UNIVERSIDADE FEDERAL DO PARANÁ



CURITIBA

2022

LUCIANO PEREIRA DA SILVA

VERIFICAÇÃO DE SOLUÇÕES NUMÉRICAS EM PROBLEMAS DIFUSIVOS RESOLVIDOS COM O MÉTODO *SMOOTHED PARTICLE HYDRODYNAMICS*

Tese apresentada ao curso de doutorado em Métodos Numéricos em Engenharia (PPGMNE), Setor de Tecnologia, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Doutor em Métodos Numéricos em Engenharia.

Orientador: Prof. Dr. Carlos Henrique Marchi Coorientador: Prof. Dr. Messias Meneguette Junior

CURITIBA 2022

Catalogação na Fonte: Sistema de Bibliotecas, UFPR Biblioteca de Ciência e Tecnologia

S586v	Silva, Luciano Pereira da Verificação de soluções numéricas em problemas difusivos resolvidos com o método Smoothed Particle Hydrodynamics [recurso eletrônico] / Luciano Pereira da Silva. – Curitiba, 2022.
	Tese - Universidade Federal do Paraná, Setor de Tecnologia, Programa de Pós-Graduação em Métodos Numéricos em Engenharia, 2022.
	Orientador: Carlos Henrique Marchi – Coorientador: Messias Meneguette Junior.
	 Equações diferenciais – Soluções numéricas. 2. Calor – Transmissão. Algoritmos algébricos. 4. Smoothed Particle Hydrodynamics. I. Universidade Federal do Paraná. II. Marchi, Carlos Henrique. III. Meneguette Junior, Messias. IV. Título.
	CDD: 515.35

Bibliotecário: Elias Barbosa da Silva CRB-9/1894



MINISTÉRIO DA EDUCAÇÃO SETOR DE CIENCIAS EXATAS UNIVERSIDADE FEDERAL DO PARANÁ PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO PROGRAMA DE PÓS-GRADUAÇÃO MÉTODOS NUMÉRICOS EM ENGENHARIA - 40001016030P0

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação MÉTODOS NUMÉRICOS EM ENGENHARIA da Universidade Federal do Paraná foram convocados para realizar a arguição da tese de Doutorado de LUCIANO PEREIRA DA SILVA intitulada: VERIFICAÇÃO DE SOLUÇÕES NUMÉRICAS EM PROBLEMAS DIFUSIVOS RESOLVIDOS COM O MÉTODO SMOOTHED PARTICLE HYDRODYNAMICS, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de doutor está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 17 de Março de 2022.

Assinatura Eletrônica 18/03/2022 22:50:31.0 MESSIAS MENEGUETTE JUNIOR Presidente da Banca Examinadora

Assinatura Eletrônica 18/03/2022 19:05:08.0 VIVIANA COCCO MARIANI Avaliador Externo (PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ)

> Assinatura Eletrônica 18/03/2022 16:29:46.0 FABIANO PETRONETTO DO CARMO Avaliador Externo

Assinatura Eletrônica 18/03/2022 11:00:43.0 LEANDRO FRANCO DE SOUZA Avaliador Externo (UNIVERSIDADE DE SÃO PAULO)

Assinatura Eletrônica 20/03/2022 12:24:46.0 ADMILSON TEIXEIRA FRANCO Avaliador Externo (UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ)

Dedico esta tese à minha irmã Elessandra e meu pai Julio (in memoriam), à minha mãe Cleunice, ao meu tio Orlando e meus amigos Dênis, Pedro, Sara e Petrônio.

AGRADECIMENTOS

À minha família, especialmente aos meus pais.

Ao meu orientador, professor Dr. Carlos Henrique Marchi.

Ao meu coorientador, professor Dr. Messias Meneguette Junior.

Aos professores, Dra. Viviana C. Mariani e Dr. Admilson T. Franco.

Aos professores, Dr. Leandro F. de Souza e Dr. Fabiano P. do Carmo.

Aos amigos do Laboratório de Experimentação Numérica (LENA).

À Universidade Estadual Paulista "Júlio de Mesquita Filho".

Ao Núcleo de Computação Científica NCC/GridUNESP.

Ao professor, Dr. Carlos Alberto Dutra Fraga Filho.

A Réverton Luis Antunes Neundorf.

À Universidade Federal do Paraná.

À professora, Dra. Roberta Suero.

Aos meus amigos pessoais.

Aos meus professores.

Ao PPGMNE.

Ao PosMAC.

À CAPES.

A Deus.

O futuro tem muitos nomes. Para os fracos é o inalcançável. Para os medrosos, o desconhecido. Para os valentes, a oportunidade. **Victor Hugo**

RESUMO

A presente pesquisa realiza a verificação de soluções numéricas de equações diferenciais que modelam alguns fenômemos da transferência de calor. As primeiras análises são realizadas por meio de um estudo detalhado de erros numéricos ao aplicar o método "Hidrodinâmica das Partículas Suavizadas" (em inglês, Smoothed Particle Hydrodynamics, SPH) para discretizar os modelos matemáticos. Adotando-se discretização uniforme do domínio, foi possível identificar novas fontes de erro numérico específicas do método SPH e por meio delas, foi possível apontar as características das soluções influenciadas por tais erros. A dedução do operador laplaciano 1D e 2D foi de suma importância para a interpretação do erro de suavização que é encontrado apenas nos casos 2D. Com a Multiextrapolação de Richardson (em inglês, Repeated Richardson Extrapolation, RRE) em precisão quádrupla, mostrou-se como aumentar a ordem de acurácia das soluções numéricas e consequentemente, reduzir o erro de discretização em até 24 ordens de magnitude. Os casos 1D produziram soluções *benchmark* usando o SPH, que eleva o pontencial desse estudo aos níveis de referência. Os modelos 2D exigem maiores esforços, a começar pela determinação de um método iterativo capaz de resolver os sistemas de equações lineares esparsos, com 25 diagonais, sendo seus elementos associados, números reais. Essa dificuldade foi contornada usando o método de Gauss-Seidel-S paralelizado para matriz compactada e também com Multinível Algébrico (em inglês, Algebraic Multilevel, AML). O AML mostrou-se uma ferramenta poderosa para resolver os sistemas, determinando speed-up de até 5136 vezes em relação ao SL paralelizado, isso contribui para que novos trabalhos considerem a não utilização de métodos explícitos para determinar as soluções numéricas com SPH. Identificou-se com o estudo, a necessidade de definir uma nova forma de discretização para que a RRE fosse aplicada até mesmo em casos onde as partículas possuem movimentação. Dessa forma, definiu-se a discretização por partícula sensor fixa e testes envolvendo discretização desordenada foram apresentados. Por fim, constatou-se que a verificação qualitativa e quantitativa das soluções numéricas por meio dos testes de coerência e métricas, são atualmente, uma metodologia de excelência para tratar problemas da transferência de calor computacional.

Palavras-chave: Hidrodinâmica das Partículas Suavizadas. Verificação. Transferência de Calor Computacional. Multinível Algébrico. Multiextrapolação de Richardson.

ABSTRACT

The present research performs verification of numerical solutions differential equations that model some heat transfer phenomena. The first analyses are performed through a detailed study of numerical errors when applying the Smoothed Particle Hydrodynamics (SPH) method to discretize the mathematical models. By adopting uniform discretization of the domain, it was possible to identify new sources of numerical error specific to the SPH method and through them, it was possible to point out the characteristics of the solutions influenced by such errors. The deduction of the 1D and 2D laplacian operator was of paramount importance for the interpretation of the smoothing error that is found only in 2D cases. With Repeated Richardson Extrapolation (RRE) in quadruple precision, it was shown how to increase the order of accuracy of the numerical solutions and consequently reduce the discretization error by up to 24 orders of magnitude. The 1D cases produced benchmark solutions using SPH, which raises the pontential of this study to benchmark levels. The 2D models require greater efforts, starting with determining an iterative method that was capable of solving the sparse linear equation systems with 25 diagonals, their associated elements being real numbers. This difficulty was circumvented using the parallel Gauss-Seidel-S method for compressed matrix and also with Algebraic Multilevel (AML). AML proved to be a powerful tool to solve the systems, determining speed-up of up to 5136 times compared to parallel SL, this contributes for new works to consider not using explicit methods to determine the numerical solutions with SPH. The study identified the need to define a new form of discretization so that the RRE could be applied even in cases where the particles are moving. Thus, the fixed sensor particle discretization was defined and tests involving disordered discretization were presented. Finally, it was found that the qualitative and quantitative verification of numerical solutions through consistency tests and metrics are currently a methodology of excellence to treat computational heat transfer problems.

Keywords: Smoothed Particle Hydrodynamics. Verification. Computational Heat Transfer. Algebraic Multilevel. Repeated Richardson Extrapolation.

LISTA DE FIGURAS

FIGURA 1 –	FLUXO DE ÓLEO DENTRO DE UM REDUTOR	27
FIGURA 2 –	A FUNÇÃO NÚCLEO (3.10) E SUAS DUAS PRIMEIRAS DERI- VADAS.	44
FIGURA 3 –	A FUNÇÃO GAUSSIANA (3.11) E SUAS DUAS PRIMEIRAS DERIVADAS	45
FIGURA 4 –	A FUNÇÃO <i>SPLINE</i> CÚBICA (3.12) E SUAS DUAS PRIMEIRAS DERIVADAS.	46
FIGURA 5 –	A FUNÇÃO NÚCLEO <i>NEW</i> QUÁRTICO (3.13) E SUAS DUAS PRIMEIRAS DERIVADAS.	47
FIGURA 6 –	A FUNÇÃO <i>SPLINE</i> QUÍNTICA (3.14) E SUAS DUAS PRIMEI- RAS DERIVADAS	48
FIGURA 7 –	A FUNÇÃO DUPLO COSSENO (3.15) E SUAS DUAS PRIMEIRAS DERIVADAS COM $\kappa = 2.0.$	49
FIGURA 8 –	PARTÍCULAS <i>GHOSTS</i> ATRIBUÍDAS NA REGIÃO EXTERNA DO DOMÍNIO.	50
FIGURA 9 –	PARTÍCULAS <i>DUMMY</i> ATRIBUÍDAS NA REGIÃO EXTERNA DO DOMÍNIO	51
FIGURA 10 –	RECRUTAMENTO OTIMIZADO DE PARTÍCULAS POR MALHA	52
FIGURA 11 –	LIMITAÇÃO SUPERIOR DE ALGORITMOS PARALELIZADOS	61
FIGURA 12 –	REPRESENTAÇÃO DE UM CICLO $V(\nu_1,\nu_2)$ DO <i>MULTIGRID</i> .	62 65
FIGURA 15 – FIGURA 14 –	REPRESENTAÇÃO DOS NÍVEIS DO AML	67
FIGURA 15 – FIGURA 16 – FIGURA 17 –	ERRO DE ARREDONDAMENTO.	71 72 77
FIGURA 18 –	REPRESENTAÇÃO DA GEOMETRIA EM FORMATO DE QUA-	
	DRADO UNITÁRIO	83
FIGURA 19 – FIGURA 20 – FIGURA 21 –	ERRO DE ITERAÇÃO	99 100
FIGURA 22 –	$h = \Delta x.$ EFEITOS DO ERRO DE INCONSISTÊNCIA SOBRE A SOLUÇÃO NUMÉRICA NUMÉRICA OBTIDA COM $N_p = 8 \to h = \Delta x.$	102 102

FIGURA 23 –	DOMÍNIO DISCRETIZADO COM ($N_{ti} = 49$) PARTÍCULAS \mathbf{x}_i INTERNAS 103
FIGURA 24 –	DOMÍNIO DE APOIO DISCRETIZADO COM $(N_{p} + 5)^{2} - (N_{p} - 1)^{2}$
110,0101121	PARTÍCULAS \mathbf{x}_i DUMMY
FIGURA 25 –	RECRUTAMENTO PARCIAL INTERNO DE PARTÍCULAS NO
	SUPORTE COMPACTO V_i SEM TRATAMENTO DE FRONTEIRA.106
FIGURA 26 –	RECRUTAMENTO PARCIAL INTERNO DE PARTÍCULAS NO
	SUPORTE COMPACTO V_i COM TRATAMENTO DA FRONTEIRA.106
FIGURA 27 –	RECRUTAMENTO PARCIAL EXTERNO DE PARTÍCULAS NO
	SUPORTE COMPACTO V_i NA REGIÃO PRÓXIMA DA FRON-
	TEIRA
FIGURA 28 –	RECRUTAMENTO COMPLETO DE PARTÍCULAS NO SUPORTE
	COMPACTO V_i NA REGIÃO PRÓXIMA DA FRONTEIRA 107
FIGURA 29 –	SOLUÇÃO ANALÍTICA PARA DOMÍNIO COMPLETO OBTIDA
	COM $N_t = 81$ PARTÍCULAS
FIGURA 30 –	SOLUÇÃO NUMÉRICA PARA DOMÍNIO COMPLETO OBTIDA
	COM $N_t = 81$ PARTÍCULAS
FIGURA 31 –	SOLUÇÃO ANALÍTICA PARA DOMÍNIO COMPLETO OBTIDA
	$COM N_t = 4225 PARTÍCULAS. \dots \dots$
FIGURA 32 –	SOLUÇÃO NUMÉRICA PARA DOMÍNIO COMPLETO OBTIDA
	$COM N_t = 4225 PARTÍCULAS. \dots \dots$
FIGURA 33 –	SOLUÇÃO ANALÍTICA E NUMÉRICA OBTIDA COM $N_{ti} = 49$
	PARTÍCULAS PARA $\mathbf{x}_1 = \mathbf{x}_1(0, 125; y)$
FIGURA 34 –	SOLUÇÃO ANALÍTICA <i>VERSUS</i> NUMÉRICA COM ERRO DE
	INCONSISTÊNCIA DE PARTÍCULAS OBTIDA COM $N_{ti} = 49$
	PARTÍCULAS PARA $\mathbf{x}_1 = \mathbf{x}_1(0, 125; y)$
FIGURA 35 –	SOLUÇÃO ANALÍTICA E NUMÉRICA OBTIDA COM $N_{ti} = 3969$
	PARTICULAS PARA $\mathbf{x}_1 = \mathbf{x}_1(0,015625; y)$
FIGURA 36 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA NO PONTO
	MEDIO USANDO $W_{c,2}^1$ COM RRE (E1-1Dp)
FIGURA 37 –	ORDEM APARENTE (p_U) DA TEMPERATURA NO PONTO
	MEDIO USANDO $W_{c,2}^1$ COM RRE (E1-1Dp)
FIGURA 38 –	ORDEM EFETIVA (p_E) DA TEMPERATURA NO PONTO MEDIO
	USANDO $W_{c,2}^1$ COM RRE (E1-1Dp)
FIGURA 39 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA MEDIA USANDO
	$W_{c,2}^{\perp}$ COM RRE (E1-1Dp)
FIGURA 40 –	ORDEM APARENTE (p_U) DA TEMPERATURA MEDIA USANDO
	$W_{c,2}^{\perp}$ COM RRE (E1-1Dp)

FIGURA 41 –	ORDEM EFETIVA (p_E) DA TEMPERATURA MÉDIA USANDO
	$W_{c,2}^1$ COM RRE (E1-1Dp)
FIGURA 42 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA NO PONTO
	MÉDIO USANDO $W_{c,2}^1$ COM RRE (E2-1Dp)
FIGURA 43 –	ORDEM APARENTE (p_U) DA TEMPERATURA NO PONTO
	MÉDIO USANDO $W_{c,2}^1$ COM RRE (E2-1Dp)
FIGURA 44 –	ORDEM EFETIVA (p_E) DA TEMPERATURA NO PONTO MÉDIO
	USANDO $W_{c,2}^1$ COM RRE (E2-1Dp)
FIGURA 45 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA MÉDIA USANDO
	$W_{c,2}^1$ COM RRE (E2-1Dp)
FIGURA 46 –	ORDEM APARENTE (p_U) DA TEMPERATURA MÉDIA USANDO
	$W_{c,2}^1$ COM RRE (E2-1Dp)
FIGURA 47 –	ORDEM EFETIVA (p_E) DA TEMPERATURA MÉDIA USANDO
	$W_{c,2}^1$ COM RRE (E2-1Dp)
FIGURA 48 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA NO PONTO
	MÉDIO USANDO $W_{c,2}^1$ COM RRE (E3-1Dp)
FIGURA 49 –	ORDEM APARENTE DA TEMPERATURA NO PONTO MÉDIO
	USANDO $W_{c,2}^1$ COM RRE (E3-1Dp)
FIGURA 50 –	ORDEM EFETIVA DA TEMPERATURA NO PONTO MÉDIO
	USANDO $W_{c,2}^1$ COM RRE (E3-1Dp)
FIGURA 51 –	ERRO DE DISCRETIZAÇÃO DA DERIVADA DA TEMPERA-
	TURA NO CONTORNO $x=0$ USANDO $W^1_{c,2}$ COM RRE (E3-1Dp).131
FIGURA 52 –	ORDEM APARENTE DA DERIVADA DA TEMPERATURA NO
	CONTORNO $x = 0$ USANDO $W_{c,2}^1$ COM RRE (E3-1Dp) 131
FIGURA 53 –	ORDEM EFETIVA DA DERIVADA DA TEMPERATURA NO
	CONTORNO $x = 0$ USANDO $W_{c,2}^1$ COM RRE (E3-1Dp) 132
FIGURA 54 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA MÉDIA USANDO
	$W_{c,2}^1$ COM RRE (E3-1Dp)
FIGURA 55 –	ORDEM APARENTE DA TEMPERATURA MÉDIA USANDO
	$W_{c,2}^1$ COM RRE (E3-1Dp)
FIGURA 56 –	ORDEM EFETIVA DA TEMPERATURA MÉDIA USANDO $W^1_{\boldsymbol{c},2}$
	COM RRE (E3-1Dp)
FIGURA 57 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA NO PONTO
	MÉDIO EM $t = 1,0s$ USANDO $W_{c,2}^1$ COM RRE (E1-1Dt) 134
FIGURA 58 –	ORDEM APARENTE (p_U) DA TEMPERATURA NO PONTO
	MÉDIO EM $t = 1.0s$ USANDO $W_{c,2}^1$ COM RRE (E1-1Dt) 135
FIGURA 59 –	ORDEM EFETIVA (p_E) DA TEMPERATURA NO PONTO MÉDIO
	EM $t = 1,0s$ USANDO $W_{c,2}^1$ COM RRE (E1-1Dt)

FIGURA 60 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA MÉDIA EM
	$t=1,0s$ USANDO $W^1_{c,2}$ COM RRE (E1-1Dt)
FIGURA 61 –	ORDEM APARENTE (p_U) DA TEMPERATURA MÉDIA EM $t =$
	1,0s USANDO $W_{c,2}^1$ COM RRE (E1-1Dt)
FIGURA 62 –	ORDEM EFETIVA (p_E) DA TEMPERATURA MÉDIA EM $t = 1,0s$
	USANDO $W_{c,2}^1$ COM RRE (E1-1Dt)
FIGURA 63 –	COMPARAÇÃO DAS ORDENS EFETIVAS (p_E) DOS EXEMPLOS
	(E3-1Dp) e (E1-1Dt)
FIGURA 64 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA NO PONTO
	MÉDIO USANDO $W_{e,24}^1$ COM RRE (E1-2Dp)
FIGURA 65 –	ORDEM APARENTE DA TEMPERATURA NO PONTO MÉDIO
	USANDO $W_{e,24}^1$ COM RRE (E1-2Dp)
FIGURA 66 –	ORDEM EFETIVA DA TEMPERATURA NO PONTO MÉDIO
	USANDO $W_{e,24}^1$ COM RRE (E1-2Dp)
FIGURA 67 –	ERRO DE DISCRETIZAÇÃO DA TEMPERATURA MÉDIA USANDO
	$W_{e,24}^1$ COM RRE (E1-2Dp)
FIGURA 68 –	ORDEM APARENTE DA TEMPERATURA MÉDIA USANDO
	$W_{e,24}^1$ COM RRE (E1-2Dp)
FIGURA 69 –	ORDEM EFETIVA DA TEMPERATURA MÉDIA USANDO $W_{e,24}^1$
	COM RRE (E1-2Dp)
FIGURA 70 –	TEMPO DE CPU VERSUS NÚMERO DE NÍVEIS DO AML (E1-
	2Dp)
FIGURA 71 –	ESCALABILIDADE DO SOLVER GAUSS-SEIDEL-S CSR (E1-2Dp).147
FIGURA 72 –	TEMPO DE CPU VERSUS NÚMERO DE PARTÍCULAS (E1-2Dp).149
FIGURA 73 –	SPEED-UP VERSUS NÚMERO DE PARTÍCULAS (E1-2Dp) 149
FIGURA 74 –	REPRESENTAÇÃO DO RAIO DE INFLUÊNCIA PARA DISCRE-
	TIZAÇÃO COM DESORDEM CANÔNICA DE PARTÍCULAS 150
FIGURA 75 –	REPRESENTAÇÃO DA DISCRETIZAÇÃO CANÔNICA COM
	PARTÍCULA SENSOR FIXA
FIGURA 76 –	SOLUÇÃO ANALÍTICA DO EXEMPLO (E1-2Dp)
FIGURA 77 –	SOLUÇÃO NUMÉRICA DO EXEMPLO (E1-2Dp)
FIGURA 78 –	SOLUÇÃO ANALÍTICA DO EXEMPLO (E2-2Dp)
FIGURA 79 –	SOLUÇÃO NUMÉRICA DO EXEMPLO (E2-2Dp)
FIGURA 80 –	TEMPO DE CPU VERSUS NÚMERO DE NÍVEIS DO AML (E1-
	2Dp)
FIGURA 81 –	TEMPO DE CPU VERSUS NÚMERO DE NÍVEIS DO AML (E2-
	2Dp)
FIGURA 82 –	TEMPO DE CPU VERSUS NÚMERO DE PARTÍCULAS (E1-2Dp).156
FIGURA 83 –	SPEED-UP VERSUS NÚMERO DE PARTÍCULAS (E1-2Dp) 156

FIGURA 84 –	DISPOSIÇÃO DE PARTÍCULAS EQUIDISTANTES NOS QUA-	
	DRANTES	7

LISTA DE TABELAS

TABELA 1 –	DEFINIÇÃO DAS VARIÁVEIS DE INTERESSE
TABELA 2 –	ÍNDICES DO POLINÔMIO DE TAYLOR EXPRESSOS NA EQUA-
	ÇÃO (5.18)
TABELA 3 –	ERRO VERDADEIRO DE $\psi_{e,24}(\mathbf{x}_1)$
TABELA 4 $-$	MOMENTOS DA DERIVADA PRIMEIRA $W_{c,2}^1$
TABELA 5 $-$	MOMENTOS DA DERIVADA PRIMEIRA $W^1_{c,2}.$
TABELA 6 $-$	MÓDULO DO ERRO DE DISCRETIZAÇÃO USANDO W^1_c COM
	RRE (E1-1Dp)
TABELA 7 $-$	COMPARAÇÃO ENTRE AS ORDENS DE ACURÁCIA DA VA-
	RIÁVEL ψ_m (E2-1Dp)
TABELA 8 –	TERMOS DO ERRO DE TRUNCAMENTO (E1-1Dp) 139
TABELA 9 –	SOLUÇÕES BENCHMARK OBTIDAS COM SPH PARA A VA-
	RIÁVEL $\psi(1/2)$
TABELA 10 -	PARÂMETROS DAS SIMULAÇÕES140
TABELA 11 –	GRAU DE ESPARSIDADE DAS MATRIZES
TABELA 12 –	TERMOS DO ERRO DE TRUNCAMENTO(E1-2Dp) 145
TABELA 13 –	ORDEM DE COMPLEXIDADE
TABELA 14 –	ORDEM DE COMPLEXIDADE USANDO DISCRETIZAÇÃO DE-
	SORDENADA (E1-2Dp)
TABELA 15 –	RESUMO DAS ORDENS DE ACURÁCIA ENCONTRADAS A
	PRIORI E A POSTERIORI
TABELA 16 –	ESTUDO DE SINAL DOS TERMOS DAS INTEGRAIS CONTIDAS
	NA EQUAÇÃO (5.21)

LISTA DE ABREVIATURAS E SIGLAS

1D	– Unidimensional
2D	– Bidimensional
AMG	– Algebraic Multigrid
AML	– Algebraic Multilevel
CDS-n	– Central Difference Scheme of nth order
CFD	- Computational Fluid Dynamics
CHT	– Computational Heat Transfer
\mathbf{CS}	- Correction Scheme
CSPM	- Corrective Smoothed Particle Method
CSR	– Compressed Sparse Row
CPU	– Central Processing Unit
CRRE	– Completed Repeated Richardson Extrapolation
DDS-n	– Downstream Difference Scheme of nth order
E1-1Dp	– Primeiro exemplo da difusão de calor em regime permamente 1D
E2-1Dp	– Segundo exemplo da difusão de calor em regime permamente 1D
E3-1Dp	– Terceiro exemplo da difusão de calor em regime permamente 1D
E1-1Dt	– Primeiro exemplo da difusão de calor em regime transiente 1D
E1-2Dp	– Primeiro exemplo da difusão de calor em regime permamente 2D
E2-2Dp	-Segundo exemplo da difusão de calor em regime permamente 2D
FAS	– Full Approximation Scheme
FDM	– Finite Difference Method
FDMA	– Five-diagonal Matrix Algorithm
FEM	– Finite Element Method
FPM	– Finite Particle Method

FRE - Full Richardson Extrapolation FVM - Finite Volume Method MM - Meshless Method – Message Passing Interface MPI MPS - Moving Particle Semi-Implicit OpenMP -Open Multi-Processing PDMA - Pentadiagonal Matrix Algorithm RAM - Random-Access Memory RBFFD - Radial Basis Function based Finite Difference Method RRE - Repeated Richardson Extrapolation SG- Single Grid SL - Single Level SPH - Smoothed Particle Hydrodynamics TDMA - Tridiagonal Matrix Algorithm Tol - Tolerância para interrupção do processo iterativo UDS-n - Upwind Difference Scheme of nth order

LISTA DE SÍMBOLOS

Símbolo	- Descrição Unidade
a_{ij}	– Coeficientes da matriz A
A	– Matriz de coeficientes
AA	– Vetor do método CSR
AD	– Vetor com os elementos da diagonal principal da matriz A que estão no vetor AA do método CSR
С	– Coeficiente da equação que determina a ordem de complexidade
C	– Conjunto de variáveis que estão na malha grossa
\overline{C}	– Coeficiente constante
C_i	– Conjunto dos pontos vizinhos da malha grossa que influenciam fortemente \boldsymbol{i}
C_{ξ}^{γ}	– Consistência do polinômio com γ -ésima derivada e ξ -ésima ordem
D_i	– Conjunto dos pontos vizinhos que não estão na malha grossa
D_i^S	– Conjunto dos pontos vizinhos da malha fina que influenciam fortemente \boldsymbol{i}
D_i^W	– Conjunto dos pontos com fraca influência sobre i
E_{ψ}	– Erro de discretização
E_{τ}	– Erro de truncamento
E_n	– Erro de iteração
E_l	– Erro de iteração com SPH
E_{π}	– Erro de arredondamento
е	– Número de <i>Euler</i>
e^h, e^H	– Erro algébrico nas malhas fina e grossa, respectivamente
f	– Termo fonte que caracteriza o vetor independente do sistema linear
f_p	– Fração paralelizada de um algoritmo
F	– Conjunto de variáveis que não estão na malha grossa

GE	– Grau de esparsidade	
h	– Comprimento de suavização	[m]
h_x	– Comprimento de suavização na direção \boldsymbol{x} para o caso 21	D [<i>m</i>]
h_y	– Comprimento de suavização na direção \boldsymbol{y} para o caso 21	D [<i>m</i>]
I_h	– Operador identidade da malha fina	
I_H^h	– Operador de restrição	
I_h^H	– Operador de prolongação	
IA	– Vetor do método CSR	
JA	– Vetor do método CSR	
k	– fator de escala do raio da base da função núcleo	[m]
kh	– raio da base da função núcleo adaptável	[m]
l	– Número de iterações	
$L_1, L_2, L_\infty,$	– Normas	[m]
m_j	– Massa da partícula na posição \mathbf{x}_j	[kg]
$M, \overline{M}, \overline{\overline{M}}$	– Momentos do núcleo SPH	adimensional
Ν	– Número de partículas	
NA	– Não se aplica	
$N_{\overline{ti}}$	– Número de partículas internas que discretizam o domín	io - caso 1D
$N_{\overline{t}}$	– Número total de partículas que discretizam o domínio i tornos - caso 1D	ncluindo con-
N_p	– Número de espaços médios uniformes entre partículas -	caso 1D
N_{p_x}	– Número de espaços médios uniformes entre partículas na direção \boldsymbol{x}	
N_{p_y}	– Número de espaços médios uniformes entre partículas na direção \boldsymbol{y}	
N_{p_z}	– Número de espaços médios uniformes idêntico nas direções x e y	
N_x	– Número de partículas internas na direção \boldsymbol{x} que discretiz	am o domínio
N_y	– Número de partículas internas na direção y que discretiz	am o domínio

N_{ti}	– Número de partículas internas que discretizam o domínio $(N_x N_y)$	
N_t	– Número total de partículas que discretizam o domínio incluindo con- tornos	
N_i	-Número de partículas dentro do suporte compacto - caso 2D	
\overline{N}_i	– Conjunto de pontos que determina a vizinhança de i	
N_L	– Número de níveis no AML	
N_D	– Número de níveis da discretização do domínio	
n	– Ordem de matriz	
n_{aa}	– Dimensão dos vetores AA e JA do método CSR	
n_{ia}	– Dimensão do vetor IA método CSR	
\overline{p}	– Ordem de complexidade	
$\overline{p_d}$	– Ordem de complexidade com discretização desordenad	a
$\overline{p_u}$	– Ordem de complexidade com discretização uniforme	
p_E	– Ordem efetiva	a dimensional
p_L	– Ordem assintótica a priori $(\min\{p_V\})$	a dimensional
p_0	– Ordem assintótica a posteriori $(\min\{p_m\})$	a dimensional
p_0, p_1, \ldots, p_m	– Ordens de acurácia equivalentes (elementos de p_U e p_E)	a dimensional
p_V	– Ordens verdadeiras	a dimensional
p_U	– Ordens aparentes	a dimensional
\mathbf{Q}	– coordenadas da partícula sensor fixa	[m]
q	– Razão de refino	[m]
\mathbb{R}	– Conjunto dos números reais	
r_{ij}	– Distância entre uma partícula \mathbf{x}_i e as \mathbf{x}_j	[m]
re	– Razão de engrossamento	[m]
S_h	– Suavizador ou <i>solver</i>	
S_i	– Conjunto de pontos que influencia fortemente \boldsymbol{i}	

S_i^D	– Conjunto de pontos que depende fortemente de ${\cal C}_i$	
S_i^T	– Conjunto de pontos que depende fortemente de i	
S_p	-Speed-up entre algoritmos seriais	
S_{τ}	-Speed-up entre algoritmos paralelizados	
t_{CPU}	– Tempo de CPU do algoritmo \varUpsilon	
t_0	– Tempo de CPU do algoritmo serial Υ	
t_{τ}	– Tempo de CPU do algoritmo paralelizado Υ	
U	– Conjunto de variáveis indefinidas	
V_i	– Conjunto que define o suporte compacto de \mathbf{x}_i	
$W_{w^*}^\gamma(\phi)$	– Expressão completa da função núcleo e suas derivadas	
$W^{\gamma}_{w^*,j}$	– Núcleo w^* com j partículas recrutadas	
$w_{w^*}^\gamma(\phi)$	– Expressão parcial da função núcleo e suas derivadas	
x	– Partícula de referência (efetiva) na forma contínua	[m]
\mathbf{x}'	– Partícula vizinha na forma contínua	[m]
\mathbf{x}_i	– Partícula de referência (efetiva) na forma discreta	[m]
\mathbf{x}_{j}	– Partícula vizinha na forma discreta	[m]
\mathbf{X}_{s}	– Partícula sensor fixa	[m]
x, y	– coordenadas da partícula de referência na forma contínua	[m]
x', y'	– coordenadas da partícula vizinha na forma contínua	[m]
x_i, y_i	– coordenadas da partícula de referência na forma discreta	[m]
x_j, y_j	– coordenadas da partícula vizinha na forma discreta	[m]
$\langle \rangle$	– Operador SPH	

Símbolos gregos

 $\begin{array}{lll} \alpha_d & & - \mbox{Constante de normalização} \\ \beta_1, \beta_2 & & - \mbox{Coeficientes constantes} \\ \gamma & & - \mbox{Ordem da derivada} \end{array}$

