA 7.26 - Passos *forward* (1 a 4) e *backward* do algoritmo de Thomas (5 a 6)

O algoritmo de Thomas é empregado neste trabalho para resolver sistemas de equações lineares resultantes das discretizações dos problemas 2D e para os sistemas de equações 3-diagonal da ETM 3D.

Para as ESW 3D, devido à formulação algorítmica (inversão de matrizes 3-diagonal) das componentes das velocidades horizontal, as expressões (4.5) e (4.6), o algoritmo utilizado é o de Cholesky. O método de Cholesky é eficiente e acumula menos erros de arredondamento que outros métodos de inversão de matrizes, devido à menor influência das operações de divisão [SZE99]. Se A é SDP, então $A = LDL^T$, em que L é a matriz triangular inferior, tal que $l_{j,j} = 1$ e D é a diagonal. Os elementos $d_j = 1$ e l_j são calculados como na fig. 7.27. Se $(A)^{-1} = (L^T)^{-1} (D)^{-1} (L)^{-1}$, basta obter os elementos \bar{l}_j de $(L)^{-1}$ e calcular $(L^T)^{-1} (D)^{-1} (L)^{-1}$. O algoritmo para a inversão de matrizes 3-diagonal SDP é mostrado na fig. 7.28.

1. $d_1 := a_{1,1}$
2. *para* $j = 2, \dots, n$
3. $l_{j-1} := a_{j,j-1} (d_{j-1})^{-1}$
4. $d_j := a_{j,j} - l_{j-1} \cdot a_{j,j-1}$

FIGURA 7.27 - Algoritmo para inversão da matriz triangular inferior

1. *para* $j = 2, \dots, n$
2. $\bar{l}_{j,j} := 1$
3. *para* $i = j, \dots, n - 1$
4. $\bar{l}_{i+1,j} := \prod_j^{i-1} (-l_j)$

FIGURA 7.28 - Algoritmo para inversão da matriz 3-diagonal SDP

Assim, o algoritmo de Cholesky é usado na obtenção da solução paralela da ESW 3D, pois a dependência implícita de dados é, nesse caso, apenas na coluna de água e, portanto, o uso de um particionamento 2D elimina quaisquer obstáculos na solução paralela do transporte escalar de substâncias 3D. Mas no caso do uso do algoritmo de Thomas na solução das ESW 2D, existe dependência implícita e global de dados, o que acarreta uma ociosidade quando implementado em arquiteturas de memória distribuída. Um modo para melhorar a eficiência do algoritmo de Thomas é usar a versão *pipeline* de Povitsky ([POV98] e [POV98a]).

7.10.1 Algoritmo de Thomas *Pipeline*

Note-se que a solução de problemas 2D pode ser obtida, considerando-se, o primeiro meio passo de tempo e a porção de dados alocada a cada processador, com o algoritmo de Thomas original. Mas o segundo meio passo de tempo exige a versão *pipeline*, pois cada processador tem apenas uma parte das colunas. A aplicação direta do algoritmo de Thomas nesse passo causaria uma grande ociosidade nos processadores, pois cada um deles depende dos resultados de seus vizinhos.

Assim, o primeiro meio passo de tempo tem alto grau de paralelismo, já que cada processador pode resolver sua parte independentemente dos outros processadores,

pois a dependência de dados é restrita às fronteiras artificialmente criadas. Nessas fronteiras, exige-se uma sobreposição de duas linhas na montagem e a solução dos sistemas de equações. Esse passo é ilustrado na fig. 7.29.

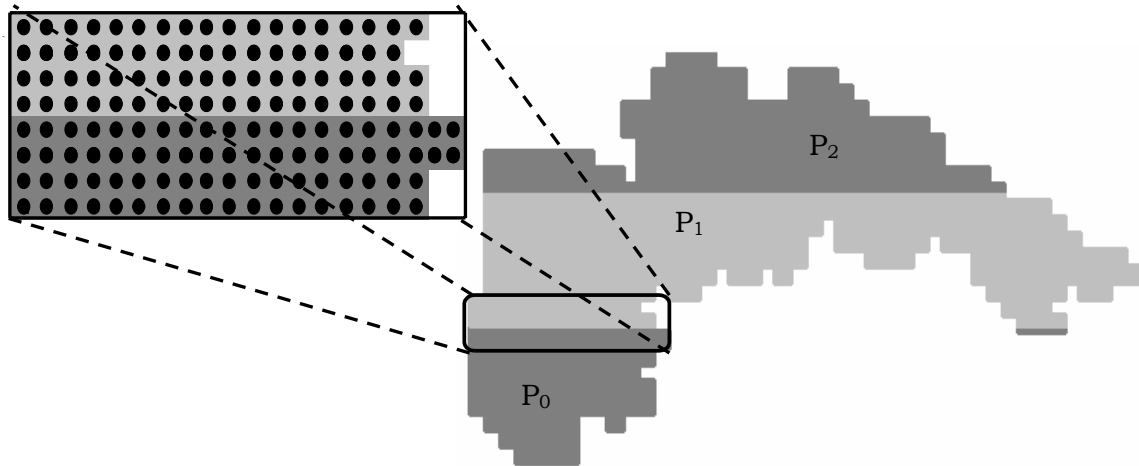


FIGURA 7.29 - Passo X: decomposição de domínio formando um sistema de equações lineares para cada linha

Note-se que a fig. 7.29 mostra uma aproximação grosseira para o Lago Guaíba, onde é ilustrado um particionamento em três faixas na direção Y . Além disso, pode-se notar, no destaque (“zoom”) das fronteiras entre os subdomínios P_0 e P_1 , uma representação para os valores nodais nessas linhas adjacentes. Tais valores foram empregados para construir as sobreposições, que são de duas linhas (pois apresentou resultados melhores daqueles quando do uso de apenas uma linha de sobreposição), quando da utilização do estêncil de 5-pontos para aproximar as EDPs.

A fig. 7.30 mostra subdomínios irregulares nos quais, para cada coluna, um sistema de equações S_i deve ser resolvido. Porém, como cada sistema de equações é distribuído entre os três processadores (P_0 , P_1 e P_2), a aplicação direta do algoritmo de Thomas é ineficiente, como se pode ver no escalonamento para a execução do algoritmo de Thomas nesse passo, como mostrado na fig. 7.27.

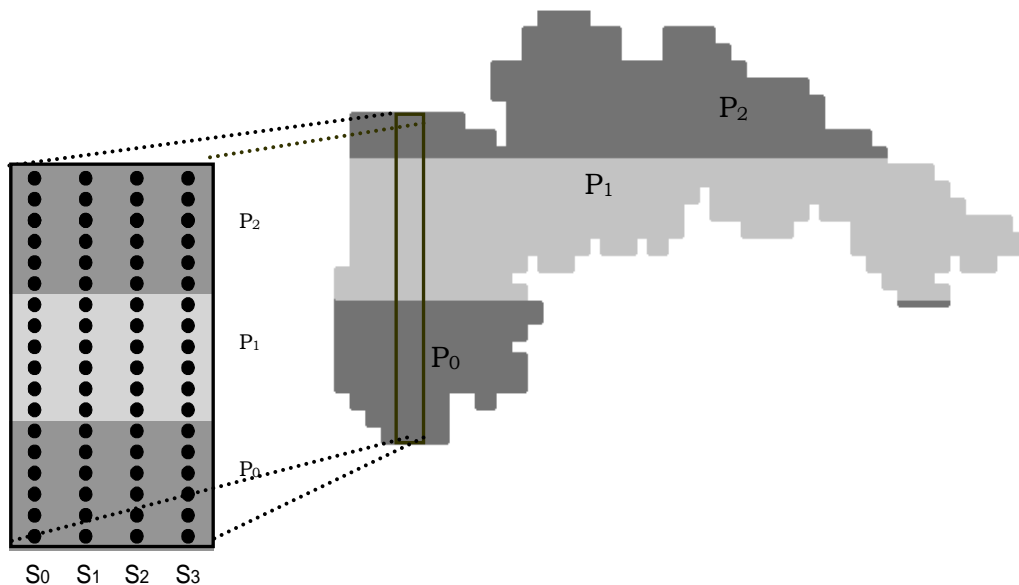


FIGURA 7.30 - Passo Y: Particionamento de dados para um subdomínio regular com 4 colunas; em cada coluna é resolvida sua porção de dados, onde S_i representa um dos quatro sistemas de equações

Essa ineficiência decorre da intrínseca dependência de dados do algoritmo de Thomas, que resolve o problema através de operações elementares entre as linhas e colunas e necessita, para a próxima operação elementar, das anteriores, acarretando uma dependência global no algoritmo.

Na fig. 7.31, F_i e B_i representam, respectivamente, a execução dos estágios *forward* e *backward* para um dado sistema de equações S_i . Inicialmente, considerando apenas duas colunas, o sistema S_1 é completamente resolvido e, somente depois dessa solução, inicia-se a solução do sistema S_2 . Como se pode observar na fig. 7.31, são necessários 12 períodos para que se resolva o primeiro dos dois sistemas, e cada processador fica 8 períodos em ociosidade. Assim, para todo o domínio, são necessários 24 períodos para que se resolvam os três sistemas durante o segundo meio passo de tempo.

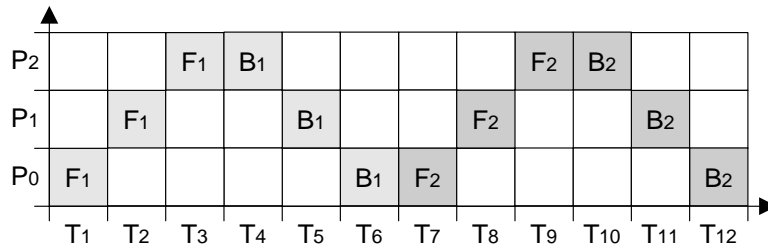


FIGURA 7.31 - Escalonamento para o algoritmo de Thomas

No segundo meio passo de tempo, a solução de uma matriz em um dado processador P_j dependerá do resultado do processador P_{j-1} , durante o estágio *forward*, e do processador P_{j+1} , durante o estágio *backward*. Essas dependências requerem mais comunicações que as do primeiro meio passo de tempo e causam mais ociosidade nos processadores.

Para reduzir essa ociosidade, aplica-se a versão do Thomas *pipeline*. Quando um processador P_j acaba um estágio *forward* F_i para um sistema de equações S_i , envia seus valores de fronteira para o próximo processador no *pipeline* (P_{j+1}) e começa a executar o estágio *forward* para o sistema de equações S_{i+1} . Quando o último processador no *pipeline* acaba o estágio *backward* B_i para um sistema S_i , ele envia seus valores na fronteira para os processadores anteriores no *pipeline* e inicia o estágio *forward* para o próximo sistema S_{i+1} . O escalonamento desse algoritmo pode ser visto na fig. 7.32, na qual o domínio dado na fig. 7.30 é resolvido com somente 12 períodos. Essa estratégia permite reduzir a ociosidade dos processadores.

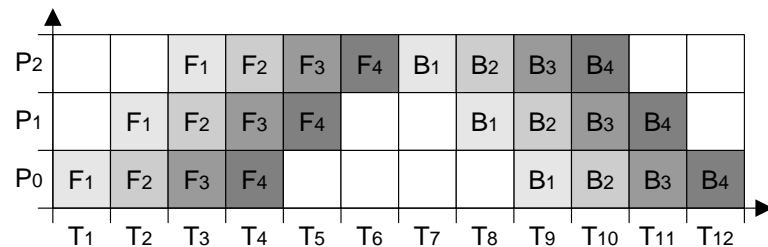


FIGURA 7.32 - Escalonamento para o algoritmo de Thomas *pipeline*

Observe-se que as fig. 7.31 e 7.32 representam o escalonamento para regiões regulares. Em subdomínios irregulares, bolhas são introduzidas no *pipeline*, reduzindo a sua eficiência. Uma estratégia para melhorar o desempenho desse algoritmo é enviar dados de mais de uma coluna para cada mensagem, o que também reduz o efeito da latência da comunicação. Neste trabalho, o número de colunas foi obtido por inspeção. Povitsky [POV98] considera esse problema, fazendo referências a trabalhos

desenvolvidos que permitem o cálculo de um número ótimo, obtido como função da relação computação/comunicação.

Dentre outras estratégias paralelas alternativas para a solução de sistemas tridiagonais obtidos com o método ADI (*Alternating Direction Implicit*) estão as empregadas por [LEC92], [VOL97] e [ZHU98], que utilizam MDD com sobreposição dos subdomínios, gerando sistemas tridiagonais que são distribuídos entre os processadores. O algoritmo de Thomas, ou o algoritmo da redução cíclica [LAK95], é aplicado a cada processador com um esquema de iterações tipo ciclos pares e ímpares. Outras abordagens podem ser vistas em [GUP95], e [KAP98].

Um particionamento que foi implementado para o modelo hidrodinâmico 2D, é aquele por divisão em faixas na direção Y , conforme mostrado na fig. 7.33. Esta heurística tem a vantagem da simplicidade de implementação do particionamento, e resulta em fronteiras retas, o que faz com que cada linha do domínio resida inteiramente em um processador, o que é vantajoso ao usar o método ADI. Além disso, o número reduzido de subdomínios vizinhos faz com que o número de mensagens trocadas seja pequeno, o que pode ser vantajoso no caso de mensagens pequenas, onde o tempo de latência é significativo em relação ao tempo total da mensagem.

Esta heurística, entretanto, pode resultar em um particionamento bastante pobre, podendo haver uma diferença grande no número de células, de um subdomínio para o outro, além de gerar subdomínios vizinhos com uma fronteira larga, o que resulta em aumento na comunicação necessária. Outro problema da heurística descrita ocorre quando é utilizada uma discretização que resulta em um único sistema pentadiagonal para todo o subdomínio. Esta discretização exige que o subdomínio seja retangular, o que faz com que um grande número de quadriculas externas sejam incluídas no subdomínio para manter a geometria retangular, acarretando cálculos desnecessários. Para minimizar o problema pode-se implementar um particionamento que, dentro de cada subdomínio retangular, detecta regiões desconexas do mesmo, dividindo o mesmo em subdomínios menores. Ver essa abordagem em [DOR2001a].

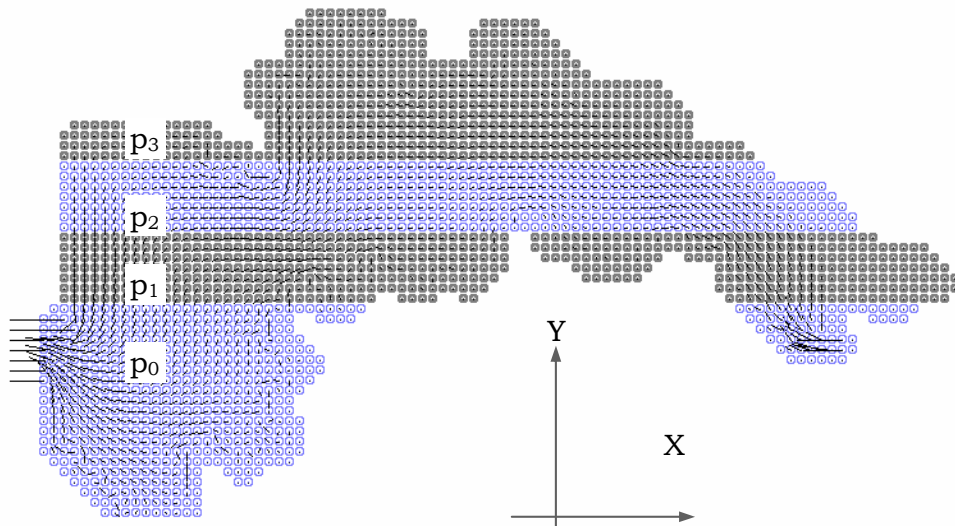


FIGURA 7.33 - Exemplo de particionamento em quatro faixas na direção Y para uma configuração aproximada do Lago Guaíba, como aquela mostrada na fig. 3.13

A fig. 7.34 mostra o campo de velocidades para o particionamento mostrado na fig. 7.33, onde se tem aproximadamente 300.000 células efetivas de cálculo.



FIGURA 7.34 - Campo de velocidades, onde $|U|=0,3$ m/s na fronteira *outflow* e $|U|=0,9$ m/s na fronteira *inflow*. Os parâmetros usados no modelo hidrodinâmico (ESW 2D-IV) foram $\eta_0 = 0,20$ m, $\Omega = 7,27 \cdot 10^{-5}$ m²/s, $C_D = 3,0 \cdot 10^{-6}$, $W_{x,y} = 6,0$ m/s², $C_z = 65,0$ e $\varepsilon_h = 10,0$ m²/s

Considerando a rede *Fast-Ethernet*, a fig. 7.35 ilustra que o aumento do *speedup* se dá com o aumento do refinamento global da malha. Especificamente, a fig. 7.35 mostra que o ganho de desempenho só é obtido a partir do refinamento de 2×2 . Isso ocorre porque a sobrecarga de comunicação na execução em paralelo é maior que o tempo ganho com a distribuição da carga de trabalho entre os processadores quando não há refinamento. Para malhas mais refinadas, obtêm-se *speedups* próximos ou superiores a dois.

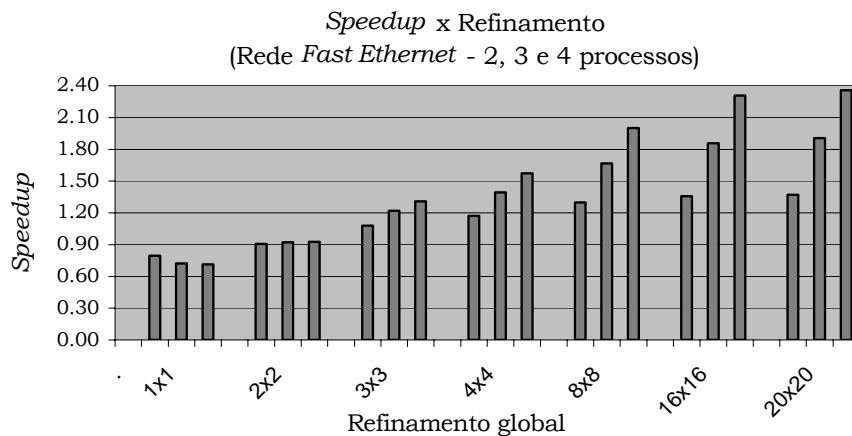


FIGURA 7.35 - *Speedup* obtido empregando 2, 3 e 4 processadores para um dado refinamento global

A fig. 7.36 ilustra a relação da eficiência de processamento com o número de processadores e o refinamento. Essa figura mostra que a eficiência é incrementada com o aumento do refinamento, pois a carga de trabalho cresce quadraticamente, enquanto a comunicação tem um aumento linear. Além disso, para cada refinamento, a eficiência diminui com mais processadores como consequência da relação comunicação/computação. Isto é, a carga de comunicação por processador mantém-se a mesma, mas a computação é distribuída entre os processadores.

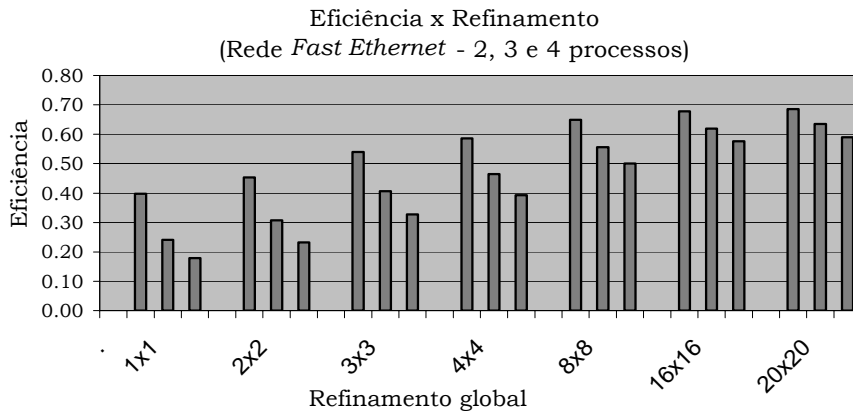


FIGURA 7.36 - Eficiência obtida com 2, 3 e 4 processadores para um dado refinamento.

Uma análise comparativa dos valores do campo das velocidades e dos níveis de água, obtidos com a solução das ESW 2D-IV e da ETM 2D-IV mostrou que os resultados produzidos com o emprego de apenas um processador, comparados com os gerados por quatro processadores, são tais que, quando do emprego da decomposição híbrida e dos algoritmos Thomas e Thomas *pipeline*, as soluções são completamente coincidentes. Isto é, o resultado obtido com um processador é igual ao resultado obtido com dois ou mais processadores.

Note-se que os modelos hidrodinâmicos 2D-IV resolvidos com o método ADI apenas foram implementados usando o particionamento em faixas, não podendo, portanto, seus resultados serem comparados com aquelas soluções paralelas obtidas com outros algoritmos de particionamento (SRCB, RCB, etc.) que foram utilizados na solução paralela dos sistemas de equações *tipo 5-diagonal*. Uma comparação possível, em termos de *speedup*, eficiência e tempo de execução, é aquela entre o MDD de Schwarz (seção 7.4.2) e entre o método Krylov-Schwarz, como visto nas fig. 7.19 e 7.20.

8 Conclusões, contribuições e trabalhos futuros

Neste trabalho construiu-se um modelo computacional paralelo multifísica para simular a hidrodinâmica e o transporte escalar de substâncias em corpos de água bidimensionais e tridimensionais. E, sendo esse o capítulo das conclusões, contribuições e trabalhos futuros, assim ele foi organizado. A título de conclusões, nesta primeira parte do texto é feita uma síntese sobre os principais assuntos discutidos nos capítulos 2 a 7 e nos anexos de 1 a 4.

8.1 Conclusões

No capítulo 2 foram apresentados os principais conceitos necessários para definir a terminologia que foi empregada ao longo de todo texto. As principais aproximações físicas usadas para obter, a partir das *equações completas*, os modelos matemáticos de escoamento de superfície livre foram, também, detalhadamente discutidas. Os modelos matemáticos analisados nesse capítulo foram: os de escoamento 3D e 2D integrado na vertical, e os modelos de transporte 3D e 2D integrado na vertical. Também apresentaram-se e discutiram-se aspectos qualitativos de modelos de parametrização de turbulência e, mais especificamente, um simples modelo de turbulência para calcular os coeficientes de viscosidade e de difusão dos modelos de escoamento e de transporte. As condições de contorno para todos esses modelos foram especificadas.

O capítulo 3 contemplou as discussões relativas ao sistema de coordenadas, aos valores nodais, à malha numérica e definiram-se as variáveis dependentes na forma discreta. Nesse capítulo foram discutidas qualitativamente algumas das principais questões associadas à determinação da malha horizontal e vertical, necessárias em um modelo tridimensional, considerando-se a representação da topografia. As condições de contorno discretas foram apresentadas e discutidas considerando-se seus diferentes tipos. Também foram abordadas as representações para os domínios computacionais, de modo a obter a discretização do domínio, relacionando-a com o armazenamento dos diferentes tipos de células, internas e de fronteira, visando às distintas estratégias de solução paralela.

Como os modelos 2D são resolvidos eficientemente, pelo menos de modo seqüencial, usando a solução combinada “método ADI e algoritmo de Thomas”, caracterizou-se essa abordagem destacando-se suas “linhas e colunas de cálculo”. Além disso, nesse capítulo também apresentaram-se e discutiram-se métodos adaptativos para a malha, e mais especificamente aquele método onde se utiliza procedimentos de interpolação espaço-temporal para obter o refinamento local para a equação do transporte.

No capítulo 4 discutiram-se os modelos, introduzidos no capítulo 2, na forma discreta. Empregando o método das diferenças finitas, são discretizados todas as equações diferenciais parciais que representam os modelos hidrodinâmico 2D e 3D, os modelos de transporte escalar de substâncias 2D e 3D, e o modelo de parametrização de turbulência. Suas condições de contorno são, também, discretizadas via diferenças finitas. Os algoritmos numéricos são, nesse capítulo, colocados sob a forma fluxo, como o caso dos modelos de transporte bi e tridimensional, ou apresentados na abordagem devida a Casulli, caso dos modelos de escoamento bi e tridimensional. Os parâmetros do modelo de parametrização são especificados a partir da literatura técnica.

O capítulo 5 abordou os métodos e técnicas de aproximação para os fluxos numéricos, sobretudo aqueles da equação do transporte escalar. A partir de uma ampla revisão bibliográfica da literatura técnica, optou-se por construir funções aproximantes para os fluxos numéricos da advecção horizontal do transporte 3D, considerando-se o método de limitação de fluxo de Sweby. Para assegurar a limitação

desses fluxos analisou-se e implementou-se uma ampla coleção de funções limitadoras de fluxo. Os resultados mostram que funções de restrição de fluxo não lineares fornecem resultados de alta qualidade numérica. Alguns desses resultados são apresentados nesse capítulo, mas no anexo 4 discorre-se sobre os todos resultados, que estão tabelados e discutidos.

Na solução da advecção vertical do modelo de transporte 3D foi empregado o método beta de Gross, sendo a difusão vertical aproximada via método Crank-Nicolson. Todos os algoritmos foram desenvolvidos para velocidades positivas ou negativas e considerando-se a sua dependência do número de Courant.

No capítulo 6, abordou-se a aplicação dos métodos e técnicas de aproximação introduzidos no capítulo 5 para construir as aproximações para os modelos discretos apresentados e discutidos no capítulo 4. O resultado é a construção dos algoritmos numéricos de modo a obter as soluções das EDPs do escoamento nos casos 2D e 3D, e as soluções das EDPs do transporte escalar nos casos 2D e 3D.

São apresentados e discutidos os resultados da aplicação de experimentos numéricos para verificação da qualidade numérica da solução dos modelos de escoamento 2D e 3D. Os resultados comparativos entre as soluções analíticas (caso 1D) e as soluções numéricas mostraram que o modelo oferece soluções positivas, acuradas e eficientes. As expressões discretas dos modelos de escoamento e de transporte são aproximadas nas fronteiras fechadas ou abertas usando, respectivamente, a técnica da supressão do termos e a técnica de extrapolação. Os resultados mostram que as condições de contorno discretas assim construídas fornecem soluções sem introdução de “ruídos” ou instabilidades ao domínio computacional. Os esquemas numéricos apresentados consideram soluções em “um só passo” e em dois “meios-passos”, onde nesta última apresentam-se os algoritmos numéricos com solução ADI para a hidrodinâmica.

O capítulo 7 contemplou as questões relativas ao paralelismo – métodos e estratégias – e ao ambiente computacional. Nele foram discutidos os paradigmas de programação, a plataforma computacional e os ambientes de programação. Nesse capítulo desenvolveram-se e apresentaram-se as abordagens que foram utilizadas na obtenção da solução paralela. Os sistemas de equações tipo 5-diagonal foram resolvidos em paralelo empregando a *decomposição de dados* e a *decomposição de domínio*. A acurácia, o *speedup* e a eficiência dos resultados das soluções paralelas obtidas sob essas duas abordagens foram analisadas e discutidas. O resultado mostrou que a qualidade numérica do método de *particionamento de dados*, ou o método Krylov-Schwarz só depende da precisão usada para resolver as operações em aritmética de ponto flutuante. A acurácia da solução paralela obtida com o método de *decomposição de domínio* depende da acurácia desejada, que é estabelecida como critério de convergência global. Além disso, são caracterizados e discutidas algumas heurísticas de particionamento de malha, estratégias de mapeamento e métodos de balanceamento de carga.

Fez-se a discussão e análise de desempenho e de qualidade dos resultados obtidos quando da aplicação do algoritmo Thomas *pipeline* na solução paralela de problemas 2D que são resolvidos com a aplicação do método ADI. Os resultados mostram que sua qualidade numérica é completamente coincidente (método direto). O desempenho computacional paralelo é, aparentemente, menor que aqueles desempenhos obtidos com o uso de “soluções em um só passo”, que são resolvidos por métodos iterativos do subespaço de Krylov, como nos sistemas *tipo* 5-diagonal.

Nesse capítulo apresentaram-se, também, os métodos iterativos e diretos de solução, relacionando-os com as estratégias e abordagens para obter a solução paralela. Métodos iterativos necessitam, geralmente, de pré-condicionamento para acelerar a solução global ou local e, desse modo, foram também discutidos alguns tipos de pré-condicionadores. O resultados indicam que pré-condicionadores de Schwarz

aproximados localmente por fatorações incompletas oferecem ótimos resultados, no sentido que diminuem o número de iterações mas, no nosso caso em particular, isso não foi o suficiente para diminuir significativamente o tempo total de execução, dado que as matrizes originais já são bem condicionadas.

O anexo 1 discorre sobre todas as propriedades e características numéricas e computacionais, visando as escolhas apropriadas das abordagens, métodos de discretização e de solução numérica dos modelos apresentados. Discutem-se em detalhes todos os atributos essenciais que o projetista dos algoritmos numéricos deve considerar quando da especificação e do projeto de modelos de escoamento de superfície livre e de transporte. Esses atributos são: a monotonicidade da solução numérica, a consistência e estabilidade do esquema de aproximação, a consistência entre a equação da continuidade e a equação do transporte, a construção de esquemas numéricos cuja difusão numérica seja menor que a difusão física, a positividade da solução numérica, a característica transportativa da solução, a conservação de massa na solução, a construção de esquemas numéricos semi-implícitos, e esquemas de integração espaço-temporal.

Não existem esquemas de solução que contemplem todos esses atributos. Quando umas ou outras dessas características são privilegiadas no algoritmo de solução, perdem-se algumas das qualidades em outras características. Porém, um objetivo central é obter um bom compromisso entre a eficiência computacional e a qualidade numérica. Com essa visão é que se desenvolveu o modelo aqui apresentado.

Nos anexos 2 e 3 são apresentadas, respectivamente, as principais métricas para a avaliação da qualidade numérica e do desempenho computacional desses tipos de modelos computacionais paralelos.

8.2 Contribuições

A contribuição deste trabalho foi a construção de um modelo computacional paralelo para o problema multifísica, onde são resolvidos de modo combinado a hidrodinâmica e o transporte de substâncias, bidimensional e tridimensional, para corpos de água com complexa e intrincada configuração geométrica. Esse modelo foi especificamente projetado para ser resolvido em paralelo em arquiteturas tipo *clusters* de PCs, e independe da aplicação, apesar do estudo de caso apresentado ser especificamente o Lago Guaíba.

O modelo apresentado decorreu das escolhas feita a partir de um abrangente levantamento bibliográfico realizado, que contemplou mais de 2.300 artigos e 220 outras referências entre livros, revistas e relatórios técnicos. Todas as referências, citadas ou não na bibliografia desta tese, estão disponíveis ou em material impresso ou gravadas em CDs. Alguns desses materiais foram coletados nas visitas feitas ao INPE de Cachoeira Paulista e de São José, à COPPE-UFRJ, ao IMPA, ao IAG-USP, ao LNCC, ao IPH-UFRGS, entre outros. Também foram feitos contatos com pesquisadores no Brasil, a exemplo dos professores Alejandro Cassalas (IPH-UFRGS), Dan Marchesin (IMPA), Alvaro Bortoli e Rudnei Cunha (IM-UFRGS), Renato Simões Silva (LNCC), entre outros. No exterior, os contatos foram feitos com Leendertse, Casulli e Gross, não por acaso autores de algumas das principais referências usadas neste trabalho.

Além disso, o Relatório de Pesquisa [RIZ2002a], não publicado, contém uma série de estudos que foram desenvolvidos a partir da revisão bibliográfica e que foram importantes para determinar as abordagens e temas apresentados nesta tese. Ele contém, por exemplo, discussões e alguns detalhes sobre o sistema de coordenadas curvilíneas, esquemas numéricos baseados no método de Leendertse, esquemas numéricos baseados no método Bott-Chlond, esquema numérico semi-Lagrangiano, trilinear, tricúbico e anti-oscilatório, baseado no trabalho da NCAR, Wang e, Harten e

Zwas, métodos de integração Euler e Runge-Kutta, e um modelo de turbulência Mellor-Yamada com funções de estabilidade de Galperin, entre outros.

Este RP contém também uma tabela com as principais características técnicas de 36 modelos já desenvolvidos. Uma análise mostrou que nenhum deles reúne as características do modelo apresentado nesta tese. A grande parte desses modelos é discretizado em diferenças finitas, usa o método ADI e o algoritmo de Thomas para a solução dos sistemas de equações, seqüencial ou emprega *decomposição de dados* para obter o paralelismo, usa coordenadas cartesianas ou curvilíneas, e muitos são modelos específicos para o escoamento, exceto nos modelos de Sheng [SHE93], Gomes e Blumberg [GOM95], Vollebregt [VOL97] e Vilar [VLA99].

Especificamente, a contribuição individual mais importante é o esquema numérico para a solução de alta qualidade numérica para a equação do transporte escalar de substâncias tridimensional, no qual a dependência implícita é restrita à coluna de água. Outras contribuições para a área de modelagem computacional paralela foram:

1. A sistematização, a partir de ampla revisão bibliográfica, das principais propriedades numéricas e computacionais que modelos de escoamento de superfície livre e de transporte escalar devem contemplar quando da sua especificação. Não há, na literatura técnica disponível, algo do gênero, o que dificulta a construção efetiva desses modelos pelos não especialistas;
2. O estudo e análise numérico-computacional de um amplo conjunto de funções limitadoras de fluxo, de modo a estabelecer quais funções são mais apropriadas para a solução numérica da advecção horizontal da equação do transporte escalar, quando do emprego o método de Sweby.
3. O desenvolvimento e implementação de esquemas numéricos semi-implícitos estáveis e computacionalmente eficientes. São seis os esquemas numéricos desenvolvidos:
 - 3.1 Um esquema numérico para a *hidrodinâmica 2D* empregando o método ADI, de modo que o algoritmo resultante é semi-implícito, estável e computacionalmente eficiente. Os sistemas de equações lineares são lineares, 3-diagonal e simétricos definidos-positivos;
 - 3.2 Um esquema numérico para a *hidrodinâmica 2D* e *3D* empregando o método desenvolvido por Casulli, de modo que o algoritmo resultante é semi-implícito, estável e computacionalmente eficiente. Os sistemas de equações são lineares, *tipo 5*-diagonal e simétricos definidos-positivos;
 - 3.3 Um esquema numérico para o *transporte de substâncias 2D*, de modo que o algoritmo resultante é computacionalmente eficiente, estável e semi-implícito. Os sistemas de equações são lineares, *tipo 5*-diagonal e não simétricos;
 - 3.4. Um esquema numérico para o *transporte de substâncias 3D*, de modo que o algoritmo resultante é computacionalmente eficiente, estável e semi-implícito. Os sistemas de equações são lineares, 3-diagonal e simétricos definidos-positivos.

Esses quatro esquemas numéricos foram apresentados e discutidos neste trabalho, porém outros dois mais foram desenvolvidos e implementados e não foram aqui apresentados e encontram-se descritos no RP [RIZ2002a], e são:

 - 3.5 Esquema numérico para a *hidrodinâmica 2D* empregando o método de Leendertse, de modo que o algoritmo resultante é semi-implícito, estável e computacionalmente eficiente. Os sistemas de equações lineares são lineares, 3-diagonal e não simétricos;
 - 3.6 Esquema numérico para o *transporte de substâncias 2D* empregando o método de Leendertse, de modo que o algoritmo resultante é semi-implícito,

estável e computacionalmente eficiente. Os sistemas de equações lineares são lineares, 3-diagonal e não simétricos.

4. O desenvolvimento e implementação, em conjunto com Ricardo Vargas Dorneles, de algoritmos de interpolação espaço-temporal para obtenção de uma malha refinada localmente, de modo a melhorar a eficiência computacional do modelo;
5. O desenvolvimento e implementação, em conjunto com Ricardo Vargas Dorneles, de estratégias de paralelização. Produziram-se duas abordagens para a solução paralela. Na primeira, a *decomposição ou particionamento de dados*, as operações e os dados foram distribuídos entre os processos e são resolvidos em paralelo. Na segunda, a *decomposição de domínio*, é aquela estratégia onde a solução do problema global foi obtida combinando as soluções de subproblemas locais. Neste trabalho empregou-se o método de decomposição de domínio aditivo de Schwarz, como método de solução, e como pré-condicionador. Sob essa última abordagem obteve-se o método Krylov-Schwarz de solução. Essas opções de solução paralela oferecem uma flexibilidade adicional ao modelo.
6. O desenvolvimento e implementação, em conjunto com Ricardo Vargas Dorneles, de um algoritmo de solução paralela híbrida, onde se empregam as decomposição de domínio e de dados para resolver sistemas de equações gerados pelo método ADI (modelos 2D). O algoritmo baseia-se na versão *pipeline* do algoritmo de Thomas e foi desenvolvido por Povitsky. Não há na literatura disponível qualquer implementação paralela dessa abordagem;

Desses estudos resultaram 20 trabalhos, em autoria ou co-autoria, os quais foram publicados (3 estão em processo de submissão) em eventos internacionais, nacionais, regionais e, dentre esses trabalhos estão artigos, trabalhos individuais, o exame de qualificação e o relatório de pesquisa. As referências dos trabalhos são apresentados na bibliografia e foram: [RIZ98], [RIZ99], [RIZ99a], [RIZ99b], [RIZ2000], [DOR2000], [DOR2000a], [ZEF2000], [DOR2001], [PIC2001], [RIZ2001], [RIZ2001a], [GLT2002], [PIC2002], [PIC2002a], [DOR2002], [PIC2002], [CAR2002], [RIZ2002] e [RIZ2002a].

Outros artigos estão sendo preparados para submissão no período 2002/2003 para: WSDP, CILAMCE, *Parallel Computing* (Elsevier Science), *Applied Numerical Mathematics* (Elsevier Science), e *Computational and Applied Mathematics* (Birkhäuser).

Essas pesquisas que visam a obtenção de soluções paralelas para modelos multifísica desenvolveram-se no GMCPAD-UFRGS sob a coordenação e orientação do prof. Dr. Tiarajú Asmuz Diverio. O grupo de pesquisa, que contribuiu de modo importante para a realização desta tese, como pode ser visto nos artigos em co-autoria e autoria, é composto pelos seguintes acadêmicos:

1. Ricardo Vargas Dorneles, doutorando junto ao PPGC da UFRGS. Em sua tese desenvolve e implementa algoritmos de particionamento de domínio e de balanceamento dinâmico de carga em arquiteturas heterogêneas, com aplicação em modelos hidrodinâmicos e de transporte de substâncias;
2. Ana Paula Canal, mestre pelo PPGC da UFRGS ([CNL2000]). Seu trabalho visou a desenvolver e implementar alguns métodos de soluções diretos e iterativos (algoritmo de Thomas e gradiente conjugado) para a solução paralela de sistemas de equações lineares, empregando a decomposição de dados (troca de mensagens e *threads*) em um *cluster* de PCs multiprocessado e heterogêneo;
3. Elias César Araújo de Carvalho, recém-mestre pelo PPGC da UFRGS (julho/2002) - convênio FACCAR-UFRGS em Rolândia - PR -, que trabalhou na sua dissertação com algoritmos de mapeamento e de modelagem do *cluster*, objetivando considerar a plataforma computacional como um grafo para a aplicação de algoritmos de mapeamento;

4. Delcino Picinin Junior, mestrando junto ao PPGC da UFRGS, que está trabalhando na paralelização do algoritmo do gradiente conjugado, explorando o paralelismo na memória distribuída, empregando a biblioteca de troca de mensagens MPI e o paralelismo de memória compartilhada (nodos multiprocessados) através do uso de *threads*.
5. André Luis Martinotto, mestrando junto ao PPGC da UFRGS, que trabalha com pré-condicionamento e métodos de decomposição de domínio.

Esse grupo fez contribuições ao presente trabalho. Porém, Ricardo Vargas Dorneles, colaborou continuamente e de modo mais importante na especificação e no desenvolvimento das estratégias de paralelismo, das estruturas de dados, e nas implementações pertinentes ao modelo.

8.3 Trabalhos futuros

No decorrer deste trabalho foram desenvolvidas atividades que visaram a construção do modelo paralelo multifísica. Porém, alguns pontos importantes não puderam ser contemplados nesta tese, o que é inevitável quando se determina os objetivos a serem atingidos dentro de um determinado tempo, e se inicia uma linha de pesquisa. A realização dessas questões serão objetos de pesquisas, algumas já em andamento e outras sendo consideradas para serem realizadas em um futuro próximo, visando ao aprimoramento do modelo construído. Especificamente, as principais atividades a serem desenvolvidas podem ser pontuadas como:

1. O desenvolvimento e implementação de esquemas de simulação para áreas alagáveis ([FAL91], [CHE93] e [BAL98]);
2. O desenvolvimento e implementação de esquema de integração temporal em subciclos para a advecção horizontal da ETM ([GRO98] e [GRO99a]);
3. O desenvolvimento e a implementação de um esquema numérico semi-Lagrangeano trilinear e tricúbico e anti-oscilatório de Wang para a solução da advecção horizontal das ESW ([WAN97] e [NAC2000]);
4. O desenvolvimento e implementação de algoritmos para malhas não estruturadas ([VLA99], [VAL2000] e [CAS2000]);
5. A discretização das EDPs empregando volumes finitos ([ADC95] e [VLA99]);
6. O desenvolvimento e implementação de esquemas de solução para a advecção da ETM 3D usando o método de Bott-Chlond ([HAR72], [BOT93] e [CHL94]);
7. O desenvolvimento e implementação de métodos de decomposição de domínio sem sobreposição ([SMI96] e [BOL98]);
8. O desenvolvimento e implementação de um modelo de parametrização de turbulência do tipo Mellor-Yamada-Canuto ([MEL82] e [CAN2000]);
9. A inclusão do modo baroclínico na EDPs das ESW e, conseqüentemente a alteração dos algoritmos numéricos ([KIM94] e [GRO99]);
10. A otimização no código computacional com sobreposição de computação e de comunicação com o uso de *threads* [VOR93];
11. O desenvolvimento e implementação de sistemas de coordenadas horizontal e sigma ou isopycnal ou eta na vertical ([LIY96], [CAS97] e [BLA94]);
12. O desenvolvimento e a implementação de métodos de refinamento local, empregando-se os métodos-r ([LIS98] e [HUA99]);

Note-se, porém, que a principal lacuna deste trabalho quanto ao desempenho computacional do modelo, é a falta de mecanismos de balanceamento dinâmico de carga para o problema multifísica. Mas essa importante questão foi objeto da tese de doutorado de Ricardo Vargas Dorneles “*Particionamento de Domínio e Balanceamento Dinâmico de Carga*”, onde estão propostas e implementadas várias estratégias e algoritmos para a solução do problema de balanceamento dinâmico de carga.

Anexo 1 Propriedades numéricas e computacionais

Em modelos computacionais, tão importante quanto seu desempenho é a qualidade do resultado das suas soluções, que depende diretamente dos atributos numérico-computacionais que seus algoritmos têm de assegurar para que sejam efetivos na simulação de fenômenos reais. Alguns atributos são mais realizáveis que outros. Por exemplo, a consistência de um esquema numérico é mais fácil de se obter; a estabilidade é mais difícil de se alcançar (e de provar); a conservação da massa é um fator crítico em algoritmos de transporte; a dissipação e a dispersão numérica são questões associadas entre si.

Nesse anexo se discutem as principais abordagens e atributos numéricos e computacionais que modelos de escoamento de superfície livre e de transporte escalar de substâncias devem considerar como suporte teórico para o desenvolvimento dos algoritmos numéricos que são efetivamente implementados. Todos os algoritmos numéricos desenvolvidos neste trabalho baseiam-se, implícita ou explicitamente nesses atributos.

Note-se que esse anexo é abrangente e engloba abordagens que não foram efetivamente contempladas neste trabalho como, por exemplo, esquemas semi-Lagrangeanos, a integração temporal em subciclos, ou esquemas para simular regiões alagáveis, entre outros. Mas algumas dessas questões são objetivos de trabalhos que se seguirão à tese e, portanto, merecedoras de uma ampla discussão e análise.

A discussão e análise desses atributos parte do fato de que o modelo computacional desenvolvido é baseado em EDPs, que exigem diferentes tipos de métodos numéricos para sua aproximação, ou seja, para a construção do esquema numérico. Essa necessidade decorre do fato que cada EDP tem características próprias e essas diferentes características físicas têm implicações fundamentais na escolha da abordagem quando da obtenção de suas soluções numéricas. Essas diferentes abordagens e métodos numéricos para os diferentes tipos de EDPs são discutidas em profundidade nos trabalhos de Colella e Puckett [COL98], Hemker [HEM99] e [HEM2000], e Gerritsma e Koren [GRT2000], entre outros.

Para compreender o motivo das diferenças entre as propriedades e as características das equações diferenciais parciais note-se, inicialmente, sua classificação. Elas podem ser classificadas analisando o “discriminante” da parte principal da EDP linear de segunda ordem com duas variáveis independentes e coeficientes reais. Então, as EDPs são classificadas em elípticas, parabólicas e hiperbólicas [IOV89]. Porém, se os coeficientes variam o tipo pode variar e, além disso, podem ser de tipos diferentes ou, ainda, serem não linear, ou de alta ordem ou dimensão. Desse modo aquela classificação não se aplica, mas é um indicativo para a escolha dos métodos de aproximação e de solução.

Considerando-se essa classificação, pode-se interpretá-la fisicamente a partir da natureza das curvas características das EDPs, que são as curvas nas quais a solução se propaga. Para o caso hiperbólico são duas as características, sendo reais e distintas, significando que há uma direção preferencial de propagação da informação. No caso parabólico as curvas características são reais e iguais, o que significa que há uma propagação na direção preferencial. No caso elíptico não existem curvas características em \mathbb{R}^2 . As curvas características são *complexas*, de modo que não há direção especial na propagação da informação, que vai-se propagar igualmente bem em todas as direções. Detalhes em [AND95], [FER97], [RIZ99], [BRL2000] e [FRT2000]

Assim, sob o ponto de vista numérico, as EDPs hiperbólicas são conservativas, e suas soluções se propagam em frentes ou choques sem dissipar. As EDPs parabólicas são dissipativas, e suas soluções tendem a tomar uma direção preferencial. Por se propagarem em todas as direções, as soluções de EDPs elípticas, não são

problemáticas. Esses resultados são fundamentais e devem ser levados em conta, sobretudo no desenvolvimento das soluções numéricas para a equação do transporte de substâncias, que é uma EDP hiperbólica não linear. Sua solução numérica deve ser acurada o suficiente para prevenir o surgimento de oscilações numéricas no entorno dos fortes gradientes de concentração, da $O(10^8)$ ou mais, que são freqüentes na solução de simulações de despejos de dejetos humanos, como neste trabalho.

Dada a natureza das equações, suas soluções numéricas requerem que sejam satisfeitas simultaneamente vários atributos visando a obtenção de soluções fisicamente corretas. Porém, devido às questões apontadas, na construção de soluções numérico-computacionais para problemas definidos no escopo da Dinâmica de Fluidos Computacional (CFD), surgem problemas como dissipação numérica, oscilação numérica e perda de positividade na solução. Soluções não físicas devem-se à dissipação numérica (viscosidade artificial) e à oscilação numérica (dispersão) que surgem da falta de estabilidade e de acurácia na solução. Esses fenômenos estão associados; quando um é minimizado, o outro torna-se mais pronunciado ([ROO87], [WAN97], [LOM2000] e [TRA2000]). A fig. A1.1 mostra uma representação desses fenômenos na solução da advecção 1D.

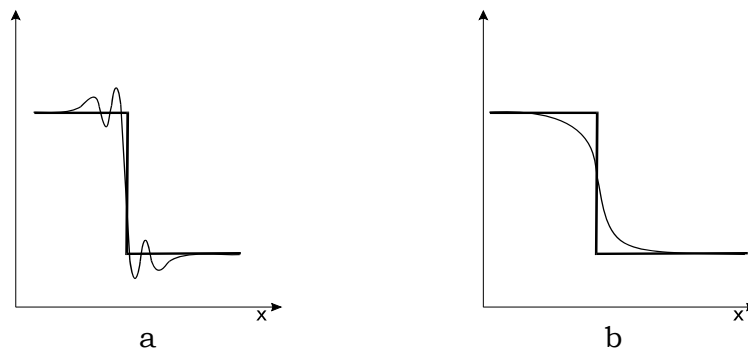


FIGURA A1.1 - (a) Oscilações numéricas quando do emprego de diferenças centrais; (b) difusão numérica com o esquema *upwind* ([MAL95])

O surgimento dessas oscilações numéricas pode ser decorrente, também, da sub e/ou da sobre-especificação dos valores nodais pelas funções interpolantes e da perda de simetria dessas funções, podendo gerar soluções cujo perfil é deslocado em relação ao perfil verdadeiro das velocidades [WAN97].

Note-se que a difusão numérica (algumas vezes denominada de viscosidade artificial) nem sempre é indesejável, pois a adição explícita de viscosidade artificial é, algumas vezes, necessária para estabilizar a solução numérica de problemas hiperbólicos complexos ([FRT2000] e [HEM2000]). Essa estratégia é, a rigor, efetivada neste trabalho quando da especificação *ad hoc* dos coeficientes de viscosidade turbulenta no termo de difusão horizontal para as ESW 2D-IV. Nessa abordagem a viscosidade é numérica e age mais como papel de elemento estabilizador da solução numérica que o de fornecer a viscosidade física como a calculada pelo modelo de fechamento de turbulência. Isso mostra que os modelos computacionais 2D, aqui discutidos e implementados, devem ter uma atenção futura nesses aspectos.

Assim, dada a importância dos atributos numéricos e computacionais visando o desenvolvimento de algoritmos, técnicas e estratégias de solução para modelos de escoamento de superfície livre e de transporte de substâncias, discute-se a seguir detalhadamente os principais atributos e/ou propriedades numéricas e computacionais visando modelos computacionais paralelos com alta eficiência computacional e alta qualidade numérica.

As propriedades, que foram empregadas, explícita ou implicitamente na construção dos algoritmos numéricos, sobretudo naqueles dos modelos

tridimensionais (3D), são: monotonicidade; consistência e estabilidade; difusão numérica menor que a difusão física; acurácia espacial e temporal de alta ordem; limitação na variação da solução; consistência entre a continuidade e o transporte; positividade da solução; transporte da solução; conservação da massa e a eficiência computacional.

Monotonicidade

Um esquema $S_i^{n+1} = \varphi(S_{i-k}^n, S_{i-k+1}^n, \dots, S_{i+k}^n)$ é dito monotônico se φ é uma função monotônica crescente de cada um dos seus argumentos, ou seja, $\partial\varphi/\partial S_j \geq 0$, $\forall i-k \leq j \leq i+k$. Isso significa que um esquema é monotônico se, dado um estado inicial monotônico, a solução permanece monotônica ([TRA2000] e [LOM2000]). Assim, um esquema numérico explícito aplicado à solução da equação da lei de conservação escalar em (A1.5), pode ser escrito, tomando $\sigma = S$, como:

$$\begin{aligned} \sigma_i^{n+1} &= \sigma_i^n - \lambda \left(f(\sigma)_{i+\frac{1}{2}} - f(\sigma)_{i-\frac{1}{2}} \right) \\ &\stackrel{not}{=} \varphi \left(\sigma_{i-k}^n, \sigma_{i-k+1}^n, \dots, \sigma_{i+k}^n \right) \end{aligned}$$

que é dito monotônico se φ é uma função monotônica crescente de cada um dos seus argumentos.

Esquemas numéricos usuais são aqueles lineares. O problema de esquemas lineares de primeira ordem é a falta de acurácia, sobretudo quando o escoamento tem altos números de Reynolds ([DEN99] e [BON2000]). Em contrapartida, esquemas lineares de alta ordem violam a condição de monotonicidade, gerando oscilações numéricas, e seu emprego requer que se previna a geração dessas oscilações, agindo diretamente no mecanismo gerador, ou na construção de um amortecimento (*damping*) através de dissipação numérica, como feito em algumas técnicas de solução para problemas definidos no escopo da aerodinâmica [RAN99].

Note-se, portanto, que a construção de um esquema monotônico não é uma questão trivial. Godunov mostrou que um esquema de solução que preserve a monotonicidade ou é de primeira ordem, ou é não-linear. Isto é, pelo teorema de Godunov, esquemas lineares que preservam a motonicidade são somente de primeira ordem de acurácia [TRA2000]. Portanto, em esquemas de alta ordem de acurácia deve-se abandonar a exigência de monotonicidade na solução satisfazendo a condição, menos restritiva, de o esquema ser TVD (*Total Variation Diminishing*). Essa questão é discutida em detalhes a seguir, quando do desenvolvimento de soluções para as leis de conservação escalar (*scalar conservation law*).

Consistência e estabilidade

A discretização tende a ser exata quando o espaçamento da malha tende a zero. A diferença entre as equações na forma discreta e na forma analítica (ou explícita) é chamada de erro de truncamento. No caso do emprego do método das diferenças finitas a estimativa dessa diferença pode ser feita usando a série de Taylor. Para um método de discretização ser *consistente*, esse erro de truncamento deve tender a zero quando o espaçamento da malha e o passo de tempo tender a zero. O erro é, geralmente, proporcional ao espaçamento da malha e ao passo de tempo. Se o termo mais significativo é proporcional a $(\Delta x_{i=1,2,3})^n$ e/ou a $(\Delta t)^n$, o método de aproximação é de n-ésima ordem.

Uma solução numérica é dita *estável* se não aumentam os erros que aparecem durante a solução numérica. Alguns dos principais problemas associados à falta de

estabilidade são os erros de propagação em amplitude, os erros de propagação na fase, as instabilidades não lineares (nas quais há transferência de energia entre os *modos*), o aparecimento de choques e os erros de transporte [TRA2000]. As abordagens mais utilizadas para fazer a análise de estabilidade são aquelas que empregam os métodos de Von Neumann ou o matricial [HIR92].

Um método numérico é dito *convergente* se a solução do sistema de equações que foi discretizado tem solução exata quando o espaçamento da malha tende a zero. Esse resultado decorre do teorema de equivalência de Lax que afirma que “*se um dado problema de valor inicial linear é corretamente posto e a aproximação via método de discretização satisfaz a condição de consistência, a estabilidade é condição necessária e suficiente para convergência*” [FER97]. Um problema é bem posto, no sentido Hadamard, se é um problema em que as equações, as CI e as CC satisfazem às condições: existe a solução; a solução é única; e solução depende continuamente dos dados iniciais e de contorno.

Porém, esse resultado (teorema de Lax) é válido somente para problemas lineares. Quando as equações são não lineares, não existem resultados gerais para o problema em questão. Nessa situação faz-se uma análise de estabilidade linear dos problemas, linearizando as EDPs através da fixação ou do desprezo dos termos não lineares ([FLE88] e [FER97]).

Assim, para as EDPs não lineares, como neste trabalho, o teorema de Lax não pode ser aplicado rigorosamente. O teorema fornece condições necessárias, mas não suficientes, para que o esquema numérico seja convergente [VLA99]. Uma vez provada a consistência do esquema, deve-se assegurar sua estabilidade e, um modo de provar a estabilidade é mostrar diretamente que isso ocorre no esquema de aproximação espaço-temporal utilizado. Essa é uma das razões por que não existe um “esquema genérico de solução” e sim uma variedade de esquemas de aproximação. Note-se, porém, que existem importantes resultados associados às soluções de leis de conservação, e esses resultados são discutidos nesse anexo.

No caso particular deste trabalho, na construção dos esquemas numéricos semi-implícitos para a hidrodinâmica, optou-se por seguir os trabalhos de Casulli [CAS90], [CAS92] e [CAS94] que, como em outros trabalhos [LEV92], [SON94] e [GRO98], desenvolveu critérios para a construção de soluções numéricas de modo matematicamente rigoroso.

Em seus trabalhos, Casulli mostrou através da análise de estabilidade das equações características das ESW 2D integradas verticalmente (2D-IV) que a celeridade (a velocidade de propagação da onda) é $c = \sqrt{gH}$. Isso significa que as ondas de gravidade propagam-se ao longo do eixos das coordenadas, em ambas as direções, e com velocidade de fase igual à celeridade [CAS90]. Considerando esse fato, e também que os termos g e $H = h + \eta$ são, respectivamente, os coeficientes dos gradientes de elevação (níveis) das $ECQM_x$ e $ECQM_y$ e das derivadas das velocidades na equação da continuidade. Então, quando esses termos são discretizados de modo implícito, a estabilidade da solução numérica não depende da celeridade. Nesse caso e, desde que os termos explícitos sejam convenientemente aproximados, o algoritmo de solução para as ESW pode ser incondicionalmente estável.

Desse modo a estabilidade dos modelos hidrodinâmicos é independente da velocidade de onda de superfície livre e assim, os esquemas semi-implícitos são estáveis, mesmo quando os termos não lineares são aproximados através da abordagem explícita. Casulli mostrou o seguinte resultado: “*o esquema semi-implícito em diferenças finitas é estável, no sentido de von Neumann, se o passo de tempo Δt satisfaz:*

$$\Delta t \leq \left[\frac{|u|}{\Delta x} + \frac{|v|}{\Delta y} + \frac{|w|}{\Delta z} + 2v_h \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right) \right]^{-1} \quad (\text{A1.1})$$

onde v_h é o coeficiente de viscosidade cinemática turbulenta horizontal [CAS92].

A condição (A1.1) é válida se os termos não lineares das ECQM das ESW são aproximados via esquema *upwind* de primeira ordem, como no caso deste trabalho. Usando o método de von Neumann, um cálculo direto com os fatores de amplificação permite estabelecer outras condições de estabilidade. Por exemplo, as condições para um esquema em diferenças centrais e um esquema semi-Lagrangiano são, respectivamente, dadas por [CAS90] e [ASE90]:

$$\Delta t \leq \min \left[\frac{\Delta x^2 \Delta y^2}{2v_h [\Delta x^2 + \Delta y^2]}, \frac{v_h}{u^2 + v^2} \right] \text{ e } \Delta t \leq \left[2v_h \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right) \right]^{-1}$$

Dadas suas características e aplicações, que são discutidas a seguir, no caso específico da equação de transporte escalar de substância (ETM 3D), estuda-se a sua estabilidade separando-se os termos horizontais e os termos verticais. Consideram-se inicialmente as componentes verticais da EDP e, em particular, a advecção vertical.

A advecção vertical é resolvida empregando o esquema $-\beta$ desenvolvido por Gross [GRO98]. Uma análise de estabilidade de von Neumann mostra que as condições $\beta = 0,0$, $\theta = 0,5$ e $C_z \leq 1$ em (A1.2) são suficientes para a estabilidade do esquema numérico para a advecção vertical. Porém, é possível determinar β de modo que se garanta a estabilidade numérica, também, quando $C_z \geq 1$, obtendo estabilidade incondicional, $\forall C_z$. Especificamente pode-se mostrar o seguinte resultado: “uma condição necessária e suficiente para a estabilidade no sentido de von Neumann é que se cumpra a condição:

$$\beta(C_z) \equiv \begin{cases} 0 & , \text{ se } |C_z| \leq 1 \\ \frac{(|C_z| - 1)^2}{4|C_z|} & , \text{ se } 1 < |C_z| \leq 3 \\ \frac{|C_z| - 2}{|C_z|} & , \text{ se } |C_z| \geq 3 \end{cases} \quad (\text{A1.2})$$

onde $\beta(C_z)$ varia no tempo e no espaço, e o método $-\beta$ depende do número de Courant $C_z = w \Delta t / \Delta z_k^n$ [GRO98].

Esse resultado é válido quando o transporte escalar não causa variação na densidade da água. Se não for esse o caso é necessário impor condições adicionais de estabilidade, como discutido em [GRO98]. Porém, isso não ocorre neste trabalho dado que a simulação computacional é somente para dejetos humanos que não modificam a densidade da água.

Para a solução da difusão vertical empregou-se o método Crank-Nicolson que é semi-implícito e incondicionalmente estável [FLE88a]. O resultado é a construção de um esquema numérico semi-implícito para o transporte vertical (advecção e difusão). Portanto, as condições de estabilidade para a ETM 3D fornecem, em princípio, estabilidade incondicional para os esquemas de solução. Mas essa estabilidade é apenas para o transporte vertical.

Assim, o uso combinado dos métodos beta e Crank-Nicolson gera um esquema numérico semi-implícito e estável para o transporte vertical. Portanto, uma solução explícita para o transporte horizontal pode fornecer um esquema numérico global que é implícito somente na coluna de água. Essa estratégia gera sistemas de equações que são 3-diagonal, ou seja, sistemas cujas matrizes são 3-diagonal de dimensão N_z , onde N_z é o número de camadas.

Com essa abordagem as dependências de dados (implícitas) ficam restritas à coluna de água. As componentes horizontais são explicitamente calculadas e, portanto, estão disponíveis no atual passo de tempo. Essa é uma abordagem importante, sob o ponto de vista da solução paralela, pois pode-se empregar um particionamento 2D no plano XY , facilitando a obtenção da solução paralela.

Outro elemento motivador para uma abordagem explícita para o transporte horizontal é que em escoamentos estratificados a condição de estabilidade imposta pelo acoplamento dos modelos hidrodinâmico e de transporte de substâncias restringe a estabilidade dos termos advectivos horizontais. Não existe, portanto, vantagem computacional, nesses casos, em tratar esses termos implicitamente. Por esses motivos, optou-se por tratar os termos horizontais de modo explícito.

Uma desvantagem da abordagem é a restrição no passo de tempo. A alternativa para esse problema é empregar a técnica de divisão do tempo (*time-splitting*) para o avanço temporal dos transportes vertical e horizontal. Através da definição de esquemas de subciclos. Esta abordagem é discutida posteriormente nesse anexo.

Difusão numérica menor que a difusão física

O objetivo básico de modelos de transporte é possibilitar o estudo da mistura vertical. Para isso, os modelos de turbulência calculam a difusão física do fenômeno. Portanto, a solução numérica da equação do transporte deve ser de tal qualidade que sua difusão numérica seja menor que a difusão física calculada pelo modelo de fechamento de turbulência. Então, um atributo importante (e difícil de se obter) em modelos de transporte é que a difusão numérica seja pelo menos uma ordem menor que a difusão física.

A questão central é, então, construir um esquema de alta qualidade para a solução numérica do transporte vertical. Uma expressão para calcular o coeficiente de dispersão (difusão) numérica ε_n do esquema numérico é determinada pela soma dos erros de discretização espacial e temporal e é dada por [ZHE95]:

$$\varepsilon_n = |\nu| \Delta h \left[\left(\frac{1}{2} - \alpha \right) + |C_r| \left(\omega - \frac{1}{2} \right) \right] \quad (\text{A1.3})$$

onde α é um fator espacial para o erro da discretização do termo advectivo, ω é um fator para o erro da discretização temporal. Além disso, $|\nu|$, Δh e $|C_r| = |\nu| \Delta t / \Delta h$ são, respectivamente, o módulo da velocidade, o espaçamento e o módulo do número de Courant para cada uma das direções coordenadas.

Com a expressão (A1.3), pode-se mostrar a relação entre o coeficiente de dispersão numérica ε_n e o coeficiente de dispersão física ε_f . Essa relação é determinada como função dos números de Péclet e de Courant, e é dada como:

$$\begin{aligned} \frac{\varepsilon_n}{\varepsilon_f} &= \frac{|\nu| \Delta h}{\varepsilon_{h,v}} \left[\left(\frac{1}{2} - \alpha \right) + |C_r| \left(\omega - \frac{1}{2} \right) \right] \\ &= P_e \left[\left(\frac{1}{2} - \alpha \right) + |C_r| \left(\omega - \frac{1}{2} \right) \right] \end{aligned}$$

onde $\varepsilon_{h,v}$ é o coeficiente de difusão horizontal ε_h ou o coeficiente de difusão vertical ε_v , e $P_e = |v|\Delta h/\varepsilon_{h,v}$ é o número de Péclet para uma dada direção coordenada. Essa expressão mostra que a relação $\varepsilon_n/\varepsilon_f$ é determinada pelo número de Péclet que controla a dispersão (difusão), e pelo número de Courant que controla a estabilidade de um esquema explícito. Assim, deve-se ter um compromisso entre a acurácia temporal e a acurácia espacial, procurando manter a estabilidade e a minimizar a difusão de um dado esquema numérico.

Essas questões estão intimamente relacionadas com os esquemas numéricos desenvolvidos. Especificamente no caso do transporte vertical, uma revisão detalhada na literatura técnica relacionada com o transporte vertical de modelos multidimensionais levou à construção e implementação, para este trabalho, de um modelo de transporte vertical baseado naquele desenvolvido por Gross [GRO98] para a advecção vertical, o esquema $-\beta$. Combinado com o método Crank-Nicolson para a difusão vertical, o esquema $-\beta$ fornece acurácia e estabilidade suficientes para minimizar a criação de difusão numérica e o aparecimento de oscilações numéricas.

O esquema $-\beta$ combina o método trapezoidal com o esquema *Quadratic Upstream Interpolation for Convective Kinematics* (QUICK) de Leonard [LEO79], gerando um esquema com pesos que tem mínima difusão numérica ([GRO98] e [GRO99]). Gross mostrou que a expressão para o coeficiente de difusão numérica para o esquema $-\beta$ é dada, considerando (A1.3), por:

$$\varepsilon_n = \frac{(\Delta z)^2}{\Delta t} |C_z| \left[\left(\theta - \frac{1}{2} \right) |C_z| + \frac{\beta}{2} \right] \quad (\text{A1.4})$$

onde se tomou em (A1.3) o fator espacial como $\alpha = 0$, e o fator temporal como $\omega = \theta$, considerando-se, ainda, $\beta/2$ na expressão da discretização temporal.

Note-se, portanto, que a expressão (A1.4) é formada pela soma do erro de discretização temporal $\left((\Delta z)^2 / \Delta t \right) (C_z)^2 (\theta - 0,5)$, mais o erro de discretização espacial $\left((\Delta z)^2 / \Delta t \right) |C_z| 0,5\beta$. Quando $\beta = 1,0$ e $\theta = 1,0$ tem-se os coeficientes máximos de difusão numérica para o erro espacial e temporal, respectivamente. Quando $\beta = 0,0$ e $\theta = 0,5$ tem-se que o coeficiente de fusão numérica do esquema β é nulo, isto é, $\varepsilon_n = 0,0$.

Porém, um bom algoritmo de solução deve ser estável para todos os números de Courant. Objetivando essa característica observa-se que enquanto o erro temporal cresce com $(C_z)^2$, o erro espacial cresce com $|C_z|$, o algoritmo de solução deve sacrificar a acurácia espacial em vez da acurácia temporal para altos números de Courant. Portanto, toma-se, no esquema $-\beta$, $\theta = 0,5$ e β variando de modo a oferecer uma solução estável, mesmo para altos números de Courant.

Apesar da grande variedade de números de Courant, comuns em modelos de transporte empregados em estuários e lagos, grande parte do movimento vertical ocorre em baixas velocidades (menor que 0,2 a 0,5 cm s⁻¹) ([GRO99] e [GRO99a]). Assim, o transporte vertical em modelos multidimensionais é, em grande parte, aquele em que se tem $C_z \leq 1$. Portanto, da relação (A1.2), tem-se que o fator β é tal que $\beta = 0,0$, que não introduz difusão numérica, justificando a escolha de que a acurácia espacial pode ser sacrificada para altos números de Courant.

Um objetivo é, então, avaliar a difusão numérica produzida pelo esquema- β , de modo a mostrar que a sua difusão tem pelo menos uma ordem de magnitude menos que a magnitude da difusão física calculada pelo modelo de parametrização de turbulência. Porém, o modelo de parametrização de turbulência desenvolvido e implementado neste trabalho (seção 2.7.2 e 4.7) é do tipo *zero equações*, que é aquele baseado em relações semi-empíricas e, por muitas vezes, aproximadas. Além disso, neste trabalho não se implementou o *modo* baroclínico nas ECQMs, que é relevante na análise dos resultados das difusões e viscosidade do modelo.

Desse modo não se justifica fazer uma análise e discussão sistemática da difusão numérica produzida pelo esquema β (ou pelo esquema FLM) comparando-as com as viscosidades cinemáticas turbulentas e as difusões turbulentas, horizontal e vertical, produzidas por tal modelo de parametrização de turbulência. Por esse motivo não se fará neste trabalho este estudo, ficando este trabalho para quando do efetivo desenvolvimento de um modelo de parametrização tipo *duas e meia equações*, conforme discutido na seção 2.7.1. Mas para efeitos de análise e de verificação da difusão numérica, quando da aplicação do esquema β , pode-se empregar uma expressão analítica para o coeficiente de difusão física para corpos de água bem misturados.

Esta estimativa é dada por $\varepsilon_f = 0,0025 h U$, onde $\varepsilon_f = \varepsilon_v$ é o coeficiente de difusibilidade vertical, h é a profundidade local, e U é a velocidade média [GRO98]. Considerando os parâmetros $h = 4,0m$ e $U = 0,5m/s$ tem-se que $\varepsilon_f = 50cm^2/s$. Como parte significativa do movimento vertical é sob números de Courant menores que a unidade, aproximadamente 75%, um valor médio β é dado por $\beta = 0,0095cm^2/s$.

Esse exemplo, representativo da média do movimento vertical em modelos tridimensionais com as características discutidas, mostra que a difusão numérica do método β é menor que difusão física. E, portanto, o esquema numérico produzido sob tal abordagem é particularmente interessante no cálculo do transporte vertical, no tocante a minimização da difusão numérica.

Outros dois aspectos fundamentais do método β devem ser considerados. O primeiro refere-se à análise das oscilações numéricas e o segundo refere-se à eficiência computacional sob a abordagem paralela. Esses casos são tratados, respectivamente, nesse anexo e no capítulo 7.

Acurácia espacial e temporal de alta ordem

No contexto do cálculo numérico e computacional, a precisão é definida como o número de dígitos da mantissa e a exatidão como a qualidade desses dígitos [CLA94]. Sob o ponto de vista da modelagem computacional, a precisão (ou acurácia) é a qualidade da aproximação que um dado esquema de discretização tem em relação à solução exata [FLE88]. Assim, a acurácia é uma medida para verificar as aproximações para a solução analítica. A exatidão expressa a qualidade dessas medidas em relação aos valores verdadeiros.

Excetuando-se uma eventual influência da aritmética de ponto-flutuante na qualidade da solução numérica, o maior problema na construção de esquemas numéricos, particularmente responsável pelo surgimentos de difusão e oscilação numérica, é a falta de acurácia espacial e temporal nas aproximações das EDPs.

Para melhorar a acurácia temporal das soluções das equações que, se de primeira ordem, introduz um amortecimento artificial (*damping*) na solução, emprega-se o método- θ , que é um caso particular do método trapezoidal, como em [PAT87],

[ABB89], [HIR92], [CAS94] e [LOM2000], de modo que se pode obter uma acurácia temporal de segunda ordem.

Desse modo, o amortecimento artificial na solução pode ser reduzido ao mínimo quando as expressões discretas são aproximadas como uma média no tempo, como no esquema Crank-Nicolson. Quando se toma $\theta=1,0$ no método- θ , o esquema resultante é de primeira ordem de acurácia no tempo. Quando se toma $\theta = 0,5$, o esquema numérico tem acurácia temporal de segunda ordem. Especificamente no caso da hidrodinâmica toma-se $\theta = 0,5$, e os gradientes de pressão barotrópica (níveis) são aproximados como uma média nos níveis de tempo n e $n+1$. Patnaik [PAT87] mostrou que os algoritmos para escoamentos de superfície livre assim construídos são estáveis para $0,5 \leq \theta \leq 1$, e que a maior acurácia e eficiência é obtida para $\theta = 0,5$. Quando $\theta < 0,5$, o método é instável.

Objetivando-se os esquemas de alta qualidade numérica (*high resolution schemes*), ou seja, de esquemas de alta ordem de acurácia, faz-se uma discussão relativa aos principais conceitos e resultados associados ao método TVD (*Total Variation Diminishing*) de Harten [HAR83], e em particular, ao método FLM (*Flux Limiter Method*) de Sweby [SWE84].

Com essa abordagem obtêm-se soluções numéricas limitadas, acuradas e estáveis, que são características numéricas e computacionais intrínsecas do método FLM, tornando-o particularmente importante na solução de complexos problemas hiperbólicos. Essa discussão é associada a alguns dos importantes conceitos da Matemática como, por exemplo, leis de conservação escalar, soluções fracas, entropia e variação limitada, entre outras, que servem como digressão para abordar os conceitos de TVD e de limitação de fluxo, sem que eles sejam introduzidos de modo *ad hoc*, já que esses conceitos estão associados com a produção de soluções fisicamente corretas para a equação do transporte escalar.