Δv_j	– Volume da partícula \mathbf{x}_j	$[m^3]$
Δm	– Incremento das ordens de acurácia	adimensional
Δx	– Tamanho do elemento de malha na direção \boldsymbol{x}	[m]
Δy	– Tamanho do elemento de malha na direção \boldsymbol{y}	[m]
Δt	– Tamanho do passo temporal	[s]
δ	– Distribuição Delta de <i>Dirac</i>	
ϵ	– Tolerância suficientemente pequena	
ε	– Fator de forte dependência na malha grossa	
ζ	– Índice do somatório	
abla.	– Operador divergente	
∇	– Operador gradiente	
$ abla^2$	– Operador laplaciano	
η	– Perturbação ou desordem escalar associada a valores ra altera as coordenadas das partículas	andômicos que $[m]$
$\overline{\eta}$	– Deslocamento da partícula	[m]
heta	– Fator de redução da malha	
κ	– Amplitude e frequência da função núcle o W_f	[m]
λ_i	– Medida de importância	
μ	– Difusividade térmica	$[m^2/s]$
$ u_1$	– Pré-suavização	
$ u_2$	– Pós-suavização	
ξ	– Grau do polinômio	
π	– Número	
$\rho(A)$	– Raio espectral da matriz A	
ρ	– Massa específica	$[kg/m^3]$
$ ho_j$	– Massa específica da partícula \mathbf{x}_j	$[kg/m^3]$

Σ	– Somatório	
au	– Número de processadores	
ϕ_{ij}	– Distância normalizada entre as partícula \mathbf{x}_i e \mathbf{x}_j	[m]
ϕ^η_{ij}	– Distância normalizada as partícula \mathbf{x}_i e \mathbf{x}_j com perturbação η	[m]
ψ^h	– Solução suavizada	[K]
$\psi(\mathbf{x})$	– Função genérica que representa a temperatura	[K]
$\psi(1/2)$	– Temperatura no ponto médio	[K]
ψ_m	– Temperatura média	[K]
$\psi^\gamma(\mathbf{x}_i)$	– γ -ésima derivada numérica da função que representa a temperat $[K]$	ura
$\psi_{a,b,c,\dots,j}$	– Função genérica usando um núcle o a, b, c, \ldots com j partículas	[K]
$\varPsi_{a,b,c,,j}$	– Solução exata numérica com núcle o a, b, c, \ldots e j partículas	[K]
Ψ	– Solução exata contínua que representa a temperatura	[K]
$arPsi^{\gamma}(\mathbf{x}_i)$	– γ -ésima derivada analítica da função que representa a temperat $[K]$	ura
Ω	– Domínio do problema	
$\partial \Omega$	– Fronteira do domínio do problema	
ω_{ij}	– Função de interpolação do AMG	
Subscritos		
h, H	– Malha fina e grossa, respectivamente	[m]
i	– Subíndice vinculado à partícula principal \mathbf{x}_i	
j	– Subíndice vinculado às partículas recrutadas \mathbf{x}_j	
w^*	– Modelo da função núcleo	
Sobrescrito	S	
h, H	– Malha fina e grossa, respectivamente	[m]
n,m	– Dimensões \mathbb{R}^n e \mathbb{R}^m	

SUMÁRIO

1	INTRODUÇÃO	26
1.1	Caracterização do problema	26
1.2	Relevância	26
1.3	Objetivos	28
1.4	Estrutura textual	28
2	REVISÃO DA LITERATURA	30
2.1	Transferência de calor computacional	30
2.2	Métodos numéricos para equações diferenciais	31
2.2.1	Métodos com malha	31
2.2.2	Métodos sem malha	31
2.2.3	Métodos híbridos	32
2.3	Método SPH	32
2.4	Método <i>multigrid</i>	34
2.5	Multiextrapolação de Richardson	35
2.6	Verificação de soluções numéricas	37
3	FUNDAMENTAÇÃO	39
3.1	Modelo numérico	39
3.1.1	Formulação SPH	39
3.1.2	Representação integral de uma função	39
3.1.3	Aproximação utilizando partículas	40
3.1.4	Propriedades da função núcleo	41
3.1.5	Função núcleo SPH: definição e modelos	43
3.1.6	Modelos clássicos de funções núcleo SPH e suas derivadas $(W^{\gamma}_{w^*}(\phi))$	43
3.2	Tratamento de fronteira	49
3.2.1	Partículas ghosts	49
3.2.2	Partículas dummy	50
3.3	Recrutamento de partículas	50
3.4	Discretização da equação da difusão de calor em regime perma-	
	nente	51
3.5	Discretização da equação da difusão de calor em regime transiente	53
3.5.1	Método de Crank-Nicolson para o SPH	53
3.5.2	Solver TDMA	54
3.5.3	Solver PDMA	56
3.5.4	Compressed Sparse Row (CSR)	56
3.5.5	Solver Gauss-Seidel-S CSR	58

3.5.6	Multigrid	50
3.5.7	Engrossamento	54
3.5.8	Prolongação	54
3.5.9	Solver Gauss-Seidel $\ldots \ldots \ldots$	6
3.6	Erro Numérico	8
3.6.1	Estimativas do erro de discretização <i>a priori</i>	i 8
3.6.2	Estimativas do erro de discretização <i>a posteriori</i>	;9
3.6.3	$Erro de truncamento \dots \dots$;9
3.6.4	Erro de iteração	0
3.6.5	Erro de arredondamento	'0
3.6.6	Erro de poluição numérica	'2
3.6.7	Erro de programação	'3
3.7	Verificação de soluções numéricas	3
3.8	Variáveis globais	4
3.8.1	Temperatura média	'4
3.8.2	Normas de vetores	'4
3.8.3	Normas de matrizes	' 4
3.9	Multiextrapolação de Richardson	5
4	MODELAGEM MATEMÁTICA	0
4.1	Modelos matemáticos	0
4.1.1	Equação da difusão de calor em regime permanente	30
4.1.2	Equação da difusão de calor em regime transiente	30
4.2	Exemplos Numéricos	0
4.2.1	Exemple $(E1-1Dp)$	30
4.2.2	Exemple (E2-1Dp) $\ldots \ldots $	31
4.2.3	Exemplo (E3-1Dp)	31
4.2.4	Exemple (E1-1Dt) \ldots 8	31
4.2.5	Exemple (E1-2Dp)	31
4.2.6	Exemplo (E2-2Dp)	32
4.3	Variáveis de interesse	2
4.4	Condições de contorno	2
4.5	Tipos de geometrias	3
5	RESULTADOS	4
5.1	Erro numérico com SPH	4
5.1.1	Estimativas do erro de discretização <i>a priori</i>	34
5.1.2	Estimativas do erro de discretização <i>a posteriori</i>	34
5.1.3	Erro de truncamento	35
5191	Digentização do operador laplaciono 1D	

6.1	Conclusão
6	CONSIDERAÇÕES FINAIS
	discretização canônica com partícula sensor fixa
5.6.3	Equação da difusão de calor em regime permanente considerando a
5.6.2	Discretização canônica com partícula sensor fixa
	das
5.6.1	Uso do AML em discretização com partículas canonicamente desordena-
5.6	Aplicação
5.5.2	Exemple (E2-2Dp) \ldots 149
5.5.1.3	Verificação quantitativa de soluções numéricas
5.5.1.2	Verificação qualitativa de soluções numéricas para variável global 143
5.5.1.1	Verificação qualitativa de soluções numéricas para variável local 141
5.5.1	Exemple $(E1-2Dp)$
5.5	Equação da difusão de calor em regime permanente 2D 140
5.4.1.2	Verificação qualitativa de soluções numéricas para variável global 136
5.4.1.1	Verificação qualitativa de soluções numéricas para variável local 134
5.4.1	Exemplo (E1-1Dt) $\ldots \ldots 134$
5.4	Equação da difusão de calor em regime transiente 1D 134
5.3.3.2	Verificação qualitativa de soluções numéricas para variável global 131
5.3.3.1	Verificação qualitativa de soluções numéricas para variável local 128
5.3.3	Exemplo (E3-1Dp)
5.3.2.2	Verificação qualitativa de soluções numéricas para variável global 124
5.3.2.1	Verificação qualitativa de soluções numéricas para variável local 123
5.3.2	Exemplo (E2-1Dp)
5.3.1.2	Verificação qualitativa de soluções numéricas para variável global 121
5.3.1.1	Verificação qualitativa de soluções numéricas para variável local 117
5.3.1	Exemplo (E1-1Dp) $\ldots \ldots 117$
5.3	Equação da difusão de calor em regime permanente 1D 117
5.2	Dados gerais das simulações numéricas
5.1.8	Erro de poluição numérica
5.1.7	Erro de suavização
5.1.6.2	Modelos matemáticos 2D $\dots \dots \dots$
5.1.6.1	Modelo matemático 1D $\dots \dots \dots$
5.1.6	Erro de inconsistência de partículas
5.1.5	Erro de arredondamento
5.1.4	Erro de iteração
5.1.3.3	Discretização do operador laplaciano 2D
5.1.3.2	Aproximação assimétrica para derivada primeira 1 D $\ldots\ldots\ldots\ldots$ 90

6.2	Contribuições $\ldots \ldots 159$
6.2.1	Contribuições originais
6.2.2	Aplicações originais
6.3	Know-how
6.4	Trabalhos Futuros
	REFERÊNCIAS 164
	APÊNDICE A – DEDUÇÕES AUXILIARES PARA A DIS- CRETIZAÇÃO DO OPERADOR LAPLACI- ANO
	APÊNDICE B–MULTIEXTRAPOLAÇÃO DE RICHARD- SON GENERALIZADA PARA O MÉTODO SPH
	APÊNDICE C – RELATÓRIO DE SAÍDA DO PROGRAMA E1-2DP 180

1 INTRODUÇÃO

1.1 Caracterização do problema

Estudos envolvendo simulações numéricas destacam-se fortemente no meio acadêmico. Isso ocorre devido a pouca exigência de recursos financeiros e pelo avanço da computação científica. Essas informações contribuem com o desenvolvimento de métodos numéricos que sejam cada vez mais eficazes para discretizar os modelos matemáticos que representam os fenômenos de interesse das engenharias. Nesse meio, é muito comum a utilização de métodos numéricos baseados em uma formulação euleriana, aqueles que necessitam de uma malha computacional para representar o domínio do problema.

Nesse estudo, escolheu-se o método numérico denominado Hidrodinâmica das Partículas Suavizadas (em inglês, *Smoothed Particle Hydrodynamics*, SPH) (GINGOLD; MONAGHAN, 1977; LUCY, 1977). O SPH é baseado na formulação lagrangiana, ou seja, o domínio computacional é representado por um conjunto de partículas independentes entre si, que para casos de escoamento, cada uma delas carrega consigo as propriedades físicas do fluido.

Para desenvolver uma pesquisa consistente, primeiramente optou-se por efetuar verificações de soluções numéricas para alguns modelos matemáticos baseadas em análises de erros. Estas simulações são necessárias para que seja possível identificar singularidades nas aproximações, e ao identificá-las, acredita-se que seja possível resolver problemas de complexidade mais elevada de modo confiável.

Uma das principais motivações é adaptabilidade deste método em geometrias irregulares e de grande complexidade como pode ser visto em alguns trabalhos (MA-RONGIU; LEBOEUF; PARKINSON, 2007; KOUKOUVINIS; ANAGNOSTOPOULOS; PAPANTONIS, 2011; JI *et al.*, 2018) e na FIGURA 1.

1.2 Relevância

Os modelos matemáticos que representam os fenômenos da transferência de calor e da fluidodinâmica são frequentemente estudados por pesquisadores e cientistas. Desde modelos simples como a difusão de calor em regime permanente, até os escoamentos mais complexos modelados pelas equações de Navier-Stokes. Resolver com determinada ordem de acurácia o problema da difusão de calor não é algo complicado, entretanto, pretende-se nesse estudo destacar as soluções com alta ordem de acurácia e consistentes utilizando o método SPH.

Em alguns casos, como aqueles onde utiliza-se o método da projeção, encontrar as soluções numéricas das equações de Navier-Stokes exige a resolução da equação da difusão



FIGURA 1 – FLUXO DE ÓLEO DENTRO DE UM REDUTOR.

FONTE: Adaptada de Ji et al. (2018).

de calor em regime permanente, ou seja, uma equação simples, porém frequentemente estudada. Essa exigência motiva a verificação das soluções numéricas para diferentes modelos matemáticos. Por consequência desses estudos, define-se mecanismos para tornar coerentes as soluções numéricas dos problemas que não possuem soluções analíticas conhecidas.

O interesse em determinar soluções numéricas coerentes utilizando o SPH para os modelos matemáticos, como aqueles que representam a difusão de calor, por exemplo, justifica-se pelo fato de que sua formulação de natureza lagrangiana possa solucionar escoamentos em geometrias mais complexas em menor tempo computacional, pois não há necessidade de geração de malha, tampouco limitações ocasionadas pela própria geometria.

Com o SPH, acredita-se que além de evitar a geração da malha computacional, haja uma facilidade em paralelizar o código computacional, pois as partículas não possuem conectividade entre si (OGER *et al.*, 2016). Encontrando-se soluções acuradas de modo mais eficiente, cria-se novas oportunidades para pesquisadores e cientistas adotarem o SPH nas simulações, como no escoamento supersônico (BERTOLDO; MARCHI, 2017).

Vale ressaltar que, como todo método, o SPH também possui limitações. A baixa acurácia, consistência fraca e o esforço computacional elevado, são característica conhecidas de tal método.

1.3 Objetivos

O objetivo principal do trabalho é reduzir o erro de discretização de soluções numéricas obtidas com o SPH. Para que isto seja possível, primeiramente desenvolve-se uma análise detalhada de erros que são gerados pelas aproximações numéricas (similar aos métodos FDM e FVM) de modelos matemáticos que envolvem fenômenos reais, tais como a transferência de calor, que é tipicamente encontrado em problemas da engenharia.

Para atingir o objetivo principal, os objetivos específicos são:

- 1. Identificar se as fontes de erro numérico já definidas para FDM e FVM permanecem as mesmas com o SPH, e apontar qualquer nova fonte de erro numérico que surja;
- 2. Realizar estimativas de erro *a priori*, encontrando-se a equação geral do erro de discretização para cada modelo matemático abordado;
- 3. Realizar estimativas de erro a posteriori para cada modelo matemático abordado;
- 4. Especificar a dinâmica das partículas a partir do recrutamento no suporte compacto da função núcleo;
- 5. Definir um *solver* genérico que seja capaz de solucionar o sistema de equações algébricas gerado;
- 6. Utilizar o método *multigrid* algébrico para acelerar a convergência das soluções numéricas obtidas com o SPH nos problemas bidimensionais;
- 7. Encontrar soluções numéricas para as variáveis de interesse com alta ordem de acurácia ao aplicar a Multiextrapolação de Richardson.

1.4 Estrutura textual

O texto que compõe esta tese foi elaborado de acordo com a seguinte estrutura:

- Capítulo 2 Revisão da Literatura: são explicitadas as contribuições mais relevantes que permitem identificar a motivação para essa pesquisa;
- Capítulo 3 Fundamentação: são apresentados os métodos numéricos e formulações matemáticas;
- Capítulo 4 Modelagem Matemática: são apresentados os modelos matemáticos, exemplos numéricos, geometrias e condições de contorno;
- Capítulo 5 Resultados: são mostradas todas as contribuições obtidas na pesquisa;

- Capítulo 6 Considereções finais e conclusão: são apontados os principais resultados obtidos na tese.
- Apêndices A, B e C Deduções auxiliares, RRE generalizada para partículas em movimento e relatório de saída do programa computacional E1-2Dp.

2 REVISÃO DA LITERATURA

2.1 Transferência de calor computacional

Segundo Patankar (1988) a Transferência de Calor Computacional (em inglês, *Computational Heat Transfer*, CHT) é uma área que concentra técnicas computacionais com o objetivo de determinar soluções numéricas para variáveis de interesse independentes, tal como, a temperatura na posição de determinado nó de malha, por exemplo. Para determinar a solução numérica de uma variável de interesse, é preciso conhecer o modelo matemático, escolher algum esquema numérico capaz de discretizar tal modelo e resolver o sistema de equações algébricas associado. Ainda segundo o autor, a precisão da solução depende apenas do esquema numérico aplicado; a eficiência é, no entanto, ditada pelo esquema numérico e também pelo algoritmo utilizado para resolver o sistema de equações algébricas. A principal consideração na derivação das equações discretas é a formulação dos termos de convecção e difusão. Uma vez que a convecção é um fenômeno assimétrico, ou seja, as condições a montante têm maior influência do que as a jusante. Dessa forma, é essencial que o esquema de discretização considere isso de alguma forma, caso contrário, podem ocorrer soluções fisicamente irreais. Há três mecanismos conhecidos para transferência de calor: radiação, condução (difusão) e convecção.

Com advento da CHT, os esforços dos pesquisadores e cientistas foram dirigidos à discretização do fluxos que combinam convecção e difusão. As formulações até então desenvolvidas possuiam consequências diretas sobre a precisão da solução para o campo de fluxo, bem como as variáveis escalares, temperatura e concentração, por exemplo. O que se determinavam eram esquemas de discretização instáveis para taxas de fluxo elevadas e a capacidade de obter soluções convergentes para problemas, estavam relacionadas com a formulação convecção-difusão. Se a solução contivesse uma quantidade excessiva de erro numérico, conhecida como falsa difusão, os resultados poderiam não reproduzir o verdadeiro fenômeno físico, tal como o modelo de turbulência. Em muitos casos, era quase impossível refinar suficientemente a malha para que os erros numéricos fossem reduzidos a níveis aceitáveis. Assim, houve a necessidade crucial de uma formulação convecção-difusão que determinasse resultados estáveis e precisos com malha grosseira (PATANKAR, 1988).

Com o desenvolvimento da ciência, a CHT passou a concentrar diversos modelos matemáticos, cada vez mais complexos e tornou-se uma área muito ampla a ponto de considerar outras propriedades físicas, como a transferência de massa, por exemplo. Além disso, a CHT pode estar incorporada em muitos problemas de Dinâmica dos Fluidos Computacional (em inglês, *Computational Fluid Dynamics*, CFD) ou fluidodinâmica, área da computação científica que estuda métodos para as simulações computacionais de fenômenos que envolvem o movimento de fluidos, incluindo ou não a transferência de calor (JALURIA; TORRANCE, 2003; GOLDSTEIN *et al.*, 2005; ASME, 2009; GOLDSTEIN *et al.*, 2010; REDDY; GARTLING, 2010). Com todo esse avanço, hoje pode-se discretizar as equações diferenciais usando métodos numéricos que consideram ou não malha computacional (SOLE-MARI *et al.*, 2019; LI *et al.*, 2021).

2.2 Métodos numéricos para equações diferenciais

Os métodos numéricos que discretizam equações diferenciais podem ser classificados levando em consideração a formulação que os define, sendo eles: métodos com malha, sem malha ou híbridos.

2.2.1 Métodos com malha

Os métodos com malha mais clássicos são: Diferenças Finitas (em inglês, *Finite Difference Method*, FDM) (SMITH, 1985; FERZIGER; PERIĆ, 2002; LEVEQUE, 2007; GALLAGHER *et al.*, 2009), Volumes Finitos (em inglês, *Finite Volume Method*, FVM) (PATANKAR, 1980; LEVEQUE, 2002) e Elementos Finitos (em inglês, *Finite Element Method*, FEM) (ZIENKIEWICZ *et al.*, 1977; CIARLET, 2002). Estes métodos discretizam o domínio contínuo utilizando um número finito de elementos de malha ou volumes de controle, de forma que o domínio computacional seja sua representação discreta. As aproximações geradas por estes métodos criam uma conectividade entre os nós das malhas, e portanto caracterizam a formulação puramente euleriana (LIU; LIU, 2003; ÇENGEL; CIMBALA, 2007; CARMO, 2008).

2.2.2 Métodos sem malha

Por outro lado, há também os métodos sem malha (em inglês, *Meshless Method*, MM) que foram introduzidos no final da década de setenta para resolver problemas em astrofísica, como o método Hidrodinâmica das Partículas Suavizadas (em inglês, *Smoothed Particle Hydrodynamics*, SPH) (GINGOLD; MONAGHAN, 1977; LUCY, 1977), e o método Semi-implícito de Partículas em Movimento (em inglês, *Moving Particle Semi-Implicit*, MPS) (KOSHIZUKA; OKA, 1996), por exemplo. Dentro desta classe de métodos (LIU; LIU, 2003; GRIEBEL; SCHWEITZER, 2003), há um deles que consiste em uma adaptação do FDM; o método de Diferenças Finitas baseado em Função de Base Radial (em inglês, *Radial Basis Function based Finite Difference Method*, RBFFD); aproximações utilizando este método podem ser encontradas em (KADALBAJOO; KUMAR; TRIPATHI, 2015). Nos MM's, o domínio espacial é representado por um conjunto de partículas (pontos materiais), não sendo discretizado por elementos ou volumes, conforme os métodos baseados em malha. Consequentemente, não há necessidade de conectividade predefinida entre as partículas, isto caracteriza a formulação puramente lagrangiana (LIU; LIU, 2003; ÇENGEL; CIMBALA, 2007; CARMO, 2008).

2.2.3 Métodos híbridos

Os métodos híbridos são aqueles que geram aproximações numéricas combinando algum método sem malha juntamente com outro que utiliza malha, como observa-se em (FULK; QUINN, 1995; LIU, 2002; ADAMCZYK *et al.*, 2014).

2.3 Método SPH

O SPH foi desenvolvido concomitantemente na década de setenta por (GINGOLD; MONAGHAN, 1977; LUCY, 1977) para resolver problemas de astrofísica. Obteve maior destaque a partir de sua aplicação em problemas de fraturas, e consequentemente, tem sido aplicado cada vez mais em problemas de engenharia (RANDLES; LIBERSKY, 1996; GRAY; MONAGHAN; SWIFT, 2001; MONAGHAN, 2005; TARTAKOVSKY; MEAKIN, 2005; MAUREL; COMBESCURE, 2008; XU *et al.*, 2010).

Ao se tratar do método SPH, o domínio do problema é discretizado utilizando-se um conjunto de partículas inicialmente distribuídas de modo uniforme (GINGOLD; MO-NAGHAN, 1977; LUCY, 1977), ou desordenado (CHAUSSONNET *et al.*, 2015; SONG; PAZOUKI; PÖSCHEL, 2018). Em Chaussonnet *et al.* (2015) realizou-se uma análise de erros ao aproximar os operadores gradiente e laplaciano com uma discretização denominada canônica, concluindo-se que há grandes impactos ao adotar qualquer distribuição desordenada para os modelos de aproximações dos operadores que foram verificados. Esta discretização canônica permite estabelecer uma configuração inicial para as partículas que pode ser uniforme ou desordenada. Isto acontece gerando-se uma desordem aleatória sobre a posição inicial das partículas dentro de uma região limitada. Existe a possibilidade de adotar uma configuração inicial totalmente caótica, o que caracteriza diversas limitações numéricas e pouco se sabe sobre este caso. Há relatos de distribuições com perturbações menos acentuadas em (MONAGHAN, 2005).

No mesmo ano, realizou-se um estudo sobre uma forma adaptativa de inserir e remover partículas dentro do suporte compacto para aproximações do operador gradiente. Estas aproximações foram realizadas utilizando-se distribuições desordenadas e verificou-se que o método adaptativo foi eficiente tornando-se mais consistente as formas discretas de se aproximar o operador gradiente (LASTIWKA; QUINLAN; BASA, 2005).

As aproximações SPH são realizadas localmente por meio de um subconjunto de partículas pertencentes a um suporte compacto associado a cada uma das partículas que discretizam o domínio. O suporte compacto é uma das propriedades das funções núcleo que são utilizadas no método SPH. Para que uma função seja um núcleo SPH ela precisa satisfazer outras propriedades, além de possuir suporte compacto.

Ao longo do tempo diversas funções núcleo surgiram com o objetivo de aumentar a consistência das aproximações e garantir a ordem de acurácia desejada (GINGOLD;

MONAGHAN, 1977; LUCY, 1977; MONAGHAN; LATTANZIO, 1985; MORRIS, 1996; LIU; LIU; LAM, 2003; YANG; PENG; LIU, 2014).

Com o desenvolvimento da pesquisa cada vez mais modelos matemáticos passaram a ser discretizados com o SPH. Em Jeong *et al.* (2003) realizou-se uma aplicação do método SPH para solucionar um modelo de equação do calor, e com isso efetuou-se uma comparação com o método de diferenças finitas, concluindo-se que os resultados foram satisfatórios e acreditando-se no desenvolvimento do método por meio de pesquisa acadêmica. Uma análise de erro foi realizada em (FATEHI; MANZARI, 2011), onde mostrou-se um novo esquema de aproximação de segunda ordem para a derivada segunda. Para a análise, aplicou-se este esquema de segunda ordem de acurácia na discretização da equação da difusão de calor em regime transiente. A formulação proposta envolve cálculos de novos somatórios e matrizes inversas, além de exigir que $h > \Delta x$. Este procedimento contribui muito para o aumento do tempo de processamento e possui acurácia fraca.

Algo semelhante à distribuição de partículas desordenadas foi realizado em (PURI; RAMACHANDRAN, 2014) para a equação de Euler, enfatizando-se a necessidade de estudos que considerem a distribuição desordenada e caótica das partículas, que ainda não possuem estudos aprimorados. A aplicação do método de partículas para escoamentos de fluidos compressíveis também é um desafio para pesquisadores e cientistas. Recentemente em Gissler *et al.* (2017), comparou-se diferentes simulações envolvendo única e também multifases, com ou sem força de arrasto. Além do escoamento multifásico com superfície livre, este trabalho abordou a deformação das partículas, algo importante para representação do fenômeno físico estudado.

Formulações SPH para escoamento turbulento estão em ascensão (VIOLEAU; ISSA, 2007; VIOLEAU, 2012; PRICE, 2012; HU; ADAMS, 2015; GHAZANFARIAN; SAGHATCHI; GORJI-BANDPY, 2016; KAZEMI *et al.*, 2017). Estes estudos contribuem para o avanço das pesquisas envolvendo escoamentos de grande complexidade.

Outro estudo envolvendo SPH foi desenvolvido para a determinação e precisão do coeficiente de arrasto em um cilindro circular (KORZANI *et al.*, 2017), avaliando-se diferentes valores de Reynolds e número de partículas.

Uma das principais motivações para o uso do SPH origina-se do artigo (LI; LUO; FAN, 2018), onde destaca-se o primeiro esforço para investigar os efeitos de partículas de inércia sobre um escoamento turbulento e transferência de calor em uma camada limite turbulenta em evolução espacial por meio de Simulação Numérica Direta (em inglês, *Direct Numerical Simulation*, DNS). Estas novas descobertas são de grande importância para melhorar a compreensão dos mecanismos físicos e transferência de calor turbulenta por partículas.

Song, Pazouki e Pöschel (2018) efetuaram uma análise sobre a influência que a

configuração inicial das partículas tem sobre a estabilidade em um modelo de escoamento de Poiseuille, identificando-se que há garantias de estabilidade para casos onde $h/\Delta x \ge 1.5$.

Em Achim, Rozas e Toledo (2021) é possível observar um método semi-desacoplado que melhora a consistência do SPH original por meio de uma correção nas aproximações do núcleo. Realizou-se alguns testes envolvendo campos vetoriais e chegou-se à conclusão que o método é computacionalmente mais vantajoso que as aproximações já conhecidas na literatura, como por exemplo: Método de Partículas Suavizadas Corretivas (em inglês, *Corrective Smoothed Particle Method*, CSPM) e Método de Partículas Finitas (em inglês, *Finite Particle Method*, FPM) (LIU; LIU, 2006; LIU; LIU, 2010; LIND; STANSBY, 2016; HUANG et al., 2018).

"Após a revisão bibliográfica, o autor não identificou estudos detalhados que mostrem os efeitos do erro de discretização sobre as aproximações SPH, tampouco encontrou-se um esquema numérico usando SPH, que fosse capaz de reduzir significativamente o erro de discretização dessas aproximações".

2.4 Método multigrid

A redução do erro de discretização geralmente é associada ao tamanho da malha, ou seja, quanto mais refinada, menor será o erro de discretização. Entretanto, malhas muito refinadas, geram sistemas de equações lineares ou não lineares de ordem elevada, exigindo maior esforço computacional para obter a solução numérica. Com isso, para amenizar o tempo computacional gerado pelo refinamento das malhas, adota-se o método *multigrid*, que acelera a convergência das soluções numéricas.

Alguns *solvers* iterativos clássicos suavizam rapidamente os modos oscilatórios dos erros na malha fina deixando apenas modos suaves, onde o método perde a efetividade. Em seguida, deve-se transferir as informações do problema para malhas mais grossas (processo denominado restrição), onde os modos suaves tornam-se mais oscilatórios, e portanto, o método volta a ser mais efetivo. Ao se atingir a malha mais grossa possível ou a mais grossa desejada, deve-se retornar com as informações do problema para a malha mais fina original (processo denominado prolongação). Esse processo acelera a convergência da solução numérica.

Dependendo do tipo de informação que se transfere entre as malhas, pode-se ter o Esquema de Correção (em inglês, *Correction Scheme*, CS), onde transfere-se apenas o resíduo, e o Esquema de Aproximação Total (em inglês, *Full Approximation Scheme*, FAS), onde transfere-se o resíduo e a solução (TROTTENBERG; OOSTERLEE; SCHULLER, 2000; WESSELING, 2004). A forma com que se percorre as diversas malhas é chamada de ciclo. Existem diversos ciclos, entre eles, os ciclos $V, W \in F$. O ciclo V possui um baixo custo computacional (TROTTENBERG; OOSTERLEE; SCHULLER, 2000; WESSELING,
2004), definido como $V(\nu_1,\nu_2)$, onde $\nu_1 \in \nu_2$ são os números de pré e pós-suavização no ciclo *multigrid*, ou seja, o número de suavizações na restrição e prolongação, respectivamente. A razão da distância entre os pontos consecutivos da malha fina (h) e da malha grossa (H), chama-se razão de engrossamento (re). É comum estudos onde re = 2, ou seja, H = 2h (TROTTENBERG; OOSTERLEE; SCHULLER, 2000; WESSELING, 2004).

A robustez do *multigrid* é garantida por suas duas formulações: o *multigrid* geométrico, desenvolvido para malhas estruturadas e o *multigrid* algébrico (em inglês, *Algebraic Multigrid*, AMG)(STÜBEN, 1983; RUGE; STÜBEN, 1987), para malhas estruturadas e não estruturadas (CHANG; WONG; FU, 1996; WATANABE; IGARASHI; HONMA, 2005; WU; ELMAN, 2006; XIAO *et al.*, 2010).

O *multigrid* geométrico é um método que resolve apenas problemas que envolvam malhas estruturadas. Ele é um caso particular do algébrico (BRANDT, 1986). Este por sua vez é baseado diretamente na característica da matriz de coeficientes, não sendo necessário grandes preocupações com condições de contornos, descontinuidades, anisotropias geométricas. A construção da matriz nos níveis com menor número de elementos trata-se de um problema puramente algébrico, tornando-se adaptável aos casos de grande complexidade geométrica.

Suero (2010) caracterizou os dois métodos *multigrid* destacando algumas informações relevantes: o *multigrid* geométrico resolve problemas contínuos, enquanto o algébrico sistemas lineares de equações algébricas; o geométrico usa a estrutura geométrica do problema, enquanto o algébrico apenas os elementos da matriz; o algoritmo do geométrico é específico para cada problema, no algébrico ele é único; a eficiência do geométrico é muito boa, por outro lado, do algébrico é considerada boa.

"A aplicação do *multigrid* algébrico para resolver os sistemas lineares de equações algébricas geradas pela discretização do modelo matemático usando SPH é desconhecida". Entretanto, pode-se encontrar estudos envolvendo a equação de Poisson, que foi resolvida com FEM e *multigrid* algébrico em malhas não estruturadas (BRAESS, 1995), destacando ganhos em número de iterações e tempo de processamento (*CPU time*). Há também os casos de malhas estruturadas (CHANG; WONG; FU, 1996), apontando tempo de processamento do ciclo V, da fase *setup* (preparação dos elementos da matriz) e de uma nova interpolação que permitiu melhorar a taxa de convergência.

2.5 Multiextrapolação de Richardson

A Multiextrapolação de Richardson (em inglês, *Repeated Richardson Extrapolation*, RRE) (ROACHE; KNUPP, 1993; MARCHI *et al.*, 2013a; MARCHI *et al.*, 2013b; MARCHI; GERMER, 2013; MARCHI; GIACOMINI; SANTIAGO, 2016) é uma técnica de pósprocessamento utilizada para diminuir o erro de discretização de soluções numéricas sem a necessidade de aumentar o número de termos na equação discreta (FDM, FVM, etc;) obtida por meio da série de Taylor. A RRE combina soluções de diferentes malhas e a cada extrapolação a solução obtida apresenta um erro de discretização menor. Essa grandeza obedece às ordens verdadeiras (MARCHI, 2001) da equação discreta usada para aproximar as derivadas da equação diferencial.

Sendo uma técnica de pós-processamento, não há impedimentos de sua aplicação em problemas com geometrias complexas, anisotropias geométricas, malhas não estruturadas e até mesmo nos casos de formulação lagrangiana, onde não há malha.