Leis de conservação escalar

A *conservação* das propriedades de um sistema é um princípio fundamental da Física, que assegura que a massa, a quantidade de movimento e a energia podem ser transportadas no sistema, porém não podem aparecer ou desaparecer na ausência de fontes ou sumidouros. Essa afirmação pode ser expressa matematicamente através da EDP escrita, com uma condição inicial e o domínio de definição, formando um problema de valor inicial (PVI), como:

$$\left. \begin{aligned} \sigma_t(x,t) + f[\sigma(x,t)]_x &= 0 \quad ; x \in \square, t > 0 \\ \sigma(x,0) &= \sigma_0(x) \quad ; x \in \square \end{aligned} \right\} \quad (A1.5)$$

A função $f(\sigma)$ é chamada de *função fluxo numérico*. Supondo que ela se anula quando $|x| \rightarrow \infty$ (função com suporte compacto), a integração da EDP do PVI (A1.5) no

intervalo $-\infty < x < \infty$ resulta em $d/dt \int_{-\infty}^{\infty} \sigma(x,t) dx = 0$. Por esse motivo, essa EDP recebe o nome *lei de conservação escalar*, pois a integral da função σ é conservada no tempo. Integrar a lei de conservação escalar no intervalo $a < x < b$ resulta em:

$$\frac{d}{dt} \int_a^b \sigma(x,t) dx = f(\sigma(b,t)) - f(\sigma(a,t))$$

E portanto, a variação da integral de σ sobre o intervalo finito é determinada pelos contornos *inflow* e *outflow* em $x = a$ e $x = b$. Ou seja, somente através das

fronteiras do domínio de definição ocorre a variação de massa, por exemplo. Essas considerações são particularmente importantes quando associadas à necessidade de construir soluções hiperbólicas não lineares para escoamentos dominados pela advecção, como no caso deste trabalho.

Soluções fortes e soluções fracas

Para obter soluções que obedeçam à lei de conservação escalar (*scalar conservation law*), considera-se inicialmente a definição usual de solução. Uma solução forte (ou clássica) do problema de valor inicial (A1.5) é uma função σ que satisfaz simultaneamente a condição de regularidade, isto é, deve ser contínua em $\square \times [0, \infty)$ e diferenciável em $\square \times \square^+$, onde $\square^+ = (0, \infty)$, e as equações em (A1.5) ([MLO91] e [SNO98]). Assim, uma solução clássica de uma EDP é uma função, necessariamente diferenciável, que substituída na EDP resulta em uma identidade válida em cada ponto do domínio de definição.

Porém existem soluções de EDPs que são somente contínuas ou mesmo descontínuas e a substituição de uma função com essas características na equação diferencial não tem sentido. A partir dessa constatação pode-se introduzir o conceito de solução fraca que, no sentido das distribuições, considera que a função candidata a solução não seja necessariamente diferenciável. Nesse caso a solução fraca pode ser obtida, multiplicando a equação diferencial por uma *função teste suave* $\phi = \phi(x, t) \in C_0^\infty(\square \times \square^+)$, de suporte compacto, resultando, via integração por partes, na expressão:

$$\int_0^\infty \int_{-\infty}^\infty \left[\frac{\partial \phi}{\partial t} \sigma + \frac{\partial \phi}{\partial x} f(\sigma) \right] dx dt = - \int_{-\infty}^\infty \phi(x, 0) \sigma(x, 0) dx \quad (A1.6)$$

Assim, uma solução fraca de (A1.5) é uma função σ que satisfaz a condição de regularidade $\sigma \in C^0(\square \times [0, \infty))$, a equação (A1.6), e a CI de (A1.5).

É possível mostrar que solução forte implica solução fraca, ou seja, se σ é solução do problema (A1.5) no sentido da definição de solução forte, então também o é no sentido da definição de solução fraca. Além disso, a noção de solução fraca não introduz soluções diferenciáveis espúrias visto que uma função diferenciável, que não for solução forte, não pode ser solução fraca. Mais especificamente tem-se que solução fraca acrescida de regularidade implica solução forte. Ou seja, se σ for solução fraca e for diferenciável em $\square \times \square^+$, então σ é uma solução clássica. Todos esses resultados podem ser vistos em [MLO91] e [SNO98]. Conceitos e resultados adicionais sobre EDPs e distribuições podem ser encontrados em [IOR88], [MLO91] e [SAT98].

Condição de entropia e unicidade da solução

A questão toda, agora, é lidar com a consequência da admissão de soluções descontínuas. Soluções fracas (numéricas) não são únicas e, portanto, deve-se buscar obter a solução fisicamente correta (única). Um modo pelo qual se determina a unicidade da solução é exigir que se satisfaça a *condição de entropia* ([OSH84], [OLS97] e [TRA2000]). Tal condição requer que a solução física para a lei de conservação escalar seja o limite da solução da equação diferencial modificada com uma adição de difusão. Ou seja, para $\sigma = \lim_{\varepsilon \rightarrow 0} \sigma_\varepsilon$ (se $\varepsilon \rightarrow 0$ chama-se *vanishing viscosity*), tem-se:

$$\frac{\partial \sigma_\varepsilon}{\partial t} + \frac{\partial f(\sigma_\varepsilon)}{\partial x} = \varepsilon \frac{\partial}{\partial x} \left(\beta(\sigma) \frac{\partial \sigma_\varepsilon}{\partial x} \right)$$

onde $\beta(\sigma) > 0$. Sob essa condição mostra-se que as descontinuidades formadas no limite da *vanishing viscosity* satisfazem a *condição E* (condição de Rankine-Hugoniot), que assegura que a solução é unicamente determinada pelos dados iniciais. Detalhes sobre as leis de conservação hiperbólicas, e suas conexões com os chamados *problemas de Riemann*, podem ser vistos em [SNO98], [TRA2000] e [LEV2001].

O emprego de métodos numéricos para a solução de leis de conservação escalar (e de seus conceitos como solução fraca, condição de entropia, etc.) é fundamental no moderno tratamento numérico de problemas em escoamentos compressíveis como aqueles sob o escopo da Dinâmica de Gases (*Gas Dynamics*). Contudo, esses mesmos métodos podem ser empregados com sucesso para construir soluções numéricas de problemas em escoamentos incompressíveis com altos gradientes, como a equação do transporte escalar de substâncias (ETM), que é uma EDP hiperbólica não linear. Sua solução pode sofrer de fortes oscilações no entorno dos altos gradientes se o esquema de solução não for apropriadamente projetado.

Consistência e variação limitada

Pode-se obter um esquema numérico explícito na forma conservativa para a EDP em (A1.5), através de sua integração sobre $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [t^n, t^{n+1}]$. Dividindo essa integral por $\Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$ (malhas equiespaçadas), obtém-se o seguinte esquema numérico explícito:

$$S_i^{n+1} = S_i^n - \lambda \left(F_{i+\frac{1}{2}}^{n+\frac{1}{2}} - F_{i-\frac{1}{2}}^{n+\frac{1}{2}} \right)$$

onde $\sigma = S$ na EDP de (A1.5), $\lambda = \Delta t^{n+\frac{1}{2}} / \Delta x_i$ e $\Delta t^{n+\frac{1}{2}} = t^{n+1} - t^n$, onde:

$$S_i^n = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} S(ix, n\Delta t) dx \quad e \quad F_{i+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} f \left[S \left(x_{i+\frac{1}{2}}, t \right) \right] dt$$

sendo que S_i^n é visto como uma aproximação espacial média na célula para $S(ix, n\Delta t)$, e $F_{i+\frac{1}{2}}^{n+\frac{1}{2}}$ é visto como o fluxo médio que atravessa a face da célula no ponto $x_{i+\frac{1}{2}}$ no intervalo de tempo Δt ([GRS99], [ZOU99], [BYU99] e [AWA00]).

A questão central é mostrar que esse esquema numérico converge para a solução correta. Não existem resultados gerais que garantam a convergência de esquemas não lineares. Porém, se o esquema satisfizer determinadas condições, existem importantes resultados nesse sentido, sobretudo para o caso 1D.

Inicialmente, note-se que o clássico teorema de equivalência de Lax afirma que, para um esquema linear consistente, a estabilidade é condição necessária e suficiente para a estabilidade. Mas esse teorema não pode ser aplicado a esquemas não lineares e não garante convergência para soluções que satisfaçam a condição de entropia (solução única).

Por outro lado, o teorema de Lax-Wendroff [TRA2000] garante que, se uma solução de um esquema conservativo converge e é limitada, então o esquema converge para a solução fraca de uma lei de conservação escalar. Porém o teorema não garante que a solução satisfaz a condição *E* de entropia. Ou seja, é possível que um esquema numérico convirja para uma solução da lei de conservação escalar, mas não atenda a condição *vanishing viscosity*, mostrando que a solução não é única. Exemplos dessa situação podem ser vistos em [SNO98].

Um resultado que garante a unicidade da solução é o teorema de Harten-Hyman-Lax ([SNO98] e [TRA2000]), que assegura que um esquema explícito monotônico

conservativo e consistente satisfaz a condição de entropia e, portanto, tem solução única. Porém, nesse caso, a solução é somente de primeira ordem de acurácia. Uma questão importante é saber, então, se a estabilidade não linear implicaria a convergência de esquemas não lineares consistentes, como no teorema de Lax. Isso é verdade, sob certas condições, pois pode-se mostrar (veja o lema 3.3.2 de [TRA2000]) que a estabilidade não linear implica, sob condições adicionais, a convergência e unicidade de soluções não lineares.

Não se discutem, neste trabalho, esse nem outros mais resultados, que podem ser obtidos na bibliografia já mencionada, sobretudo em [OSH84], [SWE84] e [TRA2000]. Conceitos e definições sobre *soluções fracas* e suas conexões com EDPs e Mecânica dos Fluidos, sobretudo com leis de conservação de sistemas hiperbólicos (*hyperbolic conservation laws*) podem ser vistos em [MLO91], [LEV96], [HEM99] e [LEV2001]. O interesse é, doravante, obter esquemas numéricos de alta ordem de acurácia para escoamento incompressíveis.

Nesse sentido, a conexão dos métodos numéricos construídos neste trabalho, com resultados anteriormente discutidos, é que o mesmo suporte teórico dado para soluções compressíveis pode ser utilizado para construir soluções numéricas de escoamentos incompressíveis sem considerações puramente heurísticas. Um resultado fundamental nessa direção é dado pelo fato de que uma aproximação numérica converge limitadamente para a solução, sob certas condições, quando o esquema numérico é consistente e a variação total da solução é limitada no domínio de integração ([SWE84] e [TRA2000]).

Esse resultado mostra que uma estimativa envolvendo a convergência em esquemas conservativos é a limitação na variação da solução. A limitação pode ser obtida considerando-se o conceito TVD, que foi introduzido por Harten [HAR83]. Ele assegura que, na ausência de forças externas, não ocorre aumento na variação total da solução, não gerando, portanto, oscilações no entorno do choque ou dos altos gradientes. Nesse caso, deve-se abandonar a exigência de monotonicidade da solução, pois, pelo teorema de Godunov, esquemas lineares que preservam a monotonicidade são somente de primeira ordem de acurácia. Portanto, esquemas que preservam a monotonicidade devem ser de alta ordem de acurácia. Um modo de obter tais soluções é satisfazer a condição, menos restritiva, de o esquema ser TVD (*Total Variation Diminishing*).

Soluções escalares TVD: acurácia de alta ordem

O conceito TVD é particularmente importante para obter esquemas advectivos que combinem acurácia e monotonicidade. Esquemas explícitos, conservativos e monotônicos são TVD, porém têm acurácia somente de primeira ordem. Um resultado importante é que esquemas conservativos TVD e não lineares preservam a monotonicidade (teorema de Harten).

Uma função, para um problema 1D na forma conservativa que mede a variação total (TV) da solução $\{\Phi\}$ no tempo é definida por:

$$TV(\Phi^n) = \sum_i |\Phi_{i+1}^n - \Phi_i^n|$$

e um método de diferenças é dito ser de variação total amortecida (*Total-Variation-Diminishing* – TVD) se produz uma solução que satisfaz a condição:

$$\sum_{i=-\infty}^{\infty} |\Phi_{i+1}^{n+1} - \Phi_i^{n+1}| \stackrel{def}{=} TV(\Phi^{n+1}) \leq TV(\Phi^n) \stackrel{def}{=} \sum_{i=-\infty}^{\infty} |\Phi_{i+1}^n - \Phi_i^n|$$

Em um esquema TVD, não são criados novos máximos e novos mínimos no avanço temporal da solução. Essa é uma importante conseqüência, pois pode-se

mostrar que todos os esquemas conservativos monotônicos são TVD e todos os esquemas TVD preservam a monotonicidade e são convergentes [TRA2000]. Esquemas lineares monotônicos são de primeira ordem de acurácia. Mas essa restrição não se aplica a esquemas TVD não lineares.

Soluções para a equação de lei conservação escalar satisfazem a propriedade de que a variação total de qualquer solução contínua fisicamente admissível

$$TV(\Phi) = \int \left| \frac{\partial \Phi}{\partial x} \right| dx,$$

não cresce no tempo (teorema de Lax [TRA2000]). Assim, é “natural” requerer que a solução de um problema contínuo $\Phi = \Phi(x, t)$ satisfaça $d/dt [TV(\Phi)] \leq 0$, como na definição da lei de conservação escalar.

Esquemas explícitos lineares 1D são monotônicos se e somente se são TVD. Já esquemas explícitos não lineares 1D e TVD preservam a monotonicidade, mas esse resultado não é aplicável a problemas multidimensionais. Goodman e LeVeque [ZIJ97] mostraram que todo esquema conservativo TVD para problemas 2D deve ter primeira ordem de acurácia. Porém experimentos numéricos mostram que soluções multidimensionais livres de oscilações podem ser obtidas empregando esquemas TVD 1D de segunda ordem de acurácia. Então, por robustez, empregam-se decomposições unidimensionais para obter soluções 2D ou 3D, a despeito do fato de que o esquema resultante não é, a rigor, TVD. Em [ZIJ95] pode-se ver um contra-exemplo mostrando que esquemas explícitos monotônicos 2D não são necessariamente TVD.

Esses esquemas numéricos explícitos TVD podem ser obtidos através do emprego do método da limitação dos fluxos (FLM), no qual o fluxo em cada face é calculado como sendo a soma de um termo *upwind* difusivo de primeira ordem, mais um termo “antidifusivo” de alta ordem [HUN96]. Para manter a monotonicidade, empregam-se funções limitadoras de fluxo. Neste trabalho, empregou-se a formulação FLM desenvolvida por Sweby [SWE84] e uma ampla coleção de funções limitadoras de fluxo. Nesse caso, esquemas numéricos TVD de segunda ordem de acurácia devem ser não lineares, mesmo quando aplicados à solução da EDP da advecção linear. Além disso, se o esquema numérico é completamente implícito, ele tem estabilidade incondicional e, portanto, não necessita de limite superior [ZIJ97].

Porém, mesmo a condição TVD pode ser, também, restritiva, dado que pode sofrer de “achatamento” (*clipping*) nos pontos críticos. Uma solução para esse problema é relaxar o conceito de variação total. Por esse motivo, desenvolveram-se esquemas tipo essencialmente não oscilatórios (*Essentially Non Oscillatory* – ENO), mas essa abordagem não é tratada neste trabalho.

Consistência entre a continuidade e o transporte

Um outro atributo que deve ser observado na construção da solução para escoamentos de superfície livre é a questão da consistência entre a aproximação da equação da continuidade (EC) e a equação do transporte de substâncias (ETM). Essa propriedade foi definida por Lin [LIN96] para garantir que o campo inicial, escalar, uniforme e na ausência de fontes ou sumidouros permaneça uniforme ao longo da simulação. Sem essa consistência o campo inicial torna-se não uniforme e, segundo Lin, “esta aparente restrição muitas vezes não é mantida em modelos de transporte e isso causa fluxos espúrios no campo dos constituintes” ([GRO99] e [BON2000]).

A consistência da equação do transporte com a da continuidade (CTC) é definida como: “dado um campo de escoamento e uma condição inicial de concentração, a discretização da equação advectiva deve-se degenerar na discretização da equação da continuidade” ([BON2000]). Ou seja, a equação da advecção discreta é CTC satisfeita se, dado um escoamento inicial constante, sua discretização degenera-se na

discretização da equação da continuidade. Para ilustrar essa propriedade, considera-se o caso 2D como em (A1.7).

$$\left. \begin{aligned} \eta_{(i,j)}^{n+1} = \eta_{(i,j)}^n - \frac{\Delta t}{\Delta x} \left[H_{(i+\frac{1}{2},j)}^n U_{(i+\frac{1}{2},j)}^{n+\theta} - H_{(i-\frac{1}{2},j)}^n U_{(i-\frac{1}{2},j)}^{n+\theta} \right] \\ - \frac{\Delta t}{\Delta y} \left[H_{(i,j+\frac{1}{2})}^n V_{(i,j+\frac{1}{2})}^{n+\theta} - H_{(i,j-\frac{1}{2})}^n V_{(i,j-\frac{1}{2})}^{n+\theta} \right] \end{aligned} \right\} \quad (A1.7)$$

Assim, (A1.7) deve ser consistente com a EC discreta como em (A1.8), significando que a quantidade de água que entra/sai em cada face (fluxo numérico para a continuidade) é a mesma que entra/sai em cada face da célula para o transporte (fluxo numérico), carregando a concentração escalar.

$$\left. \begin{aligned} S_{(i,j)}^{n+1} H_{(i,j)}^{n+1} = S_{(i,j)}^n H_{(i,j)}^n - \frac{\Delta t}{\Delta x} \left[H_{(i+\frac{1}{2},j)}^n U_{(i+\frac{1}{2},j)}^{n+\theta} S_E^n - H_{(i-\frac{1}{2},j)}^n U_{(i-\frac{1}{2},j)}^{n+\theta} S_W^n \right] \\ - \frac{\Delta t}{\Delta y} \left[H_{(i,j+\frac{1}{2})}^n V_{(i,j+\frac{1}{2})}^{n+\theta} S_N^n - H_{(i,j-\frac{1}{2})}^n V_{(i,j-\frac{1}{2})}^{n+\theta} S_S^n \right] \end{aligned} \right\} \quad (A1.8)$$

que, para um campo escalar uniforme com concentração $S_{(i,j)}^n = 1$, é escrito como:

$$\begin{aligned} S_{(i,j)}^{n+1} H_{(i,j)}^{n+1} = H_{(i,j)}^n - \frac{\Delta t}{\Delta x} \left[H_{(i+\frac{1}{2},j)}^n U_{(i+\frac{1}{2},j)}^{n+\theta} - H_{(i-\frac{1}{2},j)}^n U_{(i-\frac{1}{2},j)}^{n+\theta} \right] \\ - \frac{\Delta t}{\Delta y} \left[H_{(i,j+\frac{1}{2})}^n V_{(i,j+\frac{1}{2})}^{n+\theta} - H_{(i,j-\frac{1}{2})}^n V_{(i,j-\frac{1}{2})}^{n+\theta} \right] \end{aligned}$$

Se os fluxos S_E^n , S_W^n , S_N^n e S_S^n são, também, iguais a 1, a substituição de (A1.7) em (A1.8) resulta que $S_{(i,j)}^{n+1} = 1$, que é chamada CTC. Essa condição é suficiente para garantir que o campo inicial, escalar e uniforme (na ausência de fontes ou sumidouros), permaneça uniforme ao longo da simulação.

Isso significa que se deve aproximar as EDPs discretas de modo que o nível de tempo, a localização das componentes das velocidades e do nível de água e o espaçamento vertical da ETM coincidam com as da EC. Além disso, os fluxos numéricos para as concentrações nas faces da célula devem ser interpolados na malha de modo que coincidam com o esquema advectivo da ESW. Essas condições devem ser atendidas pelos esquemas numéricos, já que os resultados de [GRO99] e [ROS2000] mostram que a acurácia da solução de escoamentos de superfície livre é preservada somente se a condição CTC é satisfeita.

Vários experimentos numéricos foram feitos por Bonaventura [BON2000] para determinar quais os efeitos da violação da condição CTC. Os dados iniciais para a velocidade e para a altura total foram escolhidos de modo que o campo de velocidades dado pela solução analítica da equação da conservação da quantidade de movimento (*shallow water*) não linear 1D. A profundidade total do fluido é calculado através da equação da continuidade (A1.7) e a concentração escalar calculada empregando a expressão (A1.8) e usando os valores da velocidade e da profundidade calculados previamente pelas equações da conservação da quantidade de movimento e da continuidade, respectivamente.

Esses experimentos foram conduzidos de modo a produzir soluções tais que a condição CTC estivesse satisfeita e de modo que ela fosse violada. A concentração inicial é tal que $S^0 = 1$ e todas as células cuja profundidade é menor que 10^{-4} m são consideradas células secas. Os esquemas numéricos são construídos usando o método Lax-Wendroff de reconstrução do fluxo nas faces das células. Esse método é de segunda ordem de acurácia e é do tipo FLM, da classe TVD, discutido detalhadamente

nesse anexo e no cap. 5. Os resultados obtidos são como mostrados nas tab. A1.1 e A1.2.

TABELA A1.1 - Resultados para o caso 1D com dados iniciais constantes, empregando o esquema FLM Lax-Wendroff (fonte [BON2000])

Abordagem × métrica	L_1	L_2	máximo	mínimo
CTC	$1,04 \cdot 10^{-8}$	$5,84 \cdot 10^{-8}$	1,00	1,00
não CTC	$6,25 \cdot 10^{-3}$	$4,97 \cdot 10^{-2}$	1,07	$3,16 \cdot 10^{-2}$

TABELA A1.2 - Resultados para o caso 1D com dados iniciais não constantes, empregando o esquema FLM Lax-Wendroff (fonte [BON2000])

abordagem × valores máximos	depois de 10 passos de tempo	depois de 20 passos de tempo	depois de 30 passos de tempo	depois de 40 passos de tempo
CTC	0,99	0,99	0,99	0,99
não CTC	1,07	1,06	1,06	1,05

Os resultados apresentados nas tab. A1.1 e A1.2 mostram que o esquema CTC não cria novos máximos, ao contrário do esquema que não conserva a propriedade CTC. Além disso, as métricas L_1 e L_2 (veja o anexo 2) mostram que os erros do esquema não CTC são maiores que aqueles obtidos com o esquema CTC, apesar do emprego de métodos TVD. Discussões adicionais podem ser vistos em [BON2000] mas esse resultado mostra que no desenvolvimento de modelos computacionais que oferecem a solução combinada para o escoamento e o transporte de substâncias deve-se considerar essa propriedade.

Positividade da solução

Quantidades negativas não têm sentido físico e são resultados da falta de acurácia, da não conservatividade e da perda da monotonicidade na solução numérica [LOO92]. Soluções positivas podem ser obtidas suprimindo os resultados negativos, mas essa alternativa é inadequada, pois esse procedimento adiciona massa à solução e não elimina os problemas associados com a geração dessas quantidades, como as sub/sobre-especificações (*under/overshoot*) dos valores nodais. Por exemplo, um esquema 2D é positivo se, dada uma variável $\varphi = \varphi(x, y, t)$, sua solução é positiva, isto é, se [TRA2000]:

$$\forall X = (x, y) \in \Omega \subset \mathbb{R}^2; \varphi(X, 0) \geq 0 \Rightarrow \varphi(X, t) \geq 0 \quad \forall X \times t \in \Omega \times (0, T)$$

Soluções positivas são obtidas através da construção de uma discretização espacial e temporal que mantenha a positividade da solução. Assim, os esquemas de solução a serem desenvolvidos são aqueles que devem ter, também, a propriedade de serem definidos-positivos. Geralmente, essa é uma característica possível de se obter com os esquemas não-lineares de alta resolução, de modo a não considerar viscosidade artificial. Porém, mesmo esses esquemas de alta resolução necessitam de características adicionais (que é a limitação do fluxo) para manter a positividade.

Transporte da solução

Outra característica importante que um esquema numérico deve possuir é a chamada propriedade de transporte. Essa propriedade é caracterizada como definido por Roache [CHE84]. “Um esquema tem a propriedade de transporte se as perturbações

em problemas com advecção dominante se propagam na direção do escoamento”. Esquemas tipo *upwind* têm essa característica; já esquemas em diferenças finitas centrais (DFC) não a têm. Numericamente esse resultado é como mostrado na fig A1.1 a), onde oscilações surgem quando do emprego de DFC na solução da advecção 1D.

O problema para esses esquemas é que enquanto a DFC tem transporte bidirecional não realístico para problemas hiperbólicos 1D, a abordagem *upwind* tem transporte unidirecional, que é verdadeiro sob o ponto de vista físico. Fisicamente, a característica “transportativa” da solução de problemas hiperbólicos têm direção preferencial de propagação ([MAL95] e [BRL2000]), onde as perturbações do escoamento propagam-se ou a *upstream* ou a *downstream*.

Um esquema numérico que obedece a propriedade transportativa considera que, no cálculo da advecção na malha numérica empreguem-se valores nodais que estão no sentido do transporte. Ou seja, que estão somente no *domínio de influência* da EDP discreta [MES76]. Essa é a essência de métodos tipo *upwind* (*upstream* e *downstream*), o que justifica a sua popularidade na solução numérica de problemas hiperbólicos.

Assim, grande parte dos algoritmos para a advecção partem do fato de que a solução de uma EDP hiperbólica propaga-se ao longo das suas curvas características. Métodos de aproximação tipo *upstream* pertencem à classe que considera esse aspecto, e o esquema mais popular dessa classe é o *upwind*, que é definido para a advecção linear 1D, onde Φ é uma variável qualquer, como:

$$\frac{\partial \Phi}{\partial t} = - \frac{u}{\Delta x} \begin{cases} \Phi_i - \Phi_{i-1}, & \text{se } u > 0 \\ \Phi_{i+1} - \Phi_i, & \text{se } u < 0 \end{cases}$$

onde, para $u > 0$, a solução no ponto i só depende do campo de escoamento *upstream*, ou seja, somente do ponto $i-1$ que está dentro do domínio de dependência do ponto i . As propriedades do ponto $i+1$ não influenciam o ponto i . Quando $u < 0$, ocorre que a solução só depende do escoamento *downstream*. Portanto, o estêncil computacional deve ser alinhado com a direção do escoamento através do sinal de u .

Muitos dos esquemas de alta resolução empregam essa abordagem na definição de suas estratégias. Por exemplo, os esquemas FLM podem ser obtidos a partir das idéias do esquema *upwind* considerando-se uma abordagem via “reconstrução linear” como discutida em [HUN95] e [HUN96]. Nesses esquemas numéricos (FLM) os fluxos nas faces de cada elemento fluido são calculados como sendo a combinação de um termo *upwind* difusivo de primeira ordem, com um termo “antidifusivo” de alta ordem. Nesse anexo discute-se a seguir sobre essa abordagem, e na seção 5.1.3 pode-se ver os algoritmos resultantes na obtenção de esquemas FLM de alta qualidade numérica.

Note-se, porém, que técnicas de solução numérica como a *equação de Poisson para pressão* (método de acoplamento pressão-velocidades), que consistem na substituição das derivadas parciais das expressões das ECQMs na expressão da EC geram uma EDP do tipo Poisson. Nesse caso, pode-se usar na sua solução numérica uma abordagem via diferenças finitas centrais sem que surjam oscilações, como pode ser visto, por exemplo, em [KLR2000] e em [BRL2002]. Isso se deve ao fato de que a equação de Poisson é um protótipo de EDP do tipo elíptico [IOV89], que não oferece problemas quando do emprego de esquemas numéricos via DFC. Ver discussões e detalhes sobre o método da equação de Poisson para pressão na seção 7.3 de [FER97] ou na seção 5.11 de [FRT2000].

Massa conservativa

O princípio da conservação da massa enuncia que a massa não pode ser criada nem destruída. Se a vazão para dentro de uma região Ω exceder aquela que sai, a massa acumular-se-á dentro dela. A conservação da massa exige que a soma da taxa

de variação de massa dentro da região com a vazão líquida em massa saindo pela fronteira $\partial\Omega$ seja nula ([HIR92] e [AND95])

Esse princípio é formulado matematicamente através da equação da continuidade que, quando escrita empregando um volume fixo no espaço, é chamada *conservativa*, enquanto a obtida com o volume movendo-se no fluido é chamada de *não conservativa*. A segunda forma emprega a derivada Lagrangeana, que é apropriada para “equacionar” o movimento de um corpo fluido. A primeira forma é obtida com a formulação Euleriana. E, por isso, EDPs na forma conservativa e não conservativa são aquelas, respectivamente, escritas com a formulação Euleriana e com a formulação Lagrangeana.

Esquemas semi-Lagrangeanos são, portanto, não conservativos, no sentido de que os fluxos nas faces não são calculados, como o caso daqueles esquemas construídos com a abordagem Euleriana, sendo que suas quantidades são calculadas ao longo das trajetórias das partículas. Uma discussão sobre esses aspectos pode ser vista em [AND95]. Porém, cada abordagem pode ser obtida a partir de outra, ou seja, pode-se mostrar que as diferentes formas da equação da continuidade são matematicamente equivalentes.

Uma definição numérica para a questão da conservatividade da massa é tal que “*uma solução é massa conservativa se o valor total das quantidades envolvidas no interior do domínio computacional não é alterado devido aos erros numéricos*” [VLA99]. O método de discretização é, pois, fundamental para gerar esquemas conservativos. Um método apropriado é o dos “volumes finitos” ou o volume de controle finito.

No caso de diferenças finitas para obter as EDPs discretas, duas abordagens são populares. Uma emprega a expansão em série de Taylor, e a outra considera a abordagem de volume de controle [AND95]. Na primeira, as equações são escritas na forma apropriada para representar um dado modelo, e emprega-se a série de Taylor de modo puramente formal para desenvolver as aproximações em diferenças finitas. Na segunda abordagem, emprega-se o conceito de *volume de controle* finito (VC) de modo a conservar uma quantidade física específica como a massa, ou o movimento ou a energia no VC.

Essa abordagem tem a vantagem de ser aplicável mais facilmente a problemas multidimensionais, considerando as condições de contorno e conservando as propriedades físicas do problema. Uma desvantagem é que a estimativa da acurácia depende da análise das funções interpolantes nas faces do VC, enquanto o emprego de séries de Taylor fornece diretamente essa informação sob a forma da ordem de truncamento das séries.

Neste trabalho, empregou-se o método de discretização de diferenças finitas e, para manter a conservatividade fez-se uma aproximação para os fluxos, em uma célula 3D, para as concentrações nas faces leste (E), oeste (W), norte (N), sul (S), superior (U) e inferior (L), como ilustradas na fig. A1.2.

Note-se que o *fluxo numérico* é definido, para a solução da equação de transporte escalar de substâncias 3D, como sendo $f[S(h,t)] = \nu S(h,t)$, onde $h = x$ ou y ou z e $\nu = u, v$ ou w , dependendo da direção coordenada. Assim, em um esquema conservativo 3D, os valores nas faces das células na malha Arakawa C (veja a fig. 3.1) devem ser interpolados, observando o posicionamento das variáveis como mostrado na fig. A1.2. Nas faces E, W, N, S, L e U os fluxos numéricos são escritos na forma conservativa como:

$$(uS_W)_{(i+\frac{1}{2},j,k)} = (uS_E)_{(i-\frac{1}{2},j,k)}, (vS_S)_{(i,j+\frac{1}{2},k)} = (vS_N)_{(i,j-\frac{1}{2},k)} \text{ e } (wS_L)_{(i,j,k+\frac{1}{2})} = (wS_U)_{(i,j,k-\frac{1}{2})}$$

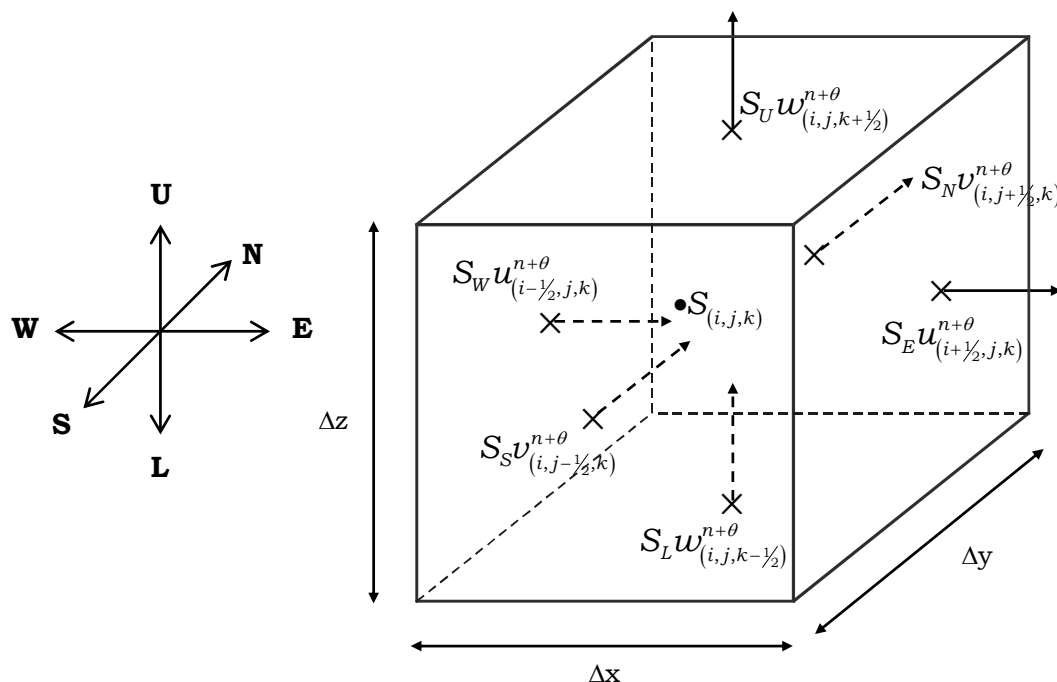


FIGURA A1.2 - Posicionamento dos fluxos numéricos nas faces da célula 3D

Porém, não obstante ser o método das diferenças finitas simples de formular, de estender para problemas multidimensionais, de ter a flexibilidade geométrica (com as técnicas de geração numérica de malhas) antes somente fornecida pelo método dos elementos finitos, e de ser ainda popular na área de *estudos ambientais* ([CAS94], [CHU96], [LIY96], [CAS97], [GRO98] e [GRO99]), o método dos volumes finitos (MFV) está cada vez mais presente nessa área como pode ser visto nos trabalhos de [ADC95], [VLA99], [VAL2000] e [CAS2000].

Nesse método, considera-se que as EDPs, na forma integral, devem ser resolvidas e discretizadas diretamente no espaço físico. O domínio solução é subdividido em um número finito de volumes elementares contíguos, nos quais a integração é executada, e as equações de conservação são aplicadas a cada elemento. Procedimentos de interpolação são empregados para representar os valores das variáveis nas faces da célula, em termos do valores nodais. Tal abordagem é conservativa por construção. Uma ampla discussão sobre aproximações de integrais de superfície e de volume, bem como de processos de discretização conservativa em volumes finitos, com aplicação à Dinâmica dos Fluidos Computacional, pode ser vista em [HIR92], [MAL95], [FER97] e [EYM97].

Note-se que o método dos volumes finitos (MVF) pode ser visto como sendo o método das diferenças finitas (MDF) aplicado às *leis de conservação* escritas na forma conservativa, com a vantagem de utilizar coordenadas arbitrárias [VAL2000]. Assim o MVF é vantajoso relativo ao MDF, pois combina a simplicidade do MDF com a flexibilidade do MEF, pois permite a aplicação às malhas não estruturadas.

Esquema semi-implícito

A discretização das EDPs deve levar em consideração o domínio de definição espacial e temporal. A discretização espacial consiste em construir uma malha através da qual o espaço contínuo é substituído por um número finito de pontos, nos quais os

valores numéricos das variáveis são determinados. A discretização temporal dos sistemas de equações determina duas famílias de métodos: os explícitos e os implícitos.

No método completamente explícito, a matriz dos coeficientes calculada a cada passo de tempo é diagonal. Sua solução é obtida eficientemente e sem dependência de dados, pois todos os valores vizinhos são conhecidos no mesmo passo de tempo. Contudo, essa vantagem é aparente, pelo fato de que as condições de estabilidade Courant-Friedrichs-Levy (CFL) impõem restrições à magnitude do passo de tempo. Essas condições CFL visam a assegurar que o domínio de dependência da solução analítica de EDPs esteja contido no domínio de dependência da solução numérica [BRT96].

Em uma abordagem completamente implícita, não há restrições no passo de tempo, pois o sistema de equações é incondicionalmente estável (estabilidade linear). Mas os sistemas de equações são não lineares e acoplados, sendo necessário resolvê-los simultaneamente, o que pode ser um problema, dada a exigência de memória e de capacidade de processamento. E, mesmo nesse caso, por acurácia, o passo de tempo não pode ser arbitrário, o que limita a utilização prática da estabilidade incondicional disponível [CAS90].