Marchi e Germer (2013) analisaram a eficiência de RRE na redução do erro de discretização para algumas soluções numéricas obtidas com diferentes malhas. Resolveu-se o modelo matemático da advecção-difusão 1D com esquemas de aproximação com ordens um, dois e três de acurácia. Os autores concluíram que RRE é eficiente na redução do erro de discretização, atingindo ordem de acurácia superior a dezoito. Identificou-se também que o Esquema de Diferenças Centrais de segunda ordem de acurácia (em inglês, *Central Difference Scheme of second order*, CDS-2) apresentou o melhor resultado com RRE.

Marchi *et al.* (2013b) aplicaram RRE para reduzir e estimar o erro de discretização de soluções numéricas em um modelo matemático que representa a condução de calor. Este modelo é a equação de Laplace 2D, que foi discretizada com FDM utilizando esquema de segunda ordem CDS-2. Além disso, utilizou-se malhas uniformes e *multigrid* geométrico para acelerar a convergência destas soluções numéricas. Os autores identificaram que RRE é capaz de reduzir significativamente o erro de discretização. Com isso, foi possível encontrar soluções com alta ordem de acurácia, evidenciando-se o menor tempo de CPU e baixa utilização de Memória de Acesso aleatório (em inglês, *Random-Access Memory*, RAM).

Em AbdelMigid *et al.* (2017), apresentou-se uma revisão de literatura, abordandose o problema da cavidade. Além disso, analisou-se algumas soluções das equações de Navier-Stokes para fluidos incompressíveis variando o número de Reynolds entre 100 e 5000. O método numérico aplicado foi FDM, com malhas uniformes e RRE. Os autores reduziram o erro de discretização e conseguiram aumentar a ordem de acurácia significativamente, passando de segunda para sexta ordem.

No mais recente estudo analisado pelos autores (SILVA *et al.*, 2020), aplicou-se a Multiextrapolação de Richardson Completa (em inglês, *Completed Repeated Richardson Extrapolation*, CRRE) para reduzir o erro de discretização espacial de soluções numéricas para um escoamento compressível. A CRRE é uma técnica que realiza a multiextrapolação em todo o campo de soluções e não sobre uma variável de interesse apenas. Os autores reduziram com CRRE sete ordens de magnitude do erro de discretização espacial.

2.6 Verificação de soluções numéricas

A verificação de soluções numéricas segue testes de coerência que permitem identificar possíveis falhas nas aproximações. Marchi (2001) abordou em sua tese diversos testes que comprovam se uma solução numérica é confiável.

Ao se tratar do FDM, por exemplo, é possível discretizar um modelo matemático levando em consideração diferentes formas. Uma aproximação pode ser realizada por Esquemas de Diferenças Centrais (em inglês, *Central Difference Scheme of nth order*, CDSn), ou descentrados, como os Esquemas de Diferenças a Montante (em inglês, *Downstream Difference Scheme of nth order*, DDS-n) e os Esquemas de Diferenças a Jusante (em inglês, *Upwind Difference Scheme of nth order*, UDS-n) (FERZIGER; PERIĆ, 2002; PLETCHER; TANNEHILL; ANDERSON JUNIOR, 2012).

Estes esquemas são construídos baseados nas aproximações por série de Taylor, onde gera-se para cada um deles uma expressão conhecida como erro de discretização (erro de truncamento local livre de outros erros), para o caso em que a equação diferencial possui apenas coeficientes constantes.

Os erros podem ser diferenciados por suas fontes (GINGOLD; MONAGHAN, 1977; LUCY, 1977; ROACHE, 1998; FERZIGER; PERIĆ, 2002; LIU; LIU, 2003; LIU; LIU, 2006; VIOLEAU, 2012; PLETCHER; TANNEHILL; ANDERSON JUNIOR, 2012; ANTUONO *et al.*, 2014; CHAUSSONNET *et al.*, 2015), sendo eles:

- 1. Erro de truncamento;
- 2. Erro de iteração;
- 3. Erro de arredondamento;
- 4. Erro de programação;
- 5. Erro de desordenação ou perturbação das partículas.

Estudos sobre a verificação de erros utilizando especificamente o método SPH ainda não possuem definições bem desenvolvidas, como nos itens anteriores (1 a 4). Em particular, o item 5 é apresentado em (CHAUSSONNET *et al.*, 2015) por meio de análises de uma medida de erro numérico. Acredita-se que uma análise mais detalhada defina novos testes de coerência para as soluções numéricas obtidas com métodos de partículas.

Diversos estudos mostrando a ordem de acurácia de soluções numéricas utilizando o método SPH já foram desenvolvidos. Entretanto, para que uma determinada ordem de acurácia seja obtida, tem-se que garantir a consistência das aproximações pelo núcleo. As técnicas mais conhecidas de restauração de consistências são: CSPM e FPM (LIU; LIU, 2006; LIU; LIU, 2010; LIND; STANSBY, 2016; HUANG *et al.*, 2018). Algumas observações sobre estes dois métodos de restauração podem receber destaque. No CSPM, as aproximações de uma função e suas derivadas são sequencialmente calculadas, embora as mesmas ordens de derivada sejam resolvidas simultaneamente. Enquanto isto, no FPM, uma função e todas as suas derivadas são simultaneamente calculadas.

No CSPM, derivadas de ordem mais baixas são consideradas essenciais nas expressões para aproximar ordens superiores. Já no FPM, exceto para os valores da função, derivadas de ordem não inferiores são as utilizadas para aproximar as derivadas de ordem superior. Por consequência, os possíveis erros numéricos existentes nas derivadas de ordem inferior não serão acumulados nas derivadas de ordem superior.

O CSPM tem consistência C^1 para o núcleo nas regiões internas e C^0 nas regiões de fronteira. O FPM tem consistência C^1 para o núcleo em ambas as regiões se apenas a derivada de primeira ordem for negligenciada na expansão em série de Taylor. Exceto para partículas uniformemente distribuídas e internas, o CSPM tem consistência C^0 . O FPM tem consistência C^1 sempre, desde que apenas a derivada de primeira ordem seja negligenciada na expansão em série de Taylor. A consistência é estudada como a capacidade do método em resolver um problema com solução polinomial, de forma a recuperar a j-ésima derivada do polinômio de i-ésima ordem e rotulado de C_{ξ}^{γ} (LIU; LIU, 2006; CHAUSSONNET *et al.*, 2015).

Muitos pesquisadores e cientistas mostraram técnicas para se obter soluções com alta ordem de acurácia, entretanto, nenhuma delas possui uma análise detalhada de convergência, tampouco possuem a eficiência de RRE. Sabe-se que são técnicas nada robustas e que contribuem para o aumento significativo do tempo de processamento, pois em alguns casos, exige inversão de matriz (LIU; LIU, 2006; OGER *et al.*, 2007; FATEHI; MANZARI, 2011; HUANG *et al.*, 2018; CHEN; HUANG; SLOAN, 2019), aumento de elementos na matriz de coeficientes (DEHNEN; ALY, 2012), somatórios duplos (FATEHI; FAYAZBAKHSH; MANZARI, 2008), ou ainda expressões adicionais que demandam a solução de sistemas lineares de equações algébricas (STRANEX; WHEATON, 2011; FRANCOMANO; PALIAGA, 2018).

3 FUNDAMENTAÇÃO

Neste capítulo encontram-se todas as informações necessárias para que sejam determinadas as soluções numéricas desejadas. Primeiramente apresenta-se a formulação do modelo numérico utilizado para discretizar os modelos matemáticos. Na sequência os *solvers* utilizados para resolver os sistemas lineares de equações algébricas dos casos unidimensionais, *solver* e método de aceleração de convergência dos casos bidimensionais. Por fim as técnicas de análises de erros e pós-processamento para diminuição do erro de discretização e aumento das ordens de acurácia.

3.1 Modelo numérico

3.1.1 Formulação SPH

Nesse capítulo apresenta-se a formulação do método SPH em duas etapas. A primeira trata da representação integral de uma função arbitrária, e para isso, utiliza-se conceitos de convolução entre uma função genérica e a distribuição Delta de Dirac. A segunda diz respeito à aproximação por partículas. Por fim, definem-se as características necessárias para que uma função seja reconhecida como um núcleo SPH.

3.1.2 Representação integral de uma função

Antes de abordar a formulação do método SPH, é importante compreender a representação matemática da partícula, que pode ser escrita como $\mathbf{x} = \mathbf{x}(\mathbf{Q},t)$, onde \mathbf{Q} é o vetor coordenadas (x,y,z) da partícula no tempo t no domínio tridimensional, por exemplo. O vetor \mathbf{Q} pode representar diferentes dimensões de domínios, como 1D e 2D. Além disso, a partícula fixa pode não depender da variável tempo, sendo reescrita na forma $\mathbf{x} = \mathbf{x}(\mathbf{Q})$. Em casos mais gerais, como a simulação mostrada na FIGURA 1, além da posição, a partícula pode carregar informações sobre as propriedades físicas do fluido. A partir desse consceito, a representação integral da função é aproximada pela soma ponderada dos valores das partículas vizinhas mais próximas (LIU; LIU, 2003). Assim, a ideia de discretização então considera um conjunto de partículas para a representação do meio. Para viabilizar isto, é necessário uma aproximação para a distribuição Delta de Dirac, que denomina-se núcleo. O conceito de representação integral de uma função $\psi(\mathbf{x})$ utilizado no método SPH começa a partir da identidade

$$\psi(\mathbf{x}) = \int_{\Omega} \psi(\mathbf{x}') \delta(\mathbf{x} - \mathbf{x}') d\mathbf{x}', \qquad (3.1)$$

onde $\psi : \Omega \longrightarrow \mathbb{R}^m$ definida num domínio aberto $\Omega \subset \mathbb{R}^n$ tal que $\mathbf{x} \in \Omega$, e ψ é contínua em Ω . Essa representação é feita através da convolução de ψ com a distribuição Delta de Dirac, dada por

$$\delta(\mathbf{x} - \mathbf{x}') = \begin{cases} \infty, & \mathbf{x} = \mathbf{x}' \\ 0, & \mathbf{x} \neq \mathbf{x}' \end{cases}.$$

Substituindo-se a distribuição Delta de Dirac na equação (3.1) por uma função núcleo $W(\mathbf{x}-\mathbf{x}';h)$ centrada nas coordenadas da partícula \mathbf{x} e dependente de um parâmetro h, obtém-se

$$\psi(\mathbf{x}) \cong \int_{\Omega} \psi(\mathbf{x}') W(\mathbf{x} - \mathbf{x}'; h) d\mathbf{x}', \qquad (3.2)$$

sendo $W : \mathbb{R}^n \longrightarrow \mathbb{R} \in h$ o comprimento de suavização que multiplicado por um fator de escala k > 0, estabelece a maior distância permitida entre as partículas $\mathbf{x} \in \mathbf{x}'$ e define o raio da base radial da função núcleo (kh).

A notação $\langle \rangle$ é convenientemente adotada para representar a aproximação para a função. Dessa forma, a equação (3.3) representa a aproximação para a função $\psi(\mathbf{x})$,

$$\langle \psi(\mathbf{x}) \rangle = \int_{\Omega} \psi(\mathbf{x}') W(\mathbf{x} - \mathbf{x}'; h) d\mathbf{x}'.$$
 (3.3)

Sendo assim, adotando-se tal notação, define-se uma aproximação para $\psi(\mathbf{x})$ por meio da representação integral da função, e para isso utiliza-se como base uma função núcleo W suficientemente suave. As propriedades de uma função núcleo W são definidas Seção 3.1.4.

3.1.3 Aproximação utilizando partículas

Assumindo-se que o meio é discreto, ou seja, formado por um número finito de partículas que possuem massa e ocupam um volume no espaço, a integral da equação (3.3) aplicada em um ponto do domínio pode ser dada por meio de um somatório sobre as partículas que se encontram recrutadas pela função núcleo. Dessa forma, considera-se na equação (3.4), o "suporte compacto" da função núcleo como o conjunto V_i centrado em \mathbf{x}_i , onde

$$V_i = \{\mathbf{x}_j, \|\mathbf{x}_i - \mathbf{x}_j\| \le kh\},\tag{3.4}$$

sendo $V_i \subset \Omega$, k > 0 um fator de escala e $j = 1 \dots N_i$, onde N_i representa o número de partículas que satisfazem a restrição $||\mathbf{x}_i - \mathbf{x}_j|| \le kh$.

Considerando-se um meio contínuo de partículas, ao substituir o volume infinitesimal $d\mathbf{x}'$ da equação (3.3) na posição da j-ésima partícula pelo volume Δv_j ocupado pela partícula \mathbf{x}_j , obtém-se a equação (3.5),

$$\langle \psi(\mathbf{x}_i) \rangle = \sum_{j=1}^{N_i} \psi(\mathbf{x}_j) W(\mathbf{x}_i - \mathbf{x}_j; h) \Delta v_j.$$
 (3.5)

Sabendo-se que o volume da partícula Δv_j pode ser representado pela razão entre a massa m_j e massa específica ρ_j , a equação (3.5) pode ser reescrita com tal substituição, obtendo-se

$$\langle \psi(\mathbf{x}_i) \rangle = \sum_{j=1}^{N_i} \psi(\mathbf{x}_j) W(\mathbf{x}_i - \mathbf{x}_j; h) \frac{m_j}{\rho_j}.$$
(3.6)

Dessa forma, a equação (3.6) determina uma aproximação para $\psi(\mathbf{x})$ em cada ponto do domínio discretizado pelas partículas.

3.1.4 Propriedades da função núcleo

Apresenta-se na literatura uma diversidade de funções núcleo SPH, entretanto é de suma importância conhecer as propriedades fundamentais para que uma função possa ser reconhecida como um núcleo SPH, ou ainda, as especificidades para se construir tal função, a qual é uma aproximação para a função Delta de Dirac. As principais propriedades são apresentadas abaixo (LIU; LIU, 2003; PAIVA *et al.*, 2009; VIOLEAU, 2012).

1. O núcleo SPH deve ser suficientemente suave

$$W(\mathbf{x} - \mathbf{x}', h) \in C^{\gamma}, \ \gamma > 1.$$

A finalidade dessa propriedade é melhorar a aproximação da função e de suas derivadas. Sendo contínua, a função núcleo garante bons resultados, e segundo (MONAGHAN, 1992; FULK, 1994) não será sensível à distribuição desordenada das partículas; os erros gerados na aproximação serão pequenos, desde que a perturbação imposta às partículas não sejam muito severas.

2. O núcleo SPH deve ser normalizado

$$\int_{\Omega} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' = 1.$$

Essa propriedade garante um resultado unitário ao integrar a função núcleo sob o domínio do suporte compacto. Propriedade herdada da distribuição Delta de Dirac.

3. O núcleo SPH deve ter suporte compacto

$$W(\mathbf{x} - \mathbf{x}', kh) = 0$$
 quando $\|\mathbf{x} - \mathbf{x}'\| > kh$,

sendo k > 0 um fator de escala. Essa propriedade transforma a aproximação de f, tida como uma operação global em uma local. É possível que para algumas funções essa propriedade não seja cumprida. Na prática, o que acontece nesse caso é que

$$W(\mathbf{x} - \mathbf{x}', kh) = \epsilon$$
 quando $\|\mathbf{x} - \mathbf{x}'\| > kh$,

onde ϵ é suficientemente pequeno. Um exemplo disso é a função gaussiana definida na Seção 3.1.6. Além disso, conforme mostrado em (PAIVA *et al.*, 2009), essa propriedade exerce influência direta representação integral da derivada de uma função, ou seja,

$$\nabla \cdot \psi(\mathbf{x}) = \int_{\Omega} \nabla \cdot [\psi(\mathbf{x}')W(\mathbf{x} - \mathbf{x}'; h)]d\mathbf{x}' , \qquad (3.7)$$
$$- \int_{\Omega} \psi(\mathbf{x}') \cdot \nabla_{\mathbf{x}'}W(\mathbf{x} - \mathbf{x}'; h)d\mathbf{x}'$$

resultando em

$$\nabla \cdot \psi(\mathbf{x}) = -\int_{\Omega} \psi(\mathbf{x}') \cdot \nabla_{\mathbf{x}'} W(\mathbf{x} - \mathbf{x}'; h) d\mathbf{x}' . \qquad (3.8)$$

4. O núcleo SPH deve ser **positivo**

$$W(\mathbf{x} - \mathbf{x}', h) \ge 0.$$

A propriedade mostra que a função núcleo deve ser não negativa sob influência da região limitada pelo suporte compacto. Isso não é necessário para a convergência, mas de suma importância para aproximações coerentes com as propriedades físicas envolvidas no problema. Sendo não negativa, a função evita, por exemplo, valores negativos para a densidade (PAIVA *et al.*, 2009).

5. O núcleo SPH deve ser decrescente

$$W(\mathbf{x}_i - \mathbf{x}_1, h) < W(\mathbf{x}_i - \mathbf{x}_2, h), \text{ se } \|\mathbf{x}_i - \mathbf{x}_1\| > \|\mathbf{x}_i - \mathbf{x}_2\|.$$

Essa propriedade garante que as partículas \mathbf{x}_j mais próximas da partícula \mathbf{x}_i exercem maior influência do que as partículas mais afastadas.

6. O núcleo SPH deve ser simétrico radial

$$W(\mathbf{x} - \mathbf{x}', h) = W(\|\mathbf{x} - \mathbf{x}'\|, h).$$

Essa propriedade determina que partículas \mathbf{x}' em diferentes posições e com mesma distância da partícula \mathbf{x} , tenham a mesma influência.

7. O núcleo SPH deve satisfazer a distribuição Delta de Dirac quando o parâmetro $h \longrightarrow 0$, ou seja:

$$\lim_{h \to 0} W(\mathbf{x} - \mathbf{x}', h) = \delta(\mathbf{x} - \mathbf{x}').$$

Por fim, essa propriedade determina que, à medida que o comprimento de suavização tende a zero, o valor aproximado se torna cada vez mais próximo do valor exato da função, ou seja, $\langle f(\mathbf{x}) \rangle \rightarrow f(\mathbf{x})$.

3.1.5 Função núcleo SPH: definição e modelos

O núcleo SPH é de extrema importância, pois não somente determina o padrão da aproximação, como também define a abrangência do suporte das partículas, determina a consistência e a precisão de ambas as aproximações: integral e por partículas.

Adota-se como modelo de núcleo SPH a função definida da seguinte forma:

$$W_{w^*}^0(\phi) = \frac{\alpha_{w^*,d}}{h^d} w_{w^*}^0(\phi),$$
$$\phi_{ij} = \frac{r_{ij}}{h} = \frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{h}.$$

onde

A variável r representa a distância entre duas partículas, h o comprimento de suavização (fixo), w é uma função definida sobre \mathbb{R}^+ , suficientemente suave e $\alpha_{w^*,d}$ uma constante de normalização específica para cada núcleo ($w^* = a, b, \ldots, f$) e dimensão (d = 1, 2, 3) (VIOLEAU, 2012).

Para d = 1, a derivada primeira da função núcleo é expressa da seguinte forma:

$$W_{w^*}^1(\phi) = \frac{\alpha_{w^*,1}}{h^2} w_{w^*}^1(\phi).$$

Para o caso bidimensional (d = 2) ou tridimensional (d = 3) procede-se da mesma forma.

De maneira análoga ao caso anterior, as derivadas de ordem superior podem ser obtidas por meio da seguinte relação:

$$W_{w^*}^{\gamma}(\phi) = \frac{\partial^{\gamma} W}{\partial \phi^{\gamma}} = \frac{\alpha_{w^*,d}}{h^{d+\gamma}} w_{w^*}^{\gamma}(\phi),$$

onde $\gamma = 0,1,2$ representa a ordem da derivada.

O gradiente da função núcleo $\nabla W(\mathbf{x}_i - \mathbf{x}_j, h)$ em termos da variável ϕ é definido na equação (3.9) como $\nabla W(\phi)$, ou seja,

$$\nabla W(\phi_{ij}) = \frac{\alpha_{w^*,d}}{h^{d+1}} w^1(\phi_{ij}) \frac{\mathbf{x}_i - \mathbf{x}_j}{r_{ij}}.$$
(3.9)

3.1.6 Modelos clássicos de funções núcleo SPH e suas derivadas $(W_{w^*}^{\gamma}(\phi))$

Os núcleos SPH são denotados por $W_{w^*}^{\gamma}(\phi)$, onde $w^* = a, b, \ldots, f$ representa o modelo da função e γ a ordem das derivadas como já mencionado anteriormente.

Diferentes núcleos SPH são encontrados na literatura. Em Lucy (1977) apresentouse a função núcleo definida na equação (3.10) e suas derivadas, vide FIGURA 2.

$$W_a^0(\phi) = \frac{\alpha_{a,d}}{h^d} \begin{cases} (1+3\phi)(1-\phi)^3 & 0 \le \phi < 1\\ 0 & \phi \ge 1 \end{cases},$$
(3.10)

com as constantes de normalização definidas por $\alpha_{a,1} = 5/4$, $\alpha_{a,2} = 5/\pi$ e $\alpha_{a,3} = 105/16$, em uma, duas e três dimensões, respectivamente. A primeira e segunda derivadas são

$$W_a^1(\phi) = \frac{\alpha_{a,d}}{h^{d+1}} \begin{cases} -12\phi(\phi-1)^2 & 0 \le \phi < 1\\ 0 & \phi \ge 1 \end{cases}$$
$$W_a^2(\phi) = \frac{\alpha_{a,d}}{h^{d+2}} \begin{cases} -36\phi^2 + 48\phi - 12 & 0 \le \phi < 1\\ 0 & \phi \ge 1 \end{cases}$$

FIGURA 2 – A FUNÇÃO NÚCLEO (3.10) E SUAS DUAS PRIMEIRAS DERIVADAS.



FONTE: O autor (2022).

Monaghan (1992) também apresentou uma função, a chamada gaussiana, definida na equação (3.11) com suas derivadas, e mostradas na FIGURA 3.

$$W_b^0(\phi) = \frac{\alpha_{b,d}}{h^d} e^{-\phi^2}.$$
 (3.11)

As constantes de normalização são dadas por $\alpha_{b,1} = 1/\pi^{1/2}$, $\alpha_{b,2} = 1/\pi$ e $\alpha_{b,3} = 1/\pi^{3/2}$, e a primeira e segunda derivadas são, respectivamente,

$$W_b^1(\phi) = \frac{\alpha_{b,d}}{h^{d+1}} \left(-2\phi e^{-\phi^2}\right)$$

е

е

$$W_b^2(\phi) = \frac{\alpha_{b,d}}{h^{d+2}} \left(-2e^{-\phi^2} + 4\phi^2 e^{-\phi^2} \right)$$

Nota-se que a função gaussiana não tem suporte compacto, mas se aproxima rapidamente de zero à medida que a distância entre duas partículas é maior. Conforme relatado em Liu e Liu (2003), essa função costuma apresentar bons resultados quando há um número pequeno de partículas no domínio de influência, o que caracteriza uma maior distância entre elas.

FIGURA 3 – A FUNÇÃO GAUSSIANA (3.11) E SUAS DUAS PRIMEIRAS DERIVADAS.



FONTE: O autor (2022).

Outros núcleos muito importantes para o SPH são as *splines*, uma distinta classe de funções. Esta classe é bastante influente e usualmente aplicada. A primeira classe apresentada, é dada pela função *spline* cúbica (MONAGHAN; LATTANZIO, 1985), definida com suas derivadas na equação (3.12) e apresentada na FIGURA 4.

$$W_c^0(\phi) = \frac{\alpha_{c,d}}{h^d} \begin{cases} 1 - \frac{3}{2}\phi^2 + \frac{3}{4}\phi^3 & 0 \le \phi < 1\\ \frac{1}{4}(2 - \phi)^3 & 1 \le \phi < 2\\ 0 & \phi \ge 2 \end{cases}$$
(3.12)

As constantes de normalização são $\alpha_{c,1} = 2/3$, $\alpha_{c,2} = 10/7\pi$ e $\alpha_{c,3} = 1/\pi$, sendo suas

primeira e segunda derivadas, respectivamente,

$$\begin{split} W_c^1(\phi) &= \frac{\alpha_{c,d}}{h^{d+1}} \begin{cases} -3\phi + \frac{9}{4}\phi^2 & 0 \le \phi < 1\\ -\frac{3}{4}(2-\phi)^2 & 1 \le \phi < 2\\ 0 & \phi \ge 2 \end{cases} \\ W_c^2(\phi) &= \frac{\alpha_{c,d}}{h^{d+2}} \begin{cases} -3 + \frac{9}{2}\phi & 0 \le \phi < 1\\ \frac{3}{2}(2-\phi) & 1 \le \phi < 2\\ 0 & \phi \ge 2 \end{cases} \end{split}$$

A *spline* cúbica é usualmente apresentada na literatura SPH por conter características muito semelhantes à gaussiana e possuir suporte compacto. Entretanto, a segunda derivada é uma função linear por partes, que pode resultar em uma desvantagem em relação aos núcleos suficientemente suaves.

FIGURA 4 – A FUNÇÃO SPLINE CÚBICA (3.12) E SUAS DUAS PRIMEIRAS DERIVADAS.



Uma pequena alteração proposta por Liu, Liu e Lam (2003) no núcleo (3.10), assegurou resultados mais precisos. A função modificada agora possui uma ordem maior, passando de terceira para quarta ordem, conforme equação (3.13). Sua representação pode ser observada na FIGURA 5.

$$W_d^0(\phi) = \frac{\alpha_{d,d}}{h^d} \begin{cases} \frac{2}{3} - \frac{9}{8}\phi^2 + \frac{19}{24}\phi^3 - \frac{5}{32}\phi^4 & 0 \le \phi < 2\\ 0 & \phi \ge 2 \end{cases}$$
(3.13)

е

e as constantes de normalização são dadas por $\alpha_{d,1} = 1$, $\alpha_{d,2} = 15/7\pi$ e $\alpha_{d,3} = 315/208\pi$, sendo suas primeira e segunda derivadas dadas por:

$$W_d^1(\phi) = \frac{\alpha_{d,d}}{h^{d+1}} \begin{cases} -\frac{9}{4}\phi + \frac{19}{8}\phi^2 - \frac{5}{8}\phi^3 & 0 \le \phi < 2\\ 0 & \phi \ge 2 \end{cases}$$

е

$$W_d^2(\phi) = \frac{\alpha_{d,d}}{h^{d+2}} \begin{cases} -\frac{9}{4} + \frac{19}{4}\phi - \frac{15}{8}\phi^2 & 0 \le \phi < 2\\ 0 & \phi \ge 2 \end{cases}$$

Para Liu, Liu e Lam (2003), a *spline* quártica têm apresentado resultados mais precisos que a *spline* cúbica, mas possui uma desvantagem: a derivada segunda não é nula na fronteira do suporte compacto.

FIGURA 5 – A FUNÇÃO NÚCLEO *NEW* QUÁRTICO (3.13) E SUAS DUAS PRIMEIRAS DERIVADAS.



A Spline quíntica proposta por Morris (1996), é definida na equação (3.14) e apresentada na FIGURA 6.

$$W_e^0(\phi) = \frac{\alpha_{e,d}}{h^d} \begin{cases} (3-\phi)^5 - 6(2-\phi)^5 + 15(1-\phi)^5 & 0 \le \phi < 1\\ (3-\phi)^5 - 6(2-\phi)^5 & 1 \le \phi < 2\\ (3-\phi)^5 & 2 \le \phi < 3\\ 0 & \phi \ge 3 \end{cases}$$
(3.14)

As constantes de normalização são dadas por $\alpha_{e,1} = 1/120$, $\alpha_{e,2} = 7/478\pi$ e $\alpha_{e,3} = 1/120\pi$, sendo suas primeira e segunda derivadas, respectivamente,

$$W_e^1(\phi) = -5\frac{\alpha_{e,d}}{h^{d+1}} \begin{cases} (3-\phi)^4 - 6(2-\phi)^4 + 15(1-\phi)^4 & 0 \le \phi < 1\\ (3-\phi)^4 - 6(2-\phi)^4 & 1 \le \phi < 2\\ (3-\phi)^4 & 2 \le \phi < 3\\ 0 & \phi \ge 3 \end{cases}$$

е

$$W_e^2(\phi) = 20 \frac{\alpha_{e,d}}{h^{d+2}} \begin{cases} (3-\phi)^3 - 6(2-\phi)^3 + 15(1-\phi)^3 & 0 \le \phi < 1\\ (3-\phi)^3 - 6(2-\phi)^3 & 1 \le \phi < 2\\ (3-\phi)^3 & 2 \le \phi < 3\\ 0 & \phi \ge 3 \end{cases}$$

FIGURA 6 – A FUNÇÃO SPLINE QUÍNTICA (3.14) E SUAS DUAS PRIMEIRAS DERIVADAS.



FONTE: O autor (2022).

Outras funções núcleo são apresentadas pela literatura, entre elas, a função Duplo Cosseno (em inglês, *Double Cosine*), proposta por (YANG; PENG; LIU, 2014). Sua principal característica é utilizar a trigonometria para garantir a simetria dentro do suporte compacto. A função duplo cosseno é definida na equação (3.15) e apresentada na FIGURA 7.

$$W_f^0(\phi) = \frac{\alpha_{f,d}}{h^d} \begin{cases} 4\cos\left(\frac{\pi}{\kappa}\phi\right) + \cos\left(\frac{2\pi}{\kappa}\phi\right) + 3 & 0 \le \phi \le \kappa \\ 0 & k < \phi \end{cases},$$
(3.15)

sendo κ um fator escala e as constantes de normalização iguais a $\alpha_{f,1} = 1/6\kappa$, $\alpha_{f,2} = \pi/[(3\pi^2 - 16)\kappa^2]$ e $\alpha_{f,3} = \pi/[(4\pi^2 - 30)\kappa^3]$. As derivadas primeira e segunda são dadas respectivamente por:

$$W_f^1(\phi) = \frac{\alpha_{f,d}}{h^{d+1}} \begin{cases} -4\frac{\pi}{\kappa} \operatorname{sen}\left(\frac{\pi\phi}{\kappa}\right) - 2\frac{\pi}{\kappa} \operatorname{sen}\left(\frac{2\pi\phi}{\kappa}\right) & 0 \le \phi \le \kappa \\ 0 & k < \phi \end{cases}$$

е

$$W_f^2(\phi) = \frac{\alpha_{f,d}}{h^{d+2}} \begin{cases} -4\frac{\pi^2}{\kappa^2} \cos\left(\frac{\pi\phi}{\kappa}\right) - 4\frac{\pi^2}{\kappa^2} \cos\left(\frac{2\pi\phi}{\kappa}\right) & 0 \le \phi \le \kappa \\ 0 & \kappa < \phi \end{cases}$$

FIGURA 7 – A FUNÇÃO DUPLO COSSENO (3.15) E SUAS DUAS PRIMEIRAS DERIVADAS COM $\kappa=2,0.$



3.2 Tratamento de fronteira

3.2.1 Partículas ghosts

O tratamento de fronteira por partículas *ghosts* consiste em criar clones das partículas de fluido que estão dentro do domínio computacional e na região que está compreendida entre (0,kh] e [1-kh,1) para um domínio unitário [0,1], por exemplo. Estas partículas clonadas são alocadas no lado externo do domínio com simetria em relação a fronteira, conforme a representação da FIGURA 8. O símbolo $W_{w^*=f,j}^{\gamma=0}$ representa a função núcleo modelo f (veja a equação (3.15)) com j partículas recrutadas. A região em destaque representa o caso 2D que considera a distribuição desordenada das partículas no interior do domínio.

Ao se tratar de um escoamento, este tipo de tratamento possui um custo elevado, pois o monitoramento das partículas é realizado continuamente.

FIGURA 8 – PARTÍCULAS GHOSTS ATRIBUÍDAS NA REGIÃO EXTERNA DO DOMÍNIO.



FONTE: O autor (2022).

3.2.2 Partículas dummy

Este tratamento de fronteira que considera partículas dummy é caracterizado for faixas de partículas fixadas com espaçamento uniforme de incremento Δx no lado externo e sobre a fronteira mesmo que a discretização interna considere a distribuição desordenada das partículas. A vantagem deste método em simulações de escoamento advém da não necessidade de monitoramento das partículas para a criação de clones. Uma representação deste tipo de tratamento pode ser observada na FIGURA 9. A região em destaque na figura, representa a discretização 2D interna que considera a distribuição desordenada das partículas. Nos casos da distribuição uniforme das partículas internas, o tratamento de fronteira por partículas ghosts e dummy ficam idênticos.

3.3 Recrutamento de partículas

O recrutamento por força bruta realiza o cálculo da distância entre a partícula \mathbf{x}_i e todas as partículas que discretizam o domínio. Com isso, a operação é caracterizada

FIGURA 9 – PARTÍCULAS DUMMY ATRIBUÍDAS NA REGIÃO EXTERNA DO DOMÍNIO.



FONTE: O autor (2022).

por um esforço computacional muito elevado, ou seja, tem ordem de complexidade $O(N_t^2)$. Para otimizar esse processo, a metodologia de recrutamento por malha uniforme foi criada (PAIVA *et al.*, 2009), ver FIGURA 10. Essa técnica consiste em fazer uma varredura no subdomínio limitado por nove elementos de malha com tamanho h nas direções $x \in y$, respectivamente. Dessa forma, o esforço computacional passa da ordem de complexidade $O(N_t^2)$ para $O(N_t)$.

3.4 Discretização da equação da difusão de calor em regime permanente

O operador laplaciano SPH (BROOKSHAW, 1985) para uma função escalar, é definido como $\nabla^2 \Psi(\mathbf{x})$. Sabendo-se que

$$\nabla^2 \Psi(\mathbf{x}_i) = 2 \sum_{j \in V_i} (\psi(\mathbf{x}_i) - \psi(\mathbf{x}_j)) \frac{m_j}{\rho_j} \Delta_{\mathbf{x}_{ij}} \cdot \nabla_i W_{ij} + O(\mathbf{x}_i - \mathbf{x}_j)^2, \qquad (3.16)$$

onde $\Delta_{\mathbf{x}_{ij}} = (\mathbf{x}_i - \mathbf{x}_j)/||\mathbf{x}_i - \mathbf{x}_j||^2$ e

$$\nabla_i W(\mathbf{x}_{ij}, h) = \frac{\alpha_{w^*, d} \mathbf{x}_{ij}}{h^{d+1} r_{ij}} \frac{\partial W_{ij}}{\partial \phi_{ij}}, \qquad (3.17)$$

onde $\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j$, $r_{ij} = ||\mathbf{x}_{ij}||$, $\alpha_{w^*,d}$ é a constante de normalização e d a dimensão do domínio. A derivada $\partial W_{ij}/\partial \phi_{ij}$ está definida na Subseção 3.1.5. Sendo assim, a aproximação SPH para a equação da difusão de calor em regime permanente é dada por $-\nabla^2 \psi(\mathbf{x}_i) = f(\mathbf{x}_i)$, ou seja,

$$-2\sum_{j\in V_i} (\psi(\mathbf{x}_i) - \psi(\mathbf{x}_j)) \frac{m_j}{\rho_j} \Delta_{\mathbf{x}_{ij}} \nabla_i W_{ij} + O(\mathbf{x}_i - \mathbf{x}_j)^2 = f(\mathbf{x}_i).$$
(3.18)

FIGURA 10 – RECRUTAMENTO OTIMIZADO DE PARTÍCULAS POR MALHA UNI-FORME.



FONTE: O autor (2022).

Fazendo

$$F(\mathbf{x}_{ij}) = \frac{\alpha_{w^*,d}}{h^{d+1}} \frac{\Delta_{\mathbf{x}_{ij}} \cdot \mathbf{x}_{ij}}{r_{ij}} \frac{\partial W_{ij}}{\partial \phi_{ij}}$$

e desprezando-se os termos do erro de truncamento, obtém-se

$$-\nabla^2 \psi(\mathbf{x}_i) = -2 \sum_{j \in V_i} \frac{m_j}{\rho_j} (\psi(\mathbf{x}_i) - \psi(\mathbf{x}_j)) F(\mathbf{x}_{ij}).$$
(3.19)

A equação (3.19) aplicada a cada partícula ido domínio computacional resulta no sistema

$$\begin{cases} -2\sum_{j\in V_1} \frac{m_j}{\rho_j} (\psi(\mathbf{x}_1) - \psi(\mathbf{x}_j)) F(\mathbf{x}_{1j}) = f(\mathbf{x}_1) \\ -2\sum_{j\in V_2} \frac{m_j}{\rho_j} (\psi(\mathbf{x}_2) - \psi(\mathbf{x}_j)) F(\mathbf{x}_{2j}) = f(\mathbf{x}_2) \\ \vdots \\ -2\sum_{j\in V_n} \frac{m_j}{\rho_j} (\psi(\mathbf{x}_n) - \psi(\mathbf{x}_j)) F(\mathbf{x}_{nj}) = f(\mathbf{x}_n) \end{cases}$$
(3.20)

sendo *n* o número de partículas que discretizam o domínio. O sistema linear de equações (3.20) é representado matricialmente por $A \cdot \psi(\mathbf{x}_{ij}) = \mathbf{b}$.

A matriz Atem dimensão $n \times n$ e é representada pelos seguintes coeficientes

 $a_{ij}, i, j \in \{1, 2, \dots, n\}$, cujos valores são definidos na equação (3.21).

$$\begin{cases}
 a_{ij} = -2\frac{m_j}{\rho_j}F(\mathbf{x}_{ij}), \ j \neq i \\
 a_{ii} = -\sum_{k \neq i} a_{ik}
\end{cases}$$
(3.21)

O vetor $\psi(\mathbf{x}_{ij}) = [\psi(\mathbf{x}_i) - \psi(\mathbf{x}_j))]$ é de dimensão *n* e conhecido como vetor solução com as componentes dadas por $\psi_i = \psi(\mathbf{x}_{ij})$.

O vetor **b** também de dimensão n é chamado de vetor independente cujas componentes são $b_i = f(\mathbf{x}_i)$.

3.5 Discretização da equação da difusão de calor em regime transiente

Diversas formas de integração temporal podem ser aplicadas na discretização do modelo matemático que representa a difusão de calor transiente. Sendo assim, para manter a mesma ordem de acurácia da discretização espacial com o método SPH, escolheu-se o método de Crank-Nicolson que possui segunda ordem de acurácia $O[\Delta x^2, \Delta t^2]$ (LIU; LIU, 2003; ÖZIŞIK *et al.*, 2017). Dessa forma, determina-se aproximações para a derivada temporal e espacial, conforme observa-se nas equações (3.22) e (3.23).

$$\frac{\partial \Psi(\mathbf{x}_i, t_n)}{\partial t} \approx \frac{\psi_i^{n+1} - \psi_i^n}{\Delta t}.$$
(3.22)

$$\nabla^2 \Psi(\mathbf{x}_i, t_n) \approx 2 \sum_{j \in V_i} \frac{m_j}{\rho_j} (\psi_i^n - \psi_j^n) \Delta_{\mathbf{x}_{ij}} \cdot \nabla_i W_{ij}.$$
(3.23)

A substituição das equações (3.22) e (3.23) na equação

$$\frac{\partial \Psi(\mathbf{x}_i, t_n)}{\partial t} = \mu \nabla^2 \Psi(\mathbf{x}_i, t_n) + f(\mathbf{x}_i), \qquad (3.24)$$

determina a equação (3.25),

$$\frac{\psi_i^{n+1} - \psi_i^n}{\Delta t} = 2\mu \sum_{j \in V_i} \frac{m_j}{\rho_j} (\psi_i^n - \psi_j^n) \Delta_{\mathbf{x}_{ij}} \cdot \nabla_i W_{ij}, \qquad (3.25)$$

onde $f(\mathbf{x}_i) = 0$.

3.5.1 Método de Crank-Nicolson para o SPH

O método de Crank-Nicolson, conforme observa-se em (CUMINATO; MENE-GUETTE JUNIOR, 2013) aproxima a função $\Psi(\mathbf{x}_i, t_n)$ por meio de uma média aritmética entre aproximações no tempo $n \in n + 1$. Dessa forma, pode-se aproximar o termo difusivo da equação do calor conforme mostrado na equação (3.26).

$$\frac{\partial^2 \Psi(\mathbf{x}_i, t_n)}{\partial x^2} \approx \frac{1}{2} \left[2 \sum_{j \in V_i} \frac{m_j}{\rho_j} (\psi_i^n - \psi_j^n) \Delta_{\mathbf{x}_{ij}} \cdot \nabla_i W_{ij} \right]
+ \frac{1}{2} \left[2 \sum_{j \in V_i} \frac{m_j}{\rho_j} (\psi_i^{n+1} - \psi_j^{n+1}) \Delta_{\mathbf{x}_{ij}} \cdot \nabla_i W_{ij} \right].$$
(3.26)

Nota-se que a derivada temporal é aproximada pelo método de Euler progressivo, equação (3.22), entretanto, ao considerar a aproximação definida para todos os termos da equação do calor, defini-se um sistema de equações implícito. Igualando o lado esquerdo e direito das equações (3.25-3.26), respectivamente, determina-se a aproximação

$$\frac{\psi_i^{n+1} - \psi_i^n}{\Delta t} = \frac{1}{2} \left[2 \sum_{j \in V_i} \frac{m_j}{\rho_j} (\psi_i^n - \psi_j^n) \Delta_{\mathbf{x}_{ij}} \cdot \nabla_i W_{ij} \right]
+ \frac{1}{2} \left[2 \sum_{j \in V_i} \frac{m_j}{\rho_j} (\psi_i^{n+1} - \psi_j^{n+1}) \Delta_{\mathbf{x}_{ij}} \cdot \nabla_i W_{ij} \right].$$
(3.27)

Considerando-se que o suporte compacto contenha as partículas $V_i = {\mathbf{x}_{i-1}, \mathbf{x}_{i+1}}$, então pode-se substituir $-\psi_j^n$ da equação (3.27) por $(-\psi_{i-1}^n - \psi_{i+1}^n)$ e $-\psi_j^{n+1}$ por $(-\psi_{i-1}^{n+1} - \psi_{i+1}^{n+1})$. Após algumas manipulações algébricas, obtém-se

$$\sigma\psi_{i-1}^{n+1} + (1-\sigma)\psi_i^{n+1} + \sigma\psi_{i+1}^{n+1} = -\sigma\psi_{i-1}^n + (1+\sigma)\psi_i^n - \sigma\psi_{i+1}^n, \qquad (3.28)$$

sendo

$$\sigma = \mu \frac{m_j}{\rho_j} \frac{\Delta t}{r_{ij}} \frac{\partial W_{ij}}{\partial \phi},$$

que é a equação discreta unidimensional da difusão de calor em regime transiente com o método SPH.

3.5.2 Solver TDMA

O Algoritmo de Matriz Tridiagonal (em inglês, *Tridiagonal Matrix Algorithm*, TDMA) é uma técnica direta de resolver sistemas lineares de equações algébricas nos casos em que a matriz de coeficientes tem a forma tridiagonal (BURDEN; FAIRES, 2005), conforme equações (3.29) e (3.30).

$$c_i x_{i-1} + d_i x_i + e_i x_{i+1} = b_i \tag{3.29}$$

d_1	e_1	0	0	0		0	$\begin{bmatrix} x_1 \end{bmatrix}$		b_1	
c_2	d_2	e_2	0	0		0	x_2		b_2	
0	C_3	d_3	e_3	0		0	x_3		b_3	
0	۰.	·	۰.	·	·	0	÷	=	÷	(3.30)
0		0	c_{n-2}	d_{n-2}	e_{n-2}	0	x_{n-2}		b_{n-2}	
0		0	0	c_{n-1}	b_{n-1}	e_{n-1}	x_{n-1}		b_{n-1}	
0		0	0	0	c_n	d_n	x_n		b_n	

A solução do sistema definido na equação (3.29), onde A é uma matriz simétrica, está descrita no ALGORITMO 1. O TDMA para matriz simétrica foi escolhido por ser um algoritmo eficiente para resolver os sistemas tridiagonais simétricos determinados pela discretização com SPH nos casos 1D e com condições de contornos prescritas. Se a discretização considerar mais de duas partículas vizinhas, o *solver* TDMA deve ser substituído por outro que considere o número de diagonais de tal discretização.

Algoritmo 1: SOLVER TDMA SIMÉTRICO.								
Entrada: A, b								
Saída: x								
1 Construção dos novos coeficientes								
2 início								
para $i = 1$ faça								
$4 \qquad \qquad e'_i = \frac{e_i}{d_i}$								
$5 \qquad b'_i = \frac{b_i}{d_i}$								
$6 \mathbf{fim}$								
para $i=2,\ldots,n$ faça								
$e_i' = \frac{e_i}{d_i - c_i e_{i-1}'}$								
9 $b'_i = \frac{b_i - c_i b'_{i-1}}{d_i - c_i e'_{i-1}}$								
10 fim								
11 fim								
12 Retrosubstituição								
13 início								
para $i = n - 1, n - 2, \dots, 1$ faça								
15 $x_n = b'_n$								
16 $x_i = b'_i - e'_i x_{i+1}$								
7 fim								
18 fim								

3.5.3 Solver PDMA

O Algoritmo de Matriz Pentadiagonal (em inglês, *Pentadiagonal Matrix Algorithm*, PDMA) ou (em inglês, *Five-diagonal Matrix Algorithm*, FDMA) é também uma técnica direta de resolver sistemas lineares de equações algébricas nos casos em que a matriz de coeficientes tem a forma pentadiagonal (ENGELN-MÜLLGES; UHLIG, 1996), conforme equação (3.31). A solução do sistema Ax = b da equação (3.31) é obtida por meio da decomposição A = LR. As matrizes $L \in R$ são apresentadas na equação (3.32), enquanto na equação (3.31), encontram-se o vetor auxiliar c.

$$A = \begin{bmatrix} d_{1} & e_{1} & f_{1} & 0 & 0 & \dots & 0 \\ h_{2} & d_{2} & e_{2} & f_{2} & 0 & \dots & 0 \\ g_{3} & h_{3} & d_{3} & e_{3} & f_{3} & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & g_{n-2} & h_{n-2} & d_{n-2} & e_{n-2} & f_{n-2} \\ 0 & \dots & 0 & g_{n-1} & h_{n-2} & d_{n-1} & e_{n-1} \\ 0 & \dots & 0 & 0 & g_{n} & h_{n} & d_{n} \end{bmatrix}, b = \begin{bmatrix} b_{1} \\ b_{2} \\ b_{3} \\ \vdots \\ b_{n-2} \\ b_{n-1} \\ b_{n} \end{bmatrix}, c = \begin{bmatrix} c_{1} \\ c_{2} \\ c_{3} \\ \vdots \\ c_{n-2} \\ c_{n-1} \\ c_{n} \end{bmatrix}$$
(3.31)

	α_1	0	0	0	0		0		1	γ_1	δ_1	0	0		0	
	β_2	α_2	0	0	0		0		0	1	γ_2	δ_2	0		0	
	ε_3	β_3	$lpha_3$	0	0		0		0	0	1	γ_3	δ_3		0	
L =	0	·	·	·	·	·	0	, R =	0	·	·	۰.	۰.	·	0	.
	0		ε_{n-2}	β_{n-2}	α_{n-2}	0	0		0		0	0	1	γ_{n-2}	δ_{n-2}	
	0		0	ε_{n-1}	β_{n-1}	α_{n-1}	0		0		0	0		1	γ_{n-1}	
	0		0	0	ε_n	β_n	α_n		0		0	0	0	0	1	
															$(3.\overline{3})$	32)

A solução do sistema Ax = b na forma pentadiagonal, onde A é uma matriz assimétrica, está descrita no ALGORITMO 2. Escolheu-se o solver PDMA assimétrico para resolver os sistemas pentadiagonais assimétricos gerados pela condição de contorno do tipo Neumman nos casos 1D. Na prática a mudança em relação aos sistemas tridiagonais ocorre apenas na linha da matriz que está associada à condição de contorno. Sendo assim, nos casos 1D mostrados nessa tese, se a condição de contorno do tipo Neumman for em x = 0, tem-se a mudança apenas na primeira linha da matriz; se for em x = 1, a mundança ocorre apenas na última linha da matriz. Todas as demais linhas da matriz preservam a forma tridiagonal.

3.5.4 Compressed Sparse Row (CSR)

O método da Linha Esparsa Compactada (em inglês, *Compressed Sparse Row*, CSR) (BULUÇ *et al.*, 2009), permite que uma matriz esparsa seja armazenada em três

```
Algoritmo 2: SOLVER PDMA ASSIMÉTRICO.
```

Entrada: A, b Saída: x 1 Fatoração A = LR2 início $\alpha_1 = d_1$ 3 $\gamma_1 = e_1/\alpha_1$ $\mathbf{4}$ $\delta_1 = f_1 / \alpha_1$ $\mathbf{5}$ $\beta_2 = h_2$ 6 $\alpha_2 = d_2 - \beta_2 \gamma_1$ $\mathbf{7}$ $\gamma_2 = (e_2 - \beta_2 \delta_1) / \alpha_2$ 8 $\delta_2 = f_2 / \alpha_2$ 9 para $i = 3, \ldots, n-2$ faça 10 $\beta_i = h_i - g_i \gamma_{i-2}$ 11 $\alpha_i = d_i - g_i \delta_{i-2} - \beta_i \gamma_{i-1}$ 12 $\gamma_i = (e_i - \beta_i \delta_{i-1}) / \alpha_i$ $\mathbf{13}$ $\delta_i = f_i / \alpha_i$ $\mathbf{14}$ fim 15 $\beta_{n-1} = h_{n-1} - g_{n-1}\gamma_{n-3}$ $\mathbf{16}$ $\alpha_{n-1} = d_{n-1} - g_{n-1}\delta_{n-3} - \beta_{n-1}\gamma_{n-2}$ $\mathbf{17}$ $\gamma_{n-1} = (e_{n-1} - \beta_{n-1}\delta_{n-2})/\alpha_{n-1}$ 18 $\beta_n = h_n - g_n \gamma_{n-2}$ 19 $\alpha_n = d_n - g_n \delta_{n-2} - \beta_n \gamma_{n-1}$ $\mathbf{20}$ para $i = 3, \ldots, n$ faça $\mathbf{21}$ $\varepsilon_i = g_i$ $\mathbf{22}$ fim 23 Atualização b = Lc $\mathbf{24}$ início $\mathbf{25}$ $c_1 = b_1 / \alpha_1$ $\mathbf{26}$ $c_2 = (b_2 - \beta_2 c_1)/\alpha_2$ $\mathbf{27}$ para $i = 3, \ldots, n$ faça $\mathbf{28}$ $c_i = (b_i - \varepsilon_i c_{i-2} - \beta_i c_{i-1})/\alpha_i$ 29 fim 30 fim 31 Retrosubstituição Rx = c32 início 33 $x_n = c_n$ $\mathbf{34}$ $\mathbf{35}$ $x_{n-1} = c_{n-1} - \gamma_{n-1} x_n$ para i = n - 2, ..., 1 faça 36 $x_i = c_i - \gamma_i x_{i+1} - \delta_i x_{i+2}$ 37 fim 38 fim 39 40 fim

vetores: AA, JA e IA. O vetor AA armazena todos os coeficientes não nulos da matriz A. Enquanto isso, JA carrega o número das colunas de cada um dos coeficientes de A

e, por fim, o vetor IA armazena o início e término de cada uma das linhas, conforme mostrado no ALGORITMO 3. A dimensão dos vetores AA e JA, equações (3.34) e (3.35), respectivamente, são as mesmas e, portanto, denotadas por (n_{aa}) . O vetor IAtem dimensão $n_{ia} = n + 1$, onde n é a ordem da matriz A definida na equação (3.33). O primeiro elemento de IA é sempre igual a unidade (IA(1) = 1) e o ultimo elemento de IAé sempre o número de elementos do vetor AA + 1 $(IA(n_{ia}) = n_{aa} + 1)$. O vetor AD não faz parte do método CSR, mas trata-se de um vetor importante para o desenvolvimento da programação paralela. Em AD, armazena-se apenas os elementos da diagonal principal da matriz A e sua dimensão é $N_x N_y = N_{ti}$, para $N_x = N_y$.

$$A = \begin{bmatrix} 3 & 4 & 0 & 0 & 0 \\ 2 & 5 & 2 & 0 & 0 \\ 0 & 7 & 9 & 3 & 0 \\ 0 & 0 & 6 & 1 & 8 \\ 0 & 0 & 0 & 7 & 3 \end{bmatrix}$$
(3.33)

$$JA = \begin{bmatrix} 1 & 2 & 1 & 2 & 3 & 2 & 3 & 4 & 3 & 4 & 5 & 4 & 5 \end{bmatrix}$$
(3.35)

$$IA = \begin{bmatrix} 1 & 3 & 6 & 9 & 12 & (n_{aa} + 1) \end{bmatrix} = \begin{bmatrix} 1 & 3 & 6 & 9 & 12 & 14 \end{bmatrix}$$
(3.36)

3.5.5 Solver Gauss-Seidel-S CSR

Os sistemas lineares de equações algébricas associados à discretização de modelos matemáticos utilizando métodos numéricos são, em geral, esparsos. O grau de esparsidade (GE), que representa a porcentagem de elementos nulos da matriz, aumenta à medida que o domínio é discretizado com um número cada vez maior de elementos de malha ou partículas, por exemplo. Dessa forma, o método CSR contribui com a economia de memória ao armazenar apenas os coeficientes não nulos da matriz e, além disso, permite realizar operações mais velozes do que nos casos em que utiliza-se a matriz esparsa completa. O ganho em velocidade acontece com ou sem paralelização, entretanto, utilizar o *solver* com a estrutura CSR e paralelizado, terá uma redução adicional no tempo de processamento que, por sua vez, é atrativa. O *solver* Gauss-Seidel-S *CSR* paralelizado está descrito no ALGORITMO 4. Essa estrutura recebe a denominação "Gauss-Seidel-S" pois trata-se de uma combinação entre Gauss-Seidel e Gauss-Jacobi que pode ser reconhecida como Gauss-Seidel modificado.

Algoritmo 3: COMPRESSED SPARSE ROW CSR COM AD.

```
Entrada: A
   Saída: AA, JA, IA, AD
IA(1) = 1
2 k0 = 0
s k1 = 0
4 início
       para i = 1, \ldots, n faça
\mathbf{5}
          para j = 1, \ldots, n faça
 6
              se A(i,j) \neq 0 então
 7
                   k0 = k0 + 1
 8
                   AA(k0) = A(i,j)
 9
                   JA(k0) = j
10
                  se i = j então
11
                      k1 = k1 + 1
\mathbf{12}
                       AD(k1) = A(i,j)
13
                  fim
14
               fim
15
               IA(i+1) = k0 + 1
16
          fim
\mathbf{17}
       fim
18
19 fim
```

Para facilitar a compreensão sobre os ganhos em tempo de processamento ao utilizar o *solver* Gauss-Seidel-S CSR paralelizado, é necessário conhecer alguma métrica de desempenho e o fenômeno conhecido como "lei de Amdahl" (AMDAHL, 1967).

Galante (2006) definiu a seguinte métrica:

Definição 1. O speed-up (S_p) é uma métrica adotada para medir o aumento da velocidade de um algoritmo Υ_1 em relação a Υ_2 .

$$S_p = \frac{t_{CPU}(\Upsilon_1)}{t_{CPU}(\Upsilon_2)}.$$
(3.37)

Em termos de paralelização, o S_p é utilizado para medir o aumento de velocidade ao comparar o tempo de CPU do algoritmo Υ com apenas um processador (Υ - algoritmo serial) e com τ processadores (Υ - algoritmo paralelizado) (TROTTENBERG; OOSTERLEE; SCHÜLLER, 2001). O S_p mostrado na equação (3.37) considera que os dois algoritmos estão sendo executados com o mesmo número de processadores. A métrica para o caso em que se utiliza τ processadores pode ser reescrita como

$$S_{\tau} = \frac{t_0(\Upsilon)}{t_{\tau}(\Upsilon)},\tag{3.38}$$

onde S_{τ} é o speed-up que mede o aumento de velocidade ao comparar o tempo de CPU t_0 do algoritmo Υ serial com um processador e o tempo de CPU t_{τ} do algoritmo Υ com τ processadores. Observe que o tempo $t_0(\Upsilon)$ é calculado sem paralelização e $t_1(\Upsilon)$ possui uma única região paralela. Para que S_{τ} seja calculado de forma coerente, o valor $S_1 = t_0(\Upsilon)/t_1(\Upsilon) \approx 1$. Quanto maior o valor de S_{τ} , menor é o tempo de CPU. Entretanto, devido a algumas regiões de cálculo serial dentro do algoritmo, o valor de S_{τ} possui um limitante que é o tempo de CPU do próprio algoritmo Υ em serial. Essa limitação é conhecida como "lei de Amdahl".

Sun e Chen (2010) mostaram uma forma mais generalizada da lei de Amdahl, descrita por

$$S_{Amdahl} = \frac{1}{(1 - f_p) + \frac{f_p}{\tau}},$$
(3.39)

onde f_p é a parcela do algoritmo que pode ser melhorada por um fator τ e $(1 - f_p)$ é a parcela que não pode ser melhorada. Nos termos do paralelismo, f_p representa a parcela do algoritmo que pode ser melhorada ao utilizar um número τ de processadores.

Na FIGURA 11, apresenta-se o *speed-up* teórico determinado pela lei de Amdahl, equação (3.39), para algoritmos $\Upsilon_1, \Upsilon_2, \ldots, \Upsilon_6$. Como forma de ilustrar tal lei, supõe-se que os algoritmos são paralelizados e que suas frações paralelizadas são, respectivamente, $f_p = 50\%$, $f_p = 80\%$, $f_p = 90\%$, $f_p = 95\%$, $f_p = 99\%$ e $f_p = 100\%$. Desse modo, ao analisar a FIGURA 11, pode-se concluir, por exemplo, que o algoritmo Υ_5 possui hipoteticamente 99% de paralelização e, quando executado com 28 processadores, seu *speed-up* é de 22 vezes em relação ao algoritmo Υ_0 executado em serial com apenas 1 processador.

3.5.6 Multigrid

O multigrid geométrico é um método capaz de suavizar o erro em uma malha, e por meio de operadores de transferência, injeta informações da malha fina na malha grossa (operador de restrição) e da malha grossa na malha fina (operador de prolongação) em um número de ciclos, que pode ser $V(\nu_1,\nu_2)$, conforme FIGURA 12, ou ainda, $W(\nu_1,\nu_2)$ ou $F(\nu_1,\nu_2)$, com ν_1 pré-suavizações e ν_2 pós-suavizações. As informações transferidas de uma malha para a outra podem ser desde apenas o resíduo (esquema CS), ou resíduo e solução (esquema FAS) (TROTTENBERG; OOSTERLEE; SCHULLER, 2000; WESSELING, 2004). Para facilitar a compreensão da transferência de informações entre malhas, destacase em vermelho a ideia do operador restrição (ida) e em azul do operador prolongação (volta).

Em suma, os modos oscilatórios do erro são suavizados ν_1 vezes tornando-se modos suaves, então transferem-se as informações para a malha imediatamente mais grossa

FIGURA 11 – LIMITAÇÃO SUPERIOR DE ALGORITMOS PARALELIZADOS COM f_p VARIANDO ENTRE 50% E 100%.



FONTE: O autor (2022).

por meio do operador de restrição. Este processo torna os modos suaves do erro mais oscilatórios novamente, permitindo que o método volte a ser eficiente.

Após atingir a malha mais grossa possível, ou desejada, faz-se a suavização dos modos oscilatórios do erro e inicia-se o retorno por meio do operador de prolongação. Este operador transfere as informações para a malha imediatamente mais fina, onde corrige-se a solução e suaviza-se ν_2 vezes. Realiza-se este procedimento até atingir a malha mais fina onde o problema é resolvido. Este processo, denominado ciclo, é repetido até que o erro atinja o critério de parada, que pode ser definido pela norma adimensional (L^l/L^0) ou número de iterações, por exemplo.

Sabendo-se que nem todos os métodos numéricos utilizam malha computacional e que um modelo mais generalizado de *multigrid* já apresentou resultados mais eficientes que o *singlegrid*, resolveu-se aplicar este modelo mais generalizado para resolver os problemas bidimensionais propostos nessa tese. O nome dado a este modelo é *Multigrid* Algébrico. O AMG utiliza informações da matriz de coeficientes para acelerar a convergência das soluções numéricas. Inicialmente seleciona-se um esquema de suavização, de forma que seja possível determinar a natureza do erro suave. Devido à falta de uma malha física, o conceito de suavização é definido algebricamente. Esta abordagem é importante para



FIGURA 12 – REPRESENTAÇÃO DE UM CICLO $V(\nu_1,\nu_2)$ DO MULTIGRID.

FONTE: O autor (2022).

diferenciar erros suaves dos não suaves (TROTTENBERG; OOSTERLEE; SCHULLER, 2000).

De acordo com Suero (2010), para facilitar a compreensão, pode-se adotar dois níveis de malha. Para a malha fina, utiliza-se o índice h, e para a mais grossa H, onde a malha H possui metade dos pontos da malha h (H = 2h). Ao resolver o sistema definido na equação (3.40),

$$A_h \Psi^h = f^h \text{ ou } \sum_{j \in \Omega^h} a^h_{ij} \Psi^h_j = f^h_i \text{ em } \Omega^h, \qquad (3.40)$$

com A_h a matriz esparsa de coeficientes do sistema linear de equações algébricas, f^h o termo fonte, encontra-se o vetor solução Ψ^h .

Para definir os pontos das malhas grossa e fina, primeiramente decompõe-se $\Omega^h = C^h \cup F^h$, onde C^h representa as variáveis que estão no nível grosso (variáveis do tipo C) e F^h é o conjunto complementar (variáveis do tipo F). Assumindo-se que $\Omega^H = C^h$, o sistema linear de equações algébricas no nível grosso é definido na equação (3.41) como

$$A_H \Psi^H = f^H \quad \text{em} \quad \Omega^H. \tag{3.41}$$

A matriz A_H é formulada pelo princípio de Galerkin, onde define-se a equação (3.42)

$$A_H = I_h^H A_h I_H^I, (3.42)$$

com I_h^H e I_H^I os operadores de restrição e prolongação, respectivamente.

Os métodos *multigrid* exigem processos de suavização, que consistem em solucionar o sistema linear de equações algébricas um número desejado de vezes. Define-se na equação (3.43) o operador de suavização linear S_h . Desta forma, obtém-se uma suavização da solução ao realizar a seguinte operação

$$\psi^{h} = S_{h} \Psi^{h} + (I_{h} - S_{h}) A_{h}^{-1} f^{h}, \qquad (3.43)$$

onde ψ^h é a solução suavizada e I_h o operador identidade. Consequentemente, define-se o erro numérico na equação (3.44) por

$$e^h = \Psi^h - \psi^h, \tag{3.44}$$

com Ψ^h sendo a solução analítica da equação (3.40).

Adota-se neste trabalho o suavizador (S_h) Gauss-Seidel-S CSR serial, definido na Seção 3.5.5. Trottenberg, Oosterlee e Schuller (2000) relatam que para boa convergência, as componentes $C \in F$ precisam ser uma interface entre a restrição na malha fina e a prolongação na malha grossa. Ainda segundo os autores, a decomposição precisa ser criada o mais uniforme possível, de modo que as variáveis em F sejam limitadas pelas variáveis em C para a restrição. O procedimento do AMG pode ser observado no ALGORITMO 5, uma simplificação de modelo que considera apenas duas malhas (BRIGGS; HENSON; MCCORMICK, 2000). Para que o AMG seja compreendido, algumas definições são fundamentais, sendo elas:

Definição 2. Um ponto *i* é definido conectado ao ponto $j \in \Omega^h$ se $a_{ij}^h \neq 0$.

Definição 3. A vizinhaça de um ponto i é dada por

$$\overline{N}_i^h = \{(i,j) \in \Omega^h / j \neq i, a_{ij} \neq 0\}.$$
(3.45)

Definição 4. Uma variável i está fortemente n-acoplada a uma variável j, se

$$-a_{ij} \le \theta \max_{a_{ik} < 0} |a_{ik}|$$
, sendo θ o fator redutor de malha $0 < \theta < 1.$ (3.46)

Definição 5. Seja S_i um conjunto de pontos com forte influência em i, definido por

$$S_i = \{ j \in \overline{N}_i / i \text{ é fortemente } n \text{-acoplada a } j \}.$$
(3.47)

Definição 6. Seja S_i^T um conjunto de pontos com forte dependência de um ponto i, definido por

$$S_i^T = \{ j \in \Omega / i \in S_j \}.$$

$$(3.48)$$

3.5.7 Engrossamento

Optou-se neste trabalho pela forma de engrossamento padrão para o AMG, entretanto há outras formas de realizá-lo (TROTTENBERG; OOSTERLEE; SCHULLER, 2000). Adotou-se esta forma padrão para que seja possível comparar os resultados obtidos ao aplicar o SPH com outros conhecidos na literatura que utilizaram FDM ou FVM, por exemplo (SUERO *et al.*, 2012).

Para o engrossamento padrão, tem-se que as partições $C \in F$ são baseadas em acoplamentos diretos, ou seja, para cada variável $i \in F$ necessita-se de um número mínimo de acoplamentos $j \in \overline{N}_i$. O procedimento é iniciado quando define-se uma variável ipara pertencer a C. As variáveis j fortemente n-acopladas a i se tornam variáveis em F. Sequencialmente, todas as variáveis indefinidas passam a ser tratadas e aquelas fortemente n-acopladas tornam-se pertencentes a F.

Definição 7. Para uma variável qualquer indefinida i, sua medida de importância λ_i é definida como

$$\lambda_i = |S_i^T \cap U| + 2|S_i^T \cap F|, \quad i \in U, \tag{3.49}$$

onde U representa apenas um conjunto de variáveis indefinidas, $|S_i^T \cap U|$ é o número de variáveis indefinidas com forte dependência de $i \in |S_i^T \cap F|$ é o número de variáveis em F com forte dependência de i.