Uma alternativa aos esquemas explícitos ou implícitos, visando a melhorar o desempenho computacional, é empregar a técnica de divisão do modo (*mode-splitting*) separando as equações verticalmente integradas (*modo externo*) das equações que contêm a estrutura vertical (*modo interno*). Essa estratégia de desacoplar as direções horizontais e verticais considera a contribuição das diferentes escalas envolvidas em cada direção ([BLU78], [SHE95], [FOR96] e [MEL98]). Isso porque as equações que governam a dinâmica da circulação contêm movimentos rápidos, causados por ondas de gravidade externas (movimentos barotrópicos), e movimentos lentos, causados por ondas de gravidade interna (movimentos baroclínicos).

Com essa abordagem, o esquema de solução tem dois intervalos de tempo distintos, um para o modo externo e outro para o modo interno. O modo externo (barotrópico) é bidimensional e usa um intervalo de tempo menor, baseado nas condições de estabilidade CFL. O modo interno (baroclínico) é tridimensional e usa um intervalo de tempo mais longo [CAM99]. Esse método é eficiente, pois permite empregar um esquema de integração temporal que use os distintos passos de tempo para calcular os dois modos.

A desvantagem dessa abordagem é a inconsistência da solução numérica desses dois *modos* [CAS92]. Embora o modo externo permita um eficiente cálculo das ondas de propagação de gravidade na superfície livre, mesmo com um pequeno passo de tempo, e a solução do modo interno, que é computacionalmente custosa, possa ser obtida com um grande passo de tempo, deve-se assegurar a consistência das soluções desses dois modos e, conseqüentemente, a consistência das quantidades físicas derivadas dos modos externo e interno.

Essa consistência significa assegurar que as velocidades integradas verticalmente no modo externo devem ser tais como na integração vertical do perfil de velocidades da solução do modo interno, e as representações do atrito no leito do corpo de água, nos modos interno e externo, devem ser também consistentes [CAS92]. Se qualquer inconsistência existir, não se pode assegurar matematicamente a convergência da solução numérica, já que as aproximações em diferenças finitas para os respectivos modos não são consistentes. Outro problema da abordagem de divisão do modo é que, geralmente, alguns dos esquemas de integração têm acurácia somente de primeira ordem [LEE89].

Uma alternativa elegante, adotada neste trabalho, é a construção de esquemas semi-implícitos, procurando equilibrar as exigências da estabilidade com a obtenção de grandes passos de tempo. Essa abordagem é atualmente popular ([LEE89], [CAS90], [LEV92], [CAS94], [SON94] e [GRO98]), e muitas das pesquisas atuais buscam

fundamentar de modo matematicamente rigoroso os critérios para a construção de tais esquemas.

Esquemas semi-implícitos surgem quando se aproximam alguns dos termos constituintes das EDPs de modo explícito e outros de modo implícito. Com essa abordagem, pode-se construir sistemas de equações que são lineares, acurados, estáveis e computacionalmente eficientes. Além disso, o grau de “explicitidade” e de “implicidade” de tais esquemas pode, ainda, ser controlado, empregando-se o método trapezoidal ([PAT87], [ABB89] e [HIR92]). Foi essa a abordagem empregada neste trabalho, como discutido nesse anexo, e os esquemas numéricos resultantes dessa abordagem são discutidos no capítulo 4.

Uma outra vantagem da abordagem semi-implícita é, que ao desacoplar os termos, permite que os termos mais críticos sejam convenientemente discretizados. Especificamente, em escoamentos hidrodinâmicos e de transporte de substâncias, nos quais a advecção domina os demais termos, os termos advectivos podem ser aproximados por diferentes esquemas de alta qualidade numérica sem que ocorram maiores alterações no código computacional. Essa abordagem foi a empregada para obter as soluções da ETM e das ESW, como pode ser visto nos capítulos 4 e 5.

Esquemas Euleriano e semi-Lagrangeano

Usando a hipótese do contínuo, pela qual se consideram os efeitos médios ou macroscópicos de muita moléculas, pode-se utilizar o conceito de velocidade de um fluido para definir que o campo de escoamento é um agregado das velocidades locais [FOX98]. Isso permite definir duas classes de formulações para as EDPs: a Euleriana e a Lagrangeana.

A Euleriana é a usual, na qual as quantidades do escoamento são funções da posição (x,y,z) e do tempo t . A formulação Lagrangeana é baseada na trajetória da partícula que se move em um campo de escoamento, de modo que os esquemas numéricos sob essa abordagem são desenvolvidos sobre malhas móveis. Tal partícula está sujeita às taxas de variações espacial e temporal. A espacial, de uma localização para outra, representada pela derivada advectiva, e a temporal, para um ponto fixo do escoamento, dada pela derivada temporal. Essas variações são representadas pela derivada Lagrangeana (ou material ou substantiva) [BTC90].

A combinação das duas classes produz uma formulação mista, Euleriana-Lagrangeana, também conhecida como semi-Lagrangeana. Essa classe concilia a simplicidade da formulação Euleriana com a flexibilidade e a acurácia da Lagrangeana. É uma abordagem importante e popular, pois resulta em esquemas estáveis e com grandes passos no tempo ([CAS90], [STA91] e [COT98]).

Porém, um esquema semi-Lagrangeano oferece dificuldades adicionais no paralelismo, como discutido a seguir e, desse modo, optou-se para neste trabalho pela abordagem Euleriana, pela qual as malhas são fixas no tempo. Porém, mesmo não implementando a abordagem semi-Lagrangeana, dada a sua relevância e visando a trabalhos futuros, faz-se a seguir uma discussão sobre suas principais características numéricas e computacionais.

Como a parte implícita é incondicionalmente estável, a restrição no passo de tempo é devido somente aos termos explícitos das equações, cujo passo de tempo depende do número de Courant. Para a hidrodinâmica, a difusão horizontal, de advecção (horizontal e vertical) e de Coriolis são aproximados explicitamente. Para o transporte de substâncias a advecção e a difusão horizontais são aproximados explicitamente. Como um esquema semi-Lagrangeano (ESL) é explícito poder-se-iam aproximar todos esses termos empregando essa abordagem.

Porém, como a equação do transporte é uma EDP hiperbólica com termo de fonte emitindo altos gradientes de concentração a cada passo de tempo, uma escolha mais

apropriada na solução do transporte horizontal é o emprego do método de limitação de fluxo. Nesse caso pode-se usar métodos de integração em subciclos para não restringir o passo de tempo global do esquema de solução. Porém, as ESW não sofrem as restrições da equação do transporte e, portanto, pode-se usar os ESL de modo a conciliar a eficiência computacional com a qualidade numérica na obtenção se sua solução, como feito por [CHE84], [CAS90], [OLI95] e [COT98].

Desse modo e, objetivando uma discussão sistemática sobre os principais atributos numéricos e computacionais para modelos computacionais paralelos, pode-se considerar que abordagens via ESL devem ser parte integrante das características desses modelos, particularmente daqueles para o escoamento.

Uma aproximação via derivada Lagrangeana considera que o escoamento é calculado ao longo da linha de corrente, definida pelo campo de escoamento, cujos pontos em geral não pertencem à malha, já que os números de Courant $C_r \equiv C_r(x, y, z, t)$, definidos por:

$$C_{r_x} = \frac{\Delta t_{ADV_H} |u|}{\Delta x}, C_{r_y} = \frac{\Delta t_{ADV_H} |v|}{\Delta y} \text{ e } C_{r_z} = \frac{\Delta t_{ADV_V} |w|}{\Delta z}$$

não são números inteiros, sendo que Δt_{ADV_H} e Δt_{ADV_V} , Δx , Δy e Δz , $|u|$, $|v|$ e $|w|$ são: os passos de tempo para as advecções horizontal e vertical, os espaçamentos nas direções X , Y e Z , e os módulos das componentes das velocidades nessas direções, respectivamente. Portanto, deve-se usar processos de interpolação para definir essa aproximação. Além disso, sendo as EDPs não-lineares, as componentes da velocidade não são constantes. Nesse caso a determinação dos números de Courant requer a solução das EDOs $u = dx/dt$, $v = dy/dt$ e $w = dz/dt$ através do emprego de métodos de integração como, por exemplo, de Euler ou de Runge-Kutta ([CAS90], [ASE90], [CHE93] e [BON98]).

Assim, as duas etapas básicas do ESL são integrar a trajetória e definir interpolações *upstream* ([SMO92] e [BAT85]). A acurácia, a estabilidade e a difusão numérica do esquema numérico dependem dos métodos de interpolação e de integração empregados. Para definir qual método de interpolação pode ser empregado, deve-se considerar a dependência de dados na solução paralela.

Os métodos de interpolação podem ser classificados como locais e globais. Nos locais, usa-se somente um número fixo de pontos vizinhos da malha. Nos globais, usa-se um número de pontos que depende da ordem do sistema. Para essas categorias, foram feitos estudos teóricos e numéricos, e uma análise comparativa entre as interpolações de Hermite cúbica, spline cúbica, Lagrange de quarta ordem, e a Fourier, mostrou que ([LOO92], [MAL96] e [THO96]):

1. Hermite é local, e pode ser difusiva, levando à perda de massa;
2. Spline é global, e conserva a massa melhor que a de Hermite, porém requer CC adicionais e é computacionalmente mais custosa que a de Hermite e a de Lagrange, pois necessita resolver sistemas 3-diagonal;
3. Fourier é global, e fornece melhor acurácia, mas é mais custosa que as interpolações locais, além de requerer CC periódicas;
4. Lagrange é local, e necessita de mais pontos que a Hermite, mas fornece melhor acurácia. Desses métodos é o de melhor relação custo/benefício.

Os métodos de interpolação global são desvantajosos em soluções paralelas devido à dependência de dados. Métodos de interpolação local têm a vantagem de que as dependências ficam restritas às fronteiras dos processadores e dependem do tipo da interpolação e da ordem do esquema construído. Pela localidade dos dados e pela sua acurácia, o método de interpolação mais popular para construir o ESL é o Lagrange na

sua forma generalizada, obtido através da decomposição dos números de Courant em suas partes inteiras e fracionárias.

Assim, na paralelização dos procedimentos de interpolação e do processo de integração, deve-se identificar o estêncil computacional já, que ESLs consistem no cálculo das trajetórias das partículas e nas interpolações nos pontos de malha que circundam o ponto de partida, ocorrendo um problema de disponibilização de dados quando o ponto de origem e o de chegada estão em processadores diferentes.

Essa dependência é caracterizada observando que o número de Courant $C_r = \max(C_{rx}, C_{ry}, C_{rz})$ determina o quanto uma partícula pode se deslocar durante um mesmo passo de tempo e, portanto, determina a quantidade de dados a serem trocados, definindo o tamanho da malha de pontos ([THO96] e [DIN98]). Ou seja, o número de Courant fornece o número de células (ou fração de uma célula) que uma dada partícula é advectada em um passo de tempo.

Esquemas Eulerianos explícitos em diferenças finitas requerem que $C_r \leq 1$, significando que uma partícula de fluido não pode se mover mais que a magnitude do comprimento de uma célula dentro do mesmo passo de tempo. E, a quantidade de dados a ser trocada deve-se somente ao estêncil computacional, pois no esquema Euleriano, os valores do campo advectado são definidos pelo estêncil computacional que contém os pontos de malha circunvizinhos. O estêncil define a dependência de dados, ou seja, as necessidades de comunicação dos valores desses pontos depois de cada passo do tempo. Assim, basta construir uma região de sobreposição em cada processador.

A solução paralela para o ESL é análoga, devendo-se construir um halo (região sobreposta) que tenha todas as informações requeridas pelo processador no qual é alocada a computação. Mas a escolha de como construir esse halo é o que diferencia a paralelização do ESL do Euleriano, pois no primeiro caso deve-se verificar o quanto as partículas movem-se no campo de escoamento.

Isso ocorre pelo fato de que ESL não são restritos pelas condições CFL, significando que o número de Courant do esquema é tal que $C_r > 1$. Portanto, o comprimento da trajetória da partícula pode ser maior que o espaçamento da malha. Assim, a dependência de dados do ESL requer um halo de dimensão $\lceil C_r \rceil + 1$, como discutido em [THO96], [PAG97] e [THO97], dado que depende da velocidade máxima do escoamento cuja informação, porém, pode não estar disponível no atual passo de tempo. Uma solução é usar uma estimativa local para a velocidade máxima para manter regiões sobrepostas de modo dinâmico, que podem variar de tamanho ([THO94], [WOL94], [PAG97] e [RAD98]).

Se a integração da trajetória *upstream* necessitar de um grande halo de sobreposição, dada a estratégia de empregar a velocidade máxima como critério de especificação da dimensão da malha, isso pode limitar o desempenho computacional. Alternativamente a essa abordagem, pode-se determinar as trajetórias das partículas somente até a fronteira do subproblema. Os valores do campo *upstream* na intersecção de uma trajetória com a fronteira interprocessador são, então, calculados com um algoritmo de interpolação espaço-temporal ([MAL96], [THO97] e [PAG97]). Discussões sobre algumas estruturas de dados utilizadas em ESL podem ser vistas em [WOL94].

Note-se que, apesar da abordagem semi-Lagrangeana não sofrer limitação no passo de tempo, ela introduz certa difusão numérica. Essa viscosidade artificial pode ser calculada explicitamente para mostrar o grau de acurácia do esquema numérico ([CAS90] e [ASE90]). Uma análise da estabilidade do ESL, quando escrito na forma generalizada na qual se consideram pontos que não pertencem à malha, é encontrada em [BAT82] e [BAT85].

Esquemas de integração temporal

A presença de termos temporais nas EDPs requer que se empregue um esquema de integração no tempo para obter a solução evolutiva. Esses esquemas de integração temporais podem ser implícitos, explícitos ou semi-implícitos (implícitos-explícitos). O esquema de integração define o tipo de avanço no tempo.

Componente horizontal: integração em subciclos

Como destacado anteriormente os termos explícitos da ESW podem ser aproximados por esquemas semi-Lagrangeanos com grande eficiência computacional e mantendo uma boa qualidade numérica. Não se implementando neste trabalho essa abordagem, os termos explícitos foram aproximados usando a abordagem *upwind* para as advecções vertical e horizontal, e DFC para a difusão horizontal. Mas considerando-se sua relevância, e que os ESLs serão desenvolvidos em trabalhos que se seguirão à tese, discute-se nessa seção um esquema de integração temporal para a equação de transporte de substâncias.

Com isso poder-se-ia ter um passo de tempo global não limitado pelo número de Courant para o problema multifísica (escoamento e transporte). Mas isso somente se o transporte de substâncias puder ser resolvido em passos cuja magnitude não sejam, também, restritos pelo número de Courant.

Note-se que a solução da ETM tem estabilidade incondicional somente para o transporte vertical, que não necessita ser calculado com o mesmo passo de tempo do transporte horizontal, que é restrito pelas condições CFL. Então, deve-se desenvolver estratégias que permitam que o transporte horizontal seja resolvido juntamente com o transporte vertical. Uma alternativa é o uso de integração em *subciclos*, na qual os termos verticais são resolvidos somente uma vez a cada passo de tempo, enquanto o transporte horizontal é resolvido em subpassos de tempo. O resultado é uma melhor eficiência computacional.

O esquema em subciclos, como discutido nessa seção, também não foi implementado. Esse esquema foi desenvolvido por Gross ([GRO98] e [GRO99]), no qual a advecção horizontal é calculada empregando N_s subpassos e cujo subciclo no passo de tempo é $\Delta t_s = \Delta t / N_s$. Nesse esquema assegura-se que a condição de consistência da equação do transporte (ETM), com a equação da continuidade (da ESW) não seja violada. Nesse caso, no esquema de integração temporal em subciclos, o cálculo das substâncias considera inicialmente que:

$$\Delta \mathbf{z}_{(i,j,k)}^{\square n+(n_s/N_s)} = \Delta \mathbf{z}_{(i,j,k)}^{n+(n_s/N_s)} - \frac{\Delta t_s}{\Delta x} \left(u_{(i+1/2,j,k)}^{n+\theta} \Delta \mathbf{z}_{(i+1/2,j,k)}^n - u_{(i-1/2,j,k)}^{n+\theta} \Delta \mathbf{z}_{(i-1/2,j,k)}^n \right)$$

e o escalar é advectado na direção X para o passo subciclo $n_s + 1$ como:

$$\begin{aligned} \mathbf{S}_{(i,j,k)}^{\square n+(n_s/N_s)} \Delta \mathbf{z}_{(i,j,k)}^{\square n+(n_s/N_s)} &= \mathbf{S}_{(i,j,k)}^{n+(n_s/N_s)} \Delta \mathbf{z}_{(i,j,k)}^{n+(n_s/N_s)} \\ &\quad - \frac{\Delta t_s}{\Delta x} \left(u_{(i+1/2,j,k)}^{n+\theta} \Delta \mathbf{z}_{(i+1/2,j,k)}^n \mathbf{S}_{E(i+1/2,j,k)}^{n+(n_s/N_s)} - u_{(i-1/2,j,k)}^{n+\theta} \Delta \mathbf{z}_{(i-1/2,j,k)}^n \mathbf{S}_{W(i-1/2,j,k)}^{n+(n_s/N_s)} \right) \end{aligned}$$

e, em seqüência, o escalar é obtido através da advecção na direção Y como:

$$\begin{aligned} \mathbf{S}_{(i,j,k)}^{\square n+(n_s+1/N_s)} \Delta \mathbf{z}_{(i,j,k)}^{\square n+(n_s+1/N_s)} &= \mathbf{S}_{(i,j,k)}^{\square n+(n_s/N_s)} \Delta \mathbf{z}_{(i,j,k)}^{\square n+(n_s/N_s)} \\ &\quad - \frac{\Delta t_s}{\Delta y} \left(v_{(i,j+1/2,k)}^{n+\theta} \Delta \mathbf{z}_{(i,j+1/2,k)}^n \mathbf{S}_{N(i,j+1/2,k)}^{n+(n_s/N_s)} - v_{(i,j-1/2,k)}^{n+\theta} \Delta \mathbf{z}_{(i,j-1/2,k)}^n \mathbf{S}_{S(i,j-1/2,k)}^{n+(n_s/N_s)} \right) \end{aligned}$$

No início do subciclo, o nível de tempo da ETM é coincidente com o do esquema numérico da EC, que é o nível de tempo n . Porém, como o fluxo vertical necessita ser calculado antes do próximo passo de tempo de modo a atualizar o espaçamento

vertical no tempo $n+1$, ou seja, toma-se o espaçamento vertical como Δz_k^{n+1} , onde a altura obtida no final do esquema de subciclo deve ser definida, portanto, por questão de consistência, como sendo $\Delta z^{n+(N_s/N_s)}$. Assim, o transporte vertical é determinado como:

$$S_{(i,j,k)}^{n+1} \Delta z_{(i,j,k)}^{n+1} = S_{(i,j,k)}^{n+(N_s/N_s)} \Delta z_{(i,j,k)}^{n+(N_s/N_s)} - \Delta t_s \left(w_{(i,j,k+\frac{1}{2})}^{n+\theta} S_U - w_{(i,j,k-\frac{1}{2})}^{n+\theta} S_L \right) + D_V$$

onde D_V é a difusão vertical.

Integração temporal: interpolação

Para determinar o esquema de avanço temporal do problema multifísica, onde a solução da hidrodinâmica fornece os dados de entrada para a solução do transporte de substâncias, os passos de tempo para as ESW e para a ETM, devem estar relacionados entre si. Essa questão não pode ser desconsiderada, dado que os passos de tempo não são, necessariamente, um múltiplo de outro. Nessa situação a alternativa é empregar procedimentos de interpolação temporal.

Se as componentes da velocidade nos tempos t_{H1} e t_{H2} representam os finais de sucessivos passos de tempo na solução da ESW, pode-se obter as componentes de velocidades (ou níveis de água) em tempos intermediários de modo a se obter os valores de entrada para a ETM. Para isso, emprega-se uma fórmula de interpolação análoga a mostrada na fig. 3.16 que é usada para obter o refinamento espacial. Especificamente, as velocidades no tempo t tal que $t_{H1} \leq t \leq t_{H2}$ podem ser estimadas como em [ZHE95] e [ZHE99]:

$$\begin{aligned} v(t) &= v(t_{H1})(1 - \alpha_t) + v(t_{H2})\alpha_t \\ u(t) &= u(t_{H1})(1 - \alpha_t) + u(t_{H2})\alpha_t \end{aligned}$$

onde $\alpha_t = t - t_{H1} / t_{H2} - t_{H1}$ é o fator de interpolação linear no tempo.

A partir dessas considerações, e especificamente no caso deste trabalho, implementou-se o procedimento de interpolação espaço-temporal para a solução do problema multifísica (veja a seção 3.6.1), de modo que se definiu o fator temporal do refinamento como $\Delta t_T = [t_T - (t_H - \Delta t_H)] / \Delta t_H$, onde Δt_H e Δt_T são, respectivamente, os passos de tempo usados na solução da hidrodinâmica e do transporte escalar, e t_T o nível de tempo da solução do transporte. Assim, enquanto t_T satisfizer a condição $\Delta t_{ETM} < \Delta t_{ESW}$, as velocidades horizontais, por exemplo para a direção X, podem ser calculadas temporalmente usando as expressões:

$$\begin{aligned} U_L &= \left(\Delta U^{n+\theta} - U_{(i-\frac{1}{2},j)}^n \right) \Delta t_T \text{ onde } \Delta U^{n+\theta} = U_{(i-\frac{1}{2},j)}^{n+1} + U_{(i-\frac{1}{2},j)}^n \\ U_R &= \left(\Delta U^{n+\theta} - U_{(i+\frac{1}{2},j)}^n \right) \Delta t_T \text{ onde } \Delta U^{n+\theta} = U_{(i+\frac{1}{2},j)}^{n+1} + U_{(i+\frac{1}{2},j)}^n \end{aligned}$$

e, em que, $\Delta U = U_R - U_L$, onde ΔU é a diferença nas velocidades horizontais (componente U) na malha original, sendo, respectivamente, U_R e U_L as velocidades à direita e à esquerda na célula como na malha de Arakawa (fig. 3.1), e R , é o refinamento adotado. O caso para a componente V (e demais variáveis dependentes) é análogo. Na seção 3.6.1 faz-se uma discussão adicional a respeito desses procedimentos de interpolação, particularmente aquele de interpolação espacial, cujo algoritmo é como mostrado na fig. 3.16.

Assim, o esquema de avanço no tempo do transporte só pode ocorrer após ser completado o passo de tempo da hidrodinâmica. Para cumprir essa exigência, requer-

se que o nível de tempo da hidrodinâmica seja maior que aquele do transporte escalar, de modo que se assegura que as componentes de velocidades (solução das ESW) estarão disponíveis para a solução da ETM, quando da execução do seu esquema de integração espaço-temporal.

Além disso, visando a minimizar o aparecimento de oscilações e dispersão numérica, pode-se exigir que em modelos explícitos, o passo de tempo da solução do transporte seja menor que o passo da solução do escoamento. Nesse caso, um passo de tempo da hidrodinâmica pode corresponder a vários passos do transporte de substâncias. Nesse caso, a estratégia de subpassos no tempo também pode ser empregada, de modo a definir o passo de tempo global como sendo a soma dos subpassos de tempo.

Em qualquer situação deve-se considerar as restrições dos passos de tempo das equações envolvidas na simulação. Especificamente, no caso de uma solução explícita para o transporte, o passo de tempo é determinado pelo número de Courant, escrevendo-o como $\Delta t \leq C_r (\Delta x/|u|)$. Assim, pode-se mostrar [ZHE95] que, em geral, os passos de tempos para a advecção e para a difusão de modelos de transporte multidimensional explícitos são dados, respectivamente, por:

$$\Delta t \leq C_r \min\left(\frac{\Delta x}{|u|}, \frac{\Delta y}{|v|}, \frac{\Delta z}{|w|}\right) \text{ e } \Delta t \leq 0,5\left(\frac{\varepsilon_x}{\Delta x} + \frac{\varepsilon_y}{\Delta y} + \frac{\varepsilon_z}{\Delta z}\right)^{-1}$$

Quando a solução global é obtida através da combinação da solução de esquemas numéricos específicos para cada termo constituinte das EDPs (ou EIs), como o caso deste trabalho, o passo de tempo global é restrito pelo menor dos passos de um particular esquema de solução para um termo constituinte da equação. Por esse motivo deve-se analisar a combinação dessas soluções e empregar abordagens, como interpolação e subciclos, para melhorar o passo de tempo global do esquema. Uma discussão mais detalhada para o passo de tempo no transporte de substâncias que envolvem reações químicas e seus termos de fonte e de decaimento pode ser vista em [ZHE95].

Observe-se que a mesma idéia usada no esquema de interpolação temporal para o problema multifísica pode ser empregada para obter soluções para os problemas que tenham distintas atividades nos fenômenos simulados. Considere-se, por exemplo, uma simulação computacional de eventos que tenham períodos com mais atividades espaciais como, por exemplo, aqueles que envolvem a maré meteorológica, ciclos de ventos, etc., e que tenham em determinados períodos menos atividades, ou seja, sejam praticamente permanentes. Nesses casos, se as soluções dos fenômenos envolvidos permitir, pode-se aumentar o passo de tempo em alguns períodos, e diminuir o passo de tempo em outros períodos. Com isso pode-se melhorar a eficiência computacional e a estabilidade da solução.

Integração temporal: método ADI

A técnica de particionamento do tempo (*time-splitting*) mais popular é a *Alternating Direction Implicit* (ADI), desenvolvida originalmente por Peaceman e Rachford [FLE88a]. Nessa abordagem, o operador implícito multidimensional é reduzido a operadores unidimensionais, que obtêm, a cada subpasso de tempo, a sua solução, que, combinada com as obtidas nos outros subpassos, gera a solução global. Por exemplo, a aplicação do método ADI decompõe um sistema de equação linear *tipo 5-diagonal* de dimensão $n \times n$, em n sistemas de equações lineares 3-diagonal de n equações com n incógnitas no primeiro nível e no segundo nível de tempo.

Essa abordagem é, ainda, popular, uma vez que os sistemas de equações lineares 3-diagonal resultantes podem ser, então, resolvidos pelo algoritmo de Thomas, que é

um método direto com complexidade computacional da $O(N)$ (seqüencial). O algoritmo de Thomas resolve as matrizes em duas varreduras. Na primeira (*forward*), transforma a matriz, eliminando a diagonal inferior e normalizando a diagonal principal. Na segunda varredura (*backward*), obtém a solução do sistema de equações através de retro-substituição. Note-se que o algoritmo de Thomas é empregado neste trabalho não somente para resolver os esquemas numéricos 2D, mas também para resolver os milhares de sistemas de equações lineares 3-diagonal da ETM 3D, como discutido no capítulo 7.

O método ADI é, portanto, um método semi-implícito por construção, que resolve implicitamente em uma das direções, enquanto a outra direção é explícita. Assim, o método permite o uso de um passo de tempo superior àquele de uma abordagem puramente explícita.

Contudo, não obstante a eficiência do algoritmo de Thomas, o emprego da estratégia ADI pode ser uma fonte de inacurácia, conhecida na literatura técnica como *efeito* ADI, que surge na solução de problemas com geometria e/ou batimetria complexas que alteram o perfil de velocidades. Então, quando esses casos ocorrem, as duas “meias soluções” construídas pelo método ADI não geram a mesma solução do esquema completo ([STE86] e [CHE93]). Mas essa questão não foi explorada neste trabalho, onde se empregou o método ADI para se obter soluções para os modelos 2D.

Anexo 2 Métricas para avaliar a qualidade numérica

Soluções numéricas são obrigatoriamente aproximadas relativamente às soluções exatas. Essas soluções têm, além dos erros que podem, eventualmente, ser introduzidos na etapa de programação, três tipos de erros que surgem de forma sistemática ([FER97] e [CLA94]):

1. erros de modelagem: que são os erros que surgem da diferença entre o escoamento real e a solução exata do modelo matemático, que quase sempre é aproximado;
2. erros de discretização: que são aqueles de aproximação ou de truncamento, e que surgem da diferença entre a solução exata das equações na forma analítica, e a solução exata do sistema de equações algébricas obtida quando da discretização das equações analíticas;
3. erros de arredondamento: que são os erros introduzidos nos cálculos que são constantemente arredondados quando do uso de aritmética de ponto flutuante.

Erros de modelagem dependem das suposições que são feitas quando se obtêm as equações, ou na definição do domínio do problema, ou através da simplificação das condições iniciais e/ou de contorno. Esses erros não são conhecidos *a priori*, sendo que somente podem ser avaliados quando comparados com dados experimentalmente precisos ou com dados obtidos a partir de modelos mais precisos.

Erros de discretização são os erros produzidos quando se substitui qualquer processo contínuo ou infinito por um discreto ou finito. As aproximações geram erros que decrescem quando a malha é refinada. Porém, para uma dada malha, existem métodos de discretização de mesma ordem que podem produzir erros que diferem muito na ordem de magnitude [FER97]. A acurácia de um método ou esquema de aproximação é dada pela ordem do truncamento da aproximação espaço-temporal.

Os erros de arredondamento produzidos pelo computador são aqueles em que o arredondamento é feito devido à aritmética de ponto-flutuante. Cláudio [CLA94] faz uma esclarecedora discussão sobre questões imprescindíveis do cálculo numérico computacional, como sistema de ponto-flutuante, tipos de arredondamentos, erros e dígitos significativos exatos e precisão (número de dígitos da mantissa) e exatidão (qualidade desses dígitos) de máquinas digitais.

Neste trabalho, são empregadas EDPs que são casos particulares da equação de Navier-Stokes e que se mostram modelos acurados na representação do escoamento real. Erros de discretização são estudados quando da construção das aproximações para as EDPs na forma discreta, onde se busca construir esquemas numéricos que tenham acurácia temporal e espacial de, pelo menos, segunda ordem dado que essa escolha fornece, em geral, um bom equilíbrio entre a acurácia e a eficiência computacional de modelos de escoamento e de transporte escalar.

Pode-se analisar e avaliar a qualidade da aproximação do modelo discreto e os erros de arredondamento gerados pela solução desse modelo discreto, através do emprego de métricas apropriadas. Essas métricas verificam a conservatividade de determinadas propriedades físicas e/ou numéricas do esquema numérico e, em geral, consideram a diferença entre o valor arredondado e o valor exato.

Assim, se φ_i^* e φ_i^e são, respectivamente, a solução numérica e a solução discreta exata, define-se o erro absoluto E_A como $E_A = |\varphi_i^* - \varphi_i^e|$ e o erro relativo E_R como $E_R = |\varphi_i^* - \varphi_i^e| / \varphi_i^*$, se $\varphi_i^* \neq 0$, ou como $E_R = |\varphi_i^* - \varphi_i^e| / \varphi_i^e$, se $\varphi_i^e \neq 0$. Erros relativos são mais apropriados já que é mais significativo saber se o erro relativo é pequeno ou grande comparando-o com, por exemplo, 1,0 pois nesse caso $E_R = 1,0$ significa que o erro absoluto é da mesma magnitude que o número que está sendo aproximado

[CLA94]. Por exemplo, se o erro relativo de φ_i^* é da $O(10^{-5})$, diz-se que φ_i^* tem precisão ou acurácia de cinco dígitos decimais.

Erros entre escalares são mensurados através do emprego de expressões para erros relativos e/ou absolutos como acima definidos. No caso de vetores e matrizes é necessário estender esse conceito para medir o tamanho ou a *norma* de um vetor ou matriz ([GOL96] e [AXE96]). As normas mais populares para mensurar os erros são L_1 , L_2 e a L_∞ .

A análise dos resultados do modelo podem ser feitas, portanto, através do emprego de métricas específicas que visam a avaliar a qualidade de algoritmos numéricos, sobretudo aqueles que são particularmente desenvolvidos para avaliar esquemas advectivos. Se φ_i^* e φ_i^e são, respectivamente, a solução numérica e a solução exata em malhas equiespaçadas, algumas das métricas mais empregadas são:

1. razão do pico (*peak ratio*): medida da preservação do pico (melhor quando 1.0).

$$\max_{1 \leq i \leq N_x}(\varphi_i^*) / \max_{1 \leq i \leq N_x}(\varphi_i^e)$$

2. conhecimento para a razão do pico (*background to peak ratio*): medida dos *ripples* (oscilações) introduzidos por esquemas não monotônicos.

$$\min_{1 \leq i \leq N_x}(\varphi_i^*) / \max_{1 \leq i \leq N_x}(\varphi_i^e)$$

3. razão de massa (*mass ratio*): medida da característica da conservação da massa.

$$\sum_{i=1}^{N_x} \varphi_i^* / \sum_{i=1}^{N_x} \varphi_i^e$$

4. razão de distribuição (*distribution ratio*): medida da retenção da forma (*shape*) (melhor quando 1.0).

$$\sum_{i=1}^{N_x} (\varphi_i^*)^2 / \sum_{i=1}^{N_x} (\varphi_i^e)^2$$

5. erro absoluto médio (*average absolute error*): medida da diferença absoluta (melhor quando 0,0).

$$\frac{1}{N} \sum_{i=1}^{N_x} |\varphi_i^* - \varphi_i^e|$$

6. erro da raiz média quadrada (*root-mean square error*): medida da distribuição do erro (melhor quando 0,0).

$$\sqrt{\frac{1}{N} \sum_{i=1}^{N_x} (|\varphi_i^* - \varphi_i^e| / \varphi_i^e)^2}$$

7. norma L_1 (do erro relativo):

$$L_1 = \sum_{i=1}^{N_x} |\varphi_i^* - \varphi_i^e| / \sum_{i=1}^{N_x} |\varphi_i^e|$$

8. norma L_2 (Euclidiana):

$$L_2 = \left[\sum_{i=1}^{N_x} (\varphi_i^* - \varphi_i^e)^2 \right]^{1/2} / \left[\sum_{i=1}^{N_x} (\varphi_i^e)^2 \right]^{1/2}$$

9. norma L_∞ (do máximo):

$$L_\infty = \max_{1 \leq i \leq N_x} |\varphi_i^* - \varphi_i^e| / \max_{1 \leq i \leq N_x} |\varphi_i^e|$$

As seis primeiras métricas estão discutidas em “*Numerical Transport Algorithms for the Community Multiscale Air Quality (CMAQ): Chemical Transport Model in*

Generalized Coordinates [BYU99]. As três últimas em “*A Semi-Implicit Method for Vertical Transport in Multidimensional Models*” de [GRO98].

Para empregar essas métricas, considera-se como *benchmark* um modelo matemático 1D ou 2D cuja solução analítica é conhecida. O emprego de tais métricas, comparando a solução numérica com a solução analítica desses simples modelos, permite a verificação de qual o comportamento da solução de modelos mais complexos que pode ser visto como expressões multidimensionais desses modelos. Exemplos de modelos que têm solução analítica podem ser vistos em [FLE88a], [BOT89], [KIM94] e [BAP95].

Porém, para modelos simples como o da equação da advecção pura, a solução explícita exata $(S_i^{n+1})^e$ pode ser obtida considerando-se como solução aquela obtida através da expressão $(S_i^{n+1})^e = (S_i^n)^e - \lambda(F_R - F_L)^e$, onde $(S_i^n)^e$ é uma determinada concentração inicial, os fluxos nas faces são *massa conservativos* de modo que $(F_R)_{i-1}^e \equiv (F_L)_i^e$, e $\lambda = |v|\Delta h/\Delta t$ é o número de Courant. Assumindo que a solução numérica exata é completamente massa conservativa, é possível, então, comparar essa solução com aquela fornecida pela solução do modelo numérico implementado. Essa abordagem é a empregada em [GRO99] e é a que se considera neste trabalho.

No desenvolvimento de modelos operacionais, não se pode deixar de exigir sua verificação, calibração e validação. A verificação é um processo qualitativo de testar a capacidade do modelo para simular casos simples e de reagir a alterações de parâmetros. A calibração tem por objetivo a escolha dos parâmetros que melhor reproduzem um conjunto de dados. A validação objetiva verificar a capacidade de previsão do modelo, simulando as condições de funcionamento com base em parâmetros calibrados com conjuntos de dados independentes dos usados na calibração. Somente após atendidos os requisitos de verificação, calibração e validação, os modelos podem ser efetivamente usados como ferramenta de pesquisa e de simulação [CHE93].

Os modelos aqui desenvolvidos são aplicados à simulação dos eventos do Lago Guaíba. Para a realização dessa simulação, os dados da geometria do modelo são obtidos em [EAS98]. Porém alguns dos dados não estão disponíveis (vento, regime hídrico, etc;) e, portanto, esses dados são fictícios. Sendo assim, os modelos não estão calibrados e validados e ainda não podem ser usados como modelos operacionais. Nesse caso a ênfase foi, neste trabalho, dada à verificação das soluções nos seus aspectos numéricos e computacionais.

Anexo 3 Métricas para avaliar o desempenho da solução paralela

A decisão pela escolha desta ou daquela estratégia para construção de esquemas numéricos e na paralelização de aplicações é, muitas vezes, baseada na experiência pessoal e na literatura técnica. Portanto são necessárias métricas para avaliar a qualidade numérica e o desempenho do modelo resultante. No caso do desempenho de aplicações paralelizadas, algumas métricas experimentais empregadas são ([BAK96], [TOL97] e [DIN98]):

1. *tempo total de execução*: que é empregado na comparação do desempenho de dois algoritmos agindo sobre o mesmo conjunto de dados. Três tempos são importantes na execução de um algoritmo como deste trabalho: o tempo para calcular todo o pré-condicionamento e executar as instruções até começar a primeira iteração; o tempo de ciclos, no qual cada ciclo é definido como o número total de iterações necessárias para obter a precisão desejada; e o tempo de comunicação, que é o tempo gasto entre as comunicações coletivas do algoritmo para cada ciclo. Note-se que em execuções em paralelo o tempo a ser considerado é aquele entre o início do primeiro processo e o término do último processo;
2. *speedup* (aceleração): o *speedup* relativo realizado por um algoritmo é definido como $S_p(N) = T_1(N)/T_p(N)$, onde N é o tamanho da entrada; p , o número de processadores; $T_1(N)$ e $T_p(N)$ são, respectivamente, o tempo de execução do algoritmo paralelo com 1 e com p processadores. Então, $S_p(N)$ mede o fator de *speedup* obtido pelo algoritmo quando p processadores são utilizados. A comunicação, a insuficiência da concorrência e o *overhead* ocorrido na sincronização das atividades dos processadores são os principais fatores da diminuição do *speedup* em um algoritmo;
3. *eficiência*: é definida, para um algoritmo paralelo, por $E_p(N) = T_1(N)/p \cdot T_p(N)$, isto é, $E_p(N) = S_p(N)/p$. Assim, a $E_p(N)$ fornece uma indicação da efetiva utilização de p processadores relativas a um dado algoritmo. Se $E_p(N) \rightarrow 1$, o algoritmo é executado com máxima eficiência.

Tais métricas devem considerar a média de determinado número de experimentos, descartados os piores e os melhores resultados, de modo a evitar que resultados anormais influenciem as medidas tomadas. Note-se, porém, que a avaliação do desempenho de um algoritmo paralelo pode indicar a eficiência de tal algoritmo, mas não pode mostrar precisamente quais os principais fatores (insuficiência da concorrência, *overhead* na sincronização dos processadores, comunicação excessiva) que limitam essa eficiência. Em aplicações como a deste trabalho, é importante que o código seja otimizado, pois as simulações envolvem a solução de milhares de sistemas de equações.

Note-se que, sob o ponto de vista prático-computacional, outras variáveis como, por exemplo, o tempo de latência, as estruturas de dados, o tempo de comunicação entre nodos, o tempo gasto na comunicação global, entre outras, são determinantes na escalabilidade da solução paralela.

Anexo 4 Funções de restrição de fluxo

As tab. A4.1 a A4.45 mostram os resultados obtidos quando da aplicação das funções de restrição de fluxo 1. a 32. às funções de interpolação de fluxo para a advecção horizontal (6.8), empregando uma conhecida função teste, a “onda quadrada” (*square wave*), como discutido na seção 5.2.1.

Todos os tempos de CPU apresentados foram normalizados com respeito ao esquema *upwind* fluxo numérico. O esquema mais caro foi o *Kappa-DPL* que custou cerca de 9% a mais que o *upwind*, seguido pelo esquema *USR-Kappa*, que custou cerca de 8,4% a mais de tempo de CPU que o *upwind*. Note-se, porém, que a abordagem fluxo numérico que foi a empregada para obter a solução da ETM3D, necessita de uma estrutura de dados e o cálculo prévio de várias expressões como, por exemplo, os números de Courant locais, os números de Péclet locais, as funções monitoras, etc., tornando o esquema *upwind* computacionalmente custoso, apesar de não necessitar dessas expressões. Esta é uma desvantagem da abordagem unificada, discutida na seção 5.1.4, pois um esquema *upwind* de primeira ordem, como aquele das seções 6.2 e 6.3 é de 3 a 5 vezes menos custoso que um esquema via limitação de fluxo como, por exemplo, o Roe-Superbee.

Os resultados mostrados nas tab. A4.1 a A4.8 foram os obtidos considerando a onda quadrada de perfil $1 \times 10 \times 10$, onde a concentração inicial é $S^0 = 1,0$ número de Courant é $C_r = 0,25$.

TABELA A4.1 - Resultados para o teste com onda quadrada para $C_r=0,25$

métrica × esquema	<i>Upwind</i>	Chakravanthy-Osher	Sweby	Minmod
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,9492492	99,9999313	99,9999237	99,9998550
razão do pico (RP)	0,5878480	0,9891190	0,9999655	0,9493801
conhecimento para a RP	0,0000000	-0,0000001	-0,0000002	-0,0000001
razão de massa	0,9995630	0,9999990	0,9999991	0,9999975
1 - razão de massa	0,0004369	0,0000009	0,0000008	0,0000024
razão de distribuição	0,4113282	0,8493388	0,9191976	0,7751510
máximo (numérico)	0,5878480	0,9891190	0,9999655	0,9493801
mínimo (numérico)	0,0000000	-0,0000001	-0,0000002	-0,0000001
erro absoluto médio	0,0105681	0,0030464	0,0019725	0,0036551
erro da RMQ	0,2606461	0,1478916	0,1054213	0,1672280
norma L_1	0,9140417	0,2639966	0,1709055	0,3169290
norma L_2	0,5782939	0,2834283	0,2403581	0,2741262
norma L_∞	0,6653625	0,5091049	0,4528807	0,5024021
tempo total (normalizado)	1,0000	1,0474	1,0574	1,0417

TABELA A4.2 - Resultados para o teste com onda quadrada para $C_r=0,25$

métrica × esquema	Kappa-PSI	R-Kappa	Albada	HQUICK
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,9999389	99,9998855	99,9998626	99,9998779
razão do pico (RP)	0,9703954	0,9989491	0,9798042	0,6765065
conhecimento para a RP	-0,0000001	-0,0000001	-0,0000001	-0,0000000
razão de massa	0,9999990	0,9999997	0,9999996	0,9999355
1 - razão de massa	0,0000009	0,0000002	0,0000003	0,0000644
razão de distribuição	0,8052187	0,8890697	0,8375179	0,4772941
máximo (numérico)	0,9703954	0,9989491	0,9798042	0,6765065
mínimo (numérico)	-0,0000001	-0,0000001	-0,0000001	-0,0000000
erro absoluto médio	0,0031174	0,0023812	0,0028122	0,0090981

erro da RMQ	0,1539266	0,1242540	0,1461791	0,1210170
norma L_1	0,2702327	0,2065113	0,2440166	0,2102026
norma L_2	0,2489435	0,2517338	0,2442347	0,2591479
norma L_∞	0,4917238	0,4806831	0,4694606	0,4984738
tempo total (normalizado)	1,0581	1,0734	1,0502	1,0481

TABELA A4.3 - Resultados para o teste com onda quadrada para $C_r=0,25$

métrica × esquema	Hundsdoerfer	23A	23B	Kappa -MPL2
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,9998550	99,9999847	99,9999008	99,9998550
razão do pico (RP)	0,9493801	0,9762439	0,9576030	0,9493801
conhecimento para a RP	-0,0000001	-0,0000001	-0,0000001	-0,0000001
razão de massa	0,9999975	0,9999987	0,9999990	0,9999975
1 - razão de massa	0,0000024	0,0000012	0,0000009	0,0000024
razão de distribuição	0,7751510	0,8324185	0,8017629	0,7751510
máximo (numérico)	0,9493801	0,9762439	0,9576030	0,9493801
mínimo (numérico)	-0,0000001	-0,0000001	-0,0000001	-0,0000001
erro absoluto médio	0,0036551	0,0031013	0,0035008	0,0036551
erro da RMQ	0,1672280	0,1529972	0,1621616	0,1672280
norma L_1	0,3161290	0,2689443	0,3034467	0,3169290
norma L_2	0,2741263	0,2697800	0,2810633	0,2741263
norma L_∞	0,5024021	0,4744279	0,4862536	0,5024021
tempo total (normalizado)	1,0692	1,0426	1,0625	1,0492

TABELA A4.4 - Resultados para o teste com onda quadrada para $C_r=0,25$

métrica × esquema	Roe- Superbee	Ospre	CHARM	ISNAS
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,9998550	99,9999008	99,9998855	99,9998855
razão do pico (RP)	0,9493801	0,9961363	0,9972438	0,9972438
conhecimento para a RP	-0,0000001	-0,0000001	-0,0000001	-0,0000001
razão de massa	0,9999975	0,9999993	0,9999988	0,9999988
1 - razão de massa	0,0000024	0,0000006	0,0000011	0,0000011
razão de distribuição	0,7751510	0,8738129	0,8766015	0,8766015
máximo (numérico)	0,9493801	0,9961363	0,9972438	0,9972438
mínimo (numérico)	-0,0000001	-0,0000001	-0,0000001	-0,0000001
erro absoluto médio	0,0036551	0,0023776	0,0026289	0,0026289
erro da RMQ	0,1672280	0,1302017	0,1138142	0,1356017
norma L_1	0,3169290	0,2059893	0,2273756	0,2278175
norma L_2	0,2741263	0,2346026	0,2658071	0,2658083
norma L_∞	0,5024021	0,4556998	0,4998423	0,4998427
tempo total (normalizado)	1,0752	1,0611	1,0469	1,0436

TABELA A4.5 - Resultados para o teste com onda quadrada para $C_r=0,25$

métrica × esquema	MUSCL	SMART	Koren	Davis
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,9999542	99,9999084	100,000007	100,000015
razão do pico (RP)	1,0000021	1,0000015	1,00000190	0,9944755
conhecimento para a RP	-0,0000002	-0,0000002	-0,0000002	-0,0000001
razão de massa	0,9999992	0,9999997	1,0000000	1,0000004
1 - razão de massa	0,0000007	0,0000002	0,0000000	0,0000004
razão de distribuição	0,9136298	0,9223230	0,9132687	0,8274675
máximo (numérico)	1,0000021	1,0000015	1,0000019	0,9944755
mínimo (numérico)	-0,0000002	-0,0000002	-0,0000002	-0,0000001

erro absoluto médio	0,0020107	0,0020330	0,0020438	0,0028192
erro da RMQ	0,0985155	0,0971163	0,0999777	0,1280482
norma L_1	0,1741387	0,1759156	0,1767740	0,2438331
norma L_2	0,2382822	0,2494198	0,2405929	0,2446223
norma L_∞	0,4596851	0,4964368	0,4677534	0,5160988
tempo total (normalizado)	1,0502	1,0780	1,0460	1,0763

TABELA A4.6 - Resultados para o teste com onda quadrada para $C_r=0,25$

métrica × esquema	UMIST	van Leer	HCUS	Kappa-USR
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,9999771	99,9999237	99,9998703	99,9999847
razão do pico (RP)	0,9999982	0,9998152	0,9996956	1,0000025
conhecimento para a RP	-0,0000001	-0,0000002	-0,0000001	-0,0000002
razão de massa	0,9999994	1,0000000	0,9999990	0,9999995
1 - razão de massa	0,0000005	0,0000000	0,0000009	0,0000004
razão de distribuição	0,8898882	0,8912259	0,8951180	0,9096163
máximo (numérico)	0,9999982	0,9998152	0,9996956	1,0000025
mínimo (numérico)	-0,0000001	-0,0000002	-0,0000001	-0,0000002
erro absoluto médio	0,0022301	0,0022000	0,0023073	0,0020346
erro da RMQ	0,1048088	0,0966844	0,1190534	0,0978036
norma L_1	0,1930242	0,1902788	0,1998395	0,1761504
norma L_2	0,2370105	0,2341371	0,2498478	0,2367933
norma L_∞	0,4584135	0,4473416	0,4809865	0,4541757
tempo total (normalizado)	1,0523	1,0710	1,0427	1,0836

TABELA A4.7 - Resultados para o teste com onda quadrada para $C_r=0,25$

métrica × esquema	Kappa-MPL1	MC	Kappa-USPL	Kappa-DPL
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	100,000007	99,9999542	99,9999160	100,000091
razão do pico (RP)	0,9152643	1,0000021	1,0000026	1,5426653
conhecimento para a RP	-0,0000000	-0,0000002	-0,0000002	-0,3802250
razão de massa	0,9999996	0,9999992	0,9999991	1,0000008
1 - razão de massa	0,0000003	0,0000007	0,0000008	0,0000008
razão de distribuição	0,6775767	0,9136298	0,9358098	1,2563618
máximo (numérico)	0,9152643	1,0000021	1,0000026	1,5426653
mínimo (numérico)	-0,0000000	-0,0000002	-0,0000002	-0,3802250
erro absoluto médio	0,0052775	0,0020107	0,0019545	0,0046004
erro da RMQ	0,1658350	0,0985155	0,0940280	2,1725196
norma L_1	0,4568598	0,1741387	0,1691091	0,3978886
norma L_2	0,3593263	0,2382822	0,2567310	0,4076789
norma L_∞	0,6133803	0,4596851	0,5023499	0,7822408
tempo total (normalizado)	1,0545	1,0609	1,0665	1,0901

TABELA A4.8 - Resultados para o teste com onda quadrada para $C_r=0,25$

métrica × esquema	Kappa-Fromm	Kappa-Linear	Kappa-Simétrico	Kappa-QUICK
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,9974594	99,9999542	99,9903640	100,000007
razão do pico (RP)	1,7598128	1,5806337	1,7960346	1,2964500
conhecimento para a RP	-0,7200535	-0,4993159	-0,7538137	-0,2477011
razão de massa	0,9999772	1,0000000	0,9998646	0,9999991
1 - razão de massa	0,0000227	0,0000000	0,0001353	0,0000008
razão de distribuição	1,6076314	1,4927718	1,4873795	1,1825431
máximo (numérico)	1,7598128	1,5806337	1,7960346	1,2964500

mínimo (numérico)	-0,7200535	-0,4993159	-0,7538134	-0,2477011
erro absoluto médio	0,0047214	0,0044436	0,0061855	0,0030853
erro da RMQ	0,1308078	0,1237403	0,2708967	0,1205898
norma L_1	0,4086001	0,3847036	0,5380776	0,2669841
norma L_2	0,4302226	0,3764522	0,5117478	0,2790092
norma L_∞	0,7598128	0,5806337	0,7960346	0,4957540
tempo total (normalizado)	1,0427	1,0542	1,0568	1,0464

A abordagem fluxo numérico conserva bem a massa total para quase todos os esquemas de solução desenvolvidos e implementados. Nesse aspecto o pior esquema foi o *upwind* que perdeu aproximadamente 0,051% de massa para um tempo total de 50s, o que é significativo.

Todos os esquemas não lineares, exceto o K-DPL e o HQUICK, preservam bem o pico, indicando que a concentração máxima não é aumentada ou diminuída. Nesse aspecto destacam-se os esquemas de Sweby, o MUSCL, o SMART, o Koren, o UMIST, o MC, o K-USR e o USPL. Já o esquema *upwind*, que é linear de primeira ordem, e todos os esquemas lineares de segunda ordem da tab. A4.8, não preservaram o pico. Enquanto os esquemas *upwind* e HQUICK suavizaram o perfil em 41,1%, 22,1% e 32,34%, respectivamente, os esquemas K- Fromm, K-Linear, K-Linear simétrico e K-QUICK adicionaram ao pico 75,98%, 58,06%, 79,60% e 29,65% respectivamente. O esquema K-DPL adicionou ao pico 54,27%, mas esse esquema só é passível de ser empregado com certa difusão.

O *conhecimento para a razão do pico* mede as oscilações introduzidas por esquemas não monotônicos. Os resultados da aplicação dessa métrica aos esquemas mostrados nas tab. A4.1 a A4.8, exceto a função K-DPL, indicam que esses esquemas não introduzem oscilações. Note-se, porém, que devido à aritmética de ponto flutuante e/ou as condições de contorno estabelecidas na malha de teste, as funções limitadoras de fluxo não evitam completamente o aparecimentos de valores negativos. Já os resultados dos esquemas lineares de alta ordem tabelados em A4.8 mostram que eles introduzem fortes oscilações. Isso ocorre devido a perda de simetria do esquema numérico e/ou a inexistência da propriedade transportativa, como visto no anexo 1.

A *razão de distribuição* indica o comportamento do formato original do perfil de concentração. Os resultados dos esquemas analisados nas tab. A4.1 a A4.7 mostram que o formato do perfil é retido. As exceções são o esquema *upwind*, e os esquemas Minmod, Hundsdorfer, K-MPL2, Roe-Superbee, MPL1 e o K-DPL que também mantém o formato do pulso onda quadrada, mas já não tão bem quanto os demais. O esquema HQUICK é o pior dentre os não lineares, distorcendo muito o perfil da concentração de substâncias. Todos os esquemas lineares da tab. A4.8 não preservam o pulso.

As métricas dos erros absoluto médio e da raiz média quadrada, e as três normas refletem as análises anteriores mostrando que as soluções obtidas com o emprego de esquemas não lineares têm um comportamento muito melhor que aqueles obtidos por esquemas lineares.

Para dar continuidade na discussão e análise da influência do campo de velocidades na qualidade dos esquemas explícitos para a advecção horizontal, fez-se outros experimentos numéricos, como em [ZIJ95] e [ZIJ97], considerando o número de Courant $C_r = 0,40$. Note-se, porém, que todos os 32 esquemas são estáveis, mas cada qual tem uma determinada condição de estabilidade, dada pelo número de Courant. Assim, alguns dos esquemas não lineares testados, como o SMART e o K-USPL, somente oferecem bons resultados para $C_r = 0,33$.

As tab. A4.9 a A4.14, e A4.15, mostram os resultados dos experimentos para os números de Courant $C_r = 0,40$ e $C_r = 0,33$, que foram obtidos para a onda quadrada com o perfil anterior, porém com o campo de velocidades constante

$(u;v) = (0,40m/s;0,0m/s)$ e $(u;v) = (0,33m/s;0,0m/s)$, respectivamente. Nesses casos, a distância de advecção é $80\Delta x$ e $66\Delta x$, respectivamente, e são obtidas depois de 200 passos de tempo. Nesses experimentos empregou-se uma malha de 150×150 pontos.

TABELA A4.9 - Resultados para o teste com onda quadrada para $C_r=0,40$

métrica × esquema	<i>Upwind</i>	Chakravan- thy-Osher	Sweby	Minmod
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,9996944	99,999885	99,999916	99,999877
razão do pico (RP)	0,5323314	0,9986966	1,0000025	0,9818968
conhecimento para a RP	0,0000000	-0,0000002	-0,0000003	-0,0000001
razão de massa	0,9999978	0,9999987	0,9999990	0,9999980
1 - razão de massa	0,0000021	0,0000012	0,0000009	0,0000019
razão de distribuição	0,3681417	0,8983882	0,9369926	0,8479105
máximo (numérico)	0,5323314	0,9986966	1,0000025	0,9818968
mínimo (numérico)	0,0000000	-0,0000002	-0,0000003	-0,0000001
erro absoluto médio	0,0096695	0,0049216	0,0049245	0,0049216
erro da RMQ	18,0546593	0,0962203	0,0912286	0,1068887
norma L_1	1,9634893	0,9993949	0,9999728	0,9993994
norma L_2	1,1556292	0,9478329	0,9679837	0,9208202
norma L_∞	0,9985898	1,0000001	1,0000025	1,0000001
tempo total (normalizado)	1,0000	1,0593	1,0672	1,0602

TABELA A4.10 - Resultados para o teste com onda quadrada para $C_r=0,40$

métrica × esquema	Kappa-PSI	R-Kappa	Albada	Hundsdofer
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,999893	99,9999237	99,999839	99,999877
razão do pico (RP)	0,9923493	0,9999930	0,9945154	0,9818968
conhecimento para a RP	-0,0000002	-0,0000002	-0,0000002	-0,0000001
razão de massa	0,9999982	0,9999989	0,9999984	0,9999980
1 - razão de massa	0,0000017	0,0000010	0,0000015	0,0000019
razão de distribuição	0,8749243	0,9174069	0,8867371	0,8479105
máximo (numérico)	0,9923493	0,9999930	0,9945154	0,9818968
mínimo (numérico)	-0,0000002	-0,0000002	-0,0000002	-0,0000001
erro absoluto médio	0,0049217	0,0049238	0,0049244	0,0049216
erro da RMQ	0,1014520	0,0928339	0,1125004	0,1068887
norma L_1	0,9994016	0,9998377	0,9999647	0,9993994
norma L_2	0,9353734	0,9578136	0,9416672	0,9208202
norma L_∞	1,0000002	1,0000002	1,0000002	1,0000001
tempo total (normalizado)	1,0772	1,0761	1,0679	1,0677

TABELA A4.11 - Resultados para o teste com onda quadrada para $C_r=0,40$

métrica × esquema	23A	23B	Kappa -MPL2	Roe- Superbee
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,999809	99,999809	99,999877	99,999877
razão do pico (RP)	0,9928336	0,9798792	0,9818968	0,9818968
conhecimento para a RP	-0,0000002	-0,0000001	-0,0000001	-0,0000001
razão de massa	0,9999980	0,9999987	0,9999980	0,9999980
1 - razão de massa	0,0000019	0,0000012	0,0000019	0,0000019
razão de distribuição	0,8792042	0,8517879	0,8479105	0,8479105
máximo (numérico)	0,9928336	0,9798792	0,9818968	0,9818968
mínimo (numérico)	-0,0000002	-0,0000001	-0,0000001	-0,0000001

erro absoluto médio	0,0049244	0,0049243	0,0049242	0,0049242
erro da RMQ	0,1144533	0,1207351	0,1184705	0,1184705
norma L_1	0,9999576	0,9999480	0,9999197	0,9999197
norma L_2	0,9376589	0,9229235	0,9208205	0,9208205
norma L_∞	1,0000001	1,0000000	1,0000001	1,0000001
tempo total (normalizado)	1,0596	1,0795	1,0844	1,0745

TABELA A4.12 - Resultados para o teste com onda quadrada para $C_r=0,40$

métrica × esquema	Ospre	CHARM	ISNAS	MUSCL
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,999824	99,999931	99,9999384	99,9999464
razão do pico (RP)	0,9998095	0,9999422	0,9999424	1,0000067
conhecimento para a RP	-0,0000002	-0,0000002	-0,0000002	-0,0000003
razão de massa	0,9999985	0,9999989	0,9999989	0,9999995
1 - razão de massa	0,0000014	0,0000010	0,0000010	0,0000004
razão de distribuição	0,9123188	0,9096885	0,9096882	0,9353405
máximo (numérico)	0,9998095	0,9999422	0,9999424	1,0000067
mínimo (numérico)	-0,0000002	-0,0000002	-0,0000002	-0,0000003
erro absoluto médio	0,0049236	0,0049236	0,0049245	0,0049235
erro da RMQ	0,0951911	0,0957071	0,1024183	0,0845029
norma L_1	0,9997862	0,9998001	0,9999742	0,9997790
norma L_2	0,9551536	0,9537758	0,9537757	0,9671299
norma L_∞	1,0000002	1,0000002	1,0000002	1,0000067
tempo total (normalizado)	1,0668	1,0687	1,0698	1,0744

TABELA A4.13 - Resultados para o teste com onda quadrada para $C_r=0,40$

métrica × esquema	Koren	Davis	UMIST	van Leer
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,999954	99,999969	100,000007	100,000015
razão do pico (RP)	1,0000064	0,9996923	1,0000065	1,0000040
conhecimento para a RP	-0,0000003	-0,0000002	-0,0000002	-0,0000002
razão de massa	0,9999996	0,9999995	0,9999996	0,9999997
1 - razão de massa	0,0000003	0,0000004	0,0000003	0,0000002
razão de distribuição	0,9320565	0,8946171	0,9236349	0,9226205
máximo (numérico)	1,0000064	0,9996923	1,0000065	1,0000040
mínimo (numérico)	-0,0000003	-0,0000002	-0,0000002	-0,0000002
erro absoluto médio	0,0049246	0,0049232	0,0049246	0,0049245
erro da RMQ	0,0906872	0,0959640	0,0912286	0,0941508
norma L_1	0,9999911	0,9997251	0,9999925	0,9999858
norma L_2	0,9654307	0,9458418	0,9610593	0,9605313
norma L_∞	1,0000064	1,0000002	1,0000065	1,0000040
tempo total (normalizado)	1,0748	1,0539	1,0591	1,0851

TABELA A4.14 - Resultados para o teste com onda quadrada para $C_r=0,40$

métrica × esquema	HCUS	Kappa-USR	Kappa-MPL1	MC
massa total (exata)	100,000000	100,000000	100,000000	100,000000
massa total (numérica)	99,999969	99,999931	99,9999384	99,999946
razão do pico (RP)	1,0000015	1,0000066	0,9674986	1,0000067
conhecimento para a RP	-0,0000003	-0,0000003	-0,0000001	-0,0000003
razão de massa	0,9999990	0,9999993	0,9999989	0,9999995
1 - razão de massa	0,0000009	0,0000006	0,0000010	0,0000004
razão de distribuição	0,9210738	0,9345550	0,7520306	0,9353405
máximo (numérico)	1,0000015	1,0000066	0,9674986	1,0000067
mínimo (numérico)	-0,0000003	-0,0000003	-0,0000001	-0,0000003

erro absoluto médio	0,0049241	0,0049245	0,0049228	0,0049235
erro da RMQ	0,0925683	0,0882097	0,1122813	0,0845029
norma L_1	0,9998963	0,9999876	0,9996357	0,9997790
norma L_2	0,9597258	0,9667238	0,8671968	0,9671299
norma L_∞	1,0000015	1,0000066	1,0000001	1,0000067
tempo total (normalizado)	1,0538	1,0687	1,0738	1.0633

TABELA A4.15 - Resultados para o teste com onda quadrada para $C_r=0,33$

métrica × esquema	SMART	Kappa-USPL
massa total (exata)	100,000000	100,000000
massa total (numérica)	99,9999614	99,9999618
razão do pico (RP)	1,0000066	-0,0000003
conhecimento para a RP	-0,0000003	1,0000073
razão de massa	0,9999996	0,9999996
1 - razão de massa	0,0000003	0,0000003
razão de distribuição	0,9292561	0,9388055
máximo (numérico)	1,0000066	1,0000073
mínimo (numérico)	-0,0000003	-0,0000003
erro absoluto médio	0,0016447	0,0016452
erro da RMQ	0,1667131	0,2311642
norma L_1	0,3339843	0,3340850
norma L_2	0,4619587	0,4721809
norma L_∞	0,8781858	0,9124545
tempo total (normalizado)	1,0616	1,0719

Analogamente aos resultados mostrados nas tab. A4.1 a A4.8, a abordagem fluxo numérico conserva bem a massa total para os esquemas de solução desenvolvidos e implementados. O esquema menos conservativo foi o *upwind*, que perdeu aproximadamente 0,00030% de massa em 80s. Note-se que os esquemas lineares de alta ordem (tab. A4.8), e os não lineares K-DPL e HQUICK, não foram tabelados, pois os seus resultados se mostraram muito piores que aqueles resultados obtidos com $C_r = 0,25$. Por exemplo, quando $C_r = 0,33$ tem-se, para os limites Fromm, K-Linear e K-Linear simétrico, que a métrica para medir as oscilações introduzidas por esquemas não monotônicos (*conhecimento para a razão do pico*), fornece os valores $-45,2470626$, $-10,5736217$ e $-15,9294815$, respectivamente, indicando fortes oscilações nesses métodos quando $C_r = 0,33$.

Todos os demais esquemas não lineares das tab. A4.9 a A4.15 preservam bem o pico, mantendo concentração máxima. Destacam-se os esquemas de Sweby, o R-Kappa, o MUSCL, o Koren, UMIST, van Leer, o HCUS, o K-USR e MC para $C_r = 0,40$. Os esquemas SMART e K-USPL também preservam bem o pico para $C_r = 0,33$. Já o esquema *upwind* não preserva o pico, suavizando-o em aproximadamente em 46,77%.

O *conhecimento para a razão do pico*, que mede as oscilações introduzidas por esquemas não monotônicos, indica que esses esquemas não introduzem oscilações, sendo que todos erros são iguais ou menores que da $O(10^{-6})$. Porém, que devido à aritmética de ponto flutuante e/ou as condições de contorno estabelecidas na malha de teste, não se evita completamente o aparecimentos de valores negativos.

A *razão de distribuição*, que indica o comportamento do formato original do perfil de concentração, mostra que os esquemas analisados nas tab. A4.9 a A4.15 retêm o perfil. As exceções são os esquemas *upwind* e K-MPL1. O resultado para o esquema *upwind* (tab. A4.9) mostra que esse esquema distorce muito o perfil (forte suavização),

mostrando a influência do campo de velocidades na manutenção, ou não, desse perfil de concentração quando de sua advecção ao longo do domínio computacional.

Já as métricas relacionadas aos erros e às normas refletem os resultados anteriores mostrando que as soluções obtidas por esquemas não lineares têm um comportamento muito melhor que aqueles comportamentos obtidos pelo esquema linear *upwind*. Isso pode ser visto sobretudo no grande erro mostrado pela métrica da raiz média quadrada. Nesse caso, o erro é aproximadamente 18 vezes maior que qualquer outro esquema.

Todos os resultados apresentados nas tabelas são com apenas sete casas decimais após a vírgula, e foram obtidos com precisão simples (*float*), padrão IEEE754. Experimentos numéricos com todas as funções implementadas neste trabalho mostram que os resultados com precisão dupla (*double*) são praticamente os mesmos que os obtidos com precisão simples. Alguns dos experimentos feitos são como mostrado nas tab. A4.16 a A4.21.

TABELA A4.16 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise da precisão dos resultados do esquema *upwind* com parâmetros 0,00025, 100x100, 200 ciclos, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=1$, <i>float</i>	$S^0=1$, <i>double</i>
massa total (exata)	100,000000000000	100,000000000000
massa total (numérica)	99,94924926758	99,94933319092
razão do pico (RP)	0,587848067284	0,587848722935
conhecimento para a RP	0,000000000000	0,000000000000
razão de massa	0,999563064575	0,999564132690
1 - razão de massa	0,000436935425	0,000435867310
razão de distribuição	0,411328204266	0,411328878031
máximo (numérico)	0,587848067284	0,587848722935
mínimo (numérico)	0,000000000000	0,000000000000
erro absoluto médio	0,010578584666	0,010578582019
erro da RMQ	0,260646184736	0,260646062900
norma L_1	0,914941787720	0,914941558838
norma L_2	0,578293997051	0,578294031318
norma L_∞	0,665362596512	0,665362477303

TABELA A4.17 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise da precisão dos resultados do esquema *upwind* com parâmetros 0,00025, 100x100, 200 ciclos, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=10^4$, <i>float</i>	$S^0=10^4$, <i>double</i>
massa total (exata)	1000000,00000000	1000000,00000000
massa total (numérica)	999492,9375000000	999493,50000000
razão do pico (RP)	0,587848046875	0,587848535156
conhecimento para a RP	0,000000000000	0,000000000000
razão de massa	0,999562875000	0,999563500000
1 - razão de massa	0,000437125000	0,000436500000
razão de distribuição	0,411329102497	0,411329421666
máximo (numérico)	5878,480468750	5878,48535156250
mínimo (numérico)	0,000000000000	0,000000000000
erro absoluto médio	105,795785639958	105,795727829807
erro da RMQ	0,318900529880	0,318900485622
norma L_1	0,915027750000	0,915027250000
norma L_2	0,578294530482	0,578294369507
norma L_∞	0,665362646484	0,665362646484

TABELA A4.18 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise da precisão dos resultados do esquema Roe-Superbee com parâmetros 0,00025, 100x100, 200 ciclos, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=1, float$	$S^0=1, double$
massa total (exata)	100,0000000000	100,0000000000
massa total (numérica)	99,99985504150	99,99990081787
razão do pico (RP)	0,949380159378	0,949380457401
conhecimento para a RP	-0,000000105838	-0,000000105838
razão de massa	0,999997558594	0,999998168945
1 - razão de massa	0,000002441406	0,000001831055
razão de distribuição	0,775151081828	0,775151398894
máximo (numérico)	0,949380159378	0,949380457401
mínimo (numérico)	-0,000000105838	-0,000000105838
erro absoluto médio	0,003664343546	0,003664341341
erro da RMQ	0,167228062855	0,167228052305
norma L_1	0,316929073334	0,316928882599
norma L_2	0,274126304784	0,274126151167
norma L_∞	0,502402186394	0,502402245998

TABELA A4.19 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise da precisão dos resultados do esquema Roe-Superbee com parâmetros 0,00025, 100x100, 200 ciclos, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=10^4, float$	$S^0=10^4, double$
massa total (exata)	1000000,0000000000	1000000,0000000000
massa total (numérica)	999998,9375000000	999999,6875000000
razão do pico (RP)	0,949379296875	0,949379882812
conhecimento para a RP	-0,0000000000006	-0,0000000000006
razão de massa	0,999999250000	0,999999687500
1 - razão de massa	0,000000750000	0,000000312500
razão de distribuição	0,775151836771	0,775153060253
máximo (numérico)	9493,7929687500	9493,79882812500
mínimo (numérico)	-0,000000062994	-0,000000062994
erro absoluto médio	36,652456208810	36,652412851197
erro da RMQ	0,243986800634	0,243986786173
norma L_1	0,317007093750	0,317006718750
norma L_2	0,274126608581	0,274126414529
norma L_∞	0,502402050781	0,502402148437

TABELA A4.20 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise da precisão dos resultados do esquema UMIST com parâmetros 0,00025, 100x100, 200 ciclos, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=1, float$	$S^0=1, double$
massa total (exata)	100,000000000000	100,000000000000
massa total (numérica)	99,99997711182	99,99996948242
razão do pico (RP)	0,999998211861	0,999996244907
conhecimento para a RP	-0,000000198382	-0,000000198867
razão de massa	0,999999465942	0,999999313354
1 - razão de massa	0,000000534058	0,000000686646
razão de distribuição	0,889888297737	0,889886712409
máximo (numérico)	0,999998211861	0,999996244907
mínimo (numérico)	-0,000000198382	-0,000000198867
erro absoluto médio	0,002231752050	0,002231760209
erro da RMQ	0,104808839929	0,104808781014
norma L_1	0,193024234772	0,193024940491

norma L_2	0,237010555258	0,237010346232
norma L_∞	0,458413541317	0,458413422108

TABELA A4.21 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise da precisão dos resultados do esquema UMIST com parâmetros 0,00025, 100x100, 200 ciclos, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=10^4$, <i>float</i>	$S^0=10^4$, <i>double</i>
massa total (exata)	999999,56250000	1000000,000000000
massa total (numérica)	1000000,000000000	999999,7500000000
razão do pico (RP)	0,999996679687	0,999994824219
conhecimento para a RP	-0,000000001243	-0,000000001322
razão de massa	0,999999562500	0,999999250000
1 - razão de massa	0,000000437500	0,000000750000
razão de distribuição	0,889888538666	0,889887634353
máximo (numérico)	9999,966796875000	9999,94824218750
mínimo (numérico)	-0,000012426864	-0,000013222519
erro absoluto médio	22,318430960227	22,318537547693
erro da RMQ	0,128587119025	0,128587029844
norma L_1	0,193032109375	0,193033031250
norma L_2	0,237010856531	0,237010463759
norma L_∞	0,458410449219	0,458410595703

Dado que algumas das métricas consideram o erro relativo (anexo 2), considerou-se para efeito de cálculo apenas aquelas células cuja concentração fosse maior ou igual a $2,5 \cdot 10^{-4}$. Com essa abordagem, os resultados mais afetados são aqueles das métricas para o erro absoluto médio e para o erro da raiz média quadrada, já que são as medidas de desempenho diretamente afetadas pelo número de células da malha de teste. Experimentos numéricos com todas as funções implementadas neste trabalho mostram que os resultados para 0,00025 e para 0,0000000025 são praticamente os mesmos, exceto as métricas relativas ao erros. Alguns dos experimentos feitos são como mostrado nas tab. A4.22 a A4.27.