3.5.8 Prolongação

De acordo com Trottenberg, Oosterlee e Schuller (2000) a convergência rápida está diretamente relacionada com uma boa aproximação ao interpolar os erros algébricos suaves. Enquanto isto, a dimensão do operador do nível grosso e seu tempo de processamento dependem do número de variáveis em C. Para garantir a eficiência global do método, o número de variáveis em C precisa ser limitado. Ainda de acordo com o autor, pode-se realizar a prolongação no AMG pelo método de interpolação direta, padrão, multipassos de Jacobi ou por truncamento da interpolação. Novamente optou-se por interpolação padrão neste trabalho para facilitar as análises que têm como base os resultados já conhecidos usando FDM ou FVM.

Para o desenvolvimento do engrossamento, supõe-se que exista um conjunto e_i com $i \in C$ variáveis na malha grossa representando um erro suave possível de ser interpolado para a malha fina $C \cup F$. Se um ponto $j \in C$ possui forte influência sobre i, então e_j exerce forte influência sobre e_i (malha fina).

Segundo Briggs, Henson e McCormick (2000), a vizinhaça de i, definida na equação (3.45) pode ser dividida em três categorias:

- 1. $C_i = C \cap S_i$ os pontos da malha grossa exercem influência sobre *i*.
- 2. $D_i^S = D_i \cap S_i$, onde $D_i = N_i C_i$, os pontos da malha fina que exercem forte influência sobre *i*.
- 3. $D_i^W = D_i S_i$, os pontos que exercem fraca influência sobre *i*. Este conjunto pode conter pontos na malha grossa e fina. Estes pontos são denominados de vizinhos com fraca conexão.

Na FIGURA 13 apresentam-se as conexões com forte e fraca influência sobre o ponto i. Os pontos da malha grossa são representados pelos círculos brancos. Os pontos pertencentes a C que exercem forte influência sobre i apresentam-se com linhas contínuas. As linhas tracejadas representam os pontos de F que exercem forte influência sobre i, enquanto isto, as linhas pontilhadas representam os pontos de F que exercem fraca influência sobre i.

FIGURA 13 – CONEXÕES FORTES E FRACAS DE UM PONTO i.



FONTE: O autor (2022).

Definição 8. O operador interpolação I_H^h é definido como

$$(I_H^h e^H)_i = \begin{cases} e_i^h, & \text{se} \quad i \in C\\ \sum_{j \in C_i} \omega_{ij} e_j^H, & \text{se} \quad i \in F \end{cases},$$
(3.50)

onde ω_{ij} são os pesos da interpolação, conforme equação (3.51).

$$\omega_{ij} = -\frac{a_{ij} + \sum_{m \in D_i^S} \left(\frac{a_{im}a_{mj}}{\sum_{k \in C_i} a_{mk}}\right)}{a_{ii} + \sum_{n \in D_i^W} a_{in}}.$$
(3.51)

Para otimizar o processo computacional é preciso realizar um truncamento no operador de interpolação, de modo que as conexões menores que o módulo da maior delas sejam desprezadas.

$$S_i^D = \left\{ j \in D_i^S / \sum_{l \in C_i} |a_{ij}| > \varepsilon \left(\frac{|a_{ij}|}{\max |a_{ik}|} \right) \max |a_{jl}|, \text{para } \varepsilon > 0 \text{ fixo} \right\}.$$
(3.52)

A constante ε é denominada fator de forte dependência na malha grossa, assim como em (SUERO, 2010). O conjunto S_i^D contém os pontos com forte dependência do conjunto C_i . Os algoritmos das fases, inicial e de solução podem ser encontrados em (SUERO, 2010), enquanto o CS para o AMG está descrito no ALGORITMO 5.

Definição 9. O operador prolongação para matriz simétrica definida positiva é

$$I_h^H = (I_H^h)^T. (3.53)$$

3.5.9 Solver Gauss-Seidel

Para resolver os sistemas lineares de equações algébricas, pode-se utilizar o *solver* Gauss-Seidel (BURDEN; FAIRES; BURDEN, 2016) conforme mostrado na equação (3.54).

$$[\psi(x_i)]^l = \frac{1}{a_{ii}} \left[-\sum_{j=1}^{i-1} (a_{ij} x_j^l) - \sum_{j=i+1}^n (a_{ij} x_j^{l-1}) + f_i \right].$$
(3.54)

O resíduo $\mathbf{r} = \mathbf{f} - A\boldsymbol{\psi}$ do sistema $A\boldsymbol{\psi} = \mathbf{f}$ é definido por

$$\mathbf{r}^l = \mathbf{f} - \boldsymbol{\psi}^l, \tag{3.55}$$

onde $l = 1, 2, ..., (L^l/L^0 \le \text{Tol})$ e

$$L^{l} = \|\mathbf{r}^{l}\|_{2} = \sqrt{\sum_{i=1}^{n} (r_{i}^{l})^{2}},$$
(3.56)

com n sendo o número de incógnitas e os valores $L^l \in L^0$ correspondem a norma euclidiana (L_2) na iteração l, e na iteração inicial l = 0, respectivamente. Entretanto, o solver mostrado na equação (3.54) possui desvantagens como: exigir armazenamento completo da matriz e coeficientes que, por sua vez torna a convergência mais lenta. Para contornar essas desvantagens, escolheu-se o solver Gauss-Seidel-S apresentado no Algorítimo 4 sem paralelização, quando utilizado o AMG.

Para efeitos de adequação, nessa tese, denomina-se a expressão Multinível Algébrico (em inglês, *Algebraic Multilevel*, AML), pois o SPH não possui malha (*grid*) computacional e cada discretização é reconhecida como nível. Sendo assim, as expressões malha grossa e fina tornam-se, nível inconsistente e consistente, respectivamente, conforme FIGURA 14.

É importante não associar o termo "nível consistente" com o fenômeno matemático chamado de "consistência". Afinal, com o SPH, pode-se calcular soluções numéricas utilizando diversos níveis consistentes de partículas sem que o fenômeno da consistência seja comprovado. Usando o FDM, por exemplo, a consistência seria confirmada ao analisar os efeitos do erro verdadeiro de uma variável local e/ou com alguma variável global à medida que $h \rightarrow 0$. No SPH, a comprovação de tal fenômeno é de interesse acadêmico, o que também justifica o estudo dessa tese.

FIGURA 14 – REPRESENTAÇÃO DOS NÍVEIS DO AML.



FONTE: O autor (2022).

3.6 Erro Numérico

O erro da solução numérica (ψ) de uma variável de interesse é definido como

$$E(\psi) = \Psi - \psi, \qquad (3.57)$$

onde Ψ é a solução analítica exata de uma variável de interesse e ψ a solução numérica. Ao calcular o erro numérico $(E(\psi))$, entende-se que este erro seja influenciado diretamente por quatro fontes principais de erro. Para caracterizar esta dependência, representa-se o erro numérico

$$E(\psi) = E(\varepsilon_{\tau}, \varepsilon_n, \varepsilon_\pi, \varepsilon_p), \qquad (3.58)$$

onde ε_{τ} é o erro de truncamento, ε_n o erro de iteração, ε_{π} o erro de arredondamento e ε_p o erro de programação (MARCHI, 2001). Ao discretizar os modelos matemáticos com o método SPH, pretende-se verificar se estas fontes de erro permanecem sendo as principais, ou se novas fontes podem surgir para afetar o erro da solução numérica.

Considera-se nessa tese a ausência de erros de programação devido as análises realizadas constantemente como parte da metodologia de verificação das soluções numéricas. Como a análise dos erros gerados pela discretização é parte fundamental nesta pesquisa, pretende-se utilizar técnicas de modo que a única fonte de erro de discretização seja o erro de truncamento.

Pode-se estimar o erro de discretização por duas diferentes formas: estimativas *a priori* ou *a posteriori* (SZABÓ; BABUŠKA, 1991; SZABÓ; BABUŠKA, 2011).

3.6.1 Estimativas do erro de discretização a priori

Estima-se o erro de discretização *a priori* tendo como base o erro de truncamento local (ε_{τ}) (LEVEQUE, 2007) obtido ao discretizar o modelo matemático com um método numérico que geralmente advém de expressões combinadas da expansão em série de Taylor.

Admitindo-se que o erro de discretização $(E(\psi))$ é gerado exclusivamente pelo erro de truncamento local, obtém-se

$$E(\psi) = \varepsilon_{\tau} = \overline{C}h^p, \qquad (3.59)$$

onde \overline{C} é um coeficiente constante desconhecido que independe de h, p é a ordem do erro de discretização e h é o tamanho do elemento de malha utilizado para discretizar o domínio $(h \to 0)$. Em outras palavras,

$$\lim_{h \to 0} E(\psi) = \lim_{h \to 0} \overline{C}h^p \longrightarrow 0.$$
(3.60)

Para facilitar a compreensão quanto ao decaimento do erro de discretização, adotase genericamente uma aproximação FDM qualquer de segunda ordem (p = 2), e ainda, duas malhas uniformes com elementos $h_1 e h_2$ para a malha grossa e fina, respectivamente. Dessa forma, como observa-se na equação (3.61), é possível estabelecer a seguinte relação

$$\frac{E(\psi_1)}{E(\psi_2)} = \frac{\overline{C}_1(h_1)^p}{\overline{C}_1(h_2)^p} = \frac{\overline{C}_1(h_1)^p}{\overline{C}_1\left(\frac{h_1}{2}\right)^p} = 2^p = 2^2 = 4.$$
(3.61)

Isto significa que o erro de discretização fica quatro vezes menor a cada refinamento ao considerar $h_2 = h_1/2$. Note que a inversão $E(\psi_2)/E(\psi_1) = 1/4$ também é verdadeira.

É importante destacar que ao se tratar de uma discretização de domínio utilizando o método SPH, as partículas são originalmente distribuídas com espaçamento uniforme, equivalente ao que acontece no FDM (GINGOLD; MONAGHAN, 1977; LUCY, 1977).

3.6.2 Estimativas do erro de discretização a posteriori

Tendo em vista o trabalho de revisão bibliográfica de (MARCHI, 2001), sabe-se que é possível estimar o erro de discretização *a posteriori* por formas diferentes, sendo que consideram-se dois grupos. O primeiro considera a análise apenas sobre a solução numérica em malha única, como por exemplo o FEM (ZHU; ZIENKIEWICZ, 1990). O segundo por sua vez, considera duas ou mais soluções obtidas em diferentes malhas com FDM e FVM (HORTMANN; PERIĆ; SCHEUERER, 1990).

3.6.3 Erro de truncamento

A expressão do erro de truncamento local (ε_{τ}) característica nos métodos FDM e FVM (FERZIGER; PERIĆ, 2002) é dada por

$$\varepsilon_{\tau}(\psi) = \overline{C}_1(h)^{p_L} + \overline{C}_2(h)^{p_2} + \overline{C}_3(h)^{p_3} + \ldots + \overline{C}_n(h)^{p_n}$$
(3.62)

A expressão (3.62) é denominada equação geral do erro de truncamento. As potências $p_L < p_2 < p_3 < \ldots < p_n$ são as ordens verdadeiras apenas dos termos não nulos na equação do erro de truncamento. Estas ordens verdadeiras são números inteiros positivos que geralmente caracterizam uma série aritmética sendo $p_L \ge 1$ a menor delas e chamada de ordem assintótica. Dessa forma, pode-se escrever a solução analítica exata como a soma entre solução numérica e o erro de truncamento, de acordo com equação

$$\Psi(x_i) = \psi(x_i) + \varepsilon_\tau(\psi(x_i)). \tag{3.63}$$

Tomando-se como exemplo uma aproximação CDS-2 por FDM para o operador diferencial de segunda ordem, obtém-se as equações

$$\nabla^2 \Psi_i = \frac{\Psi_{i-1} - 2\Psi_i + \Psi_{i+1}}{h^2} - \Psi_i^{iv} \frac{h^2}{12} - \Psi_i^{vi} \frac{h^4}{360} - \Psi_i^{viii} \frac{h^6}{20160} - \dots - \Psi_i^n \frac{h^{n-2}}{n!/2}, \quad (3.64)$$

$$\nabla^2 \Psi_i = \psi_i + \varepsilon_\tau(\psi_i), \qquad (3.65)$$

onde

$$\psi_i = \frac{\Psi_{i-1} - 2\Psi_i + \Psi_{i+1}}{h^2} \tag{3.66}$$

е

$$\varepsilon_{\tau}(\psi_i) = -\Psi_i^{iv} \frac{h^2}{12} - \Psi_i^{vi} \frac{h^4}{360} - \Psi_i^{viii} \frac{h^6}{20160} - \dots - \Psi_i^n \frac{h^{n-2}}{n!/2}.$$
 (3.67)

3.6.4 Erro de iteração

O erro de iteração (ε_n) (MARCHI, 2001) é definido como a diferença entre solução exata e aproximada do sistema de equações algébricas em uma determinada iteração (ROACHE, 1998; FERZIGER; PERIĆ, 2002), conforme equação

$$\varepsilon_n(\psi(x_i)) = \Psi(x_i) - [\psi(x_i)]^n, \qquad (3.68)$$

onde n representa o número da iteração atual no processo de solução do sistema de equações lineares ou não lineares.

Pode-se gerar este tipo de erro ao utilizar qualquer método iterativo para solucionar o modelo matemático com a formulação discreta. É comum em casos de equações não lineares, em *solvers* como Gauss-Seidel, em aplicações de métodos *multigrid*, pseudo-transientes, entre outros. Os métodos TDMA (BURDEN; FAIRES, 2005) e PDMA (ENGELN-MÜLLGES; UHLIG, 1996) são diretos, logo não possuem erros de iteração.

A FIGURA 15 representa o erro de iteração adimensional L_2^l/L_2^0 atingindo a ordem de magnitude da precisão quádrupla (precisão estendida ou *Real*16*) em que os cálculos são processados. Em uma simulação numérica, quando o erro de iteração oscila no limite da precisão, diz-se que não há erros de iteração. Os valores $L_2^l \in L_2^0$ correspondem à norma euclidiana (L_2) vetorial, definida na equação (3.69), na iteração l e na iteração inicial l = 0, respectivamente.

$$L_2 = \sqrt{\sum_{i=1}^{n} |E_i|^2}.$$
(3.69)

3.6.5 Erro de arredondamento

O erro de arredondamento (ε_{π}) é caracterizado pela representação finita dos números reais no ambiente computacional (BAUER, 1974; LINNAINMAA, 1976; STEWART, 1979; WALTER, 1993). São vinculados ao compilador utilizado para criar e executar o


FIGURA 15 – ERRO DE ITERAÇÃO.

código numérico computacional (*software*) e também ao computador (*hardware*). A relação é direta com o número de algarismos (*bytes*) adotados ao efetuar o cálculo no computador. Outra fonte de erro de arredondamento é a utilização de funções nativas do compilador que são aproximadas por séries infinitas mas que consideram apenas um número finito de termos ao utilizá-las.

Para mostrar o erro de arredondamento em uma solução numérica há duas estratégias: a primeira é caracterizada pelo gráfico do $|E(\psi)|$ versus h em escala log-log com $h \to 0$; a segunda pelo gráfico do erro de iteração (adimensional ou não) versus número de iterações em escala y-log com $l \to \infty$. Em problemas 1D a primeira opção pode ser a única, pois em geral utiliza-se algum método direto para determinar a solução numérica. Por outro lado, em problemas 2D ou 3D, é mais fácil encontrar o erro de arredondamento no processo iterativo do solver, pois atingir o erro de arredondamento em soluções numéricas para $h \to 0$ requer om tempo computacional elevado se nenhum método *multigrid* for utilizado. Para mostrar o erro de arredondamento, escolheu-se a segunda estratégia, onde apresenta-se o erro de iteração adimensional para valores de lsuficientemente elevados. Na FIGURA 16, nota-se o erro de iteração $\varepsilon_l \to 0$ à medida que $l \to \infty$, até atingir um valor de l suficientemente grande. A partir de então, o erro de arredondamento se torna maior que o erro de iteração. Em outras palavras, ε_l diminui até atingir um determinado l ótimo, enquanto isso, ε_{π} aumenta a partir do l ótimo. Portanto é esperada uma mudança de comportamento a partir de um l que pode ser denominado ótimo. A FIGURA 16 é uma ampliação da FIGURA 15 a partir de l = 149.



FIGURA 16 - ERRO DE ARREDONDAMENTO.

FONTE: O autor (2022)

3.6.6 Erro de poluição numérica

O erro de poluição numérica (ε_{np}) baseado em (MARCHI; SILVA, 2002; BURDEN; FAIRES; BURDEN, 2016) é caracterizado pelo erro verdadeiro ($E(\psi_i) = \Psi_i - \psi_i$) da variável de interesse nas coordenadas do nó x_i , quando aplicado no esquema numérico utilizado para discretizar o modelo matemático. Para facilitar o entendimento, apresentase o erro de poluição numérica para o operador laplaciano e para a regra do trapézio, discretizados com FDM, respectivamente.

$$\nabla^2 \Psi_i = \frac{\psi_{i-1} - 2\psi_i + \psi_{i+1}}{h^2} + \varepsilon_{np}(\psi(x_i)) + \varepsilon_\tau(\psi(x_i)), \qquad (3.70)$$

onde

$$\varepsilon_{np}(\psi(x_i)) = \frac{E(\psi_{i-1}) - 2E(\psi_i) + E(\psi_{i+1})}{h^2}$$
(3.71)

е

$$\psi_m = \frac{h}{2L} \sum_{i=2}^{N} (\psi_{i-1} + \psi_i) + \varepsilon_{np}(\psi_m) + \varepsilon_{\tau}(\psi_m), \qquad (3.72)$$

em que

$$\varepsilon_{np}(\psi_m) = \frac{h}{2L} \sum_{i=2}^{N} (E(\psi_{i-1}) + E(\psi_i)).$$
(3.73)

Note que nas duas aproximações numéricas, o termo do erro de truncamento foi acoplado. Isso quer dizer que, ao adicionarmos o erro de poluição numérica e o erro de truncamento (com seus infinitos termos) à aproximação numérica calculada, obtém-se a solução analítica numericamente dentro da ordem de magnitude do erro de arredondamento. Em outras palavras, se as duas únicas fontes de erro são o erro de poluição numérica e o erro de truncamento, então o acoplamento deles na solução numérica determina o erro verdadeiro nulo, ou seja, $E(\psi_i) = 0$. Na prática, não é necessário utilizar infinitos termos do erro de truncamento porque a magnitude do erro de arredondamento é em torno de 1,0E - 32.

O erro de poluição numérica pode gerar uma solução numérica com melhores resultados do que o previsto, e isso só pode ser observado quando é realizada a verificação das soluções numéricas por meio dos testes de coerência.

3.6.7 Erro de programação

O erro de programação (ε_p) relatado por (ROACHE, 1998) é caracterizado por:

- 1. Uso incorreto de um método numérico para discretizar o modelo matemático;
- 2. Implementação incorreta do código computacional
- 3. Utilização incorreta do código computacional para geração de resultados;
- 4. Outras fontes.

3.7 Verificação de soluções numéricas

Define-se como erro verdadeiro (ou erro numérico), a diferença entre a solução analítica Ψ e numérica ψ , conforme equação (3.74) (MARCHI, 2001).

$$E = \Psi - \psi. \tag{3.74}$$

A ordem efetiva equivalente (p_E) baseada em duas soluções numéricas, caracteriza a inclinação da reta correspondente ao decaimento do erro *E versus h*. Ela pode ser avaliada por meio da equação (3.75),

$$p_E = \frac{\log\left(\frac{|E_1|}{|E_2|}\right)}{\log\left(\frac{h_1}{h_2}\right)}.$$
(3.75)

Nos casos onde a solução analítica é desconhecida, pode-se avaliar a ordem de acurácia por meio da ordem aparente equivalente (p_U) , equação (3.76), que combina três soluções numéricas em sua formulação.

$$p_U = \frac{\log\left(\frac{|\psi_2 - \psi_1|}{|\psi_3 - \psi_2|}\right)}{\log\left(\frac{h_1}{h_2}\right)},\tag{3.76}$$

onde ψ_1 é a solução obtida com a discretização mais grosseira com h_1 , ψ_2 a intermediária com h_2 , e ψ_3 a solução numérica obtida com a discretização mais refinada com h_3 .

3.8 Variáveis globais

As variáveis globais, que constituem o estágio secundário são a temperatura média (ψ_m) , as normas L_1 , $L_2 \in L_{\infty}$.

3.8.1 Temperatura média

Realiza-se o cálculo da temperatura média (ψ_m) pela regra do trapézio (KREYS-ZIG, 1999), conforme descrito na equação (3.77),

$$\psi_m = \frac{h}{2L} \sum_{i=2}^{N} (\psi_{i-1} + \psi_i).$$
(3.77)

3.8.2 Normas de vetores

O cálculo das normas de vetores pode ser realizado de acordo com (NOBLE; DANIEL, 1986; BURDEN; FAIRES, 2005). Seja E um vetor de dimensão n. Então define-se,

$$L_1 = \sum_{i=1}^{n} |E_i|. \tag{3.78}$$

$$L_2 = \sqrt{\sum_{i=1}^{n} |E_i|^2}.$$
(3.79)

$$L_{\infty} = \max_{1 \le i \le n} |E_i|. \tag{3.80}$$

3.8.3 Normas de matrizes

O cálculo das normas de matrizes pode ser realizado de acordo com (NOBLE; DANIEL, 1986; BURDEN; FAIRES, 2005). Seja \mathbb{A} uma matriz $n \times n$. Então define-se,

$$L_{1} = \max_{x \neq 0} \left\{ \frac{\|\mathbb{A}\mathbf{x}\|_{1}}{\|\mathbf{x}\|_{1}} \right\},$$
(3.81)

onde

$$\|\mathbb{A}\mathbf{x}\|_{1} = \max_{1 \le j \le n} \sum_{i=1}^{n} |a_{ij}|.$$
(3.82)

$$L_{2} = \max_{x \neq 0} \left\{ \frac{\|\mathbb{A}\mathbf{x}\|_{2}}{\|\mathbf{x}\|_{2}} \right\},$$
(3.83)

sendo

$$\|\mathbf{A}\mathbf{x}\|_2 = \text{valor singular máximo de } \mathbf{A}. \tag{3.84}$$

$$L_{\infty} = \max_{x \neq 0} \left\{ \frac{\|\mathbb{A}\mathbf{x}\|_{\infty}}{\|\mathbf{x}\|_{\infty}} \right\},\tag{3.85}$$

е

$$\|\mathbb{A}\mathbf{x}\|_{\infty} = \max_{1 \le i \le n} \sum_{j=1}^{n} |a_{ij}|.$$
(3.86)

3.9 Multiextrapolação de Richardson

A Repeated Richardson Extrapolation (RRE) (ROACHE; KNUPP, 1993; MARCHI et al., 2013a; MARCHI et al., 2013b; MARCHI; GERMER, 2013; MARCHI; GIACOMINI; SANTIAGO, 2016) é uma técnica de pós-processamento utilizada para diminuir o erro de discretização de soluções numéricas sem a necessidade de aumentar o número de termos na equação discreta do método numérico, geralmente obtida por meio da série de Taylor.

A inovação neste trabalho considera a aplicação da RRE em soluções obtidas com o método SPH. Sabendo-se que ela sempre foi aplicada em simulações com métodos clássicos que utilizam malhas, adota-se este esquema com a mesma finalidade, diminuir o erro de discretização. Entretanto, agora o método numérico não necessita de malha, pois a discretização do domínio computacional é realizada apenas com partículas em uma formulação puramente lagrangiana.

A RRE clássica combina soluções de diferentes malhas e a cada extrapolação a solução obtida apresenta um erro de discretização menor. Quando utiliza-se o FDM, por exemplo, essa grandeza obedece as ordens verdadeiras da equação de diferenças usada para aproximar as derivadas da equação diferencial, mas também pode ser baseada em ordens aparentes equivalentes.

Sem prejuízo algum, substitui-se as malhas por diversos níveis discretizações por partículas que tenham o mesmo perfil, adotando-se a razão de refino constante.

Neste caso, utiliza-se como referência o modelo de discretização CDS-2 por FDM onde $p_L = 2$, com ordens verdadeiras $p_V = \{2,4,6,8,\ldots\}$ deduzidas *a priori*, de modo a obter o máximo possível de extrapolações. A expressão da RRE é

$$\psi_{g,m} = \psi_{g,m-1} + \frac{\psi_{g,m-1} - \psi_{g-1,m-1}}{q^{p_{m-1}} - 1},$$
(3.87)

onde ψ é a solução numérica da variável de interesse, g é a malha/nível de que a solução numérica foi encontrada, m é o número de extrapolações de Richardson e p_m são as

ordens verdadeiras da equação do erro de discretização e $q = h_{g-1}/h_g$ é a razão de refino (MARCHI *et al.*, 2013a).

Generalizando a equação (3.87) para o caso em que as ordens aparentes são aplicadas, obtém-se as equações (3.88a)-(3.88b). Este procedimento da RRE encontra-se descrito no ALGORITMO 6.

$$(p_U)_{g,m} = \frac{\log\left(\frac{|\psi_{g-1,m} - \psi_{g-2,m}|}{|\psi_{g,m} - \psi_{g-1,m}|}\right)}{\log(q)}$$
(3.88a)

$$\psi_{g,m} = \psi_{g,m-1} + \frac{\psi_{g,m-1} - \psi_{g-1,m-1}}{q^{(p_U)_{g,m-1}} - 1}.$$
(3.88b)

Para a equação (3.88a), a varição entre o número de discretizações e extrapolações pode ser descrita por $3 \leq g \leq G$ e $0 \leq m \leq [$ parte inteira (g-3)/2], onde G é o número máximo de discretizações. Enquanto isto, na equação (3.88b) m = 0 não se aplica, resultando em $\psi_{g,0}$ que são as soluções não-extrapoladas. A partir disso, $3 \leq g \leq G$ e $1 \leq m \leq [$ parte inteira (g-1)/2]. Para aplicar a RRE utilizando as ordens verdadeiras deduzidas *a priori*, basta substituir p_U por p_V nas equações ((3.88a)) e ((3.88b)) e definir a nova variação $2 \leq g \leq G$.

Para diferenciar as ordens de acurácia deduzidas *a priori* das obtidas *a posteriori*, usa-se a seguinte padronização:

- $p_V \coloneqq$ conjunto de ordens de acurária verdadeiras, deduzidas *a priori*;
- p_U := conjunto de ordens de acurária aparentes equivalentes, obtido a posteriori, que pode ser calculado mesmo se a solução analítica for desconhecida;
- $p_E \coloneqq$ conjunto de ordens de acurária efetivas equivalentes, obtido *a posteriori*, que pode ser calculado apenas se a solução analítica for conhecida;
- os elementos de p_U e p_E são $\{p_0, p_1, p_2, \ldots, p_m\};$
- $p_L := \min\{p_V\}$, onde p_L é chamada de ordem assintótica deduziada *a priori*;
- $p_0 \coloneqq \min\{p_U\} \in p_0 \coloneqq \min\{p_E\}$, onde p_0 é chamada de ordem assintótica equivalente, obtida *a posteriori*;
- o min $\{p_V\}$, min $\{p_U\}$ e min $\{p_E\}$ são ordens assintóticas equivalentes;
- $\Delta m \coloneqq$ incremento constante das ordens de acurácia $(p_m p_{m-1} \forall m \ge 1)$.

A FIGURA 17 apresenta o fluxograma para determinação das soluções numéricas da equação da difusão de calor em regime permanente 2D.



FIGURA 17 – FLUXOGRAMA DA SIMULAÇÃO.

Algoritmo 4: GAUSS-SEIDEL-S CSR PARALELIZADO.

```
Entrada: AA, JA, IA, AD, f, tol, N_{ti}, itemax
   Saída: \psi, erro, erro1, ite
 1 alocar: res_{L2}(1:itemax), erro1(1:itemax)
 2 ite = 0
 \mathbf{s} \ L_2 sum = 0
 4 res_{L2} = \vec{0}
 5 res00_{L2} = 0
 erro = 0
 \mathbf{7} \ erro1 = \vec{0}
 8 início
       enquanto erro >= tol faça
 9
            ite = ite + 1
\mathbf{10}
            !$OMP BARRIER
11
            para k = 1, \ldots, N_{ti} (realizado em paralelo por cada thread) faça
\mathbf{12}
                cont = IA(k) - 1
\mathbf{13}
                c = IA(k+1) - IA(k)
\mathbf{14}
                s = 0
15
                para i = 1, \ldots, c faça
16
                     cont = cont + 1
\mathbf{17}
                     j = JA(cont)
\mathbf{18}
                     se j > k então
19
                      s = s + AA(cont)\psi_{old}(j)
\mathbf{20}
                     fim
\mathbf{21}
                     se j < k \ e \ k \neq AD(k) então
22
                      s = s + AA(cont)\psi(j)
23
                     fim
\mathbf{24}
                fim
25
                 \psi(k) = [1/AA(AD(k))][f(k) - s]
26
                 L_2 sum = L_2 sum + [\psi(k) - \psi_{old}(k)]^2
\mathbf{27}
                 \psi_{old}(k) = \psi(k)
\mathbf{28}
            fim
\mathbf{29}
            região serial realizado por uma única thread faça
30
                 !$OMP SINGLE
\mathbf{31}
                 res_{L2}(ite) = \sqrt{L_2 sum}
\mathbf{32}
                res00_{L2} = res_{L2}(1)
33
                erro = res_{L2}(ite)/res00_{L2}
34
                erro1(ite) = erro
35
                 L_2 sum = 0
36
                 !$OMP END SINGLE
\mathbf{37}
            fim
38
            se ite == itemax então
39
                pare
40
            senão
\mathbf{41}
                continue
\mathbf{42}
            fim
43
       fim
44
45 fim
```

Algoritmo 5: ESQUEMA CS COM AMG UTILIZANDO DUAS MA-LHAS.

Entrada: ψ_0^h, A_h, f^h Saída: ψ^h 1 Ida - Restrição 2 início Iterar ν_1 vezes $A_h \psi^h = f^h$ com estimativa inicial ψ_0^h ; 3 Calcular o resíduo em $\Omega^h : r^h = f^h - A_h \psi^h;$ $\mathbf{4}$ Restringir o resíduo para $\Omega^H : r^H = I_h^H r^h;$ Resolver $A_H e^H = r^H$ em Ω^H , com estimativa inicial $\psi_0^H = 0;$ $\mathbf{5}$ 6 7 fim Volta - Prolongação 8 início 9 Interpolar o erro para $\Omega^h : e^h = I_H^h e^H;$ 10 Corrigir a aproximação na malha fina $\psi^h \leftarrow \psi^h + e^h$; 11 Iterar ν_2 vezes $A_h \psi^h = f^h$ com estimativa inicial ψ^h . 1213 fim

Algoritmo 6: MULTIEXTRAPOLAÇÃO DE RICHARDSON (RRE).

Entrada: ψ soluções em G discretizações por partículas, q Saída: ψ 1 início **Definir:** $\psi_{1,0} = \psi_1, \psi_{2,0} = \psi_2, \psi_{3,0} = \psi_3, \dots, \psi_{G,0} = \psi_G$ $\mathbf{2}$ para $q = 3, \ldots, G$ faça 3 para $m = 0, \ldots, [parte inteira (g-1)/2]$ faça $\mathbf{4}$ se $m \ll [parte inteira (q-3)/2]$ então $\mathbf{5}$ $(p_U)_{g,m} = \frac{\log\left(\frac{|\psi_{g-1,m} - \psi_{g-2,m}|}{|\psi_{g,m} - \psi_{g-1,m}|}\right)}{\log(q)}$ 6 fim 7 se m >= 1 então $\psi_{g,m} = \psi_{g,m-1} + \frac{\psi_{g,m-1} - \psi_{g-1,m-1}}{q^{(p_U)_{g,m-1}} - 1}$ 8 9 fim $\mathbf{10}$ fim 11 fim $\mathbf{12}$ 13 fim

4 MODELAGEM MATEMÁTICA

Neste capítulo encontram-se os modelos matemáticos descritos nas equações (4.1) e (4.2), as variáveis de interesse na TABELA 1, as condições de contorno e geometrias escolhidas para o desenvolvimento das simulações numéricas. O domínio computacional para os casos 1D é definido como o intervalo fechado [0,1] em todos os casos, enquanto nos casos 2D as geometrias encontram-se descritas na Seção 4.5.

Os modelos numéricos mostrados nesse capítulo recebem a seguinte codificação:

- Exemplo 01 da difusão de calor 1D em regime permanente (E1-1Dp);
- Exemplo 02 da difusão de calor 1D em regime permanente (E2-1Dp);
- Exemplo 03 da difusão de calor 1D em regime permanente (E3-1Dp);
- Exemplo 01 da difusão de calor 1D em regime transiente (E1-1Dt);
- Exemplo 01 da difusão de calor 2D em regime permanente (E1-2Dp);
- Exemplo 02 da difusão de calor 2D em regime permanente (E2-2Dp).