TABELA A4.22 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise dos resultados quando da imposição da restrição para erro relativo no esquema *upwind* com parâmetros 100x100, 200 ciclos, $S^0 \times 10 \times 10$, *float*.

métrica × concentração	$S^0=1$ e 0,00025	$S^0=1$ e 0,0000000025
massa total (exata)	100,000000000000	100,000000000000
massa total (numérica)	99,94924926758	99,94924926758
razão do pico (RP)	0,587848067284	0,587848067284
conhecimento para a RP	0,000000000000	0,000000000000
razão de massa	0,999563064575	0,999563064575
1 - razão de massa	0,000436935425	0,000436935425
razão de distribuição	0,411328204266	0,411328204266
máximo (numérico)	0,587848067284	0,587848067284
mínimo (numérico)	0,000000000000	0,000000000000
erro absoluto médio	0,010578584666	0,010579576160
erro da RMQ	0,260646184736	0,318900540944
norma L_1	0,914941787720	0,915027542114
norma L_2	0,578293997051	0,578293997051
norma L_∞	0,665362596512	0,665362596512

TABELA A4.23 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise dos resultados quando da imposição da restrição para erro relativo no esquema *upwind* com parâmetros 100x100, 200 ciclos, $S^0 \times 10 \times 10$, *float*.

métrica × concentração	$S^0=10^4, 0,00025$	$S^0=10^4, 0,0000000025$
massa total (exata)	1000000,000000000	1000000,000000000
massa total (numérica)	999492,937500000	999492,937500000
razão do pico (RP)	0,587848046875	0,587848046875
conhecimento para a RP	0,000000000000	0,000000000000
razão de massa	0,999562875000	0,999562875000
1 - razão de massa	0,000437125000	0,000437125000
razão de distribuição	0,411329102497	0,411329102497
máximo (numérico)	5878,48046875000	5878,48046875000
mínimo (numérico)	0,000000000000	0,000000000000
erro absoluto médio	105,795785639958	105,795785639958
erro da RMQ	0,318900529880	0,362507874056
norma L_1	0,915027750000	0,915027750000
norma L_2	0,578294530482	0,578294530482
norma L_∞	0,665362646484	0,665362646484

TABELA A4.24 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise dos resultados quando da imposição da restrição para erro relativo no esquema Roe-Superbee com parâmetros 100x100, 200 ciclos, $S^0 \times 10 \times 10$, *float*.

métrica × concentração	$S^0=1, 0,00025$	$S^0=1, 0,0000000025$
massa total (exata)	100,000000000000	100,000000000000
massa total (numérica)	99,99985504150	99,99985504150
razão do pico (RP)	0,949380159378	0,949380159378
conhecimento para a RP	-0,000000105838	-0,000000105838
razão de massa	0,999997558594	0,999997558594
1 - razão de massa	0,000002441406	0,000002441406
razão de distribuição	0,775151081828	0,775151081828
máximo (numérico)	0,949380159378	0,949380159378
mínimo (numérico)	-0,000000105838	-0,000000105838
erro absoluto médio	0,003664343546	0,003665245948
erro da RMQ	0,167228062855	0,250994358445
norma L_1	0,316929073334	0,317007122040
norma L_2	0,274126304784	0,274126304784
norma L_∞	0,502402186394	0,502402186394

TABELA A4.25 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise dos resultados quando da imposição da restrição para erro relativo no esquema Roe-Superbee com parâmetros 100x100, 200 ciclos, $S^0 \times 10 \times 10$, *float*.

métrica × concentração	$S^0=10^4, 0,00025$	$S^0=10^4, 0,0000000025$
massa total (exata)	1000000,000000000	1000000,000000000
massa total (numérica)	999998,937500000	999998,937500000
razão do pico (RP)	0,949379296875	0,949379296875
conhecimento para a RP	-0,000000000006	-0,000000000006
razão de massa	0,999999250000	0,999999250000
1 - razão de massa	0,000000750000	0,000000750000
razão de distribuição	0,775151836771	0,775151836771
máximo (numérico)	9493,792968750000	9493,79296875000
mínimo (numérico)	-0,000000062994	-0,000000062994

erro absoluto médio	36,652456208810	36,652459821945
erro da RMQ	0,243986800634	0,290694921281
norma L_1	0,317007093750	0,317007125000
norma L_2	0,274126608581	0,274126608581
norma L_∞	0,502402050781	0,502402050781

TABELA A4.26 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise dos resultados quando da imposição da restrição para erro relativo no esquema UMIST com parâmetros 100×100 , 200 ciclos, $S^0 \times 10 \times 10$, *float*.

métrica × concentração	$S^0=1, 0,00025$	$S^0=1, 0,0000000025$
massa total (exata)	100,000000000000	100,000000000000
massa total (numérica)	99,99997711182	99,99997711182
razão do pico (RP)	0,999998211861	0,999998211861
conhecimento para a RP	-0,000000198382	-0,000000198382
razão de massa	0,999999465942	0,999999465942
1 - razão de massa	0,000000534058	0,000000534058
razão de distribuição	0,889888297737	0,889888297737
máximo (numérico)	0,999998211861	0,999998211861
mínimo (numérico)	-0,000000198382	-0,000000198382
erro absoluto médio	0,002231752050	0,002231841364
erro da RMQ	0,104808839929	0,176031817927
norma L_1	0,193024234772	0,193031959534
norma L_2	0,237010555258	0,237010555258
norma L_∞	0,458413541317	0,458413541317

TABELA A4.27 - Experimento numérico para a onda quadrada com $C_r=0,25$. Análise dos resultados quando da imposição da restrição para erro relativo no esquema UMIST com parâmetros 100×100 , 200 ciclos, $S^0 \times 10 \times 10$, *float*.

métrica × concentração	$S^0=10^4, 0,00025$	$S^0=10^4, 0,0000000025$
massa total (exata)	1000000,0000000000	1000000,0000000000
massa total (numérica)	1000000,0000000000	1000000,0000000000
razão do pico (RP)	0,999996679687	0,999996679687
conhecimento para a RP	-0,000000001243	-0,000000001243
razão de massa	0,999999562500	0,999999562500
1 - razão de massa	0,000000437500	0,000000437500
razão de distribuição	0,889888538666	0,889888538666
máximo (numérico)	9999,966796875000	9999,966796875000
mínimo (numérico)	-0,000012426864	-0,000012426864
erro absoluto médio	22,318430960227	22,318430960227
erro da RMQ	0,128587119025	0,186244532583
norma L_1	0,193032109375	0,193032109375
norma L_2	0,237010856531	0,237010856531
norma L_∞	0,458410449219	0,458410449219

Além disso, foram testados outros perfis para a onda quadrada, como os $10 \times 10 \times 10$, $100 \times 10 \times 10$, $1000 \times 10 \times 10$, $1 \times 5 \times 5$, $10 \times 5 \times 5$, $100 \times 5 \times 5$, $1000 \times 5 \times 5$, $1 \times 20 \times 20$, $10 \times 20 \times 20$, $100 \times 20 \times 20$ e $1000 \times 20 \times 20$, etc. Os resultados foram análogos aos apresentados anteriormente, sendo que o *erro absoluto médio* aquela métrica que melhor caracteriza as diferenças entre os perfis que apresentam grandes variações na concentração inicial S^0 . Foram feitos experimentos numéricos com todas as 32 funções limitadoras de fluxos implementadas neste trabalho. Alguns dos experimentos feitos são como mostrado nas tab. A4.28 a A4.45.

TABELA A4.28 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema Roe-Superbee com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=1$	$S^0=10$	$S^0=10^2$
massa total (exata)	100,000000	1000,000000	10000,000000
massa total (numérica)	99,999855	999,9981079	9999,9824218
razão do pico (RP)	0,9493801	0,9493795	0,9493795
conhecimento para a RP	-0,0000000	-0,0000000	-0,0000000
razão de massa	0,9999975	0,9999990	0,9999979
1 - razão de massa	0,0000024	0,0000009	0,0000020
razão de distribuição	0,7751510	0,7751514	0,7751511
máximo (numérico)	0,9493801	9,4937953	94,9379501
mínimo (numérico)	-0,0000000	-0,0000000	-0,0000000
erro absoluto médio	0,0036643	0,0366518	0,3665239
erro da RMQ	0,1672280	0,1920103	0,2109476
norma L_1	0,3169290	0,3170016	0,3170065
norma L_2	0,2741263	0,2741263	0,2741264
norma L_∞	0,5024021	0,5024021	0,5024020

TABELA A4.29 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema Roe-Superbee com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=10^3$	$S^0=10^4$	$S^0=10^5$
massa total (exata)	100000,000000	1000000,000000	10000000,000000
massa total (numérica)	99999,8359375	999998,9375000	9999985,000000
razão do pico (RP)	0,9493794	0,9493792	0,9493793
conhecimento para a RP	-0,0000000	-0,0000000	-0,0000000
razão de massa	0,9999988	0,9999992	0,9999985
1 - razão de massa	0,0000011	0,0000007	0,0000015
razão de distribuição	0,7751514	0,7751518	0,7751504
máximo (numérico)	949,3794555	9493,7929687	94937,9375000
mínimo (numérico)	-0,0000000	-0,0000000	-0,0000000
erro absoluto médio	3,6652446	36,6524562	366,5246849
erro da RMQ	0,2283196	0,2439868	0,2571379
norma L_1	0,3170070	0,3170070	0,3170072
norma L_2	0,2741264	0,2741266	0,2741264
norma L_∞	0,5024020	0,5024020	0,5024021

TABELA A4.30 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para esquema Roe-Superbee com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=10^6$	$S^0=10^7$
massa total (exata)	100000000,000000	1000000000,000000
massa total (numérica)	99999808,000000	999999040,000000
razão do pico (RP)	0,9493793	0,9493790
conhecimento para a RP	-0,0000000	-0,0000000
razão de massa	0,9999991	0,9999996
1 - razão de massa	0,0000008	0,0000003
razão de distribuição	0,7751514	0,7751503
máximo (numérico)	949379,3750000	9493790,000000
mínimo (numérico)	-0,0000000	-0,0000000
erro absoluto médio	3665,2445369	36652,4832928
erro da RMQ	0,2698627	0,2799565

norma L_1	0,3170070	0,3170073
norma L_2	0,2741264	0,2741264
norma L_∞	0,5024022	0,5024021

TABELA A4.31 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema *Upwind* com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=1$	$S^0=10$	$S^0=10^2$
massa total (exata)	100,000000	1000,000000	10000,000000
massa total (numérica)	99,9492492	999,4917602	9994,9052734
razão do pico (RP)	0,5878480	0,5878479	0,5878480
conhecimento para a RP	0,0000000	0,0000000	0,0000000
razão de massa	0,9995630	0,9995642	0,9995639
1 - razão de massa	0,0004369	0,0004357	0,0004360
razão de distribuição	0,4113282	0,4113285	0,4113281
máximo (numérico)	0,5878480	5,8784799	58,784805
mínimo (numérico)	0,0000000	0,0000000	0,0000000
erro absoluto médio	0,0105681	0,1057949	1,0579581
erro da RMQ	0,2606461	0,2761541	0,2924221
norma L_1	0,9140417	0,9150202	0,9150280
norma L_2	0,5782939	0,5782942	0,5782939
norma L_∞	0,6653625	0,6653626	0,6653625

TABELA A4.32 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema *upwind* com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=10^3$	$S^0=10^4$	$S^0=10^5$
massa total (exata)	100000,000000	1000000,000000	10000000,000000
massa total (numérica)	99949,2578125	999492,9375000	9994917,0000000
razão do pico (RP)	0,5878479	0,5878480	0,5878480
conhecimento para a RP	0,0000000	0,0000000	0,0000000
razão de massa	0,9995631	0,9995628	0,9995641
1 - razão de massa	0,0004368	0,0004371	0,0004359
razão de distribuição	0,4113281	0,4113291	0,4113278
máximo (numérico)	587,8479614	5878,4804687	58784,8085937
mínimo (numérico)	0,0000000	0,0000000	0,0000000
erro absoluto médio	10,5795738	105,7957856	1057,9581454
erro da RMQ	0,9150273	0,3189005	0,3299485
norma L_1	0,9150273	0,9150277	0,9150280
norma L_2	0,5782940	0,5782945	0,5782938
norma L_∞	0,6653625	0,6653626	0,6653625

TABELA A4.33 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema *Upwind* com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=10^6$	$S^0=10^7$
massa total (exata)	100000000,000000	1000000000,000000
massa total (numérica)	99949352,0000000	999492544,0000000
razão do pico (RP)	0,5878479	0,5878480
conhecimento para a RP	0,0000000	0,0000000
razão de massa	0,9995635	0,9995641
1 - razão de massa	0,0004364	0,0004358
razão de distribuição	0,4113281	0,4113284
máximo (numérico)	587847,9375000	5878480,5000000

mínimo (numérico)	0,0000000	0,0000000
erro absoluto médio	10579,5759047	105795,8404439
erro da RMQ	0,3416550	0,3519895
norma L_1	0,9150275	0,9150282
norma L_2	0,5782941	0,5782940
norma L_∞	0,6653625	0,6653626

TABELA A4.34 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para esquema UMIST com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=1$	$S^0=10$	$S^0=10^2$
massa total (exata)	100,000000	1000,000000	10000,0000000
massa total (numérica)	99,9999771	999,9998779	9999,9980468
razão do pico (RP)	0,9999982	0,9999964	0,9999967
conhecimento para a RP	-0,0000001	-0,0000000	-0,0000000
razão de massa	0,9999994	0,9999995	1,0000001
1 - razão de massa	0,0000005	0,0000004	0,0000001
razão de distribuição	0,8898882	0,8898880	0,8898885
máximo (numérico)	0,9999982	9,9999647	99,9996795
mínimo (numérico)	-0,0000001	-0,0000002	-0,0000002
erro absoluto médio	0,0022301	0,0223183	0,2231844
erro da RMQ	0,1048088	0,1112309	0,1173020
norma L_1	0,1930242	0,1930314	0,1930321
norma L_2	0,2370105	0,2370107	0,2370106
norma L_∞	0,4584135	0,4584103	0,4584110

TABELA A4.35 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema UMIST com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=10^3$	$S^0=10^4$	$S^0=10^5$
massa total (exata)	100000,000000	1000000,000000	1000000,000000
massa total (numérica)	100000,0546875	1000000,000000	9999996,000000
razão do pico (RP)	0,9999968	0,9999966	0,9999964
conhecimento para a RP	-0,0000000	-0,0000000	-0,0000000
razão de massa	1,0000003	0,9999995	0,9999999
1 - razão de massa	0,0000003	0,0000004	0,0000001
razão de distribuição	0,8898883	0,8898885	0,8898874
máximo (numérico)	999,9968261	9999,9667968	99999,6484375
mínimo (numérico)	-0,0000005	-0,0000124	-0,0001264
erro absoluto médio	2,2318446	22,3184309	223,1845589
erro da RMQ	0,1230739	0,1285871	0,1338734
norma L_1	0,1930322	0,1930321	0,1930323
norma L_2	0,2370106	0,2370108	0,2370105
norma L_∞	0,4584106	0,4584104	0,4584106

TABELA A4.36 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para esquema UMIST com parâmetros 0,00025, 100x100, float, $S^0 \times 10 \times 10$

métrica × concentração	$S^0=10^6$	$S^0=10^7$
massa total (exata)	100000000,00000	1000000000,0000
massa total (numérica)	99999944,000000	999999744,00000
razão do pico (RP)	0,9999957	0,9999947
conhecimento para a RP	-0,0000000	-0,0000000
razão de massa	0,9999996	0,9999995

1 - razão de massa	0,0000004	0,0000004
razão de distribuição	0,8898874	0,8898856
máximo (numérico)	999995,750000	9999947,000000
mínimo (numérico)	-0,0013027	-0,0106831
erro absoluto médio	2231,8503873	22318,5496589
erro da RMQ	0,1389587	0,1438643
norma L_1	0,1930327	0,1930331
norma L_2	0,2370105	0,2370101
norma L_∞	0,4584105	0,4584104

TABELA A4.37 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema UMIST com parâmetros 0,00025, 100x100, float, $S^0 \times 20 \times 20$

métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	400,0000000	4000000,000000	4000000000,000
massa total (numérica)	399,9998779	3999998,50000	399998156,0000
razão do pico (RP)	0,9999998	0,9999982	0,9999982
conhecimento para a RP	-0,0000000	-0,0000000	-0,0000000
razão massa	0,9999984	0,9999987	1,0000010
1 - razão de massa	0,0000015	0,0000012	0,0000010
razão de distribuição	0,9447599	0,9447613	0,9447596
máximo (numérico)	0,9999998	9999,9824218	9999982,00000
mínimo (numérico)	-0,0000000	-0,0000154	-0,0123558
erro absoluto médio	0,0045507	45,5092062	45509,3092843
erro da RMQ	0,1436588	0,1755345	0,1960697
norma L_1	0,0983977	0,0984022	0,0984029
norma L_2	0,1674878	0,1674877	0,1674878
norma L_∞	0,4584137	0,4584132	0,4584132

TABELA A4.38 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema *upwind* com parâmetros 0,00025, 100x100, float, $S^0 \times 20 \times 20$

métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	400,0000000	4000000,00000	4000000000,000
massa total (numérica)	393,2933959	3932937,750000	3932932864,000
razão do pico (RP)	0,8999948	0,8999948	0,8999950
conhecimento para a RP	0,0000000	0,0000000	0,0000000
razão de massa	0,9846802	0,9846783	0,9846831
1 - razão de massa	0,0153197	0,0153216	0,0153168
razão de distribuição	0,6539777	0,6539786	0,6539790
máximo (numérico)	0,8999948	8999,9482421	8999950,000000
mínimo (numérico)	0,0000000	0,0000000	0,0000000
erro absoluto médio	0,0220083	220,0953578	220095,4302231
erro da RMQ	0,2994133	0,3665969	0,4025252
norma L_1	0,4758745	0,4759011	0,4759035
norma L_2	0,3750895	0,3750898	0,3750897
norma L_∞	0,6189880	0,6189880	0,6189881

TABELA A4.39 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema Roe-Superbee com parâmetros 0,00025, 100x100, float, $S^0 \times 20 \times 20$

métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	400,0000000	4000000,000000	4000000000,000
massa total (numérica)	399,9911193	3999906,00000	3999913984,000

razão do pico (RP)	0,9996355	0,9996349	0,9996348
conhecimento para a RP	-0,0000000	-0,0000000	-0,0000000
razão de massa	0,9999824	0,9999820	0,9999857
1 - razão de massa	0,0000175	0,0000180	0,0000142
razão de distribuição	0,8914871	0,8914870	0,8914891
máximo (numérico)	0,9996355	9996,3496093	9996348,00000
mínimo (numérico)	-0,0000000	-0,0000000	-0,0000000
erro absoluto médio	0,0072197	72,2101688	72210,0626662
erro da RMQ	0,2243776	0,2931831	0,3269305
norma L_1	0,1561099	0,1561364	0,1561369
norma L_2	0,1904938	0,1904938	0,1904937
norma L_∞	0,5105758	0,5105754	0,5105756

TABELA A4.40 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema Roe-Superbee com parâmetros 0,00025, 100x100, float, $S^0 \times 5 \times 5$

métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	25,0000000	25000,000000	250000000,0000
massa total (numérica)	25,0000019	24999,9960937	249999904,0000
razão do pico (RP)	0,6435598	0,6435595	0,6435594
conhecimento para a RP	-0,0000001	-0,0000000	-0,0000000
razão de massa	0,9999993	0,9999994	0,9999991
1 - razão de massa	0,0000006	0,0000005	0,0000008
razão de distribuição	0,4984892	0,4984892	0,4984891
máximo (numérico)	0,6435598	643,5595092	6435594,00000
mínimo (numérico)	-0,0000001	-0,0000001	-0,0000000
erro absoluto médio	0,0022234	2,2238217	22238,20557289
erro da RMQ	0,1328199	0,1804332	0,2330080
norma L_1	0,7692102	0,7693533	0,7693529
norma L_2	0,5201496	0,5201501	0,5201500
norma L_∞	0,5891063	0,5891063	0,5891064

TABELA A4.41 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema *upwind* com parâmetros 0,00025, 100x100, float, $S^0 \times 5 \times 5$

métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	25,0000000	25000,000000	250000000,0000
massa total (numérica)	24,9985027	24998,5058593	249985168,0000
razão do pico (RP)	0,3190271	0,3190272	0,3190272
conhecimento para a RP	0,0000000	0,0000000	0,0000000
razão de massa	0,9999497	0,9999504	0,9999508
1 - razão de massa	0,0000502	0,0000495	0,0000491
razão de distribuição	0,2180794	0,2180794	0,2180795
máximo (numérico)	0,3190271	319,02725	3190272,250000
mínimo (numérico)	0,0000000	0,0000000	0,0000000
erro absoluto médio	0,0038509	3,8519278	38519,2979535
erro da RMQ	0,2303963	0,2750062	0,3189962
norma L_1	1,3322869	1,3326129	1,3326136
norma L_2	0,7742757	0,7742755	0,7742754
norma L_∞	0,7880090	0,7880090	0,7880090

TABELA A4.42 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema UMIST com parâmetros 0,00025, 100x100, float, $S^0 \times 5 \times 5$

métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	25,0000000	25000,0000000	250000000,0000
massa total (numérica)	24,999982	25000,0000000	249999984,0000
razão do pico (RP)	0,9067085	0,9067085	0,9067087
conhecimento para a RP	-0,0000001	-0,0000000	-0,0000000
razão de massa	0,9999990	1,0000001	1,0000002
1 - razão de massa	0,0000009	0,0000001	0,0000002
razão de distribuição	0,7319003	0,7319011	0,7319011
máximo (numérico)	0,9067085	906,7085571	9067087,00000
mínimo (numérico)	-0,0000001	-0,0000006	-0,0100141
erro absoluto médio	0,0012152	1,2153394	12153,3929934
erro da RMQ	0,0780331	0,0922886	0,1084184
norma L_1	0,4204382	0,4204588	0,4204587
norma L_2	0,3474281	0,3474284	0,3474286
norma L_∞	0,4638689	0,4638695	0,4638696

TABELA A4.43 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema Roe-Superbee com parâmetros 0,00025, 100x100, float, $S^0 \times 1 \times 1$

métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	1,0000000	10000,000000	10000000,00000
massa total (numérica)	1,0000000	9999,9951171	9999997,00000
razão do pico (RP)	0,0944487	0,0944485	0,0944485
conhecimento para a RP	-0,0000001	-0,0000000	-0,0000000
razão de massa	1,0000002	0,9999997	0,9999992
1 - razão de massa	0,0000002	0,0000002	0,0000008
razão de distribuição	0,0966196	0,0966198	0,0966197
máximo (numérico)	0,0944487	944,48565	944485,5625000
mínimo (numérico)	-0,0000001	-0,0000001	-0,0000000
erro absoluto médio	0,0001874	1,8780152	1878,0145681
erro da RMQ	0,1059718	0,1528731	0,1872106
norma L_1	1,6209386	1,6242954	1,6242948
norma L_2	0,8926875	0,8926882	0,8926883
norma L_∞	0,6565963	0,6565964	0,6565965

TABELA A4.44 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema *upwind* com parâmetros 0,00025, 100x100, float, $S^0 \times 1 \times 1$

métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	1,0000000	10000,000000	10000000,00000
massa total (numérica)	0,9999878	9999,8769531	9999882,00000
razão do pico (RP)	0,0456115	0,0456115	0,0456115
conhecimento para a RP	0,0000000	0,0000000	0,0000000
razão de massa	0,9999903	0,9999902	0,9999896
1 - razão de massa	0,0000096	0,0000097	0,0000104
razão de distribuição	0,0408588	0,0408588	0,0408588
máximo (numérico)	0,0456115	456,1152954	456115,3750000
mínimo (numérico)	0,0000000	0,0000000	0,0000000
erro absoluto médio	0,0002096	2,1026613	2102,6613481
erro da RMQ	0,2080893	0,2604087	0,2914181
norma L_1	1,8131337	1,8185917	1,8185918

norma L_2	0,9466566	0,9466570	0,9466570
norma L_∞	0,7049124	0,7049125	0,7049125

TABELA A4.45 - Resultados para o teste com onda quadrada para $C_r=0,25$ analisando a influência do perfil para o esquema UMIST com parâmetros 0,00025, 100x100, float, $S^0 \times 1 \times 1$

Métrica × concentração	$S^0=1$	$S^0=10^4$	$S^0=10^7$
massa total (exata)	1,0000000	10000,000000	10000000,00000
massa total (numérica)	0,9999994	9999,9980468	10000000,00000
razão do pico (RP)	0,1367194	0,1367190	0,1367190
conhecimento para a RP	-0,0000004	-0,0000000	-0,0000000
razão de massa	0,9999992	1,0000000	1,0000000
1 - razão de massa	0,0000007	0,0000000	0,0000000
razão de distribuição	0,1460935	0,1460936	0,1460936
máximo (numérico)	0,1367194	1367,1901855	1367190,00000
mínimo (numérico)	-0,0000001	-0,0000007	-0,0011959
erro absoluto médio	0,0001684	1,6848263	1684,8263383
erro da RMQ	0,0733714	0,0843663	0,0922232
norma L_1	1,4567965	1,4572063	1,4572063
norma L_2	0,8446833	0,8446844	0,8446843
norma L_∞	0,6153217	0,6153221	0,6153222

Os resultados apresentados nas tab. A4.28 a A4.45 mostram que a solução numérica desses esquemas mantém a qualidade numérica, como as apresentados nas tabelas anteriores, mesmo sob situações onde ocorrem elevados gradientes de concentração. As diferenças dos resultados fornecidos pelas métricas relativas aos erros e as normas apenas refletem as diferenças das quantidades envolvidas nos testes das tabelas anteriores e nessas tabelas.

Alguns dos resultados apresentados nas tabelas A4.1 a A4.45 são qualitativamente como aqueles obtidos nos trabalhos de [ZIJ95] e [ZIJ97]. A partir dos resultados aqui obtidos pode-se afirmar que, em geral, os limites lineares K-QUICK, K-Fromm, K-Linear e K-Linear Simétrico, todos lineares de segunda ordem, sofrem de fortes oscilações para os números de Courant testados. Um conjunto de resultados com essas funções é como mostrado na tab. A4.8. As oscilações ocorrem pois esse esquemas não são monotônicos e/ou limitados e não possuem a propriedade de transporte (veja anexo 1) como, por exemplo, o esquema *upwind*, que é linear e de primeira ordem. Portanto, não são recomendados para o cálculo da solução da equação de transporte escalar. Além disso, a função K-DPL, com $K=0,75$, também sofre de fortes oscilações, mas isso deve-se ao fato de que não atende a condição (5.4), discutida na seção 5.1.1.

Note-se, portanto, que as funções de restrição de fluxo aplicada às funções de interpolação (método de Sweby) para obtenção da solução da advecção horizontal são particularmente atraentes sob o ponto de vista numérico, pois atributos, discutidos nos anexos, como a conservatividade, positividade, o transporte da solução, etc., são características dessa classe de soluções. A principal desvantagem dessa abordagem é a limitação do passo de tempo, que acaba deteriorando a eficiência computacional. Uma solução para essa questão é empregar uma integração temporal em *subciclos*, discutida no anexo 1, na qual os termos horizontais são resolvidos em frações do passo de tempo global.

Bibliografia

- [ABB89] ABBOTT, M. B.; BASCO, D. R. **Computational Fluid Dynamics: an Introduction for Engineers**. London: Longmann Scientific & Technical, 1989. 425 p.
- [ADC95] ADCROFT, A. **Representing Topography - the Finite Volume Approach**. 1995. p. 104-132 Ph.D. Dissertation (Computer Science) – University of California, Los Angeles. Disponível em: <<http://www.atmos.ucla.edu/~alastair/html.papers/thesis/>>. Acesso em: 22 nov. 2001.
- [AGO94] AGOSHKOV, V. I.; QUARTERONI, A.; SALERI, F. Recent Developments in the Numerical Simulation of Shallow Water Equations I: Boundary Conditions. **Applied Numerical Mathematics**, [S.l.], v.15, p. 175-200, 1994.
- [ALO2002] ALONSO, J.; FATICA, M. **Parallel Methods in Numerical Analysis: Domain Decomposition Methods I**. 2002. 28 p. (Resume) – Aerospace Computing Laboratory, UMIST, Stanford University, Stanford. Disponível em: <<http://www.stanford.edu/class/cs238/syllabus.html>> Acesso em: 01 abr. 2002.
- [ALU97] ALURU, S.; SEVILGEN, F. E. Parallel Domain Decomposition and Load Balancing Using Space-Filling Curves. In: INTERNATIONAL CONFERENCE ON HIGH PERFORMANCE COMPUTING, HiPC, 4., 1997. **Proceedings...** [S.l.]: IEEE Computer Society, 1997. Disponível em: <<http://www.computer.org/proceedings/8067/80670230abs.htm>>. Acesso em: 25 nov. 1999.
- [AME77] AMES, W. F. **Numerical Methods for Partial Differential Equations**. 2nd ed. New York: Academic Press, 1977. 365 p.
- [AND95] ANDERSON, J. D. **Computational Fluid Dynamics**. New York: McGraw-Hill, 1995. 547 p.
- [APS2001] APSLEY, D. **Computational Hydraulics**. Manchester: Manchester Centre for Civil and Construction Engineering, 2001. 93 p. Disponível em: <<http://personalpages.umist.ac.uk/staff/david.d.apsley/lectures/comphydr/index.htm>>. Acesso em: 20 out. 2001.
- [ARA97] ARAÚJO, E. R. **Métodos Iterativos em Álgebra Linear Computacional**. Rio de Janeiro: LNCC, 1997. 104 p. Trabalho apresentado na Escola de Verão em Computação Científica. 2., 1997.
- [ARA97a] ARAÚJO, E. R. **Métodos Iterativos para a Solução de Grandes Sistemas de Equações em Engenharia Estrutural**. 1997. 179 p. Tese (Doutorado). COPPE/UFRJ-Departamento de Engenharia Civil, Rio de Janeiro.
- [ASE90] ASENJO, E. O. B. **Métodos Numéricos em Diferenças Centrais, Upwind e Euler-Lagrange para as Equações de Navier-Stokes**. 1990. 76 p. Dissertação (Mestrado em Matemática) – Departamento de Matemática, UFRGS, Porto Alegre.
- [AWA2000] AWAD, M. ; TURNER, I. Flux-limiting and Non-linear Solution Techniques for Simulation of Transport in Porous Media. **Anziam Journal**, Brisbane, v. 42, p. 157-182, Nov. 2000.
- [AXE96] AXELSSON, A. **Iterative Solution Methods**. Cambridge: University Press, 1996. 654 p.
- [BAI92] BAINES, M. J. **MASTERFUL Monotonic Adaptive Solutions of Transient Equations using Recovery, Fitting, Upwinding and Limiters**. Reading: Department of Mathematics, University of Reading, 1993. 16 p. (Numerical Analysis Report 5/92).

- [BAK96] BAKER, L.; SMITH, B. J. **Parallel Programming**. New York: McGraw-Hill, 1996. 381 p.
- [BAL98] BALZANO, A. Evaluation of Methods for Numerical Simulation of Wetting and Drying in Shallow Water Flow Models. **Coastal Engineering**, [S.l.], v.34, p. 83-107, 1998.
- [BAP95] BAPTISTA, A. M.; ADAMS, E. E.; GRESHO, P. **Benchmarks for the Transport Equations: The Convection-Diffusion Forum and Beyond**. 1995. Disponível em: <<http://www.ccalmr.ogi.edu/CDF/paper-cdf2.html>>. Acesso em: 05 mar. 1999.
- [BAR94] BARRETT, R.; BERRY, M.; CHAN, T. F.; DEMMEL, J.; DONATO, J. M.; DONGARRA, J.; ELJKHOUT, V. POZO, R.; ROMINE, C.; VORST, H. V. **Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods**, 1994. Disponível em: <<http://www.netlib.org/templates>> . Acesso em: 02 abr. 1999.
- [BAT82] BATES, J. R.; McDONALD, A. Multiply-Upstream, Semi-Lagrangian Advective Schemes: Analysis and Applications to a Multi-Level Primitive Equation Model. **Monthly Weather Review**, [S.l.], v. 110, p. 1831-1842, Dec. 1982.
- [BAT85] BATES, J. R. Semi-Lagrangian Advective Schemes and Their Use in Meteorological Modeling. **Lectures in Applied Mathematics**, [S.l.], v. 22, p. 1-29, 1985.
- [BTC90] BATCHELOR, G. K. **An Introduction to Fluid Dynamics**. Cambridge: Cambridge University Press, 1990. 615 p.
- [BEN97] BENZI, M. **Preconditioning Methods for Sparse Linear Systems. Short Course on Iterative Linear Solvers and Preconditioners**. Zürich: ETH, 1997. Disponível em: <<http://www.c3.lanl.gov/~benzi.html>>. Acesso em : 20 out. 1999.
- [BEN99] BENZI, M.; TUMA, M. A Comparative Study of Sparse Approximate Inverse Preconditioners. **Applied Numerical Mathematics**, [S.l.], v. 30, p. 305-340, 1999. Disponível em: <http://www.mathcs.emory.edu/~benzi/Web_papers/pubs.html> . Acesso em: 14 out. 1999.
- [BJO95] BJORSTAD, P.; MOE, R.; OLUFSEN, R.; VAINIKKO, E. **Domain Decomposition Techniques in Parallelization of the 3-Dimensional FRONTSIM Code**. Bergen, Norway: Institutt for Informatikk, University of Bergen, 1995. 14 p. (EC-ESPRIT III / EUROPORT2 project).
- [BLA94] BLACK, T. L. The New NMC Mesoscale Eta Model: Description and Forecast Examples. **Weather and Forecasting**, [S.l.], v. 9, p. 265-277, June 1994.
- [BLE98] BLECK, R. **Ocean Modeling in Isopycnic Coordinates**. Los Alamos: Advanced Computing Laboratory (ACL), 1998. p. 423-448. (Report). Disponível em: <<http://www.acl.lanl.gov/CHAMMP/micom.html>>. Acesso em: 03 maio 1999.
- [BLO94] BLOM, J. G.; HUNSDORFER, W.; VEEERWER, J. G. **Vectorization Aspects of a Spherical Advection Scheme on a Reduced Grid**. Amsterdam: Centrum voor Wiskunde en Informatica, 1994. 11 p. (Report NM-R9418). Disponível em:<http://dbs.cwi.nl:8080/cwwwi/owa/cwwwi.print_reports2?ID=269&type=2>. Acesso em: 17 set. 1999.
- [BLU78] BLUMBERG, A. F.; MELLOR, G. L. A Coastal Ocean Numeric Model. In: MATH. MODELING OF ESTUARINE PHYSICS. INT. SYMP, 1978, Hamburg. **Proceedings...** Berlin: Springer, 1978. p. 102-132.
- [BOL98] BOLLHÖFER, M. **Algebraic Domain Decomposition**. 1998. 191 p. Ph.D. Dissertation (Mathematik). Technischen Universität Chemnitz, Fakultät

- für Mathematik. Disponível em: <<http://www.mathematik.tu-chemnitz.de/preprints/sfb393.html>>. Acesso em: 19 ago. 1999.
- [BON98] BONAVENTURA, L. **A Second Order Scheme for Semi-Lagrangian Advection with Accurate Approximation of Trajectories**. 1998. Disponível em: <<http://www.ing.unitn.it/~bonavent/archi.html>>. Acesso em: 13 set. 1999.
- [BON2000] BONAVENTURA, L.; GROSS, E. S. **Constancy Preserving, Conservative Methods for Free-Surface Models**. 2000. Disponível em : < <http://www.ing.unitn.it/~bonavent/archi.html>>. Acesso em : 15 jan. 2001.
- [BOO75] BOOK, D. L.; BORIS, J. P.; HAIN, K. Flux-Corrected Transport II. Generalizations of the Method. **Journal of Computational Physics**, [S.l.], v. 18, p. 248-283, 1975.
- [BOR73] BORIS, J. P.; BOOK, D. L. Flux-Corrected Transport I. SHASTA, A Fluid Transport Algorithm That Works. **Journal of Computational Physics**, [S.l.], v. 11, p. 38-69, 1973.
- [BOR76] BORIS, J. P.; BOOK, D. L. Flux-Corrected Transport III. Minimal-error FCT Algorithm. **Journal of Computational Physics**, [S.l.], v. 20, p. 397-431, 1976.
- [BOT89] BOTT, A. A Positive Definite Advection Scheme Obtained by Nonlinear Renormalization of the Advective Fluxes. **Monthly Weather Review**, [S.l.], v. 117, p. 1006-1015, 1989.
- [BOT89a] BOTT, A. Reply. **Monthly Weather Review**, [S.l.], v. 117, p. 2633-2636, Nov. 1989.
- [BOT92] BOTT, A. Monotone Flux Limitation in the Area-preserving Flux-Form Advection Algorithm. **Monthly Weather Review**, [S.l.], v. 120, p. 2592-2602, Nov. 1992.
- [BOT93] BOTT, A. The Monotone Area-Preserving Flux-Form Advection Algorithm: Reducing the Time-Splitting Error in Two-Dimensional Flow Fields. **Monthly Weather Review**, [S.l.], v. 121, p. 2627-2641, Sept. 1993.
- [BOU2000] BOUNAÏM, A. **Domain Decomposition Methods**. 2000. 49 p. (course) – University of Oslo, Oslo. Disponível em: <<http://www.ifi.uio.no/~aichab/>>. Acesso em: 20 nov. 2001.
- [BRL2000] BORTOLI, A. L. De. **Introdução à Dinâmica de Fluidos Computacional**. Porto Alegre: Ed. da UFRGS, 2000. 134 p.
- [BRL2002] BORTOLI, A. L. De; CUNHA, R. A Transient Simulation of the Carotid Artery Flow. In: VECTOR AND PARALLEL PROCESSING, VECPAR, 2002. **Proceedings...** Cidade do Porto: [s.n.], 2002. p. 17-27.
- [BRT96] BARTELLO, P.; THOMAS, S. J. The Cost-Effectiveness of Semi-Lagrangian Advection. **Monthly Weather Review**, [S.l.], v. 124, p. 2883-2897, Dec. 1996.
- [BUR95] BURCHARD, H.; BAUMERT, H. On the Performance of a Mixed-layer Model Based on the *k-varepsilon* Turbulence Closure. **Journal Geophys. Res**, [S.l.], v.100, p. 8523-8540, 1995. Disponível em: <<http://www.gotm.net>>. Acesso em: 20 nov. 2001.
- [BUR97] BURCHARD, H.; PETERSEN, O. **Models of Turbulence in the Marine Environment a Comparative Study of Two-equation Turbulence Models**. 1997. Disponível em:< <http://www.gotm.net/>>. Acesso em: 12 abr. 1999.
- [BUR98] BURCHARD, H.; PETERSEN, O.; RUPPETH., T. P. **Comparing the Performance of the Mellor-Yamada and the κ - ϵ Two-equation Turbulence Models**. Disponível em: <<http://www.gotm.net/>>. Acesso em: 04 maio 1999.

- [BUR98a] BURCHARD, H.; BOLDING, K.; VILLARREAL, M. **GOTM, a General Ocean Turbulence Model. Theory, Implementation and Test Cases.** 1998. Disponível em: <<http://www.gotm.net/>>. Acesso em: 10 ago. 1999.
- [BUR2000] BURCHARD, H.; DELEERSNIJDER, E. Investigating the Stability of Algebraic Non-equilibrium Second-moment Turbulence Closure Models. Submitted to **Ocean Modelling Online**, revised version, Sept. 2000. Disponível em: <<http://www.gotm.net/>>. Acesso em 01 dez. 2000.
- [BUR2000a] BURCHARD, H. Note on the q^2 by Mellor and Yamada [1982]. Accepted for publication in **Journal of Physical Oceanography**, Aug., 2000. Disponível em: <<http://www.gotm.net/>>. Acesso em: 20 dez. 2000.
- [BUR2000b] BURCHARD, H.; BOLDING, K. Comparative Analysis of Four Second-Moment Turbulence Closure Models for the Oceanic Mixed Layer. Accepted for publications in **Physical Oceanography**, July 2000. Disponível em : <<http://www.gotm.net/>>. Acesso em: 25 out. 2000.
- [BYU99] BYUN, D. W.; YOUNG, J.; PLEIM J. **Numerical Transport Algorithms for the Community Multiscale Air Quality (CMAQ) Chemical Transport Model in Generalized Coordinates.** [S.l.]: National Oceanic and Atmospheric Administration, 1999. 55 p. (EPA/600/R-99/030).
- [CAM99] CAMPOS, E. J. D. **O Modelo Oceânico da Universidade de Princeton (POM).** 1999. 16 p. Mini-Curso no Workshop sobre Ambientes Costeiros e Mudanças Globais – Instituto Oceanográfico, Universidade de São Paulo.
- [CAN2000] CANUTO, V. M.; HOWARD, A.; CHENG, Y.; DUBOVIKOV, M. S. Ocean Turbulence I: One-point Closure Model. Momentum and Heat Vertical Diffusivities With and Without Rotation. **Journal of Physical Oceanography**, submitted 2000.
- [CAR2002] CARVALHO, E. C. A.; DORNELES, R. V.; RIZZI, R. L.; DIVERIO, T. A. Modelagem de um Cluster Heterogêneo para ser Utilizado em Particionamento e Mapeamento de Domínios de Aplicações Paralelizáveis, Baseado em Programas Específicos de Benchmarks. WPERFORMANCE, 2002. **Anais...** Florianópolis: SBC, 2002, 12 p.
- [CAR2002a] CARVALHO, E. C. **Algoritmos e Software para Particionamento de Domínios e Mapeamento em Clusters Heterogêneos.** 2002. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.
- [CAS90] CASULLI, V. Semi-Implicit Finite Difference Methods for the Two-Dimensional Shallow Water Equations. **Journal of Computational Physics**, [S.l.], v. 86, p. 56-74, 1990.
- [CAS92] CASULLI, V.; CHENG, R. T. Semi-Implicit Finite Difference Methods for Three-Dimensional Shallow Water Flow. **International Journal for Numerical Methods in Fluids**, [S.l.], v. 15, p. 629-648, 1992.
- [CAS94] CASULLI, V.; CATTANI, E. Stability, Accuracy and Efficiency of a Semi-Implicit Method for Three-Dimensional Shallow Water Flow. **Computers Math. Applic.**, [S.l.], v. 27, n. 4, p. 99-112, 1994.
- [CAS97] CASULLI, V. Numerical Simulation of Three-Dimensional Free Surface Flow in Isopycnal Co-ordinates. **International Journal for Numerical Methods in Fluids**, [S.l.], v. 25, p. 645-658, 1997.
- [CAS2000] CASULLI, V.; WALTERS, R. An Unstructured Grid, Three-dimensional Model Based on the Shallow Water Equations. **International Journal for Numerical Methods in Fluids**, [S.l.], v. 32, p. 331-348, 2000.
- [CAT98] CATABRIGA, L. **Estudo de Pré-Condicionadores para o Método GMRES Usando Estruturas de Dados por Arestas.** 1998. 57 p. (Seminário de Qualificação ao doutorado) - COPPE, UFRJ, Rio de Janeiro. Disponível em: <<http://www.coc.ufrj.br/~luci/>>. Acesso em: 17 nov. 1999.

- [CET2001] CETESB. Disponível em : <http://www.cetesb.sp.gov.br/Agua/praias_balneabilidade.htm>. Acesso em: 08 fev. 2002.
- [CHA94] CHAN, T. F.; MATHEW, T. P. Domain Decomposition Algorithms. **Acta Numerica**, Los Angeles, p. 61-143, Aug. 1994.
- [CHE84] CHENG, R. T.; CASULLI, V.; MILFORM, S. N. Eulerian-Lagrangian Solution of the Convection-Dispersion Equation in Natural Coordinates. **Water Resources Research**, [S.l.], v. 20, n. 7, p. 944-952, July 1984.
- [CHE93] CHENG, R. T.; CASULLI, V.; GARTNER, J. W. Tidal, Residual, Intertidal Mudflat (TRIM) Model and its Applications to San Francisco Bay, California. **Estuarine, Coastal and Shelf Science**, [S.l.], v. 36, p. 235-280. 1993.
- [CHF94] CHEFTER, J. G.; CHU, C. K.; KEYES, D. E. Domain Decomposition for the Shallow Water Equations. In: INTERNATIONAL CONFERENCE ON DOMAIN DECOMPOSITION METHODS IN SCIENTIFIC AND ENGINEERING COMPUTING, 9., 1994. **Proceedings...** [S.l.]: Ed. Domain Decomposition Methods Org., 1994. Disponível em: <<http://www.ddm.org/DD9>>. Acesso em: 19 nov. 1999.
- [CHL94] CHLOND, A. Locally Modified Version of Bott's Advection Scheme. **Monthly Weather Review**, [S.l.], v. 122, p. 111-115, 1994.
- [CHO97] CHOW, E. T. **Robust Preconditioning for Sparse Linear Systems**. 1997. 163 p. Ph.D. Dissertation (Computer Science) - Graduate School of the University of Minnesota, Minneapolis. Disponível em: <http://www.llnl.gov/CASC/people/chow/chow_pubs.html>. Acesso em: 23 out. 1999.
- [CHR90] CHRISTARA, C. C. **Parallel Computation of Partial Differential Equations on Distributed Memory Machines**. Toronto : Department of Computer Sciences, 1990. Disponível em: <<http://www.cs.toronto.edu/~ccc>>. Acesso em: 16 nov. 1999.
- [CHU96] CHAU, K. W.; JIN., H.S.; SIN, Y.S. A Finite Difference Model of Two-Dimensional Tidal Flow in Tolo Harbor, Hong Kong. **Applied Math. Modelling**, Hung Hom, v. 20, p. 321-328, July 1996.
- [CLA94] CLAUDIO, D. M.; MARINS, J. M. **Cálculo Numérico Computacional: Teoria e Prática**. São Paulo: Atlas, 1994. 464 p.
- [CMO99] CAMPOS, J. L. E. **Análise Numérica do Transporte de Contaminantes em Meios Porosos com Reações Químicas**. 1999. 95 p. Tese (Doutorado em Engenharia Civil) - Departamento de Engenharia Civil, PUC-RJ, Rio de Janeiro.
- [CNL2000] CANAL, A. P. **Paralelização de Métodos de Resolução de Sistemas Lineares Esparsos com o DECK em um Cluster de PCs**. 2000. 117 p. Dissertação (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.
- [COL84] COLELLA, P.; WOODWARD, P. R. The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations. **Journal Comp. Phys.**, [S.l.], n. 54, p. 174, 1984.
- [COL98] COLELLA, P.; PUCKETT E. G. **Modern Numerical Methods for Fluid Flow**. [S.l.:s.n.], 1998. 149 p. Disponível em: <www.amath.unc.edu/Faculty/minion/class/puckett/C_P_Notes.pdf>. Acesso em: 25 jul. 1999.
- [COL98] COLELLA, P.; PUCKETT E. G. **Modern Numerical Methods for Fluid Flow**. [S.l.:s.n.], 1998. 149 p. Disponível em: <www.amath.unc.edu/Faculty/minion/class/puckett/C_P_Notes.pdf>. Acesso em: 01 jul. 1999.
- [CON98] CONCUS, P.; GOLUB, G. H.; SUN, Y. **An Object Oriented Parallel Algorithm for 3D Isopycnal Flow Simulations**. 1998. Disponível em: <<http://www-sccm.stanford.edu/~ysun/Research/papers/Isopycnal.html>> Acesso em: 03 nov. 1999.

- [COT98] CÔTÉ, J. et al. The Operational CMC-MRB Global Environmental Multiscale (GEM) Model. Part I: Design Considerations and Formulation. **Monthly Weather Review**, [S.l.], v.126, p.1373-1395, June 1998.
- [CRU97] CRUZ, G. A. M. **Algoritmos Paralelos Iterativos do Tipo Quasi-Newton para a Minimização de Funções Multivariadas**. 1997. p. 94. Dissertação (Mestrado em Matemática) - Instituto de Matemática, UFRGS, Porto Alegre.
- [CSL85] CASSALAS, A. B. **Modelo Matemático de Correntologia do Estuário do Rio Guaíba**. Porto Alegre: Instituto de Pesquisas Hidráulicas, UFRGS. 1985. 68 p. (Relatório Técnico, n.12).
- [CUM2000] CUMINATO, J. A.; MENEGUETTE Jr., M. **Discretização de Equações Diferenciais Parciais – Técnica de Diferenças Finitas**. 2000. 232 p. Disponível em: <<http://www.icmc.sc.usp.br/~jacumina/>>. Acesso em: 11 dez. 2000.
- [CUN92] CUNHA, R. D. **A Study on Iterative Methods for the Solution of Systems of Linear Equations on Transputer Networks**. 1992. 215 p. Ph.D. Dissertation- Computing Laboratory, University of Kent, Canterbury.
- [DEB98] DEBREU, L.; BLAYO, E. On the Schwarz Alternating Method for Oceanic Models on Parallel Computers. **Journal of Computational Physics**, [S.l.], v. 141, p. 93-111, 1998.
- [DEM96] DEMMEL, J. **CS267 - Applications of Parallel Computers**. 1996. Disponível em : <<http://www.cs.berkeley.edu/~demmel/cs267/>>. Acesso em :29 dez. 2000.
- [DEM99] DEMMEL, J. **CS267 – Applications of Parallel Computers**. Lecture 23: Load Balancing and Scheduling. 1999. Disponível em : <http://www.cs.berkeley.edu/~demmel/cs267_Spr99/> . Acesso em: 10 jan. 2000.
- [DEN99] DENARO, F. M.; MARRA, F. S.; CONTINILLO, G. **High Order Numerical Schemes with Positivity-Preserving Properties for the Solution of the Transport Equations in Reactive Flows**. 1999. Disponível em: <http://www.irc.na.cnr.it/gc/html_papers/filippo/paper.html>. Acesso em: 27 fev. 2000.
- [DIE97] DIEKMANN, R.; MONIEN, B.; PREIS, R. Load Balancing Strategies for Distributed Memory Machines. **World Scientific**, [S.l.], p. 1-12, 1997.
- [DIE97a] DIEKMANN, R.; MUTHUKRISHNAN, S.; NAYAKKANKUPPAM, M. V. Engineering Diffusive Load Balancing Algorithms Using Experiments. In: INTERNATIONAL SOLVING IRREGULARLY STRUCTURED PROBLEMS IN PARALLEL, IRREGULAR, 1997. **Proceedings...** Berlin: Springer Verlag, 1997. p. 111-122. (Lectures Notes in Computer Science, v. 1253).
- [DIN98] DING, X. **Numerical Solution of the Shallow-Water Equations on Distributed Memory Systems**. 1998. 73 p. Master of Science (Computer Science). University of Toronto, Toronto. Disponível em: <<http://www.cs.toronto.edu/NA/reports.html>>. Acesso em: 19 nov. 1999.
- [DOR2000] DORNELES, R. V.; RIZZI, R. L.; ZEFERINO, C. A.; DIVERIO, T. A.; NAVAU, P. O. A.; BAMPI, S.; SUZIN, A. A. A PC Cluster Implementation of a Mass Transport Two Dimensional Model. In: SYMPOSIUM ON COMPUTER ARCHITECTURE AND HIGH PERFORMANCE COMPUTING, SBAC-PAD, 12., 2000. **Proceedings...** São Pedro: SBC, 2000. p. 191-198.
- [DOR2000a] DORNELES, R. V.; RIZZI, R. L.; ZEFERINO, C. A.; DIVERIO, T. A.; BAMPI, S.; SUZIN, A. A.; NAVAU, P. O. A. Parallel Solution for Shallow Water Equations Using Data Decomposition. In: WORKSHOP DE SISTEMAS DISTRIBUIDOS Y PARALELISMO, WSDP, 4., 2000. **Proceedings...** Santiago: Ed. da Universidade de Santiago, 2000.

- [DOR2001] DORNELES, R. V.; RIZZI, R. L.; DIVERIO, T. A.; NAVAU, P. O. A. h-Refined Krylov-Schwarz Solution for Hydrodynamics and Mass Transport in a PC Cluster. In: SYMPOSIUM ON COMPUTER ARCHITECTURE AND HIGH PERFORMANCE COMPUTING, SBAC-PAD, 13., 2001. **Proceedings...** Pirenópolis: SBC, 2001. p. 17-24.
- [DOR2001a] DORNELES, R. V. **Particionamento de Domínio e Balanceamento Dinâmico de Carga em Arquiteturas Heterogêneas: Aplicação a Modelos Hidrodinâmicos e de Transporte de Massa 2D e 3D.** 2001. 73 p. Proposta de Tese - Instituto de Informática, UFRGS, Porto Alegre.
- [DOR2002] DORNELES, R. V. **Particionamento de Domínio e Balanceamento Dinâmico de Carga.** 2002. Tese (Doutorado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.
- [DRY89] DRYJA, M.; WIDLUND, O. B. Towards a Unified Theory of Domain Decomposition Algorithms for Elliptic Problems. In: INTERNATIONAL SYMPOSIUM ON DOMAIN DECOMPOSITION METHODS FOR PARTIAL DIFFERENTIAL EQUATIONS, 3., 1989. **Proceedings...** Houston: SIAM Publications, 1990. p. 3-21.
- [DUF99] DUFF, I. S.; VORST, H. A. Van der. Developments and Trends in the Parallel Solution of Linear Systems. **Parallel Computing**, [S.l.], v. 25, p. 1931-1970, Feb. 1999.
- [DYE88] DYER, K.R. Fine sediment particle transport in estuaries. In: DROKERS, J.; LEUSSEN, W.V. (Ed.). **Physical Processes in Estuaries.** New York. Springer-Verlag, 1988. p. 295-310.
- [EAS98] EASTEMANN, J. R. **IDRISE – Manual do Usuário, versão digital.** Porto Alegre: Centro de Recursos IDRISI Brasil, 1998. Disponível em: < <http://www.ecologia.ufrgs.br/idrise/>>. Acesso em : 04 nov. 1999.
- [EYM97] EYMARD, R.; GALLOUËT, T.; HERBIN, R. **Finite Volume Methods.** Marseille: [s.n.], 1997. 234 p. Disponível em: <<http://www-gm3.univ-mrs.fr/~gallouet/publi.html>>. Acesso em: 22 nov. 1999.
- [FAL91] FALCONER, R. A.; CHEN, Y. An Improved Representation of Flooding and Drying and Wind Stress Effects in a Two-dimensional Tidal Numerical Model. In: INST. CIV. ENGRS, 1991. **Proceedings...** [S.l.]: Maritime Board, 1991. part 2. p. 659-678.
- [FER97] FERZIGER, J. H.; PERIC, M. **Computational Methods for Fluid Dynamics.** Berlin: Springer, 1997. 364 p.
- [FLE88] FLETCHER, C. A. J. **Computational Techniques for Fluid Dynamics.** New York: Springer-Verlag, 1988. v. 1.
- [FLE88a] FLETCHER, C. A. J. **Computational Techniques for Fluid Dynamics.** New York: Springer-Verlag, 1988. v. 2.
- [FOR96] FORTUNATO, A. B. **Three-Dimensional Modeling of Coastal Flows Using Unstructured Grids.** 1996. 234 p. Ph.D. Dissertation (Environmental Science and Engineering) - Oregon Graduate Institute of Science & Technology, Portland.
- [FOR99] FORTUNATO, A. B.; OLIVEIRA, A. An Adaptive Grid Technique for the Vertical Structure of Shallow Water Models. **Applied Mathematics Modelling**, [S.l.], v. 23, p. 639-650, 1999.
- [FOX98] FOX, R. W.; McDONALD, A. T. **Introdução à Mecânica dos Fluidos.** Rio de Janeiro: LTC, 1998. 662 p.
- [FRT2000] FORTUNA, A. de O. **Técnicas Computacionais para Dinâmica dos Fluidos: Conceitos Básicos e Aplicações.** São Paulo :EDUSP, 2000. 426 p.
- [GAL88] GALPERIN, B.; KANTHA, L. H.; HASSID, S.; ROSATI, A. A Quasi-Equilibrium Turbulent Energy Model for Geophysical Flows. **Journal Atmos. Sci**, [S.l.], v. 45, p. 55-62, 1988.

- [GAR79] GAREY, M. R.; JOHNSON, D. S. **Computer and Intractability**: a Guide to the Theory of NP-completeness. San Francisco: Freeman, 1979. 338 p.
- [GER85] GERALD, C. F.; WHEATLEY, P. O. **Applied Numerical Analysis**. 3rd ed. New York: Addison-Wesley, 1985. 576 p.
- [GHO96] GHOSH, B.; MUTHUKRISHNAN, U. W.; SCHULTZ, M. H. First and Second Order Diffusive Methods for Rapid, Coarse, Distributed Load Balancing. In: ANNUAL ACM SYMPOSIUM ON PARALLEL ALGORITHMS AND ARCHITECTURES, 8., 1996. **Proceedings...** Padua: ACM, 1996. p. 72-81.
- [GLT2002] GALANTE, G.; BALBINIT, J. I.; RIZZI, R. L.; DORNELES, R. V.; DIVERIO, T. A. Processamento de Alto Desempenho Aplicado a Modelos Computacionais de Dinâmica de Fluidos Ambiental. In: ESCOLA REGIONAL DE ALTO DESEMPENHO, ERAD, 2., 2000. **Anais...** São Leopoldo: SBC, 2002. p. 271-274.
- [GOL96] GOLUB, G. H.; VAN LOAN, C. F. **Matrix Computations**. Baltimore: John Hopkins University Press, 1996. 694 p.
- [GOM95] GOMEZ-REYES, E.; BLUMBERG, A. F. Pollutant Transport in Coastal Water Bodies. In: INTERNATIONAL CONF. ON COMP. MODELLING OF SEARS AND COAST. REGIONS, 2., 1995. **Proceedings...** Boston: Computational Mechanics Publications, 1995. p. 87-94.
- [GOO96] GOOSSENS, S.; TAN, K.; ROOSE, D. An Efficient FGMRES Solver for the Shallow Water Equations Based on Domain Decomposition. In: INTERNATIONAL CONFERENCE ON DOMAIN DECOMPOSITION METHODS IN SCIENTIFIC AND ENGINEERING COMPUTING, 9., 1996. **Proceedings...** Disponível em: < <http://www.ddm.org>.1999>. Acesso em : 29 nov. 1999.
- [GOO99] GOOSSENS, S.; CAI, X. C. Lower Dimensional Interpolation in Overlapping Composite Mesh Difference Methods. In: INTERNATIONAL CONFERENCE ON DOMAIN DECOMPOSITION METHODS IN SCIENTIFIC AND ENGINEERING COMPUTING, 10., 1999. **Proceedings...** Disponível em: <<http://www.ddm.org>>. Acesso em : 12 mar. 2000.
- [GRI2000] GRIFFIES, S. M. et al. Developments in Ocean Climate Modelling. **Ocean Modelling**, [S.l.], v. 2, p. 123-192, 2000.
- [GRO98] GROSS, E. S.; CASULLI, V.; BONAVENTURA, L.; KOSEFF, J. A Semi-Implicit Method for Vertical Transport in Multidimensional Models. **International Journal for Numerical Methods in Fluids**, [S.l.], v. 28, p. 157-186, 1998.
- [GRO99] GROSS, E.; KOSEFF, J. R.; MONISMITH, S. G. Evaluation of Advective Schemes for Estuarine Salinity Simulation. **Journal of Hydraulic Engineering**, [S.l.], v. 125, p. 32-46, Jan. 1999.
- [GRO99a] GROSS, E. S.; KOSEFF, J. R.; MONISMITH, S. Three-Dimensional Salinity Simulations of South San Francisco Bay. **Journal of Hydraulic Engineering**, [S.l.], v. 125, n. 11, p. 1199-1209, Nov. 1999.
- [GRP94] GROPP, W. **Parallel Computing and Domain Decomposition**. Mathematics and Computer Science Division, Argonne National Laboratory, 1994. Disponível em: <<http://www-unix.mcs.anl.gov/~gropp/papers.html>>. Acesso em: 25 nov. 1999.
- [GRP98] GROPP, W. D.; KEYES, D. E.; MCINNES, L. C.; TIDRINI, M. D. **Globalized Newton-Krylov-Schwarz Algorithms and Software for Parallel Implicit CFD**. Langley, Hampton: NASA Langley Res. Center, 1998. 37 p. (ICASE Report 98-24, NASA/CR-1998-208435). Disponível em: <<http://www.mcs.anl.gov/~gropp>>. Acesso em: 07 nov. 1999.

- [GRP2000] GROPP, W.; LUSK, E. **Installation Guide to Mpich, a Portable Implementation of MPI Version 1.2.1**. 2000. Disponível em: <<http://www.mcs.anl.gov/mpi/mpich/download>>. Acesso em: 24 jan. 2001.
- [GRS99] GERISH, A.; GRIFFITHS, D. F.; WEINER, R.; CHAPLAIN, M. A. J. **Splitting Methods for Mixed Hyperbolic-Parabolic Systems**. Dundee: University of Dundee, 1999. 26 p. (Numerical Analysis Report NA/186).
- [GRT2000] GERRITSMA, M. I.; KOREN, B. **Introduction to Computational Fluid Dynamics**. [S.l.:s.n.], 2000. 80 p. Disponível em: <<http://www.hsa.lr.tudelft.nl/~marc/ICFD.html>>. Acesso em: 30 jan. 2001.
- [GUP95] GUPTA, A.; KUMAR, V. **Parallel Algorithms for Forward and Back Substitution in Direct Solution of Sparse Linear Systems**. 1995. Disponível em: <<http://www.cs.umn.edu/users/kumar/paper.html>>. Acesso em : 20 dez. 1999.
- [HAR72] HARTEN, A.; ZWAS, G. Self-adjusting Hybrid Schemes for Shock Computations. **Journal of Computational Physics**, [S.l.], v.9, p. 568-583, 1972.
- [HAR83] HARTEN, A. High Resolution Schemes for Hyperbolic Conservation Laws. **Journal of Computational Physics**, [S.l.], v.49, p. 357-393, 1983.
- [HAW91] HAWKEN, D. F.; GOTTLIEB, J. J.; HANSEN, J. S. Review of Some Adaptive Node-Movement Techniques in Finite-Elements and Finite-Difference Solutions of Partial Differential Equations. **Journal of Computational Physics**, [S.l.], v. 95, p. 254-302, 1991.
- [HEA2001] HEATH, M. T. **Scientific Computing an Introductory Survey**. 2001. 48 p. Chapter 1. (polígrafo) - Computational Science and Engineering (CSE), University of Illinois. Disponível em: <<http://www.cse.uiuc.edu/cse301/>>. Acesso em: 27 abr. 2002.
- [HEI99] HEITLAGER, I.; ENGELEN, R.; WOLTERS, L. **The Construction of Flux-Limiting Advection Algorithms through Program Generation**. 1999. Disponível em : <<http://www.wi.leidenuniv.nl/TechRep/1998/tr98-04.html>>. Acesso em : 14 jan. 2000.
- [HEM99] HEMKER, P.W. **Eight Lectures on Discretisation Principles and Hyperbolic Conservation Laws**. [S.l.:s.n.], 1999. 128 p. Disponível em: <<http://www.>>. Acesso em: 25 nov. 2000.
- [HEM2000] HEMKER, P.W. **Numerical Hydrodynamics or Computational Fluid Dynamics**. [S.l.:s.n.], 2000. 55 p. Disponível em: <<http://www.>>. Acesso em: 25 nov. 2000.
- [HEN93] HENDRICKSON, B.; LELAND, R. **Multidimensional Spectral Load Balancing**. 1993. Disponível em: <<http://www.cs.sandia.gov/~bahendr/>>. Acesso em: 28 nov. 1999.
- [HEN95] HENDRICKSON, B.; LELAND, R. **The Chaco User's Guide. Version 2.0**. 1995. Disponível em : <<http://www.cs.sandia.gov/~bahendr/>>. Acesso em : 20 nov. 1999.
- [HIR92] HIRSCH, C. **Numerical Computation of Internal and External Flows: Fundamentals of Numerical discretization**. Chichester: John Wiley & Sons, 1992. v. 1.
- [HOS80] HOSSAIN M. S. Hossain. **Mathematische Modellierung von Turbulenten Auftriebsströmungen**. 1980. 145 p. PhD thesis - Universität Karlsruhe, Karlsruhe.
- [HUA94] HUANG, W.; REN, Y.; RUSSEL, R. D. Moving Mesh Methods Based on Moving Mesh Partial Differential Equations. **Journal of Computational Physics**, [S.l.], v. 113, p. 279-290, 1994.
- [HUA94a] HUANG, W.; REN, Y.; RUSSEL, R. D. Moving Mesh Partial Differential Equations (MMPDES) Based on the Equidistribution Principle. **SIAM Journal of Numerical Analysis**, [S.l.], v. 31, n. 3, p. 709-730, June 1994.

- [HUA97] HUANG, W.; RUSSEL, R. D. **A High Dimensional Moving Mesh Strategy.** 1997. Disponível em : <<http://titania.math.ukans.edu/~reports/1997.html>>. Acesso em : 04 nov. 1999.
- [HUA99] HUANG, W. **Practical Aspects of Formulations and Solutions of Moving Mesh Partial Differential Equations.** 1999. Disponível em: <<http://titania.math.ukans.edu/~reports/1999.html>>. Acesso em: 22 jan. 2000.
- [HUN95] HUNDSORFER, W.; KOREN, B.; LOON, M. V; VERWER, J. G. A Positive Finite-Difference Advection Scheme. **Journal of Computational Physics**, [S.l.], v.117, p.35-46, 1995.
- [HUN96] HUNSDORFER, W. **Numerical Solution of Advection-diffusion-Reaction Equations.** 1996. 89 p. (Lecture Notes for Ph.D. Course) - Thomas Stieltjes Institute, Amsterdam. (Report NM-N9603). Disponível em: <<http://www.cwi.nl/~willem/>>. Acesso em 20 out. 1999.
- [IOR88] IÓRIO JÚNIOR, R.; IÓRIO, V. M. **Equações Diferenciais Parciais: Uma Introdução.** Rio de Janeiro: IMPA, 1988. 366 p.
- [IOV89] IÓRIO, V. M. **EDP: Um Curso de Graduação.** Rio de Janeiro: IMPA, 1989. 285 p.
- [JOS97] JOSHI, M. V.; GUPTA, A.; KARYPIS, G.; KUMAR, V. **A High Performance Two Dimensional Scalable Parallel Algorithm for Solving Triangular Systems.** 1997. Disponível em: <<http://www.cs.umn.edu/users/kumar/paper.html>>. Acesso em : 13 dez. 1999.
- [JÚD96] JÚDICE, J. J.; PATRICIU, J. M. **Sistemas de Equações Lineares.** Coimbra : Departamento de Matemática da Universidade de Coimbra, 1996. 151 p.
- [KAL99] KALLOS, K. **Model Dynamics.** Atmospheric Modeling & Weather Forecasting Group. Disponível em: <<http://forecast.uoa.gr/etaski.html>>. Acesso em: 24 nov. 1999.
- [KAP98] KAPLAN, E. **A Shallow Water Model Distributed Using Domain Decomposition.** 1998. 74 p. (Master's Thesis of Numerical Analysis and Computing Science). Royal Institute of Technology, [S.l.]. Disponível em: <<http://www.fing.edu.uy/~elias/>>. Acesso em: 25 nov. 1999.
- [KAR98] KARYPIS, G.; KUMAR, V. **METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices.** 1998. Disponível em : <<http://www.cs.umn.edu/~karypis>>. Acesso em: 14 jun. 1999.
- [KEL95] KELLEY, C. T. **Iterative Methods for Linear and Nonlinear Equations.** Philadelphia : SIAM, 1995. 165 p.
- [KER70] KERNIGHAN, B.; LIN, S. An Effective Heuristic Procedure for Partitioning Graphs. **The Bell System Technical Journal**, [S.l.], p. 291-308, Feb. 1970.
- [KEY95] KEYES, D. E. Aerodynamic Applications of Newton-Krylov-Schwarz Solvers. In: INTERNATIONAL CONFERENCE ON NUMERICAL METHODS IN FLUID DYNAMICS, 14., 1995. **Proceedings...** New York: Springer, 1995. p. 1-20. Disponível em: <<http://www.math.odu.edu/~keyes/>>. Acesso em: 19 jul. 1999.
- [KEY97] KEYES, D. E.; VENKATAKRISHNAN, V. **Newton-Krylov-Schwarz Methods: Interfacing Sparse Linear Solvers with Nonlinear applications.** Norfolk : Department of Computer Science, Old Dominion University, Norfolk and ICASE, 1997. Disponível em: <<http://www.math.odu.edu/~keyes/>>. Acesso em : 11 dez. 1999.

- [KIM94] KIM, C.; LEE, J. A Three-dimensional PC-based Hydrodynamic Model Using an ADI Scheme. **Coastal Engineering**, [S.l.], v. 23, p. 271-287, 1994.
- [KLR2000] KELLER, S. C. **Desenvolvimento de uma técnica Numérica pra Análise de Escoamentos em Cavidades para Rayleigh entre 10^2 e 10^5** . 2000. 108 p. Dissertação (Mestrado em Matemática Aplicada) – Instituto de Matemática, UFRGS, Porto Alegre.
- [KNU93] KNUPP, P.; STEINBERG, S. **Fundamentals of Grid Generation**. Boca Raton : CRC Press, 1993. 286 p.
- [KUM2002] KUMAR, V. **Graph Partitioning & Applications in High Performance Computing**. 2002. 52 p. (Resumo) – Army High Performance Computing Research Center & Depto. Of Computer Science & Engineering, University of Minnesota, Minnesota.
- [LAK95] LAKSHMIVARAHAN, S.; DHALL, K, S. **Analysis and Design of Parallel Algorithms**. New York: McGraw-Hill, 1995. 657 p.
- [LAN97] LANGROIS, G.; SERPETTE, A. **MIAMI Isopycnic Coordinate Ocean Model: Analytical and Numerical Study of the Theoretical the Thermodynamic Foundations of the Code**. 1997. Disponível em: <<http://www.acl.lanl.gov/CHAMMP/micom.html>>. Acesso em: 04 nov. 1999.
- [LEC92] LECA, P.; MANE, L. A 3D ADI Algorithm on Distributed Memory Multiprocessors. In: SIMOM, Host D. (Ed). **Parallel Computational Fluid Dynamics**. Cambridge : The MIT Press, 1992. p. 150-164.
- [LEE67] LEENDERTSE, J. J. **Aspects of a Computational Model For Long-Period Water-wave Propagation**. Santa Monica: The Rand Corp., 1967. 165p. (Technical Report RM-5294-RC).
- [LEE70] LEENDERTSE, J. J. **A Water-Quality Simulation Model for Well-Mixed Estuaries and Coastal Seas**. Santa Monica: The Rand Corp., 1970. v. 1. (Principles of Computation. Technical Report RM-6230-RC).
- [LEE71] LEENDERTSE, J. J.; GRITTON, E. C. **A Water-Quality Simulation Model for Well-Mixed Estuaries and Coastal Seas**. Santa Monica: The Rand Corp. 1971. v. 2. (Computation Procedures. Technical Report R-708-NYC).
- [LEE89] LEENDERTSE, J. J. **A New Approach to Three-dimensional Free-Surface Flow Modeling**. Santa Monica: The Rand Corp., 1989. 51 p. (Technical Report R-3712-NETH/RC).
- [LEO79] LEONARD, B. P. A Stable and Accurate Convective Modeling Procedure Based on Quadratic Upstream Interpolation. **Comput. Methods. Appl. Mech. Eng.**, [S.l.], v. 19, p. 59-98, 1979.
- [LER85] LEER, B. van. **Large-Scale Computation in Fluid Mechanics**. Providence: American Math. Soc. 1985. 327 p.
- [LEV2001] LEVY, D. **Computational Methods for Fronts, Waves and Interfaces: Conservation Laws**. 2001. 8 p. (course) - Department of Mathematics, Stanford University, Stanford.
- [LEV92] LEVEQUE, R. J. **Numerical Methods for Conservation Laws**. Zürich: Birhhäuser Verlag, 1992.
- [LEV96] LEVEQUE, R. J. High-Resolution Conservative Algorithms for Advection in Incompressible Flow. **SIAM Journal Numerical Analysis**, [S.l.], v. 33, n. 2, p. 627-665, Apr. 1996.
- [LIE2001] LIE, K. **Introduction to Domain Decomposition Methods**. 2001. 43 p. (course in Winter School on Parallel Computing) – SINTEF Applied Mathematics, Department of Numerical Simulation, Department of Informatics, University of Oslo, Oslo.