4.1 Modelos matemáticos

4.1.1 Equação da difusão de calor em regime permanente

$$-\nabla^2 \Psi(\mathbf{x}) = f(\mathbf{x}) \tag{4.1}$$

4.1.2 Equação da difusão de calor em regime transiente

$$\frac{\partial \Psi(\mathbf{x},t)}{\partial t} = \mu \nabla^2 \Psi(\mathbf{x},t) + f(\mathbf{x})$$
(4.2)

4.2 Exemplos Numéricos

4.2.1 Exemplo (E1-1Dp)

O exemplo (E1-1Dp), definido pela equação (4.3), considera como termo fonte uma funçao suave, ou seja, com derivada de qualquer ordem, C^{∞} e sem mudança de sinal (FUKUCHI, 2021).

$$\nabla^2 \Psi(\mathbf{x}) = e^{\mathbf{x}},\tag{4.3}$$

com solução analítica $\Psi(\mathbf{x}) = e^{\mathbf{x}}$, condições de contorno $\Psi(0) = 1$, $\Psi(1) = e$ e temperatura média analítica $\Psi_m = (e - 1)$.

4.2.2 Exemplo (E2-1Dp)

O exemplo (E2-1Dp), definido pela equação (4.4), considera como termo fonte uma função suave de classe C^{∞} e mudança de sinal (FUKUCHI, 2021).

$$\nabla^2 \Psi(\mathbf{x}) = -\pi^2 \operatorname{sen}(\pi \mathbf{x}), \tag{4.4}$$

com solução analítica $\Psi(\mathbf{x}) = \operatorname{sen}(\pi \mathbf{x})$, condições de contorno $\Psi(0) = 0$, $\Psi(1) = 0$ e temperatura média analítica $\Psi_m = 2/\pi$.

4.2.3 Exemplo (E3-1Dp)

O exemplo (E3-1Dp), definido pela equação (4.5), considera como termo fonte uma função suave de classe C^{∞} , sem mudança de sinal e com condição de contorno do tipo Neumann em x = 0 (FUKUCHI, 2021).

$$\nabla^2 \Psi(\mathbf{x}) = e^{\mathbf{x}},\tag{4.5}$$

com solução analítica $\Psi(\mathbf{x}) = e^{\mathbf{x}}$, condições de contorno $\Psi^1(0) = 1$, $\Psi(1) = e$ e temperatura média analítica $\Psi_m = (e - 1)$.

4.2.4 Exemplo (E1-1Dt)

O exemplo (E1-1Dt), definido pela equação (4.6), considera difusividade térmica (μ) constante e fonte de calor nula (MARCHI; SILVA, 2005).

$$\frac{\partial\Psi(\mathbf{x},t)}{\partial t} = \mu\nabla^2\Psi(\mathbf{x},t),\tag{4.6}$$

com solução analítica $\Psi(\mathbf{x},t) = \operatorname{sen}(\pi \mathbf{x})e^{-\pi^2 t}$ (usa-se t = 1,0s), $\mu = 1$, condições de contorno $\Psi(0) = 0$, $\Psi(1) = 0$ e condição inicial $\Psi(\mathbf{x},0) = \operatorname{sen}(\pi \mathbf{x})$; a temperatura média analítica em t = 1,0s é $\Psi_m = (2/\pi)e^{-\pi^2}$.

4.2.5 Exemplo (E1-2Dp)

O exemplo (E1-2Dp), definido pela equação (4.7), considera como termo fonte uma função polinomial (com finitas derivadas) e sem mudança de sinal (SUERO *et al.*, 2012).

$$\nabla^2 \Psi(\mathbf{x}) = -2[(1 - 6x^2)y^2(1 - y^2) + (1 - 6y^2)x^2(1 - x^2)], \qquad (4.7)$$

com solução analítica $\Psi(\mathbf{x}) = (x^2 - x^4)(y^4 - y^2)$, condições de contorno $\Psi(x,0) = \Psi(x,1) = \Psi(0,y) = \Psi(1,y) = 0$ e temperatura média analítica $\Psi_m = -4/225$.

4.2.6 Exemplo (E2-2Dp)

O exemplo (E1-2Dp), definido pela equação (4.8), considera como termo fonte uma função senoidal de classe C^{∞} com mudança de sinal (SHI; CAO; CHEN, 2012).

$$\nabla^2 \Psi(\mathbf{x}) = sen(\pi x)sen(\pi y), \tag{4.8}$$

com solução analítica $\Psi(\mathbf{x}) = -1/(2\pi^2) sen(\pi x) sen(\pi y)$, condições de contorno $\Psi(x,0) = \Psi(x,1) = \Psi(0,y) = \Psi(1,y) = 0$ e temperatura média analítica $\Psi_m = -2/\pi^4$.

4.3 Variáveis de interesse

Na TABELA 1 as variáveis $\Psi(1/2)$ e $\psi(1/2)$ representam a temperatura no ponto médio do domínio. A rigor, por tratar-se de discretização por partículas, a temperatura é calculada sobre o ponto material médio (partícula na coordenada do ponto médio x = 1/2). Em casos bidimensionais, as variáveis são descritas como $\Psi(1/2,1/2)$ e $\psi(1/2,1/2)$. As variáveis Ψ_m e ψ_m representam a temperatura média, que consideram o valor da temperatura em todos os pontos materiais. Enquanto isso, as avariáveis Ψ^{γ} e ψ^{γ} representam as derivadas das funções Ψ e ψ , respectivamente (condição de contorno do tipo Neumann).

TABELA 1 – DEFINIÇÃO DAS VARIÁVEIS DE INTERESSE.

Tipo de variável	Analítica	Numérica	Operação	Estágio da variável			
Dependente	Ψ	ψ	local	primário			
Dependente	$\Psi(1/2)$	$\psi(1/2)$	local	primário			
Média da variável dependente	Ψ_m	ψ_m	global	secundário			
Derivada da variável dependente	Ψ^{γ}	ψ^γ	local	secundário			

FONTE: O autor (2022).

4.4 Condições de contorno

Para que seja possível resolver numericamente um modelo matemático de equação diferencial, necessita-se que exista uma única solução e que dependa continuamente das condições de contorno e/ou iniciais (FORTUNA, 2012). Por outro lado, problemas modelados por sistemas de equações não lineares geralmente possuem mais de uma solução. Nesta pesquisa, escolheu-se modelos matemáticos que possuem solução analítica conhecida e única.

Em problemas modelados por equações diferenciais parciais, pode-se especificar três tipos de condições sobre uma variável dependente Ψ na fronteira $\partial \Omega$ de uma região Ω .

1. Condição de contorno do tipo *Dirichlet*: o valor de Ψ é prescrito em $\partial \Omega$;

- 2. Condição de contorno do tipo Neumann: o gradiente normal $\partial \Psi / \partial n = f$ ou tangencial $\partial \Psi / \partial s = g$ é prescrito em $\partial \Omega$;
- 3. Condição de contorno do tipo *Robin*: uma combinação dos dois casos anteriores é prescrita em $\partial \Omega$; por exemplo, $\beta_1(\partial \Psi/\partial n) + \beta_2 \Psi = f$.

Escolheu-se para as análises desta pesquisa problemas que envolvam condições de contorno do tipo *Dirichlet* e *Neumann*. Em matemática, a condição de contorno de *Dirichlet* refere-se ao caso em que o valor de uma variável de interesse encontra-se prescrito no contorno. Os casos mais realistas, como os problemas da termodinâmica, a condição de contorno de *Neumann* representa a taxa de condução de calor sobre a fronteira (BERGMAN *et al.*, 2011).

4.5 Tipos de geometrias

Resolvem-se os problemas unidimensionais sobre a reta no intervalo fechado [0,1], enquanto os bidimensionais no quadrado unitário $[0,1] \times [0,1]$, conforme FIGURA 18.

FIGURA 18 – REPRESENTAÇÃO DA GEOMETRIA EM FORMATO DE QUADRADO UNI-TÁRIO.



5 RESULTADOS

5.1 Erro numérico com SPH

5.1.1 Estimativas do erro de discretização a priori

Estima-se o erro de discretização *a priori* tendo como base o erro de truncamento local (ε_{τ}) (LEVEQUE, 2007) obtido ao discretizar o modelo matemático com um método numérico que geralmente advém de expressões combinadas da expansão em série de Taylor.

Admitindo-se que o erro de discretização $(E(\psi))$ é gerado exclusivamente pelo erro de truncamento local, obtém-se

$$E(\psi) = \varepsilon_{\tau} = \overline{C} [kh(\mathbf{x} - \mathbf{x}')]^p, \qquad (5.1)$$

onde \overline{C} é um coeficiente constante desconhecido que independe de $h = \Delta x$, p é a ordem do erro de discretização e Δx é o tamanho do elemento de malha utilizado para discretizar o domínio (base para distribuição das partículas no domínio). Em outras palavras,

$$\lim_{h \to 0} E(\psi) = \lim_{h \to 0} \overline{C} [kh(\mathbf{x} - \mathbf{x}')]^p \longrightarrow 0.$$
(5.2)

Para facilitar a compreensão quanto ao decaimento do erro de discretização, adotase genericamente uma aproximação SPH qualquer de segunda ordem (p = 2), e ainda, dois níveis de discretização uniforme, com comprimentos de suavização h_1 e h_2 para o nível inconsistente e intermediário (equivalente a malha grossa e imediatamente mais fina, ou seja, com o dobro de nós), respectivamente. Dessa forma, como observa-se na equação (5.3), é possível estabelecer a seguinte relação

$$\frac{E(\psi_1)}{E(\psi_2)} = \frac{\overline{C}_1[kh_1(\mathbf{x} - \mathbf{x}')]^p}{\overline{C}_1[kh_2(\mathbf{x} - \mathbf{x}')]^p} = \frac{\overline{C}_1[kh_1(\mathbf{x} - \mathbf{x}')]^p}{\overline{C}_1\left[\frac{kh_1}{2}(\mathbf{x} - \mathbf{x}')\right]^p} = 2^p = 2^2 = 4.$$
(5.3)

Isto significa que o erro de discretização fica quatro vezes menor a cada refinamento ao considerar $h_2 = h_1/2$. Note que a inversão $E(\psi_2)/E(\psi_1) = 1/4$ também é verdadeira.

É importante destacar que ao se tratar de uma discretização de domínio utilizando o método SPH, as partículas podem ser originalmente distribuídas com espaçamento uniforme, equivalente ao que acontece no FDM (GINGOLD; MONAGHAN, 1977; LUCY, 1977).

5.1.2 Estimativas do erro de discretização *a posteriori*

Tendo em vista o trabalho de revisão bibliográfica de (MARCHI, 2001), sabe-se que é possível estimar o erro de discretização *a posteriori* por formas diferentes, sendo que

considera-se dois grupos. O primeiro considera a análise apenas sobre a solução numérica em malha única, como por exemplo o FEM (ZHU; ZIENKIEWICZ, 1990). O segundo por sua vez, considera duas ou mais soluções obtidas em diferentes malhas com FDM e FVM (HORTMANN; PERIĆ; SCHEUERER, 1990).

5.1.3 Erro de truncamento

As análises do erro de truncamento local (ε_{τ}) características nos métodos FDM e FVM (FERZIGER; PERIĆ, 2002) podem ser estendidas ao método SPH (QUINLAN; BASA; LASTIWKA, 2006), conforme equação (5.4).

$$\varepsilon_{\tau}(\psi) = \overline{C}_1 \left[kh(\mathbf{x} - \mathbf{x}') \right]^{p_L} + \overline{C}_2 \left[kh(\mathbf{x} - \mathbf{x}') \right]^{p_2} + \overline{C}_3 \left[kh(\mathbf{x} - \mathbf{x}') \right]^{p_3} + \ldots + \overline{C}_m \left[kh(\mathbf{x} - \mathbf{x}') \right]^{p_m}$$
(5.4)

A expressão (5.4) é denominada equação geral do erro de truncamento. As potências $p_L < p_2 < p_3 < \ldots < p_m$ são as ordens verdadeiras apenas dos termos não nulos na equação do erro de truncamento. Estas ordens verdadeiras são números inteiros positivos que geralmente caracterizam uma série aritmética sendo, $p_L \ge 1$ a menor delas e chamada de ordem assintótica. Dessa forma, pode-se escrever a solução analítica exata como a soma entre a solução numérica e o erro de truncamento, de acordo com equação (5.5),

$$\Psi(\mathbf{x}_i) = \psi(\mathbf{x}_i) + \varepsilon_\tau(\psi(\mathbf{x}_i)). \tag{5.5}$$

5.1.3.1 Discretização do operador laplaciano 1D

A expressão para o operador laplaciano utilizada nessa tese é obtida a partir da expansão em série de Taylor. Para uma função $\Psi(\mathbf{x}')$, a série de Taylor até o γ -ésimo termo envolvendo γ -ésima derivada, ao redor da partícula \mathbf{x} , é dada por

$$\Psi(\mathbf{x}') = \Psi(\mathbf{x}) + (\mathbf{x}' - \mathbf{x})\Psi^{1}(\mathbf{x}) + \frac{(\mathbf{x}' - \mathbf{x})^{2}}{2!}\Psi^{2}(\mathbf{x}) + \frac{(\mathbf{x}' - \mathbf{x})^{3}}{3!}\Psi^{3}(\mathbf{x}) +$$

+
$$\frac{(\mathbf{x}' - \mathbf{x})^4}{4!} \Psi^4(\mathbf{x}) + \frac{(\mathbf{x}' - \mathbf{x})^5}{5!} \Psi^5(\mathbf{x}) + \frac{(\mathbf{x}' - \mathbf{x})^6}{6!} \Psi^5(\mathbf{x}) + \frac{(\mathbf{x}' - \mathbf{x})^7}{7!} \Psi^7(\mathbf{x}) + \frac{(\mathbf{x}' - \mathbf{x})^8}{6!} \Psi^5(\mathbf{x}) + \frac{(\mathbf{x}' - \mathbf{x})^7}{7!} \Psi^7(\mathbf{x})$$

+
$$\frac{\gamma}{8!} \Psi^{(\mathbf{x})} + \ldots + \frac{\gamma}{\gamma - 1!} \Psi^{(-1)}(\mathbf{x}) + \frac{\gamma}{\gamma!} \Psi^{(-1)}(\mathbf{x}) + \frac{\gamma}{\gamma!} \Psi^{(-1)}(\mathbf{x})$$
 +

$$+ O(\mathbf{x}' - \mathbf{x})^{\gamma+1}.$$
(5.6)

Integrando termo a termo da equação (5.6) em relação às partículas do suporte compacto, obtém-se

$$\begin{split} \underbrace{\int_{\Omega} \Psi(\mathbf{x}') \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}'}_{1} &= \underbrace{\int_{\Omega} \Psi(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}'}_{2} + \\ &+ \underbrace{\int_{\Omega} (\mathbf{x}'-\mathbf{x}) \Psi^{1}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}'}_{3} + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{2}}{2!} \Psi^{2}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}'}_{4} + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{3}}{3!} \Psi^{3}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}'}_{5} + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{3}}{3!} \Psi^{4}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}'}_{7} + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{5}}{5!} \Psi^{5}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}'}_{7} + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{6}}{6!} \Psi^{6}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}'}_{8} + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{7}}{7!} \Psi^{7}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}'}_{10} + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{8}}{8!} \Psi^{8}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}' + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{7}}{7!} \Psi^{7}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}' + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{7}}{7!} \Psi^{7}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}' + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{7}}{7!} \Psi^{7}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}' + \\ &+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}'-\mathbf{x})^{7}}{7!} \Psi^{7}(\mathbf{x}) \nabla_{\mathbf{x}'} W(\mathbf{x}'-\mathbf{x},h) d\mathbf{x}' + \\ &+ O(\mathbf{x}'-\mathbf{x})^{\gamma+1}. \end{split}$$

As integrais 3, 5, 7 e 9 do lado direito da equação (5.7) são nulas devido à propriedade de simetria do núcleo. Dessa forma, multiplicando os termos restantes por $\Delta_{\mathbf{x}}$, onde $\Delta_{\mathbf{x}} = (\mathbf{x} - \mathbf{x}')/||\mathbf{x} - \mathbf{x}'||^2$, obtém-se

$$\underbrace{\int_{\Omega} \Psi(\mathbf{x}') \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{1} = \underbrace{\int_{\Omega} \Psi(\mathbf{x}) \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{2} + \\
+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}' - \mathbf{x})^{2}}{2!} \Psi^{2}(\mathbf{x}) \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{4} + \\
+ \underbrace{\int_{\Omega} \frac{(\mathbf{x}' - \mathbf{x})^{4}}{4!} \Psi^{4}(\mathbf{x}) \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{6} + \\
+ \dots + \\
+ \int_{\Omega} \frac{(\mathbf{x}' - \mathbf{x})^{\gamma}}{n!} \Psi^{\gamma}(\mathbf{x}) \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}' + \\
+ O(\mathbf{x}' - \mathbf{x})^{\gamma+1}.$$
(5.8)

Agrupando as integrais 1 e 2 no lado esquerdo e retirando as derivadas de $\Psi({\bf x})$ de dentro das integrais, pois são constantes, obtém-se

$$\underbrace{\int_{\Omega} [\Psi(\mathbf{x}') - \Psi(\mathbf{x})] \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{1} = \frac{\Psi^{2}(\mathbf{x})}{2!} \underbrace{\int_{\Omega} (\mathbf{x}' - \mathbf{x})^{2} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{4} + \\
+ \frac{\Psi^{4}(\mathbf{x})}{4!} \underbrace{\int_{\Omega} (\mathbf{x}' - \mathbf{x})^{4} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{6} + \\
+ \frac{\Psi^{6}(\mathbf{x})}{6!} \underbrace{\int_{\Omega} (\mathbf{x}' - \mathbf{x})^{6} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{8} + \\
+ \dots + \\
+ \frac{\Psi^{\gamma}(\mathbf{x})}{\gamma!} \int_{\Omega} (\mathbf{x}' - \mathbf{x})^{\gamma} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}' + \\
+ O(\mathbf{x}' - \mathbf{x})^{\gamma+1}.$$
(5.9)

Após algumas manipulações algébricas, determina-se

$$\underbrace{\int_{\Omega} [\Psi(\mathbf{x}') - \Psi(\mathbf{x})] \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{1} = \underbrace{\Psi^{2}(\mathbf{x})}_{2!} \underbrace{\int_{\Omega} (\mathbf{x}' - \mathbf{x})^{2} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{4} + \\
+ \underbrace{\Psi^{4}(\mathbf{x})}_{4!} \underbrace{\int_{\Omega} (\mathbf{x}' - \mathbf{x})^{4} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{6} + \\
+ \underbrace{\Psi^{6}(\mathbf{x})}_{6!} \underbrace{\int_{\Omega} (\mathbf{x}' - \mathbf{x})^{6} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}'}_{8} + \\
+ \dots + \\
+ \underbrace{\Psi^{\gamma}(\mathbf{x})}_{\gamma !} \int_{\Omega} (\mathbf{x}' - \mathbf{x})^{\gamma} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h) d\mathbf{x}' + \\
+ O(\mathbf{x}' - \mathbf{x})^{\gamma + 1}.$$
(5.10)

A integral 4 é igual a -1 (-1 para $\nabla_{\mathbf{x}'}W$ e 1 para $\nabla_{\mathbf{x}}W$) devido à normalização do núcleo. Outra propriedade importante é a do gradiente do núcleo, ela diz que $\nabla_{\mathbf{x}}W = -\nabla_{\mathbf{x}'}W$. Nota-se que o gradiente da integral 1 está relacionado a \mathbf{x}' e não a \mathbf{x} . Dessa forma, ao efetuar a mudança, tem-se

$$2\int_{\Omega} [\Psi(\mathbf{x}) - \Psi(\mathbf{x}')] \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' = \Psi^{2}(\mathbf{x}) + + \frac{\Psi^{4}(\mathbf{x})}{12} \int_{\Omega} (\mathbf{x} - \mathbf{x}')^{4} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + + \frac{\Psi^{6}(\mathbf{x})}{360} \int_{\Omega} (\mathbf{x} - \mathbf{x}')^{6} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + + \frac{\Psi^{8}(\mathbf{x})}{20160} \int_{\Omega} (\mathbf{x} - \mathbf{x}')^{8} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + + \dots + + \frac{\Psi^{\gamma}(\mathbf{x})}{\gamma!/2} \int_{\Omega} (\mathbf{x} - \mathbf{x}')^{\gamma} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + + O(\mathbf{x} - \mathbf{x}')^{\gamma+1}.$$
(5.11)

Isolando-se o termo da derivada de segunda ordem na equação (5.11), obtém-se

$$\begin{split} \Psi^{2}(\mathbf{x}) &= 2 \int_{\Omega} [\Psi(\mathbf{x}) - \Psi(\mathbf{x}')] \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + \\ &- \frac{\Psi^{4}(\mathbf{x})}{12} \int_{\Omega} (\mathbf{x} - \mathbf{x}')^{4} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + \\ &- \frac{\Psi^{6}(\mathbf{x})}{360} \int_{\Omega} (\mathbf{x} - \mathbf{x}')^{6} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + \\ &- \frac{\Psi^{8}(\mathbf{x})}{2160} \int_{\Omega} (\mathbf{x} - \mathbf{x}')^{8} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + \\ &- \dots - \\ &- \frac{\Psi^{\gamma}(\mathbf{x})}{\gamma!/2} \int_{\Omega} (\mathbf{x} - \mathbf{x}')^{\gamma} \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + \\ &- O(\mathbf{x} - \mathbf{x}')^{\gamma+1}. \end{split}$$
(5.12)

Sendo assim, a aproximação para o operador laplaciano 1D é dada por

$$\nabla^2 \Psi(\mathbf{x}) = 2 \int_{\Omega} [\Psi(\mathbf{x}) - \Psi(\mathbf{x}')] \Delta_{\mathbf{x}} \nabla_{\mathbf{x}} W(\mathbf{x} - \mathbf{x}', h) d\mathbf{x}' + O(\mathbf{x} - \mathbf{x}')^2.$$
(5.13)

A integral na equação (5.13) pode ser representada na forma de somatório,

$$\nabla^2 \Psi(\mathbf{x}_i) = \sum_{j \in V_i} 2 \frac{m_j}{\rho_j} (\Psi(\mathbf{x}_i) - \Psi(\mathbf{x}_j)) \Delta_{\mathbf{x}_{ij}} \nabla_i W(\mathbf{x}_{ij}; h) + O(\mathbf{x}_i - \mathbf{x}_j)^2, \qquad (5.14)$$

onde $d\mathbf{x}' = m_j/\rho_j = h$ e $\nabla_i W(\mathbf{x}_{ij}; h) = (\alpha_d/h^{d+1})W^1(\phi_{ij})[(\mathbf{x}_i - \mathbf{x}_j)/||(\mathbf{x}_i - \mathbf{x}_j)||]$. Dessa forma, a equação geral do erro de truncamento é dada por

$$\varepsilon_{\tau}(\psi(\mathbf{x})) = \sum_{\gamma=2\zeta+2}^{+\infty} -\frac{\Psi^{\gamma}(\mathbf{x})}{\gamma!/2} \int_{\Omega} (\mathbf{x} - \mathbf{x}')^{\gamma} \Delta_{\mathbf{x}} \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||} \frac{\alpha_d}{h^{d+1}} W^1(\phi) d\mathbf{x}' + O(\mathbf{x} - \mathbf{x}')^{\gamma+1}$$
$$= \sum_{\gamma=2\zeta+2}^{+\infty} \frac{\Psi^{\gamma}(\mathbf{x})}{\gamma!/2} h^{\gamma-2} + O(\mathbf{x} - \mathbf{x}')^{\gamma},$$
(5.15)

onde $\zeta=1,\!2,\!3,\ldots,$ consequentemente, $\gamma=\{4,\!6,\!8,\ldots\}$ e

$$\int_{\Omega} (\mathbf{x} - \mathbf{x}')^{\gamma} \Delta_{\mathbf{x}} \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||} \frac{\alpha_d}{h^{d+1}} W^1(\phi) d\mathbf{x}' = -(h^{\gamma-2}).$$
(5.16)

5.1.3.2 Aproximação assimétrica para derivada primeira 1D

A aproximação para derivada primeira unidimensional a seguir é uma adaptação do Esquema de Diferenças a Jusante de segunda ordem (em inglês, *Downstream Difference Scheme of second order*, DDS-2) de acurácia (MARCHI, 2001). Essa adaptação foi desenvolvida para que todas as linhas do sistema fossem descritas pelo método SPH. Esse novo esquema pode ser utilizado para aproximar uma função no contorno esquerdo e também no direito, basta mudar o sinal dos coeficientes e aplicá-los na última linha do sistema.

$$\begin{cases}
A(1,2) = -\sum_{j \in V_i} 8 \frac{m_j}{\rho_j} (\psi(\mathbf{x}_i) - \psi(\mathbf{x}_j)) \Delta_{\mathbf{x}_{ij}} \frac{\mathbf{x}_{ij}}{||\mathbf{x}_{ij}||} W^1(\phi;h), se \phi = ||\mathbf{x}_{ij}||/h = 1 \\
A(1,3) = \sum_{j \in V_i} 2 \frac{m_j}{\rho_j} (\psi(\mathbf{x}_i) - \psi(\mathbf{x}_j)) \Delta_{\mathbf{x}_{ij}} \frac{\mathbf{x}_{ij}}{||\mathbf{x}_{ij}||} W^1(\phi - 1;h), se \phi = ||\mathbf{x}_{ij}||/h = 2 \\
A(1,1) = -[A(1,2) + A(1,3)] \\
\frac{\partial \Psi(\mathbf{x}_i)}{\partial x} = \frac{A(1,1)\psi(\mathbf{x}_i) + A(1,2)\psi(\mathbf{x}_{i+1}) + A(1,3)\psi(\mathbf{x}_{i+2})}{2h(N_p)^2} + O(\mathbf{x}_i - \mathbf{x}_j)^2 = f(\mathbf{x}_i) \\
(5.17)
\end{cases}$$

5.1.3.3 Discretização do operador laplaciano 2D

A dedução do operador laplaciano bidimensional utilizada nessa tese é baseada na aproximação de uma função pelo polinômio de Taylor (BURDEN; FAIRES; BURDEN, 2016; FRAGA FILHO, 2019), representado pela equação (5.18), onde $\mathbf{x} = (x,y)$, $\mathbf{x}' = (x',y')$ e P_{γ} é o grau do polinômio de Taylor. Os índices ip e jp dos somatórios da equação (5.18) para $P_{\gamma} = 5$ encontram-se detalhados na TABELA 2.

$$\Psi_{P_{\gamma}}(\mathbf{x}') = \sum_{ip=0}^{P_{\gamma}} \sum_{jp=0}^{P_{\gamma}-ip} \frac{\frac{\partial^{(ip+jp)}}{\partial x^{ip} \partial y^{jp}} \Psi(\mathbf{x})}{ip!jp!} (x'-x)^{ip} (y'-y)^{jp}.$$
(5.18)

ip = 0, jp	ip = 1, jp	ip = 2, jp	ip = 3, jp	ip = 4, jp	ip = 5, jp
ip = 0, jp = 0	ip = 1, jp = 0	ip = 2, jp = 0	ip = 3, jp = 0	ip = 4, jp = 0	ip = 5, jp = 0
ip = 0, jp = 1	ip = 1, jp = 1	ip = 2, jp = 1	ip = 3, jp = 1	ip = 4, jp = 1	
ip = 0, jp = 2	ip = 1, jp = 2	ip = 2, jp = 2	ip = 3, jp = 2		
ip = 0, jp = 3	ip = 1, jp = 3	ip = 2, jp = 3			
ip = 0, jp = 4	ip = 1, jp = 4				
$ip = 0, \ jp = 5$					

FONTE: O autor (2022).

Com base nos índices ip e jp para $P_{\gamma} = 5$ apresentados na TABELA 2, pode-se

determinar uma aproximação para o polinômio de Taylor envolvendo as derivadas de até quinta ordem. A expansão do polinômio de Taylor de grau 5 encontra-se na equação (5.19).

$$\begin{split} \Psi_{5}(\mathbf{x}') &= \frac{\partial^{0}\Psi(\mathbf{x})}{\partial x^{0}\partial y^{0}} \frac{(x'-x)^{0}(y'-y)^{0}}{0!0!} + \frac{\partial^{1}\Psi(\mathbf{x})}{\partial x^{0}\partial y^{1}} \frac{(x'-x)^{0}(y'-y)^{1}}{0!1!} + \\ &+ \frac{\partial^{2}\Psi(\mathbf{x})}{\partial x^{0}\partial y^{2}} \frac{(x'-x)^{0}(y'-y)^{2}}{0!2!} + \frac{\partial^{3}\Psi(\mathbf{x})}{\partial x^{0}\partial y^{3}} \frac{(x'-x)^{0}(y'-y)^{3}}{0!3!} + \\ &+ \frac{\partial^{4}\Psi(\mathbf{x})}{\partial x^{0}\partial y^{4}} \frac{(x'-x)^{0}(y'-y)^{4}}{0!4!} + \frac{\partial^{5}\Psi(\mathbf{x})}{\partial x^{0}\partial y^{5}} \frac{(x'-x)^{0}(y'-y)^{5}}{0!5!} + \\ &+ \frac{\partial^{1}\Psi(\mathbf{x})}{\partial x^{1}\partial y^{0}} \frac{(x'-x)^{1}(y'-y)^{0}}{1!0!} + \frac{\partial^{2}\Psi(\mathbf{x})}{\partial x^{1}\partial y^{1}} \frac{(x'-x)^{1}(y'-y)^{1}}{1!1!} + \\ &+ \frac{\partial^{3}\Psi(\mathbf{x})}{\partial x^{1}\partial y^{2}} \frac{(x'-x)^{1}(y'-y)^{2}}{1!2!} + \frac{\partial^{4}\Psi(\mathbf{x})}{\partial x^{1}\partial y^{1}} \frac{(x'-x)^{2}(y'-y)^{0}}{1!3!} + \\ &+ \frac{\partial^{5}\Psi(\mathbf{x})}{\partial x^{1}\partial y^{4}} \frac{(x'-x)^{2}(y'-y)^{1}}{1!4!} + \frac{\partial^{4}\Psi(\mathbf{x})}{\partial x^{2}\partial y^{0}} \frac{(x'-x)^{2}(y'-y)^{2}}{2!0!} + \\ &+ \frac{\partial^{5}\Psi(\mathbf{x})}{\partial x^{2}\partial y^{1}} \frac{(x'-x)^{2}(y'-y)^{1}}{2!1!} + \frac{\partial^{4}\Psi(\mathbf{x})}{\partial x^{3}\partial y^{0}} \frac{(x'-x)^{3}(y'-y)^{0}}{3!0!} + \\ &+ \frac{\partial^{4}\Psi(\mathbf{x})}{\partial x^{2}\partial y^{1}} \frac{(x'-x)^{3}(y'-y)^{1}}{3!1!} + \frac{\partial^{5}\Psi(\mathbf{x})}{\partial x^{3}\partial y^{0}} \frac{(x'-x)^{3}(y'-y)^{2}}{3!2!} + \\ &+ \frac{\partial^{4}\Psi(\mathbf{x})}{\partial x^{4}\partial y^{0}} \frac{(x'-x)^{4}(y'-y)^{0}}{4!0!} + \frac{\partial^{5}\Psi(\mathbf{x})}{\partial x^{4}\partial y^{1}} \frac{(x'-x)^{4}(y'-y)^{1}}{4!1!} + \\ &+ \frac{\partial^{5}\Psi(\mathbf{x})}{\partial x^{5}\partial y^{0}} \frac{(x'-x)^{5}(y'-y)^{0}}{5!0!} + O[(x'-x)^{6}(y'-y)^{6}]. \end{split}$$

Multiplicando os termos da equação (5.19) por $(\mathbf{x}' - \mathbf{x})/||\mathbf{x}' - \mathbf{x}||^2$ e integrando termo a termo em relação às partículas do suporte compacto na posição \mathbf{x}' , obtém-se

$$\begin{split} \int_{\Omega} \Psi(\mathbf{x}') \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' &= \int_{\Omega} \Psi(\mathbf{x}) \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' &+ \\ &+ \int_{\Omega} \frac{\partial^1 \Psi(\mathbf{x})}{\partial x^0 \partial y^1} \frac{(x'-x)^0 (y'-y)^1}{0! 1!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' &+ \end{split}$$

$$\begin{split} &= \int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^0 \partial y^2} \frac{(x'-x)^0 (y'-y)^2}{0!2!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^0 \partial y^3} \frac{(x'-x)^0 (y'-y)^3}{0!3!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^4 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x'-x)^0 (y'-y)^4}{0!4!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x'-x)^1 (y'-y)^0}{0!5!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^1 \Psi(\mathbf{x})}{\partial x^1 \partial y^0} \frac{(x'-x)^1 (y'-y)^0}{1!0!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^1 \partial y^1} \frac{(x'-x)^1 (y'-y)^1}{1!1!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^1 \partial y^2} \frac{(x'-x)^1 (y'-y)^2}{1!2!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^4 \Psi(\mathbf{x})}{\partial x^1 \partial y^3} \frac{(x'-x)^1 (y'-y)^3}{1!3!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^1 \partial y^4} \frac{(x'-x)^1 (y'-y)^3}{1!3!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x'-x)^2 (y'-y)^0}{2!0!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^2 \partial y^1} \frac{(x'-x)^2 (y'-y)^3}{2!1!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^2 \partial y^2} \frac{(x'-x)^2 (y'-y)^2}{2!2!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^2 \partial y^3} \frac{(x'-x)^2 (y'-y)^2}{2!3!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^2 \partial y^3} \frac{(x'-x)^2 (y'-y)^2}{2!3!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^2 \partial y^3} \frac{(x'-x)^2 (y'-y)^2}{3!0!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^2 \partial y^3} \frac{(x'-x)^3 (y'-y)^0}{3!0!} \frac{(x'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' + \\ &+ \int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^2 \partial y^3} \frac{(x'-x)^2 (y'-y)^2}{3!0!} \frac{(x'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf$$

$$+\int_{\Omega} \frac{\partial^{4}\Psi(\mathbf{x})}{\partial x^{3}\partial y^{1}} \frac{(x'-x)^{3}(y'-y)^{1}}{3!1!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'}Wd\mathbf{x}' + \\ +\int_{\Omega} \frac{\partial^{5}\Psi(\mathbf{x})}{\partial x^{3}\partial y^{2}} \frac{(x'-x)^{3}(y'-y)^{2}}{3!2!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'}Wd\mathbf{x}' + \\ +\int_{\Omega} \frac{\partial^{4}\Psi(\mathbf{x})}{\partial x^{4}\partial y^{0}} \frac{(x'-x)^{4}(y'-y)^{0}}{4!0!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'}Wd\mathbf{x}' + \\ +\int_{\Omega} \frac{\partial^{5}\Psi(\mathbf{x})}{\partial x^{4}\partial y^{1}} \frac{(x'-x)^{4}(y'-y)^{1}}{4!1!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'}Wd\mathbf{x}' + \\ +\int_{\Omega} \frac{\partial^{5}\Psi(\mathbf{x})}{\partial x^{5}\partial y^{0}} \frac{(x'-x)^{5}(y'-y)^{0}}{5!0!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'}Wd\mathbf{x}' + \\ +O[(x'-x)^{6},(y'-y)^{6}].$$

$$(5.20)$$

Nota-se na equação (5.20) que a expressão $\nabla_{\mathbf{x}'} W$ é equivalente a $\nabla_{\mathbf{x}'} W(\mathbf{x}' - \mathbf{x}, h)$. Isolando os termos de derivada de ordem zero na equação (5.20), determina-se a equação (5.21).

$$\begin{split} \int_{\Omega} [\Psi(\mathbf{x}') - \Psi(\mathbf{x})] \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' &= \underbrace{\int_{\Omega} \frac{\partial^1 \Psi(\mathbf{x})}{\partial x^0 \partial y^1} \frac{(x' - x)^0 (y' - y)^1}{0!1!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^0 \partial y^2} \frac{(x' - x)^0 (y' - y)^2}{0!2!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^3 \Psi(\mathbf{x})}{\partial x^0 \partial y^3} \frac{(x' - x)^0 (y' - y)^3}{0!3!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^4 \Psi(\mathbf{x})}{\partial x^0 \partial y^4} \frac{(x' - x)^0 (y' - y)^4}{0!4!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x' - x)^0 (y' - y)^5}{0!5!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x' - x)^0 (y' - y)^5}{0!5!} \frac{(\mathbf{x}' - \mathbf{x})^2}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x' - x)^0 (y' - y)^5}{0!5!} \frac{(x' - \mathbf{x})^2}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x' - x)^0 (y' - y)^5}{0!5!} \frac{(x' - \mathbf{x})^2}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x' - x)^0 (y' - y)^5}{0!5!} \frac{(x' - \mathbf{x})^2}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x' - x)^0 (y' - y)^5}{0!5!} \frac{(x' - \mathbf{x})^2}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x' - x)^0 (y' - y)^5}{0!5!} \frac{(x' - \mathbf{x})^2}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x' - x)^0 (y' - y)^5}{0!5!} \frac{(x' - x)^2 (x' - x)^2}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{1} + \\ &+ \underbrace{\int_{\Omega} \frac{\partial^5 \Psi(\mathbf{x})}{\partial x^0 \partial y^5} \frac{(x' - x)^2 (x' - x)^2 (x' - x)^2}{0!5!} \frac{(x' - x)^2 (x' - x)^2 (x' - x)^2}{0!5!} \frac{(x' - x)^2 (x' - x)^2 (x' - x)^2}{0!5!} \frac{(x' - x)^2 (x' - x)^2 (x' - x)^2}{0!5!} \frac{(x' - x)^2 (x' - x)^2 (x' - x)^2}{0!5!} \frac{(x' - x)^2 (x' - x)^2 (x' - x)^2}{0!5!} \frac{(x' - x)^2 (x' - x)^2}{0!5!}$$

$$+ \underbrace{\int_{\Omega} \frac{\partial^{1} \Psi(\mathbf{x})}{\partial x^{1} \partial y^{0}} \frac{(x'-x)^{1}(y'-y)^{0}}{1!0!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{6}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{2} \Psi(\mathbf{x})}{\partial x^{1} \partial y^{1}} \frac{(x'-x)^{1}(y'-y)^{1}}{1!1!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{7}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{3} \Psi(\mathbf{x})}{\partial x^{1} \partial y^{2}} \frac{(x'-x)^{1}(y'-y)^{2}}{1!2!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{8}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{4} \Psi(\mathbf{x})}{\partial x^{1} \partial y^{3}} \frac{(x'-x)^{1}(y'-y)^{3}}{1!3!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{9}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{1} \partial y^{4}} \frac{(x'-x)^{1}(y'-y)^{4}}{1!4!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{10}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{2} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{0}} \frac{(x'-x)^{2}(y'-y)^{0}}{2!0!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{12}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{3} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{1}} \frac{(x'-x)^{2}(y'-y)^{1}}{2!1!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{12}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{4} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{1}} \frac{(x'-x)^{2}(y'-y)^{2}}{2!2!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{13}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{2}} \frac{(x'-x)^{2}(y'-y)^{3}}{2!2!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{14}^{0} + \\ \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{2}} \frac{(x'-x)^{2}(y'-y)^{3}}{2!2!} \frac{(\mathbf{x}'-\mathbf{x})}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{14}^{0} + \\ \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{2}} \frac{(x'-x)^{2}(y'-y)^{3}}{2!3!} \frac{(\mathbf{x}'-\mathbf{x})^{2}}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{14}^{0} + \\ \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{3}} \frac{(x'-x)^{2}(y'-y)^{3}}{2!3!} \frac{(\mathbf{x}'-\mathbf{x})^{2}}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{14}^{0} + \\ \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{3}} \frac{(x'-x)^{2}(y'-y)^{3}}{2!3!} \frac{(x'-\mathbf{x})^{2}}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{14}^{0} + \\ \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{3}} \frac{(x'-x)^{2}(y'-y)^{3}}{2!3!} \frac{(x'-\mathbf{x})^{2}}{|\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{14}^{0} + \\ \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{2} \partial y^{3}} \frac{(x'-x)^{2}(y'-y)$$

$$+ \underbrace{\int_{\Omega} \frac{\partial^{3} \Psi(\mathbf{x})}{\partial x^{3} \partial y^{0}} \frac{(x'-x)^{3} (y'-y)^{0}}{3!0!} \frac{(\mathbf{x}'-\mathbf{x})}{\|\mathbf{x}'-\mathbf{x}\|^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{15}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{4} \Psi(\mathbf{x})}{\partial x^{3} \partial y^{1}} \frac{(x'-x)^{3} (y'-y)^{1}}{3!1!} \frac{(\mathbf{x}'-\mathbf{x})}{\|\mathbf{x}'-\mathbf{x}\|^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{16}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{3} \partial y^{2}} \frac{(x'-x)^{3} (y'-y)^{2}}{3!2!} \frac{(\mathbf{x}'-\mathbf{x})}{\|\mathbf{x}'-\mathbf{x}\|^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{17}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{4} \partial y^{0}} \frac{(x'-x)^{4} (y'-y)^{0}}{4!0!} \frac{(\mathbf{x}'-\mathbf{x})}{\|\mathbf{x}'-\mathbf{x}\|^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{18}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{4} \partial y^{1}} \frac{(x'-x)^{4} (y'-y)^{1}}{4!1!} \frac{(\mathbf{x}'-\mathbf{x})}{\|\mathbf{x}'-\mathbf{x}\|^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{19}^{0} + \\ + \underbrace{\int_{\Omega} \frac{\partial^{5} \Psi(\mathbf{x})}{\partial x^{5} \partial y^{0}} \frac{(x'-x)^{5} (y'-y)^{0}}{5!0!} \frac{(\mathbf{x}'-\mathbf{x})}{\|\mathbf{x}'-\mathbf{x}\|^{2}} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{20}^{0} + \\ + O[(x'-x)^{6}, (y'-y)^{6}]. \tag{5.21}$$

Após a eliminação das integrais nulas da equação (5.21), obtém-se

$$\begin{split} \int_{\Omega} [\Psi(\mathbf{x}') - \Psi(\mathbf{x})] \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}' &= \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^0 \partial y^2} \frac{(x' - x)^0 (y' - y)^2}{0!2!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{2} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^4 \Psi(\mathbf{x})}{\partial x^0 \partial y^4} \frac{(x' - x)^0 (y' - y)^4}{0!4!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{4} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)^0}{2!0!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)^0}{2!0!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)^0}{2!0!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)^0}{2!0!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)^0}{2!0!} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)}{2!0!} \frac{(x' - x)}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)}{2!0!} \frac{(x' - x)}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)}{2!0!} \frac{(x' - x)}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)}{2!0!} \frac{(x' - x)}{||\mathbf{x}' - \mathbf{x}||^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)^2 (y' - y)}{(x' - \mathbf{x})^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)}{(x' - \mathbf{x})^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)}{(x' - \mathbf{x})^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)}{(x' - \mathbf{x})^2} \nabla_{\mathbf{x}'} W d\mathbf{x}'}_{11} &+ \\ &+ \underbrace{\int_{\Omega} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2 \partial y^0} \frac{(x' - x)}{(x' - \mathbf{x})^2} \nabla_{\mathbf{x}'} W d$$

$$+\underbrace{\int_{\Omega} \frac{\partial^{4}\Psi(\mathbf{x})}{\partial x^{4}\partial y^{0}} \frac{(x'-x)^{4}(y'-y)^{0}}{4!0!} \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^{2}} \nabla_{\mathbf{x}'}Wd\mathbf{x}'}_{18} + O[(x'-x)^{6},(y'-y)^{6}].$$
(5.22)

Aplicando-se a mudança $\nabla_{\bf x} W = -\nabla_{\bf x'} W$ e a troca de sinal conforme equações (A.12) e (A.13), obtém-se

$$\begin{split} \int_{\Omega} [\Psi(\mathbf{x}) - \Psi(\mathbf{x}')] \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}' &= \frac{1}{2} \frac{\partial^2 \Psi(\mathbf{x})}{\partial y^2} \underbrace{\int_{\Omega} (y - y')^2 \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{2}^{-1} + \\ &+ \frac{1}{24} \frac{\partial^4 \Psi(\mathbf{x})}{\partial y^4} \underbrace{\int_{\Omega} (y - y')^4 \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{4} &+ \\ &+ \frac{1}{2} \frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2} \underbrace{\int_{\Omega} (x - x')^2 \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{1} &+ \\ &+ \frac{1}{24} \frac{\partial^4 \Psi(\mathbf{x})}{\partial x^4} \underbrace{\int_{\Omega} (x - x')^4 \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{18} &+ \\ &+ O[(x - x')^6, (y - y')^6]. \end{split}$$
(5.23)

Agrupando-se os termos das derivada de segunda ordem,

$$\begin{split} \int_{\Omega} [\Psi(\mathbf{x}) - \Psi(\mathbf{x}')] \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}' &= \frac{1}{2} \left(\frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2} + \frac{\partial^2 \Psi(\mathbf{x})}{\partial y^2} \right) + \\ &+ \frac{1}{24} \frac{\partial^4 \Psi(\mathbf{x})}{\partial y^4} \underbrace{\int_{\Omega} (y - y')^4 \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{4} + \\ &+ \frac{1}{24} \frac{\partial^4 \Psi(\mathbf{x})}{\partial x^4} \underbrace{\int_{\Omega} (x - x')^4 \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{18} + \\ &+ O[(x - x')^6, (y - y')^6]. \end{split}$$
(5.24)

Realizando-se algumas manipulações algébricas na equação (5.24)

$$2\int_{\Omega} [\Psi(\mathbf{x}) - \Psi(\mathbf{x}')] \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}' = \underbrace{\frac{\partial^2 \Psi(\mathbf{x})}{\partial x^2} + \frac{\partial^2 \Psi(\mathbf{x})}{\partial y^2}}_{\nabla^2(\mathbf{x})} + \frac{1}{12} \frac{\partial^4 \Psi(\mathbf{x})}{\partial y^4} \underbrace{\int_{\Omega} (y - y')^4 \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{4} + \frac{1}{12} \frac{\partial^4 \Psi(\mathbf{x})}{\partial x^4} \underbrace{\int_{\Omega} (x - x')^4 \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^2} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{18} + O[(x - x')^6, (y - y')^6].$$

$$(5.25)$$

Dessa forma, ao isolar o termo com as derivadas de segunda ordem na equação (5.25), obtém-se a aproximação para o operador laplaciano com seus respectivos termos do erro de truncamento,

$$\nabla^{2}(\mathbf{x}) = 2 \int_{\Omega} [\Psi(\mathbf{x}) - \Psi(\mathbf{x}')] \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^{2}} \nabla_{\mathbf{x}} W d\mathbf{x}' + - \frac{1}{12} \frac{\partial^{4} \Psi(\mathbf{x})}{\partial y^{4}} \underbrace{\int_{\Omega} (y - y')^{4} \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^{2}} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{4} + - \frac{1}{12} \frac{\partial^{4} \Psi(\mathbf{x})}{\partial x^{4}} \underbrace{\int_{\Omega} (x - x')^{4} \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||^{2}} \nabla_{\mathbf{x}} W d\mathbf{x}'}_{18} + - O[(x - x')^{6}, (y - y')^{6}], \qquad (5.26)$$

onde

$$\begin{split} \varepsilon_{\tau}(\psi(\mathbf{x})) &= -\frac{1}{4!/2} \frac{\partial^4 \Psi(\mathbf{x})}{\partial y^4} \int_{\Omega} (y - y')^4 \Delta_{\mathbf{x}} \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||} \frac{\alpha_d}{h^{d+1}} W^1(\phi) d\mathbf{x}' &+ \\ &- \frac{1}{4!/2} \frac{\partial^4 \Psi(\mathbf{x})}{\partial x^4} \int_{\Omega} (x - x')^4 \Delta_{\mathbf{x}} \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||} \frac{\alpha_d}{h^{d+1}} W^1(\phi) d\mathbf{x}' &+ \\ &- O[(x - x')^6, (y - y')^6]. \end{split}$$

As integrais contidas nos termos do erro de truncamento para $h_x = h_y = h$ (discretização uniforme nas duas direções) resultam em

$$\int_{\Omega} (y - y')^{\gamma} \Delta_{\mathbf{x}} \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||} \frac{\alpha_d}{h^{d+1}} W^1(\phi) d\mathbf{x}' = -(h^{\gamma - 2}),$$

$$\int_{\Omega} (x - x')^{\gamma} \Delta_{\mathbf{x}} \frac{(\mathbf{x} - \mathbf{x}')}{||\mathbf{x} - \mathbf{x}'||} \frac{\alpha_d}{h^{d+1}} W^1(\phi) d\mathbf{x}' = -(h^{\gamma - 2}),$$
(5.27)

pois,

$$\lim_{h \to 0} \sum_{j \in V_i} (y_i - y_j)^{\gamma} \Delta_{\mathbf{x}_{ij}} \frac{(\mathbf{x}_i - \mathbf{x}_j)}{||\mathbf{x}_i - \mathbf{x}_j||} \frac{\alpha_d}{h^{d+1}} W^1(\phi_{ij}) \frac{m_j}{\rho_j} = -(h^{\gamma-2}),$$

$$\lim_{h \to 0} \sum_{j \in V_i} (x_i - x_j)^{\gamma} \Delta_{\mathbf{x}_{ij}} \frac{(\mathbf{x}_i - \mathbf{x}_j)}{||\mathbf{x}_i - \mathbf{x}_j||} \frac{\alpha_d}{h^{d+1}} W^1(\phi_{ij}) \frac{m_j}{\rho_j} = -(h^{\gamma-2}).$$
(5.28)

Sendo assim, a equação geral do erro de truncamento é dada por

$$\varepsilon_{\tau}(\psi(\mathbf{x})) = \sum_{\gamma=2\zeta+2}^{+\infty} \left(\frac{1}{\gamma!/2} \frac{\partial^{\gamma} \Psi(\mathbf{x})}{\partial x^{\gamma}} + \frac{1}{\gamma!/2} \frac{\partial^{\gamma} \Psi(\mathbf{x})}{\partial y^{\gamma}} \right) h^{\gamma-2} + O[(x-x')^{\gamma}, (y-y')^{\gamma}],$$
(5.29)

onde $\zeta = 1, 2, 3, ...,$ consequentemente, $\gamma = \{4, 6, 8, 10, ...\}, m_j / \rho_j = h^2 \in \nabla_i W(\mathbf{x}_{ij}; h) = (\alpha_d / h^{d+1}) W^1(\phi_{ij})(\mathbf{x}_i - \mathbf{x}_j) / ||\mathbf{x}_i - \mathbf{x}_j||.$

Dessa forma, a expressão numérica para o laplaciano bidimensional é dada por

$$\nabla^2 \Psi(\mathbf{x}_i) = \sum_{j \in V_i} 2 \frac{m_j}{\rho_j} (\Psi(\mathbf{x}_i) - \Psi(\mathbf{x}_j)) \Delta_{\mathbf{x}_{ij}} \frac{(\mathbf{x}_i - \mathbf{x}_j)}{||(\mathbf{x}_i - \mathbf{x}_j)||} \nabla_i W(\mathbf{x}_{ij}; h) + O[(x_i - x_j)^2, (y_i - y_j)^2].$$
(5.30)

A equação (5.26), sem os termos do erro de truncamento coincidem com a aproximação do operador laplaciano, conforme (FRAGA FILHO, 2019). Para o caso da equação da difusão de calor em regime permanente, tem-se que $-\nabla^2 \Psi(\mathbf{x}_i) = f(\mathbf{x}_i)$, ou seja, os termos do erro de truncamento também sofrem a mudança de sinal.

As deduções auxiliares para se chegar ao cálculo das integrais contidas na aproximação dos operadores laplaciano 1D e 2D são apresentadas no Apêndice A.

5.1.4 Erro de iteração

O erro de iteração (ε_n) (MARCHI, 2001) nessa tese será chamado de (ε_l) e definido como a diferença entre a solução exata do sistema de equações algébricas e a solução numérica em uma determinada iteração (ROACHE, 1998; FERZIGER; PERIĆ, 2002), conforme equação (5.31). Para efeitos de simplificação, considere $\psi(\mathbf{x}_i) = \psi_{w^*}(\mathbf{x}_i)$.

$$\varepsilon_l(\psi(\mathbf{x}_i)) = \Psi(\mathbf{x}_i) - [\psi(\mathbf{x}_i)]^l, \qquad (5.31)$$

onde l representa o número da iteração atual no processo de solução do sistema de equações lineares ou não lineares.

Pode-se gerar este tipo de erro ao utilizar qualquer método iterativo para solucionar o modelo matemático com a formulação discreta. É comum em casos de equações não lineares, em *solvers* como Gauss-Seidel, em aplicações de métodos *multigrid*, pseudo-transientes, entre outros. Os métodos TDMA (BURDEN; FAIRES, 2005) e PDMA (ENGELN-MÜLLGES; UHLIG, 1996) são diretos, logo não possuem erros de iteração.

A FIGURA 19 representa o erro de iteração adimensional L_2^l/L_2^0 atingindo a ordem de magnitude da precisão quádrupla (precisão estendida ou *Real*16*) em que os cálculos são processados. Em uma simulação numérica, quando o erro de iteração oscila no limite da precisão, diz-se que não há erros de iteração. Os valores $L_2^l \in L_2^0$ correspondem a norma euclidiana (L_2) vetorial, definida na equação (3.79), do erro na iteração l e na iteração inicial l = 0, respectivamente.

A linha vertical pontilhada representa aproximadamente o início do erro de arredondamento, outra fonte de erro numérico, que foi definido na Seção 3.6.5.



FIGURA 19 – ERRO DE ITERAÇÃO.

Erro de arredondamento

5.1.5

O erro de arredondamento (ε_{π}) é caracterizado pela representação finita dos números reais no ambiente computacional (BAUER, 1974; LINNAINMAA, 1976; STEWART, 1979; WALTER, 1993). São vinculados ao compilador utilizado para criar e executar o código numérico computacional (*software*) e também ao computador (*hardware*). A relação é direta com o número de algarismos (*bytes*) adotados ao efetuar o cálculo no computador. Outra fonte de erro de arredondamento é a utilização de funções nativas do compilador que são aproximadas por séries infinitas mas que consideram apenas um número finito de termos ao utilizá-las.

Para mostrar o erro de arredondamento em uma solução numérica há duas estratégias: a primeira é caracterizada pelo gráfico do $|E(\psi)|$ versus h em escala log-log com $h \to 0$; a segunda pelo gráfico do erro de iteração (adimensional ou não) versus número de iterações em escala y-log com $l \to \infty$. Em problemas 1D a primeira opção pode ser a única, pois em geral utiliza-se algum método direto para determinar a solução numérica. Por outro lado, em problemas 2D ou 3D, é mais fácil encontrar o erro de arredondamento no processo iterativo do *solver*, pois atingir o erro de arredondamento em soluções numéricas para $h \to 0$ requer um tempo computacional elevado se nenhum método *multigrid* for utilizado. Para mostrar o erro de arredondamento, escolheu-se a segunda estratégia, onde apresenta-se o erro de iteração adimensional para valores de lsuficientemente elevados. Na FIGURA 20, nota-se o erro de iteração $\varepsilon_l \to 0$ à medida que $l \to \infty$, até atingir um valor de l suficientemente grande. A partir de então, o erro de arredondamento se torna maior que o erro de iteração. Em outras palavras, ε_l diminui até atingir um determinado l ótimo, enquanto isso, ε_{π} aumenta a partir do l ótimo. Portanto é esperada uma mudança de comportamento a partir de um l que pode ser denominado ótimo. A FIGURA 20 é uma ampliação da FIGURA 19 a partir de l = 385.



FIGURA 20 - ERRO DE ARREDONDAMENTO.

FONTE: O autor (2022).

5.1.6 Erro de inconsistência de partículas

O erro de inconsistência de partículas $(\varepsilon_{\mathbf{x}_i})$ é caracterizado pela insuficiência de partículas \mathbf{x}_j dentro do suporte compacto (V_i) associado à alguma partícula \mathbf{x}_i . Para identificar o fenômeno matemático da inconsistência de partículas, primeiramente é necessário compreender o processo de recrutamento de partículas, e a partir desse conceito pode-se isolar $\varepsilon_{\mathbf{x}_i}$ das demais fontes de erro numérico sem grande dificuldade. O Processo de recrutamento das partículas encontra-se detalhado na Subseção 5.1.6.2, para um modelo matemático 2D.

5.1.6.1 Modelo matemático 1D

Como exemplo, adota-se o modelo matemático da difusão de calor unidimensional definido pela equação

$$-\nabla^2 \Psi = f,\tag{5.32}$$

onde $f = -\pi^2 sen(\pi \mathbf{x})$. A solução analítica é dada por $\Psi(\mathbf{x}) = sen(\pi \mathbf{x})$ e contornos $\Psi(0) = \Psi(1) = 0$. O domínio é definido pela semirreta $[0,1] \in \mathbb{R}$, discretizado com 8 espaços médios uniformes entre partículas $(N_p = 8)$. Com isso, obtém-se $N_{\overline{t}} = 9$ partículas, sendo $N_{\overline{ti}} = 7$ internas e as outras duas sobre os contornos. Sabendo-se que as variáveis são conhecidas nos contornos, utiliza-se apenas as $N_{\overline{ti}} = 7$ partículas internas que definem as variáveis no sistema de equações lineares.

Em problemas unidimensionais, o número ideal para a densidade do suporte compacto é $N_i = 2$. Nesses casos 1D, o erro é mais influente devido ao baixo número de partículas N_i em V_i associado à cada partícula \mathbf{x}_i . Por exemplo, se $N_1 = 1 \neq N_{i\geq 2} = 2$ (considerando apenas a inconsistência de partículas no contorno esquerdo), a solução estará completamente contaminada pelo erro de inconsistência de partículas. Se houver dois ou mais $N_i < 2$, será impossível determinar uma solução numérica. Pode-se realizar a análise do erro de inconsistência de partículas em modelos 1D ao considerar apenas um dos contornos de cada vez. Isso acontece porque é impossível determinar a solução numérica com o método TDMA (BURDEN; FAIRES, 2005) para matriz simétrica se houver um número reduzido de partículas em dois ou mais conjuntos V_i .

Para ilustrar os efeitos do erro $\varepsilon_{\mathbf{x}_i}$, apresenta-se primeiramente na FIGURA 21 a solução analítica $\Psi(\mathbf{x})$ versus solução numérica $\psi(\mathbf{x})$ para o caso em que $N_i = 2 \forall V_i$ e $\mathbf{x} = \mathbf{x}(x)$. Nesse caso, há apenas influência do erro de discretização. Enquanto isso, na FIGURA 22 a solução numérica foi obtida com $N_1 = 1$ em V_1 e $N_{i\geq 2} = 2$ para $V_{i\geq 2}$. A exclusão da partícula \mathbf{x}_j (localizada sobre o contorno esquerdo) de dentro do suporte compacto V_1 foi suficiente para contaminar toda a solução numérica para as $N_{ti} = 7$ partículas. Esse perfil se mantém no contorno direito e não foi mostrado para evitar repetição. No caso do contorno direito, a solução numérica ficaria espelhada em relação ao eixo y, com a divergência iniciando em $\psi(\mathbf{x}_7)$.



FIGURA 21 – SOLUÇÃO ANALÍTICA E NUMÉRICA OBTIDA COM $N_p=8$ E $h=\varDelta x.$

FIGURA 22 – EFEITOS DO ERRO DE INCONSISTÊNCIA SOBRE A SOLUÇÃO NUMÉRICA NUMÉRICA OBTIDA COM $N_p = 8 \to h = \Delta x$.



FONTE: O autor (2022).

5.1.6.2 Modelos matemáticos 2D

Como exemplo, adota-se o modelo matemático da difusão de calor bidimensional em regime permanente, definido pela equação (4.7). O domínio é definido pelo quadrado unitário $[0,1] \times [0,1] \in \mathbb{R}^2$, discretizado com 8 espaços médios uniformes entre partículas $(N_{p_z} = 8)$ em cada direção. Como as variáveis são conhecidas nos contornos, utiliza-se apenas as N = 7 partículas por direção $(N_x = N_y = 7)$, totalizando $N_{ti} = 49$ variáveis no sistema de equações lineares $(A_{N_{ti} \times N_{ti}} \psi_{N_{ti},1} = f_{N_{ti},1})$.

O volume de cada partícula \mathbf{x}_j é definido como $\Delta V_j = m_j/\rho_j = (\Delta x \Delta y) = (\Delta x)^2 = h^2$. A discretização uniforme nos diz que $h_x = \Delta x$, $h_y = \Delta y$ e $h = h_x = h_y$.

O processo de discretização é realizado em duas etapas; na etapa (1) criamse todas as partículas \mathbf{x}_i (partículas efetivas que definem as coordenadas em que as variáveis são calculadas) em ordem lexicográfica, conforme apresenta-se na FIGURA 23; na etapa (2) criam-se as partículas definidas pelo tipo de tratamento de fronteira (partículas ghosts, dummy, etc). No exemplo, criam-se partículas \mathbf{x}_j do tipo dummy, também em ordem lexicográfica, conforme pode-se observar na FIGURA 24. O sobrescrito $\partial\Omega$ sobre a numeração em ordem lexicográfica, indica que a partícula pertence ao tratamento de fronteira.

FIGURA 23 – DOMÍNIO DISCRETIZADO COM $(N_{ti} = 49)$ PARTÍCULAS \mathbf{x}_i INTERNAS.



FONTE: O autor (2022).

Ao iniciar o processo de construção dos coeficientes da matriz associada ao sistema de equações lineares (ou não lineares, de acordo com o modelo matemático) gerado pela discretização do modelo matemático com o método SPH, inicia-se o trabalho de FIGURA 24 – DOMÍNIO DE APOIO DISCRETIZADO COM $(N_{p_z} + 5)^2 - (N_{p_z} - 1)^2$ PARTÍCULAS \mathbf{x}_j DUMMY.



FONTE: O autor (2022).

recrutamento de partículas \mathbf{x}_j dentro do suporte compacto V_i associado a cada uma das partículas \mathbf{x}_i . Observando-se a FIGURA 25, entende-se que a partícula $\mathbf{x}_i = \mathbf{x}_1$ e $V_i = \{\mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_8, \mathbf{x}_9, \mathbf{x}_{10}, \mathbf{x}_{15}, \mathbf{x}_{16}, \mathbf{x}_{17}, \mathbf{x}_{22}\}$ ou $V_i = \{2, 3, 4, 8, 9, 10, 15, 16, 17, 22\}$. A partícula \mathbf{x}_i é denominada partícula de referência ou efetiva, enquanto \mathbf{x}_j é a partícula vizinha. O recrutamento obedece a restrição $\mathbf{x}_j \in V_i$ se $||\mathbf{x}_i - \mathbf{x}_j|| \leq kh$, onde h é o comprimento de suavização e k é um escalar para aumentar o raio da base (funções núcleo têm base radial). Em geral adota-se $h = \kappa \Delta x$ ($\Delta x = 1/N_{p_z}$) para facilitar a manipulação da variável ϕ . Em outras palavras, o escalar κ permite gerar valores não inteiros para a variável ϕ . Nessa tese, adota-se $\kappa = 1$ em todos os casos mostrados, ou seja, $h = \Delta x$.

Simultaneamente ao processo de recrutamento, calcula-se a derivada de alguma função núcleo usando cada uma das partículas $\mathbf{x}_j \in V_i$, conforme mostrou-se na equação (5.30). Escolheu-se neste exemplo a função núcleo *Spline* quíntica proposta por (MORRIS, 1996). A definição da *Spline* cúbica e sua primeira e segunda derivadas, respectivamente, encontram-se na equação (5.33).

$$W_e^0(\phi) = \frac{\alpha_{e,d}}{h^d} \begin{cases} (3-\phi)^5 - 6(2-\phi)^5 + 15(1-\phi)^5 & 0 \le \phi < 1\\ (3-\phi)^5 - 6(2-\phi)^5 & 1 \le \phi < 2\\ (3-\phi)^5 & 2 \le \phi < 3\\ 0 & \phi \ge 3 \end{cases}$$
(5.33)

As constantes de normalização para modelos matemáticos 1D, 2D e 3D são dadas por

 $\alpha_{e,1} = 1/120$, $\alpha_{e,2} = 7/478\pi$ e $\alpha_{e,3} = 1/120\pi$, sendo suas primeira e segunda derivadas, respectivamente,

$$W_e^1(\phi) = -5\frac{\alpha_{e,d}}{h^{d+1}} \begin{cases} (3-\phi)^4 - 6(2-\phi)^4 + 15(1-\phi)^4 & 0 \le \phi < 1\\ (3-\phi)^4 - 6(2-\phi)^4 & 1 \le \phi < 2\\ (3-\phi)^4 & 2 \le \phi < 3\\ 0 & \phi \ge 3 \end{cases}$$

е

$$W_e^2(\phi) = 20 \frac{\alpha_{e,d}}{h^{d+2}} \begin{cases} (3-\phi)^3 - 6(2-\phi)^3 + 15(1-\phi)^3 & 0 \le \phi < 1\\ (3-\phi)^3 - 6(2-\phi)^3 & 1 \le \phi < 2\\ (3-\phi)^3 & 2 \le \phi < 3\\ 0 & \phi \ge 3 \end{cases}$$

A variável independente discreta da função núcleo é definida por $\phi_{ij} = r_{ij}/h = ||\mathbf{x}_i - \mathbf{x}_j||/h$. Observando-se a derivada primeira $W_e^1(\phi)$ na equação (5.33), pode-se identificar que todas as partículas \mathbf{x}_j em que a resultante $\phi \geq 3$, determina $W_e^1(\phi) = 0$. Desta forma, as partículas \mathbf{x}_4 e \mathbf{x}_{22} são automaticamente dispensadas do suporte compacto V_i . Esta informação encontra-se na FIGURA 25, onde as partículas dispensadas estão apontadas.