- [LIN96] LIN, S.; ROOD, R. B. Multidimensional Flux-Form Semi-Lagrangian Transport Schemes. **Monthly Weather Review**, [S.l.], v. 119, p. 2046-2070, 1996.
- [LIS98] LI, S.; PETZOLD, L.; REN, Y. Stability of Moving Mesh Systems of partial Differential Equations. **SIAM Journal Sci. Comput**, [S.l.], v. 20, n. 2, p. 719-738, 1998.
- [LIY96] LI, Y. S.; ZHANG, M. Y. A Semi-implicit Three-dimensional Hydrodynamic Model Incorporating the Influence of Flow-dependent Eddy Viscosity, Bottom Topography and Wave-current Interaction. **Applied Ocean Research**, [S.l.], v. 18, p. 173-185, 1996.
- [LOM2000] LOMAX, H.; PULLIAM, T. H.; ZINGG, D. W. **Fundamentals of Computational Fluid Dynamics**. 2000. Disponível em: <http://www.nas.nasa.gov/~pulliam/Classes/New_notes/notes.html.2000>. Acesso em : 09 jan. 2001.
- [LOO92] LOON, M. **Tests on Semi-Lagrangian Transport and Interpolation**. 1992. Amsterdam: Centrum voor Wiskunde en Informatica, 1992. (Report NM-R9217). Disponível em: <http://dbs.cwi.nl:8080/cwwwi/owa/cwwwi.print_reports2>. Acesso em: 17 maio 1999.
- [LUC2000] LUCAS, B. **Applications of Parallel Computers. Lecture 14 (CS267): Graph Partitioning II**. 2000. Disponível em: <<http://www.nerisc.gov/~dhbailey/cs267>>. Acesso em : 17 fev. 2001.
- [MAC76] MACEDO, H. **Dicionário de Física**. Rio de Janeiro: Nova Fronteira, 1976. 367 p.
- [MAL95] MALISKA, C. R. **Transferência de Calor e Mecânica dos Fluidos Computacional**. Rio de Janeiro: LTC, 1995. 424 p.
- [MAL96] MALEVSKY, A. V.; THOMAS, S. J. Parallel Algorithms for Semi-Lagrangian Advection. **J. Numer. Methods Fluids**. Disponível em: <<http://www.scd.ucar.edu/css/staff/thomas/>>. Acesso em: 19 nov. 1999.
- [MAN2001] MANTOVANELLI, A. **Caracterização dos Fluxos Hídricos e de Material Particulado em Suspensão nas Baías de Antonina e Paranaguá (PR, Brasil)**. Projeto de Pesquisa de Mestrado. Disponível em: <<https://www.cem.ufpr.br/fisica/pguanto.html>>. Acesso em: 20 de fev. 2001.
- [MAR2002] MARTINOTTO, A. L.; FRIZZO, E. J.; DORNELES R. V.; DIVERIO, T. A. Paralelização do Método do GMRES com MPI e *Threads*. **SCIENTIA**, São Leopoldo, v.1, ago. 2002.
- [MAT99] MATOS, D. M. M. **Estudo Comparativo de Funções de Interpolação na Simulação de Escoamentos Supersônicos para Mach 2.0, 2.8 e 3.75**. 1999. 89 p. Dissertação (Mestrado em Matemática Aplicada) – Instituto de Matemática, UFRGS, Porto Alegre.
- [MEI97] MEIER, H. E. M. **Embedding a κ - ϵ Turbulence Model Into a Three-dimensional Model of the Western Baltic Sea**. 1997. Disponível em: <<http://www.ifm.uni-kiel.de/to/baltex/mm2-turbulence-modelo>>. Acesso em: 24 jul. 1999.
- [MEL74] MELLOR, G. L.; YAMADA, T. A Hierarchy of Turbulence Closure Models for Planetary Boundary Layers. **Journal of the Atmospheric Sciences**, [S.l.], v. 32, p. 1791-1806, 1974.
- [MEL82] MELLOR, G. L.; YAMADA, T. Development of a Turbulence Closure Model for Geophysical Fluid Problems. **Rev. Geophys**, [S.l.], v. 20, p. 851-875, 1982.
- [MEL98] MELLOR, G. L. **Users Guide for A Three-Dimensional, Primitive Equation, Numerical Ocean Model**. Princeton: Program in Atmospheric

- and Oceanic Sciences Princeton University, 1998. Disponível em: <<http://www.aos.princeton.edu/WWWPUBLIC/htdocs.pom/PubOnLine>> . Acesso em: 03 out. 1999.
- [MEL98a] MELLOR, G. L.; OEY, L. Y.; EZER, T. Sigma Coordinate Pressure Gradient Errors and the Seamount Problem. **Journal of Atmospheric and Oceanic Technology**, [S.l.], v. 15, p. 1122-1131, Oct. 1998.
- [MEN98] MENEGAT, R.; PORTO, M. L.; CARRARO, C. C.; FERNANDES, L. A. D. (Coord.). **Atlas Ambiental de Porto Alegre**. Porto Alegre : Ed. da UFRGS, 1998. 228 p.
- [MES76] MESINGER, F.; ARAKAWA, A. Numerical Methods Used in Atmospheric Models. **Global Atmospheric Research Program (GARP)**, [S.l.], v. 1, n. 17, p. 1-64, Sept. 1976.
- [MES84] MESINGER, F. A Blocking Technique for Representation of Mountains in Atmospheric Models. **Rev. Meteor. Aeronaut.**, [S.l.], v. 44, p. 195-202, 1984.
- [MES98] MESINGER, F. **Numerical Methods: The Arakawa Approach**, Horizontal Grid. Global and Limited-Area Modeling. Camp Springs: Academic Press, 1998. 53 p.
- [MLO91] MELO, S.T.; NETO, F.M. Mecânica dos Fluidos e Equações Diferenciais. In: COLÓQUIO BRASILEIRO DE MATEMÁTICA, IMPA-SBM, 18., 1991. **Anais...** Rio de Janeiro: IMPA, 1991.
- [MOO92] MOORE, P.; FLAHERTY, J. E. Adaptive Local Overlapping Grid Methods for Parabolic Systems in Two Dimensions. **Journal of Computational Physics**, [S.l.], v. 98, p. 54-63, 1992.
- [MOU2001] MOURA, C. A. Parallel Numerical Methods for Differential Equations : a Survey. In: CORREA, E. et al. (Ed.). **Models for Parallel and Distributed Computation: Theory, Algorithmic Techniques and Applications**. [S.l.]: Kluwer, 2001. p. 38.
- [NAC2000] NACHBIN, A.; ALBUQUERQUE, D. L.; MARCHESIN, D. Global Semi-Lagrangian Advection. In: CONGRESSO BRASILEIRO DE METEOROLOGIA, SBMET, 11., 2000. **Anais...** [S.l.:s.n.], 2000. p. 3530-3537. Disponível em:<<http://www.fluidimpa.br/Portugues/Publicacoes.html>>. Acesso em: 01 jan. 2001.
- [NEI94] NEIL, B. **An Investigation of the Dynamics of Several Equidistribution Schemes**. 1994. 61 p. Thesis (Master in Mathematics) - Department of Mathematics, University of Reading, Reading.
- [NOB86] NOBLE, B.; DANIEL, J. W. **Álgebra Linear Aplicada**. Rio de Janeiro: Prentice-Hall do Brasil, 1986. 378 p.
- [OLI95] OLIVEIRA, A.; BAPTISTA, A. M. A Comparison of Integration and Interpolation Eulerian-Lagrangian Methods. **International Journal for Numerical Methods in Fluids**, [S.l.], v. 21, p. 183-204, 1995.
- [ORL83] ORLOB, G. T. (Ed.). **Mathematical Modeling of Water Quality: Streams, Lakes, and Reservoirs**. Chichester: John Wiley & Sons, 1983. 518 p.
- [OSH84] OSHER, S.; CHAKRAVARTHY, S. High Resolution Schemes and the Entropy Condition. **SIAM Journal Numerical Analysis**, [S.l.], v. 21, n. 5, p. 955-984, Oct. 1984.
- [PAC97] PACHECO, P. S. **Parallel Programming with MPI**. San Francisco: Morgan Kaufmann, 1997. 419 p.
- [PAG97] PAGLIERI, L.; AMBROSI, D.; FORMAGGIA, L.; QUARTERONI, A.; SCHEININE, A. L. Parallel Computation for Shallow Water Flow: A Domain Decomposition Approach. **Parallel Computing**, [S.l.], v. 23, p. 1261-1277, 1997.

- [PAT87] PATNAIK, G.; GUIRGUIS, R. H.; BORIS, J. P.; ORAN, E. S. A Barely Implicit Correction for Flux-Corrected Transport. **Journal of Computational Physics**, [S.l.], v. 71, n.1, p. 1-20, July 1987.
- [PTK81] PATANKAR, S. V. **Numerical Heat Transfer and Fluid Flow**. New York: Hemisphere Publishing Co., 1981. 420 p.
- [PEL96] PELLEGRINI, F. **SCOTCH 3.1 User's Guide**. 1996. Disponível em: <[http:// dept-info.labri.u-bordeaux.fr/~pelegrin/scotch](http://dept-info.labri.u-bordeaux.fr/~pelegrin/scotch)>. Acesso em: 23 jan. 2001.
- [PIC2001] PICININ JUNIOR, D.; DORNELES, R. V.; RIZZI, R. L.; DIVERIO, T. A. Paralelização de Métodos Iterativos do Subespaço de Krylov em Clusters Multiprocessados. In: CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL, CNMAC, 24., 2001. **Anais...** Belo Horizonte: SBMAC, 2001. p. 325.
- [PIC2002] PICININ JÚNIOR, D.; RIZZI, R. L.; DORNELES, R. V.; DIVERIO, T. A. Parallelizing Conjugate Gradient Method for Clusters Using MPI and Threads. In: INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED PROCESSING TECHNIQUES AND APPLICATIONS, PDPTA, 2002. **Proceedings...** Las Vegas: [s.n.], 2002.
- [PIC2002a] PICININ JÚNIOR., D.; MARTINOTTO, A. L.; RIZZI, R. L.; DORNELES, R. V.; DIVERIO, T. A.; NAVAUX, P. O. A. Ordenamento de Mensagens e Pré-Condicionamento na Solução Paralela do Gradiente Conjugado em Clusters de PCs Multiprocessados. In: WORKSHOP EM SISTEMAS COMPUTACIONAIS DE ALTO DESEMPENHO, WSCAD, 3., 2002. **Anais...** Vitória : SBC, 2002. 8 p.
- [PIE84] PIELKE, R. A. **Mesoscale Meteorological Modeling**. New York: Academic Press, 1984. 612 p.
- [PIL96] PILKINGTON, J. R. Dynamic Partitioning of Non-Uniform Structured Workloads with Spacefilling Curves. **IEEE Transactions on Parallel and Distributed Systems**, [S.l.], v. 7, n. 3, p. 288-300, Mar. 1996.
- [POL92] POLLARD, A. D. **Preconditioned Conjugate Gradient: Methods for Serial and Parallel Computers**. 1992. 93 p. Thesis (Master in Mathematic) - Department of Mathematics, University of Reading, Reading.
- [POV98] POVITSKY, A. **Parallelization of the Pipelined Thomas Algorithm**. Hampton: Institute for Computer Application in Science and Engineering, 1998. 21 p. (ICASE Report n. 98-48).
- [POV98a] POVITSKY, A. **Parallelization of the Pipelined Thomas Algorithm for Sets of Linear Banded Systems**. Hampton: Institute for Computer Application in Science and Engineering, 1998. 41 p. (ICASE Report n. 98-45).
- [QUA99] QUARTERONI, A.; VALLI, A. **Domain Decomposition Methods for Partial Differential Equations**. Oxford, 1999. 376 p.
- [QUI94] QUINN, M.J. **Parallel Computing - Theory and Practice**. New York: McGraw-Hill, 1994. 446 p.
- [RAD98] RADI, B.; ESTRADE, J. Adaptive Parallelization Techniques in Global Weather Models. **Parallel Computing**, [S.l.], v. 24, p. 1167-1175, 1998.
- [RAN99] RANGO, S.; ZINGG, D. **Aerodynamic Computations Using a Higher-Order Algorithm**. Reno: American Institute of Aeronautics and Astronautics, 1999. 13 p. (AIAA 99-0167).
- [RIZ98] RIZZI, R. L. **Simulação Numérica de Modelos Meteorológicos Parte 1: Cálculo de Várias Variáveis, Equações Diferenciais e Mecânica dos Fluidos**. 1998, 109 p. Trabalho Individual I (Doutorado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.
- [RIZ99] RIZZI, R. L.; DORNELES, R. V.; DIVERIO, T. A. Solução da Equação do Transporte Bidimensional usando a Técnica ADI. In: CONGRESSO

- NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL, CNMAC, 22., 1999. **Anais...** Santos: SBMAC, 1999. p. 86.
- [RIZ99a] RIZZI, R. L. **Simulação Numérica de Modelos Meteorológicos Parte 2: Modelos Matemáticos, Discretização e Computação Paralela**. 1999, 117 p. Trabalho Individual II (Doutorado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.
- [RIZ99b] RIZZI, R. L. **Técnicas Matemáticas e Computacionais para Modelos Meteorológicos**. 1999. 149 p. Exame de Qualificação (Doutorado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.
- [RIZ2000] RIZZI, R. L.; ZEFERINO, C. A.; DORNELES, R. V.; NAVAU, P. O. A.; BAMPI, S.; SUZIN, A. A.; DIVERIO, T. A. Fluvial Flowing of Guaíba River Estuary: A Parallel Solution for the Shallow Water Equations Model. In: VECTOR AND PARALLEL PROCESSING, VECPAR, 2000. **Proceedings...** Cidade do Porto: Faculdade de Engenharia da Universidade do Porto, 2000. v.3, p. 895-896.
- [RIZ2001] RIZZI, R. L. **Modelo Computacional Paralelo para a Hidrodinâmica e para o Transporte de Massa Bidimensional e Tridimensional**. 2001. 152 p. Proposta de Tese (Doutorado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.
- [RIZ2001a] RIZZI, R. L.; DORNELES, R. V.; CANAL, A. P.; GOULART, P. C.; DIVERIO, T. A. Aplicações de Alto Desempenho: Modelos Computacionais 2D e 3D para a Hidrodinâmica e o Transporte de Massa. In: ESCOLA REGIONAL DE ALTO DESEMPENHO, ERAD, 1., 2001. **Anais...** Gramado: SBC, 2001. p. 226-228.
- [RIZ2002] RIZZI, R. L.; DORNELES, R. V.; DIVERIO, T. A.; NAVAU, P. O. A. Parallel Solution in PC Clusters by the Schwarz Domain Decomposition for Three-Dimensional Hydrodynamics Models. In: VECTOR AND PARALLEL PROCESSING, VECPAR, 2002. **Proceedings...** Cidade do Porto: [s.n.], 2002, p. 655-667.
- [RIZ2002a] RIZZI, R. L.; DORNELES, R. V.; DIVERIO, T. A. **Esquemas Numéricos para a Hidrodinâmica e o Transporte de Massa 2D e 3D e Modelos de Parametrização de Turbulência**. 2001. 160 p. (Relatório Técnico não publicado) - Instituto de Informática, UFRGS, Porto Alegre.
- [ROD80] RODI, W. Turbulence Models and Their Applications in Hydraulics. A State of the Art Review. In: EXPERIMENTAL AND MATHEMATICAL FLUID DYNAMICS, IAHR, 1980. **Proceedings...** [S.l.]: IAHR, 1980.
- [ROE97] ROEST, M. R. T. **Partitioning for Parallel Finite Difference Computations in Coastal Water Simulation**. 1997. 167 p. Ph.D. Dissertation (Hydraulic and Environmental Engineering)- Delft University of Technology, Delft. Disponível em: <<http://ta.twi.tudelft.nl/wagm/papers/pub97.html>>. Acesso em: 07 jun. 1999.
- [ROO87] ROOD, R. B. Numerical Advection Algorithms and Their Role in Atmospheric Transport and Chemistry Models. **Reviews of Geophysics**, [S.l.], v. 25, n. 1, p. 71-100, Feb. 1987.
- [ROS2000] ROSATTI, G.; BONAVENTURA, L.; VILLA, D. A. Three-Dimensional Numerical Modelling of Pollutant Transport in the Lagoon of Venice. In: CONVEGNO DI IDRAULICA E COSTRUZIONI IDRAULICHE, IDRA, 27., 2000. **Proceedings...** Trento: Ed. da Università degli Studi di Trento, 2000. Disponível em: <<http://www.ing.unitn.it/~bonavent/archi.html>>. Acesso em : 08 jan. 2001.
- [SAA96] SAAD, Y. **Iterative Methods for Sparse Linear Systems**. Boston: PWS Publishing Company, 1996. 447 p.

- [SAA99] SAAD, Y. **High-Performance Iterative Solution Methods**. 1999. Lecture Notes for CscSci 8314 and Scicom 8002. Disponível em: <ftp://ftp.cs.umn.edu/dept/users/saad/reports/index.html>. Acesso em: 01 dez. 2000.
- [SAA2002] SAAD, Y. **Iterative Methods for Large Sparse Matrix Problems: Parallel Implementations**. 84 p. (course in Winter Workshop on Iterative Methods for Large Sparse Matrix Problems) 2002. Brisbane. Disponível em: <ftp://ftp.cs.umn.edu/dept/users/saad/reports/index.html>. Acesso em: 04 Jan. 2002.
- [SAT98] SATTINGER, D. H. **Partial Differential Equations of Applied Mathematics**. Department of Mathematics Utah State University and Yale University. Disponível em: <http://http://www.math.umn.edu/~dhs/>. Acesso em: 27 nov. 2000.
- [SCE91] SCHETTINI, E. B. C. **Modelo Matemático Bidimensional de Transporte de Massa em Elementos Finitos com Ênfase em Estuários**. 1991. 89 p. Dissertação (Mestrado em Recursos Hídricos e Saneamento) - Instituto de Pesquisas Hidráulicas, UFRGS, Porto Alegre.
- [SCH98] SCHLOEGEL, K.; KARYPIS, G.; KUMAR, V. **A Performance Study of Diffusive vs Remapped Load-Balancing Schemes**. 1998. Disponível em: <http://www-users.cs.umn.edu/~karypis/publications/partitioning.html>. Acesso em : 18 nov. 2000.
- [SCH99] SCHLOEGEL, K.; KARYPIS, G.; KUMAR, V. **Dynamic Repartitioning of Adaptively Refined Meshes**. 1999. Disponível em: <http://www.arc.umn.edu/publications/preprints/abstracts99.html#99-006>. Acesso em: 01 fev. 2000.
- [SCH2000] SCHLOEGEL, K.; KARYPIS, G.; KUMAR, V. **Multilevel Repartitioning of Adaptive Meshes**. 2000. Disponível em: <http://www-users.cs.umn.edu/~karypis/publications/Talks/adaptive>. Acesso em : 18 nov. 2000.
- [SCH2000a] SCHLOEGEL, K.; KARYPIS, G.; KUMAR, V. **Graph Partitioning for High Performance Scientific Simulation**. 2000. Disponível em: <http://www-users.cs.umn.edu/~karypis/publications/partitioning.html>. Acesso em: 08 mar. 2001.
- [SHE83] SHENG, Y. P. **Mathematical Modeling of Three-Dimensional Coastal Currents and Sediment Dispersion: Model Development and Application**. Princeton : Aeronautical Research Associates of Princeton, Inc., 1983. 288p. (Technical Report CERC-83-2).
- [SHE89] SHENG, Y. P.; LEE, H. K.; WANG, K. H. On Numerical Strategies of Estuarine and Coastal Modeling. **Estuarine and Coastal Modeling**, [S.l.], p. 291-301, 1989.
- [SHE95] SHENG, Y. P.; PEENE, S.; YASSUDA, E.; DAVIE, J.; SCHOFIELD, S. **A Field and Modeling Study on Circulation and Transport in Sarasota Bay**. Florida: Coastal & Oceanographic Engineering Department, 1995. 265 p.
- [SHW94] SHEWCHUCK, J. R. **An Introduction to the Conjugate Gradient Method Without the Agonizing Pain**. School of Computer Science. Carnegie Mellon University, 1994. Disponível em: <ftp://WARP.CS.CMU.EDU>. Acesso em: out. 1999.
- [SIL86] SILVEIRA, A. L. L. **Modelo Hidrodinâmico Bidimensional com Aplicação ao Rio Guaíba**. 1986. 237. Dissertação (Mestrado em Recursos Hídricos e Saneamento) – Instituto de Pesquisas Hidráulicas, UFRGS, Porto Alegre.
- [SLA97] SLAVIEIRO, V. M. P. **O Método do Gradiente Conjugado com Produto Interno Geral**. 1997. 68 p. Dissertação (Mestrado em Matemática) - Instituto de Matemática, UFRGS, Porto Alegre.

- [SLV97] SILVA, R. S.; ALMEIDA, R. C.; GALEÃO, A. C. N.; COUTINHO, A. **Iterative Local Solvers for distributed Krylov-Schwarz Methods Applied to Convection-Diffusion Problems.** 1997. Disponível em: <<http://www.lncc.br/~rssl/renato.html>>. Acesso em : 03 abr. 1999.
- [SMT92] SMITH, C. J. **Adaptive finite difference Solutions for Convection and Convection-Diffusion Problems.** 1992. 50p. Thesis (Mestrado em Matemática) - Department of Mathematics, University of Reading, Reading.
- [SMI92] SMITH, B. F. **Domain Decomposition methods for Partial Differential Equations.** Mathematics and Computer Science Division, Argonne National Laboratory, 1992. Disponível em: <<http://www-fp.mcs.anl.gov/division/publications/abstracts/abstracts92.htm>>. Acesso em: 03 nov. 1999.
- [SMI96] SMITH, B.; BJORSTAD, P.; GROPP, W. **Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations.** Cambridge: Cambridge University, 1996. 224 p.
- [SMO82] SMOLARKIEWICZ, P. K. The Multi-Dimensional Crowley Advection Scheme. **Monthly Weather Review**, [S.l.], v. 110, p. 1968-1983, Dec. 1982.
- [SMO83] SMOLARKIEWICZ, P. K. A Simple Positive Definite Advection Scheme with Small Implicit Diffusion. **Monthly Weather Review**, [S.l.], v. 111, p. 479-486, Mar. 1983.
- [SMO84] SMOLARKIEWICZ, P. K. A Fully Multidimensional Positive Definite Advection Transport Algorithm with Small Implicit Diffusion. **Journal of Computational Physics**, [S.l.], v. 54, p. 325-362, 1984.
- [SMO90] SMOLARKIEWICZ, P. K.; GRABOWSKI, W. The Multidimensional Positive Definite Advection Transport Algorithm: Nonoscillatory Option. **Journal of Computational Physics**, [S.l.], v. 86, p. 355-375, 1990.
- [SMO92] SMOLARKIEWICZ, P. K.; PUDYKIEWICZ, J. A. A Class of Semi-Lagrangian Approximations for Fluids. **Journal of the Atmospheric Sciences**, [S.l.], v. 49, n. 22, p. 2082-2095, Nov. 1992.
- [SNO98] SCALAR Conservation Law. 1998. 121 p. Disponível em: <<http://www>>. Acesso em: 20 nov. 1999.
- [SON2000] SONG, B.; LIU, G. R.; LAM, K. Y.; AMANO, R. S. On a Higher-order Bounded Discretization Scheme. **International Journal for Numerical Methods in Fluids**, [S.l.], v. 32, p. 881-897, 2000.
- [SON94] SONG, Y.; HAIDVOGEL, D. A Semi-implicit Ocean Circulation Model Using a Generalized Topography-Following Coordinate System. **Journal of Computational Physics**, [S.l.], v. 115, p. 228-244, 1994.
- [SPA72] SPALDING, D. B. A Novel Finite-difference Formulation for Differential Expressions Involving Both First and Second Derivatives. **International Journal Numerical Methods Eng.**, [S.l.], v. 4, p. 551-567, 1972.
- [SPA2001] SPALDING, B. **Numerical Convection Schemes in Phoenix.** 2001. Disponível em: http://www.cham.co.uk/phoenix/d_polis/d_lects/numerics/scheme.htm>. Acesso em: 25 abr. 2002.
- [SPE96] SPEE, E. J.; ZEEUW, P. M.; VERWER, J. G.; BLOM, J. G.; HUNSDORFER, W. H. **Vectorization and Parallelization of a Numerical Scheme for 3D Global Atmospheric Transport-chemistry Problems.** Amsterdam: Centrum voor Wiskunde en Informatica, 1996. 24 p. (Report NM-R9620). Disponível em: <<http://dbs.cwi.nl:8080/cwwwi/owa/reports2>> Acesso em: 25 nov. 1999.
- [STA91] STANFORTH, A.; CÔTÉ, J. Semi-Lagrangian Integration Schemes for Atmospheric Models – A Review. **American Meteorological Society**, [S.l.], v. 119, p. 2206-2223, Sept. 1991.

- [STE86] STELLING, G. S.; WIERSMA, A. K.; WILLEMSE, J. M. Practical Aspects of Accurate Tidal Computations. **Journal of Hydraulic Engineering**, [S.l.], v. 112, n. 9, p. 802-816, Sept. 1986.
- [STR94] STERN, J. M. **Esparsidade, Estrutura, Estabilidade e Escalonamento em Álgebra Linear Computacional**. Recife: UFPE, 123 p. Trabalho apresentado na Escola de Computação, 9., 1994
- [SWE84] SWEBY, P.K. High Resolution Schemes Using Flux Limiters for Hyperbolic Conservation Laws. **SIAM Journal Numerical Analysis**, [S.l.], v.21, n. 5, p. 995-1011, Oct. 1994.
- [SZE99] SZEREMETA, J. F.; PETERS, S. **Análise Numérica I**: Laboratório de Computação Algébrica e Simbólica. Florianópolis: Departamento de Informática e Estatística da Universidade Federal de Santa Catarina. 1999. Disponível em : <<http://www.inf.ufsc.br/~szeremeta/>>. Acesso em: 09 mar. 2000.
- [SZW99] SZEWCZYK, R.; FERENCZ, A.; WEINSTEIN, J.; WILKENING, J. **Graph Bisection**. University of Berkeley, 1999. 10 p. (CS270, Final Report). Disponível em: <<http://www.cs.berkeley.edu/szewcyk/CS/270/>>. Acesso em: 30 dez. 2000.
- [TAL94] TALLEC, P. L. Domain Decomposition Methods in Computational Mechanics. **Computational Mechanics Advances**, Le Chesnay Cedex, v. 1, p. 121-220, Oct. 1994.
- [THM86] THOMPSON, J. F.; MARSI, Z. U. A.; MASTIN, C. W. **Numerical Grid Generation**. 1986. Disponível em : <<http://www.erc.msstate.edu/>>. Acesso em: 20 set. 1999.
- [THO94] THOMAS, S.; CÔTE, J.; STANFORTH, A.; LIE, I.; SKALIN, R. A Massively Parallel Semi-Lagrangian Advection. In: ECMWF WORKSHOP ON THE USE OF PARALLEL PROCESSORS IN METEOROLOGY, 1994. Disponível em : <<http://www.scd.ucar.edu/css/staff/thomas/>>. Acesso em 03 ago. 1999.
- [THO96] THOMAS, S.; CÔTE, J. Massively Parallel Semi-Lagrangian Advection. 1996. **Journal Simulation Theory and Practice**, [S.l.], v. 3, p. 223-238, 1996. Disponível em: <<http://www.scd.ucar.edu/css/staff/thomas/>>. Acesso em: 01 dez. 1999.
- [THO97] THOMAS, S. J.; MALEVSKY, A. V.; DESGAGNÉ, M.; BENOIT, R.; PELLERIN, P.; VALIN, M. Massively Parallel Implementation of the Mesoscale Compressible Community Model. **Parallel Computing**, [S.l.], v. 23, p. 2143-2160, 1997.
- [TOL97] TOLEDO, E.; SIMÕES, R. **Introdução à Computação Paralela**. Rio de Janeiro: LNCC/CNPq, 1997. 226 p. Trabalho apresentado na Escola de Verão em Computação Científica, 2., 1997.
- [TOS97] TOSCANI, L. V.; LEAL, L. dos S.; ROSA, D. S. **Complexidade Computacional**. Pelotas: UFPEL, 1997. 36 p. Mini-Curso na Escola de Métodos Formais para Qualidade de Software, 1997.
- [TRA2000] TRANGENSTEIN, J. A. **Numerical Solution of Partial differential Equations**. Durham: Department of Mathematics, Duke University, 2000. Disponível em: <<http://www.eos.duke.edu/Research/hydro/trangen.html>>. Acesso em : 12 jan. 2001.
- [TUC98] TUCCI, C. E. M. **Modelos Hidrológicos**. Porto Alegre: Ed. da UFRGS, 1998. 669 p.
- [UMG96] UMGIESSER, G. **Modeling Residual Currents in the Venice**. 1996. Disponível em: <<http://flux.isdgm.ve.cnr.it/~georg/venice/circul/pecs.html> > . Acesso em : 06 dez. 2000.

- [VAL2000] VALIANI, A.; CALEFFI, V.; ZANNI, A. **Finite Volume Scheme for 2D Shallow-water Equations Application to a Flood Event in the Toce River.** 2000. Disponível em: <<http://www.hrwallingford.co.uk/projects/CADAM/Contents.html>>. Acesso em: 21 dez. 2000. Zaragoza
- [VAR99] VARGA, R. S. **Matrix Iterative Analysis.** 2 nd ed. Berlin: Springer Verlag, 1999. 359p.
- [VIL2000] VILLARREAL, M.R. **Parameterization of Turbulence in the Ocean and Application of a 3D baroclinic Model to the Rio de Pontevedra.** Ph.D. Dissertation (Física) – Departamento de Física de la Materia Condensada, Grupo de Física no Lineal, Universidade de Santiago de Compostela, Santiago de Compostela. Disponível em: <http://www.gotm.net/html/4-HTML/PAPER_MAIN.html#other>. Acesso em: 03 jan. 2001.
- [VIN91] VINZON, S. B. **Modelação de Transporte de Substâncias Passivas em Corpos de Água Rasos.** 1991. 104 p. Dissertação (Mestrado em Engenharia Costeira e Oceanográfica) - COPPE, Programa de Engenharia Oceânica, UFRJ, Rio de Janeiro.
- [VLA99] VILAR, P.M. **Estudio de la Hidrodinámica de la Ría de Vigo Mediante un Modelo de Volúmenes Finitos.** Ph.D. Dissertation (Física) – Departamento de Física de la Materia Condensada, Grupo de Física no Lineal, Universidade de Santiago de Compostela, Santiago de Compostela. Disponível em: <<http://www.gotm.net/html/>>. Acesso em: 20 set. 2000.
- [VOL97] VOLLEBREGT, E. A. H. **Parallel Software Development Techniques for Shallow Water Models.** 1997. 133p. Ph.D. Dissertation (Hydraulic and Environmental Engineering) - Delft University of Technology, Delft.
- [VOL97a] VOLLEBREGT, E. A. H.; ROEST, M. R. T.; CATE, H. H.; LIN, H. X. **Performance Analysis of Parallel TRIWAQ I: Cluster of Workstations and ParsytecXplores.** [S.l.]: Delft University of Technology, 1997. 18 p. (Technical Report 97-09).
- [VOR93] VORST, Henk A. Van der. **Parallel Numerical Linear Algebra.** Utrecht: Mathematical Institute, University of Utrecht, 1993. Disponível em: <<http://www.math.uu.nl/people/vorst/>>. Acesso em: 05 abr. 1999.
- [VOR94] VORST, H. A. Van der. **Parallel Iterative Solution Methods for Linear Systems Arising from Discretized PDEs.** Utrecht: Mathematical Institute, University of Utrecht, 1994. Disponível em: <<http://www.math.uu.nl/people/vorst/>>. Acesso em: 04 out. 1999.
- [VOR94a] VORST, H. A. Van der. **Lectures Notes on Iterative Methods.** Utrecht: Mathematical Institute, University of Utrecht. 1994, Disponível em: <<http://www.math.uu.nl/people/vorst/>>. Acesso em: 20 jul. 1999.
- [VRE94] VREUGDENHIL, C.B. Introduction. In: VREUGDENHIL, C. B.; KOREN, B. (Ed.). **Numerical Methods for Advection – Diffusion Problems.** [S.l.]: Vieweg, 1994. 371 p.
- [WAL2000] WALSHAW, C. **The Jostle User Manual: Version 2.2.** 2000. Disponível em: <<http://www.gre.ac.uk/jostle>>. Acesso em: 07 mar. 2001.
- [WAL95] WALSHAW, C.; CROSS, M.; EVERETT, M. G. **Dynamic Mesh Partitioning: a Unified Optimization and Load-Balancing Algorithm.** Disponível em: <<http://www.gre.ac.uk/~c.walshaw/papers>>. Acesso em: 28 nov. 1999.
- [WAN97] WANG, H. Q.; LACROIX, M. Interpolation Techniques Applied to the Eulerian-Lagrangian Solution of the Convection-Dispersion Equation in Natural Coordinates. **Computers & Geosciences**, [S.l.], v. 23, n. 6, p. 677-688, 1997.

- [WAS86] WASHINGTON, W. M.; PARKINSON C. L. **An Introduction to Three-Dimensional Climate Modeling**. Mill Valley: Ed. University Science Books, 1986. 422 p.
- [WAT95] WATTS, J. **A Practical Approach to Dynamic Load Balancing**. 1995. 47 p. Thesis (Mestrado em Ciências da Computação) - California Institute of Technology, Scalable Concurrent Programming Laboratory, California.
- [WEI92] WEIYAN, T. **Shallow Water Hydrodynamics: Mathematical Theory and Numerical Solution for a Two-dimensional System of Shallow Water Equations**. Amsterdam : Water & Power Press, 1992. 434 p.
- [WOL94] WOLTERS, L.; CATS, G.; GUSTAFSSON, N.; WILHELMSSON, T. **Dataparallel Semi-Lagrangian Numerical Weather Forecasting**. 1994. Disponível em: <<http://www.wi.leidenuniv.nl/TechRep/1994/>>. Acesso em: 02 fev. 2001.
- [WUJ95] WU, J.; TSANIS, I. K. A Vertical/Horizontal Integration Wind-induced Circulation Model (VWI3D): A Method for Including Surface and Bottom Logarithmic Profiles. **Advances in Water Resources**, [S.l.], v.18, n.2, p. 77-87, 1995.
- [XUJ92] XU, J. Iterative Methods By Space Decomposition and Subspace Correction. **SIAM Review**, [S.l.], v.34, n. 4, p. 581-613, 1992.
- [ZEE93] ZEEUW, D.; POWELL, K. G. An Adaptively Refined Cartesian Mesh Solver for the Euler Equations. **Journal of Computational Physics**, [S.l.], v. 104 , p. 56-68, 1993.
- [ZEF2000] ZEFERINO, C.; RIZZI, R. L.; DORNELES, R. V.; BAMPI, S.; SUZIN, A. Parallel Simulation of the Hydrodynamic of Guaíba River. In: MICROELECTRONIC SEMINAR, SIM, 15., 2000. **Proceedings...** Torres: SBC, 2000. p.21-24.
- [ZHE95] ZHENG, C.; BENNETT, G. D. **Applied Contaminant Transport Modeling**. New York: Van Nostrand Reinhold, 1995. 440 p.
- [ZHE99] ZHENG, C.; WANG, P. P. **MT3DMS: A Modular Three-Dimensional Multispecies Transport Model for Simulation of Advection, Dispersion, and Chemical Reactions of Contaminants in Groundwater Systems**. Vicksburg: U.S. Army Corps of Engineers, 1999. 170 p. (Documentation and User's Guide. Report SERDP-99-1).
- [ZHU98] ZHU, J.; JOHNSON, B.; BANGALORE, P.; HUDDLESTON, D.; SKJELLUM, A. **On the Parallelization of CH3D**. 1998. Disponível em: <http://www.erc.msstate.edu/news_and_publications/public_pres_1999/zhu/CH3D>. Acesso em: 29 dez. 1999.
- [ZIJ95] ZIJLEMA, M.; WESSELING, P. **Higher Order Flux-limiting Methods for Steady-state, Multidimensional, Convection-dominated Flow**. Delft: Delft University of Thechnology, Faculty of Technical Mathematics and Informatics, 1995. 28 p. (Report 95-131).
- [ZIJ97] ZIJLEMA, M. **Flux-limiting Schemes for the Convection-diffusion Equation**. Delft: Delft University of Thechnology, Faculty of Technical Mathematics and Informatics, 1997. 24 p. (Report 97-02).
- [ZOU99] ZOPPOU, C.; ROBERTS, S. **Simulation Techniques for the Advection Equation**. Disponível em: <<http://room.anu.edu.au/~steve/publications.html>>. Acesso em: 21 nov. 2000.