O recrutamento associado à etapa (1) seleciona apenas as partículas \mathbf{x}_j internas, enquanto isso, as partículas \mathbf{x}_j dummy são desprezadas, como observa-se na FIGURA 26. Por outro lado, na FIGURA 27, pode-se perceber que o recrutamento associado à etapa (2), consiste em selecionar apenas as partículas \mathbf{x}_j dummy pertencentes ao domínio de apoio.

O conjunto \overline{V}_i é chamado de complemento de V_i e só existe para as partículas \mathbf{x}_i localizadas a uma distância menor ou igual a kh, em nosso exemplo, adota-se k = 3, ou seja, a distância é menor igual 3h dos contornos. A união entre os conjuntos V_i e \overline{V}_i pode ser vista na FIGURA 28. Essa união torna o suporte compacto denso, ou seja, não há erro de inconsistência de partículas ($\varepsilon_{\mathbf{x}_i}$). Em suma, quando alguma partícula \mathbf{x}_i estiver fora da região em que a função núcleo sofre truncamento por fronteira, o suporte compacto V_i é naturalmente denso e não há \overline{V}_i . Ainda na FIGURA 28, nota-se também que a ordem lexicográfica é independente para cada uma das etapas, de modo que pode haver repetição da numeração, mas nunca a utilização de uma mesma partícula por mais de uma vez dentro do mesmo suporte compacto. Para facilitar a distinção, utiliza-se o sobrescrito $\partial\Omega$ nas partículas pertencentes ao tratamento de fronteira. Por fim, ao incorporar \overline{V}_i a V_i e dispensando-se todas as partículas associadas à $\phi \geq 3$ ($V_i = \{4,22,4^{\partial\Omega},40^{\partial\Omega}\}$), obtém-se o conjunto final $V_i =$ $\{2,3,8,9,10,15,16,17,15^{\partial\Omega},16^{\partial\Omega},17^{\partial\Omega},18^{\partial\Omega},19^{\partial\Omega},28^{\partial\Omega},29^{\partial\Omega},30^{\partial\Omega},31^{\partial\Omega},32^{\partial\Omega},41^{\partial\Omega},42^{\partial\Omega},47^{\partial\Omega},$ $48^{\partial\Omega},53^{\partial\Omega},54^{\partial\Omega}\}$. Este conjunto V_i gera uma matriz de coeficientes simétrica, com 25 diagonais associada ao sistema de equações lineares.

FIGURA 25 – RECRUTAMENTO PARCIAL INTERNO DE PARTÍCULAS NO SUPORTE COMPACTO V_i SEM TRATAMENTO DE FRONTEIRA.



FONTE: O autor (2022).

FIGURA 26 – RECRUTAMENTO PARCIAL INTERNO DE PARTÍCULAS NO SUPORTE COMPACTO V_i COM TRATAMENTO DA FRONTEIRA.



FONTE: O autor (2022).

Sabendo-se como a matriz de coeficientes é construída, pode-se isolar o erro de inconsistência de partículas para analisá-lo. A análise será realizada por meio do erro verdadeiro da variável $E(\psi(\mathbf{x}_1)) = \Psi(\mathbf{x}_1) - \psi(\mathbf{x}_1)$ com retirada de partículas do conjunto \overline{V}_i .
FIGURA 27 – RECRUTAMENTO PARCIAL EXTERNO DE PARTÍCULAS NO SUPORTE COMPACTO V_i NA REGIÃO PRÓXIMA DA FRONTEIRA.



FONTE: O autor (2022).

FIGURA 28 – RECRUTAMENTO COMPLETO DE PARTÍCULAS NO SUPORTE COMPACTO V_i NA REGIÃO PRÓXIMA DA FRONTEIRA.



FONTE: O autor (2022).

Na TABELA 3 apresenta-se o erro verdadeiro associado a partícula \mathbf{x}_1 em três níveis distintos de discretização, à medida que as partículas \mathbf{x}_j dummy são retiradas do conjunto \overline{V}_i . A metodologia consiste em retirar uma partícula por vez, partindo sempre da partícula \mathbf{x}_j mais próxima de \mathbf{x}_1 (há grupos de partículas \mathbf{x}_j com mesma distância de \mathbf{x}_1). A escolha das partículas com mesma distância pode ser aleatória, pois a posição não afetará o resultado. Na primeira linha da TABELA 3 o conjunto complementar \overline{V}_i possui 16 partículas, enquanto V_i tem 8. A união dos dois conjuntos gera um total de 24 partículas, que é o valor da densidade do suporte compacto para $h = \kappa \Delta x$, com $\kappa = 1$ e kh = 3h, ou seja, k = 3. Não discutiu-se ainda os valores para a densidade ótima do suporte compacto nos modelos 2D. A partícula $\mathbf{x}_i \notin V_i$. A última linha, ainda da TABELA 3, mostra o conjunto $\overline{V}_i = \{\emptyset\}$ e o erro verdadeiro numérico atingindo seu maior valor se for considerado apenas a existência dos erros de inconsistência de partículas e discretização. O erro de inconsistência de partículas pode ser desprezado se o número de elementos de $V_i \cup \overline{V}_i, i = 1, 2, \dots, N_{ti} - 1, N_{ti}$ for sempre o mesmo. A razão entre os erros verdadeiros de $\psi(\mathbf{x}_1)$ à medida que se retira as partículas, tende a 1. Acredita-se que tal informação confirme a propriedade que o núcleo SPH deve ser decrescente, como definição abaixo

 $W(\mathbf{x}_i - \mathbf{x}_1, h) > W(\mathbf{x}_i - \mathbf{x}_2, h), \text{ se } \|\mathbf{x}_i - \mathbf{x}_1\| < \|\mathbf{x}_i - \mathbf{x}_2\|.$

$V_{i} = 16$	$N_{p_z} = 8$	$N_{p_z} = 16$	$N_{p_z} = 32$
16	2,41294444976275146E - 05	9,73167569090202797E - 06	1,36712988345996509E - 06
15	8,45049581258682307E - 05	1,54950607362761122E - 05	1,90451076196233359E - 06
14	1,81273402283575465E - 04	2,47351495390228667E - 05	2,76607713132439879E - 06
13	2,38097164665481406E - 04	3,01625908566579772E - 05	3,27215293053721527E - 06
12	3,12808776599961377E - 04	3,73002927997840841E - 05	3,93771039431544162E - 06
11	4,15432556598215974E - 04	4,71078329443721390E - 05	4,85223690924044873E - 06
10	4,31133132895397619E - 04	4,86086311190137388E - 05	4,99218421581785991E - 06
09	4,47608522976494455E - 04	5,01835860691391864E - 05	5,13904709276602468E - 06
08	4,52790103772121976E - 04	5,06789359048721823E - 05	5,18523805672899093E - 06
07	4,58050768124248313E - 04	5,11818556571509517E - 05	5,23213496679234660E - 06
06	4,63392340467369403E - 04	5,16925201868305034E - 05	5,27975413125909747E - 06
05	4,68816701789767209E - 04	5,22111097821400675E - 05	5,32811236465532904E - 06
04	4,74325791841940438E - 04	5,27378103709000255E - 05	5,37722700752579796E - 06
03	4,79921611449337162E - 04	5,32728137427748526E - 05	5,42711594716576143E - 06
02	4,79932956107391423E - 04	5,32738983904955372E - 05	5,42721709037946188E - 06
01	4,79944301124705030E - 04	5,32749830726098533E - 05	5,42731823680063263E - 06
00	4,79955646501295049E - 04	5,32760677891194368E - 05	5,42741938642942625E - 06

TABELA 3 – ERRO VERDADEIRO DE $\psi_{e,24}(\mathbf{x}_1)$.

FONTE: O autor (2022).

A FIGURA 29 mostra a solução analítica para todas as partículas do domínio incluindo os contornos. A FIGURA 30 mostra a solução numérica para todas as partículas do domínio, também incluindo os contornos. Em ambos os casos utilizou-se $N_{p_z} = 8$.

A FIGURA 31 mostra a solução analítica para todas as partículas do domínio incluindo os contornos. A FIGURA 32 mostra a solução numérica para todas as partículas do domínio, também incluindo os contornos. Em ambos os casos utilizou-se $N_{pz} = 64$.

Na FIGURA 33 encontra-se o gráfico da temperatura para $x = \Delta x$ fixo e y variável. Nesse caso não há erro de inconsistência de partículas na aproximação numérica associada

FIGURA 29 – SOLUÇÃO ANALÍTICA PARA DOMÍNIO COMPLETO OBTIDA COM $N_t=81$ PARTÍCULAS.





FIGURA 30 – SOLUÇÃO NUMÉRICA PARA DOMÍNIO COMPLETO OBTIDA COM $N_t=81$ PARTÍCULAS.



FONTE: O autor (2022).

à partícula \mathbf{x}_1 . Enquanto isso, na FIGURA 34 foram retiradas todas as partículas \mathbf{x}_j dummy do suporte compacto V_1 associado à \mathbf{x}_1 . Fica evidente o erro numérico verdadeiro para a variável nas coordenadas dessa partícula.

FIGURA 31 – SOLUÇÃO ANALÍTICA PARA DOMÍNIO COMPLETO OBTIDA COM ${\cal N}_t=4225$ PARTÍCULAS.



FONTE: O autor (2022).

FIGURA 32 – SOLUÇÃO NUMÉRICA PARA DOMÍNIO COMPLETO OBTIDA COM ${\cal N}_t=4225$ PARTÍCULAS.



FONTE: O autor (2022).

Para observar a convergência da solução, apresenta-se na FIGURA 35 o mesmo gráfico da temperatura em $x = \Delta x$ fixo e y variável, mas agora com $N_{p_z} = 64$. É fácil notar que a solução numérica converge para a analítica.

FIGURA 33 – SOLUÇÃO ANALÍTICA E NUMÉRICA OBTIDA COM $N_{ti} = 49$ PARTÍCULAS PARA $\mathbf{x}_1 = \mathbf{x}_1(0, 125; y)$.



FIGURA 34 – SOLUÇÃO ANALÍTICA VERSUS NUMÉRICA COM ERRO DE INCONSIS-TÊNCIA DE PARTÍCULAS OBTIDA COM $N_{ti} = 49$ PARTÍCULAS PARA $\mathbf{x}_1 = \mathbf{x}_1(0,125; y).$



5.1.7 Erro de suavização

O erro de suavização (ε_W) é caracterizado pela substituição da Distribuição *Delta* de *Dirac* por uma função núcleo, suficientemente suave, na representação integral de uma

FIGURA 35 – SOLUÇÃO ANALÍTICA E NUMÉRICA OBTIDA COM $N_{ti} = 3969$ PARTÍCU-LAS PARA $\mathbf{x}_1 = \mathbf{x}_1(0,015625; y).$



função com o método SPH. Tal erro independe do termo fonte utilizado. Além disso, esse erro só poderá ser analisado se não houver erro de inconsistência de partícula. Para isso, verifica-se o valor de N_i para todo V_i e sendo o mesmo, a partir dessa informação, avalia-se os momentos do núcleo SPH de acordo com a formulação adotada na discretização do modelo matemático. Para o modelo da difusão de calor 1D em regime permanente, adotou-se um modelo numérico que considera a derivada primeira da função núcleo. Dessa forma, obtém-se

$$\begin{cases}
M_0^1 = \int_{\Omega} W^1(\mathbf{x}_i - \mathbf{x}_j; h) d\mathbf{x}_j = 0 \\
M_1^1 = \int_{\Omega} (\mathbf{x}_i - \mathbf{x}_j) W^1(\mathbf{x}_i - \mathbf{x}_j; h) d\mathbf{x}_j = 1 \\
M_2^1 = \int_{\Omega} (\mathbf{x}_i - \mathbf{x}_j)^2 W^1(\mathbf{x}_i - \mathbf{x}_j; h) d\mathbf{x}_j = 0 \\
\vdots \\
M_n^1 = \int_{\Omega} (\mathbf{x}_i - \mathbf{x}_j)^n W^1(\mathbf{x}_i - \mathbf{x}_j; h) d\mathbf{x}_j = 0
\end{cases}$$
(5.34)

Usando o exemplo 1D mostrado na Seção 5.1.6, foi possível identificar que os três primeiros momentos da equação (5.34) são perfeitamente cumpridos com os núcleos *spline* cúbica (MONAGHAN; LATTANZIO, 1985) e *new* quártico (LIU; LIU; LAM, 2003). Para efeitos de simplificação, apresenta-se na TABELA 4, o cálculo numérico para os três primeiros momentos definidos na equação (5.34), usando o núcleo *spline* cúbica.

Se houver erro de inconsistência de partículas, tem-se uma dificuldade adicional

V_i	N_i	M_{0}^{1}	M_1^1	M_2^1
V_1	2	0,0000000E + 00	1,0000000E + 00	0,0000000E + 00
V_2	2	0,0000000E + 00	1,00000000E + 00	0,00000000E + 00
V_3	2	0,0000000E + 00	1,00000000E + 00	0,00000000E + 00
V_4	2	0,0000000E + 00	1,00000000E + 00	0,00000000E + 00
V_5	2	0,0000000E + 00	1,00000000E + 00	0,00000000E + 00
V_6	2	0,0000000E + 00	1,00000000E + 00	0,00000000E + 00
V_7	2	0,0000000E + 00	1,0000000E + 00	0,00000000E + 00
			- ()	

TABELA 4 – MOMENTOS DA DERIVADA PRIMEIRA $W_{c,2}^1$.

FONTE: O autor (2022).

em determinar os momentos apresentados na TABELA 4. Como exemplo, apresenta-se na TABELA 5 os momentos para o mesmo modelo matemático, mas com $N_1 = 1$ em V_1 . Os efeitos desse erro são mostrados na Seção 5.1.6, FIGURA 22.

TABELA 5 – MOMENTOS DA DERIVADA PRIMEIRA $W_{c,2}^1$.

V_i	N_i	M_{0}^{1}	M_{1}^{1}	M_{2}^{1}
V_1	1	-4,0000000E + 00	5,00000000E - 01	-6,25000000E - 02
V_2	2	0,00000000E + 00	1,00000000E + 00	0,00000000E + 00
V_3	2	0,00000000E + 00	1,00000000E + 00	0,00000000E + 00
V_4	2	0,00000000E + 00	1,00000000E + 00	0,00000000E + 00
V_5	2	0,00000000E + 00	1,00000000E + 00	0,00000000E + 00
V_6	2	0,00000000E + 00	1,00000000E + 00	0,00000000E + 00
V_7	2	0,00000000E + 00	1,00000000E + 00	0,00000000E + 00

FONTE: O autor (2022).

5.1.8 Erro de poluição numérica

O erro de poluição numérica (ε_{np}) baseado em (MARCHI; SILVA, 2002; BURDEN; FAIRES; BURDEN, 2016) é caracterizado pelo erro verdadeiro ($E(\psi(\mathbf{x}_i)) = \Psi(\mathbf{x}_i) - \psi(\mathbf{x}_i)$) da variável de interesse nas coordenadas de uma partícula \mathbf{x}_i , quando aplicado no esquema numérico utilizado para discretizar o modelo matemático. Para facilitar o entendimento, apresenta-se o erro de poluição numérica para o operador laplaciano SPH e para a regra do trapézio, respectivamente.

$$-\nabla^2 \Psi(\mathbf{x}_i) = -2\sum_{j \in V_i} (\psi(\mathbf{x}_i) - \psi(\mathbf{x}_j)) \frac{m_j}{\rho_j} \Delta_{\mathbf{x}_{ij}} \cdot \nabla_i W_{ij} - \varepsilon_{np}(\psi(\mathbf{x}_i)) - \varepsilon_\tau(\psi(\mathbf{x}_i)), \quad (5.35)$$

onde

$$\varepsilon_{np}(\psi(\mathbf{x}_i)) = -2\sum_{j \in V_i} (E(\psi(\mathbf{x}_i)) - E(\psi(\mathbf{x}_j))) \frac{m_j}{\rho_j} \Delta_{\mathbf{x}_{ij}} \cdot \nabla_i W_{ij}$$
(5.36)

е

$$\psi_m = \frac{h}{2L} \sum_{i=2}^{N} (\psi_{i-1} + \psi_i) + \varepsilon_{np}(\psi_m) + \varepsilon_{\tau}(\psi_m), \qquad (5.37)$$

sendo

$$\varepsilon_{np}(\psi_m) = \frac{h}{2L} \sum_{i=2}^{N} (E(\psi_{i-1}) + E(\psi_i)).$$
(5.38)

Note que nas duas aproximações numéricas, o termo do erro de truncamento foi acoplado. Isso quer dizer que, ao adicionarmos o erro de poluição numérica e o erro de truncamento (com seus infinitos termos) à aproximação numérica calculada, obtém-se a solução analítica numericamente dentro da ordem de magnitude do erro de arredondamento. Em outras palavras, se as duas únicas fontes de erro são o erro de poluição numérica e o erro de truncamento, então o acoplamento deles na solução numérica determina o erro verdadeiro nulo, ou seja, $E(\psi(\mathbf{x}_i)) = 0$. Na prática, não é necessário utilizar infinitos termos do erro de truncamento porque a magnitude do erro de arredondamento é em torno de 1,0E-32.

O erro de poluição numérica pode gerar uma solução numérica com melhores resultados do que o previsto, e isso só pode ser observado quando é realizada a verificação das soluções numéricas por meio dos testes de coerência.

5.2 Dados gerais das simulações numéricas

Todas as figuras dessa tese foram criadas usando Matlab[®] versão R2016a. As simulações numéricas foram obtidas inicialmente por meio de linguagem Fortran 95 usando Microsoft[®] Visual Studio[®] 2008 compiler v. 9.0.21022.8 RTM com precisão quádrupla (ou estendida) e executadas em um computador com processador 3.40 GHz Intel[®] CoreTMi7-6700, 16GB de memória RAM e Windows[®] 10 de 64-bit. Após um período, as simulações foram migradas para o cluster GridUNESP por meio de projeto de pesquisa dos autores, submetido e aprovado. As configurações do GridUNESP são:

Architecture:	x86_64
CPU op-mode(s):	32-bit, 64-bit
Byte Order:	Little Endian
CPU(s):	56
On-line CPU(s) list:	0-55
Thread(s) per core:	2
Core(s) per socket:	14
Socket(s):	2
NUMA node(s):	2

Slab:

901168 kB

Vendor ID:	G	enuineIr	ntel					
CPU family:	6							
Model:	79	9						
Model name:	I	ntel(R)	Xeon(R)	CPU	E5-2680	v4	0	2.40GHz
Stepping:	1							
CPU MHz:	2	399.853						
CPU max MHz:	24	400.000)					
CPU min MHz:	1:	200.000)					
BogoMIPS:	4	789.08						
Virtualization:	V	Г-х						
L1d cache:	3:	2K						
L1i cache:	3:	2K						
L2 cache:	2	56K						
L3 cache:	3	5840K						
NUMA node0 CPU(s	s): 0 [.]	-13,28-4	11					
NUMA node1 CPU(s	s): 14	4-27,42-	-55					
MemTotal:	13144262	4 kB						
MemFree:	33873384	kB						
MemAvailable:	126528908	8 kB						
Buffers:	60	kB						
Cached:	92674044	kB						
SwapCached:	10740	kB						
Active:	6197088	kB						
Inactive:	89171980	kB						
Active(anon):	2531368	kB						
<pre>Inactive(anon):</pre>	216944	kB						
Active(file):	3665720	kB						
<pre>Inactive(file):</pre>	88955036	kB						
Unevictable:	0	kB						
Mlocked:	0	kB						
SwapTotal:	15624188	kB						
SwapFree:	15466924	kB						
Dirty:	60	kB						
Writeback:	4	kB						
AnonPages:	2685140	kB						
Mapped:	73832	kB						
Shmem:	53344	kB						

SReclaimable:	660804	kВ	
SUnreclaim:	240364	kB	
KernelStack:	14352	kВ	
PageTables:	23040	kВ	
NFS_Unstable:	12	kB	
Bounce:	0	kB	
WritebackTmp:	0	kВ	
CommitLimit:	81345500	kВ	
Committed_AS:	39838908	kВ	
VmallocTotal:	343597383	367	kВ
VmallocUsed:	514740	kВ	
VmallocChunk:	342918451	L16	kВ
Percpu:	13568	kВ	
HardwareCorrupte	ed: 0	kВ	
AnonHugePages:	331776	kВ	
CmaTotal:	0	kВ	
CmaFree:	0	kВ	
HugePages_Total:	: 0		
HugePages_Free:	0		
HugePages_Rsvd:	0		
HugePages_Surp:	0		
Hugepagesize:	2048	kВ	
DirectMap4k:	196480	kВ	
DirectMap2M:	6725632	kВ	
DirectMap1G:	128974848	3 kI	3

A verificação das soluções numéricas obtidas com o SPH está separada em duas partes, sendo elas: 1) verificação qualitativa das soluções numéricas; 2) verificação quantitativa das soluções numéricas. A verificação qualitativa das soluções numéricas aborda os testes de coerência, ou seja, são realizadas análises sobre o decaimento do erro de discretização versus aumento do número de partículas que discretizam o domínio. Além disso, são calculadas as ordens efetivas (p_E) e aparentes (p_U) de acurácia, obtidas *a posteriori*. Enquanto isso, a verificação quantitativa das soluções numéricas aborda aspectos sobre tempo de processamento, coerência entre resultados obtidos com paralelização e teorias (apropriada para casos 2D ou 3D).

Usando precisão quádrupla, as soluções numéricas com SPH são determinadas em diferentes níveis de discretização, sempre com razão de refinamento constante (q = 2). Os níveis de discretização são obtidos variando o número de espaços médios uniformes entre partículas $N_p = 8,16,32,64,\ldots,32768,65536$ e $N_{p_z} = 8,16,32,64,\ldots,2048,4096$ (N_p para o caso 1D e N_{p_z} para 2D). Utiliza-se o comprimento de suavização $h = \Delta x = 1/N_{p_i}$, o raio da base da função núcleo kh = 2h, o incremento temporal uniforme $\Delta t = \Delta x = h$ e $m_j/\rho_j = h$ para o caso 1D. Para o caso 2D, $h = \Delta x = \Delta y = 1/N_{p_{z_i}}$ e kh = 3h e $m_j/\rho_j = h^2$. Os modelos matemáticos 1D da difusão de calor em regime permanente foram resolvidos com N_p variando até 65536 e o modelo da difusão de calor em regime transiente até 32768. Para o caso 2D, a discretição com $N_{p_z} = 4096$, determina 4097 partículas por direção, resultando em um total de 16.785.409 partículas.

5.3 Equação da difusão de calor em regime permanente 1D

5.3.1 Exemplo (E1-1Dp)

5.3.1.1 Verificação qualitativa de soluções numéricas para variável local

Os resultados apresentados nas simulações consideram as derivadas dos núcleos spline cúbica (W_c) (MONAGHAN; LATTANZIO, 1985) e new quártico spline (W_d) (LIU; LIU; LAM, 2003), pois as demais funções núcleo SPH mostradas na Seção 3.1.6 não apresentaram resultados coerentes ao adotar $h = \Delta x$. A convenção $h = \Delta x$ foi escolhida para manter a razão de refino constante. Nessa seção, avaliam-se os resultados das simulações para a variável primária local (temperatura no ponto médio $\psi(1/2)$) e secundária global (temperatura média ψ_m).

O primeiro resultado apresentado na FIGURA 36 mostra em escala log-log a diminuição do erro de discretização *versus* o aumento do número de partículas. No nível m = 0 a RRE não foi aplicada, de modo que, a solução numérica foi obtida ao resolver o sistema linear de equações algébricas gerado pelo modelo numérico apresentado nas equações (3.20) e (3.21). O nível m = 1 representa a diminuição do erro de discretização versus h após aplicar uma única extrapolação de Richardson usando a equação (3.87). Nos níveis sucessores, de m = 2 até m = 7 foram aplicadas múltiplas extrapolações de Richardson usando também a equação (3.87). A RRE é finalizada no nível m = 7 porque o erro de discretização atingiu a mesma ordem de magnitude que o erro de arredondamento (aproximadamente 1,0E-32). Analisando-se cuidadosamente os resultados, percebe-se que o módulo do erro de discretização foi eficientemente reduzido de 1,67E-08 com 1.025 partículas até 3.46E-33 com quatro extrapolações de Richardson (m = 4) e 1025 partículas (incluindo os contornos). A RRE diminuiu a magnitude do erro de discretização aproximadamente 24 ordens e para isso utilizou-se um número relativamente pequeno de partículas. A influência do erro de arredondamento pode ser observada nos níveis m = 5, m = 6 e m = 7. Curiosamente, os resultados obtidos com a função núcleo new quártico spline foram numericamente idênticos aos gerados pela spline cúbica e por esse motivo, escolheu-se não apresentá-los, evitando assim, repetições.





Na TABELA 6 é possível observar a diminuição do módulo do erro de discretização em cada uma das extrapolações. Para efeitos de apresentação, os valores possuem apenas duas casas decimais. Entretanto, é importante entender que os cálculos foram realizados com precisão quádrupla (Real*16, ou estendida), de modo que, todas as operações utilizam 32 casas decimais. A sigla não se aplica (NA) é utilizada para marcar a perda de solução à medida que as multiextrapolações de Richardson são aplicadas. Observa-se que a quantidade de soluções perdidas é idêntica ao número do nível, ou seja, no nível m = 10, foram perdidas 10 soluções obtidas inicialmente em m = 0. Na quarta extrapolação (m = 4) determinou-se o menor erro de discretização (3,46E-33), que está associado a discretização com 1025 partículas. Isso quer dizer que, não é necessário resolver sistemas de ordens muito elevadas para se determinar soluções com a maior ordem de acurácia possível, sempre de acordo com a limitação computacional. Nota-se que os erros de discretização associados aos valores de h = 6,10E-05 e h = 3,05E-05, respectivamente, já estão oscilando no nível m = 2.

Observação: seja um conjunto qualquer de ordens verdadeiras deduzidas *a priori* $p_V = \{2,4,6,8,10,\ldots\}$. Em geral, lê-se que a primeira ordem de acurácia é 2, a segunda 4, assim sucessivamente. Nessa tese, todas as vezes que for mencionado a ordem de acurácia, essa relação será descrita de forma diferente, considerando-se que a segunda ordem de acurácia é 2, a quarta ordem é 4, assim por diante. Ou seja, não estamos enfatizando a ordenação do conjunto e assim a ordem do método. Por exemplo, supondo-se que após 3 extrapolações as soluções estão com ordem de acurácia convergindo para 8. Na forma geral, denota-se que atingiu-se a quarta ordem de acurácia (ordenação do conjunto de ordens verdadeiras, p_V) que é 8, por outro lado, na nova convenção, fica definido que atingiu-se a

h	m = 0	m = 1	m = 2	m = 3	m = 4	m = 5	m = 6	m = 7	m = 8	m = 9	m = 10
1,25E - 01	2,73E - 04	NA									
6,25E - 02	6,84E - 05	5,34E - 08	NA								
3,12E - 02	1,71E - 05	3,34E - 09	2,07E - 12	NA							
1,56E - 02	4,28E - 06	2,09E - 10	3,23E - 14	1,77E - 17	NA						
7,81E - 03	1,07E - 06	1,30E - 11	5,06E - 16	6,92E - 20	3,50E - 23	NA	NA	NA	NA	NA	NA
3,90E - 03	2,67E - 07	8,16E - 13	7,91E - 18	2,70E - 22	3,42E - 26	1,66E - 29	NA	NA	NA	NA	NA
1,95E - 03	6,68E - 08	5,10E - 14	1,23E - 19	1,05E - 24	3,40E - 29	5,81E - 31	5,77E - 31	NA	NA	NA	NA
9,76E - 04	1,67E - 08	3,18E - 15	1,93E - 21	4,12E - 27	3,46E - 33	3,67E - 32	3,69E - 32	3,69E - 32	NA	NA	NA
4,88E - 04	4,18E - 09	1,99E - 16	3,01E - 23	1,88E - 29	2,78E - 30	NA	NA				
2,44E - 04	1,04E - 09	1,24E - 17	4,71E - 25	7,09E - 30	7,04E - 30	7,05E - 30	NA				
1,22E - 04	2,61E - 10	7,78E - 19	7,37E - 27	2,68E - 30	2,66E - 30						
6,10E - 05	6,53E - 11	4,86E - 20	5,84E - 29	5,75E - 29	5,77E - 29	5,78E - 29					
3,05E - 05	1,63E - 11	3,04E - 21	1,36E - 28	1,40E - 28							
1,52E - 05	4,08E - 12	1,90E - 22	2,32E - 28	2,38E - 28	2,39E - 28	2,40E - 28					

TABELA 6 – MÓDULO DO ERRO DE DISCRETIZAÇÃO USANDO W_c^1 COM RRE (E1-1Dp).

FONTE: O autor (2022).

oitava ordem de acurácia após 3 extrapolações, sem vincular a ordenação do conjunto p_V .

Apresenta-se na FIGURA 37 o cálculo das ordens de acurácia usando a ordem aparente equivalente (p_U) . Isso significa que é possível calcular as ordens de acurácia mesmo nos casos em que a solução analítica é desconhecida. A ordem aparente (p_U) é uma técnica equivalente a ordem efetiva (p_E) , porém mais robusta, já que p_E só é possível ser calculada se a solução analítica for conhecida. Mostrou-se que é possível utilizar as duas técnicas para realizar a verificação qualitativa das soluções numéricas obtidas com o método SPH. A verificação consiste em analisar o decaimento do módulo do erro de discretização versus aumento do número de partículas e ordens de acurácia p_U e p_E . Essas três técnicas são conhecidas como testes de coerência. Em suma, a verificação qualitativa das soluções numéricas consiste da análise de resultados obtidos nos testes de coerência. Iniciando-se em m = 0, onde $p_0 = 2 \in p_U$, determina-se 3 extrapolações de Richardson (m = 3), onde atingiu-se $p_3 = 8$, ou seja, foi possível obter oitava ordem de acurácia para as soluções numéricas $\psi(1/2)$ usando a ordem aparente equivalente. Nota-se no lado direito da figura que são necessárias 3 (para p_E apenas 2, conforme FIGURA 38) soluções numéricas para que uma nova solução seja determinada no nível de extrapolação sucessor. Essa informação também pode ser observada na equação (3.88a).

A FIGURA 38 mostra as ordens efetivas de acurácia (p_E) . Iniciando-se em m = 0onde $p_0 = 2 \in p_E$, foi possível obter 4 extrapolações de Richardson (m = 4), determinandose $p_4 = 10$, ou seja, soluções de décima ordem de acurácia para a variável $\psi(1/2)$. É possível obter ordens mais elevadas de acurácia ao utilizar p_E e isso somente é possível, porque como mencionado anteriormente, para o seu cálculo são utilizadas apenas a combinação entre 2 soluções consecutivas. A degeneração das ordens que aparece no lado esquerdo da figura acontece devido ao fato que o módulo do erro de discretização já atingiu a magnitude do erro de arredondamento. No lado direito sempre nota-se as perdas de soluções devido ao avanço das extrapolações, enquanto que no lado esquerdo acontece o fenômeno da degeneração de ordem por erro de arredondamento. Percebe-se que, por exemplo, para obter a primeira solução de décima ordem, resolveu-se 6 sistemas de equações lineares

FIGURA 37 – ORDEM APARENTE (p_U) DA TEMPERATURA NO PONTO MÉDIO USANDO $W_{c,2}^1$ COM RRE (E1-1Dp).



(soluções no nível m = 0), sendo eles: 7 × 7, 15 × 15, 31 × 31, 63 × 63, 127 × 127 e 255 × 255. A solução $\psi(1/2)$ de cada um desse sistemas é então armazenada e em seguida aplica-se a RRE.

FIGURA 38 – ORDEM EFETIVA (p_E) DA TEMPERATURA NO PONTO MÉDIO USANDO $W_{c,2}^1$ COM RRE (E1-1Dp).



5.3.1.2 Verificação qualitativa de soluções numéricas para variável global

Nesta seção realiza-se a verificação das soluções numéricas para a variável secundária global ψ_m que é determinada a partir do modelo numérico apresentado na equação (3.77). Nota-se na FIGURA 39 que para m = 6 o módulo do erro de discretização da temperatura média já está influenciado pelo erro de arredondamento. Isso significa que não há necessidade de continuar aplicando a RRE, uma vez que o método atingiu o limite da precisão do compilador para variáveis do tipo real no programa.

Na FIGURA 40 pode-se perceber que foi possível determinar a oitava ordem de acurácia ($p_3 = 8$) para ψ_m em m = 3 usando p_U . Em m = 4, o valor de p_4 é completamente degenerado. Nesses casos, a ordem de acurácia mostra a degeneração porque o módulo do erro de discretização associado a um dado h atingiu o seu menor valor possível e no nível de discretização sucessor, aumentou novamente. Mais uma vez, esse fenômeno se repete porque atingiu-se o limite da precisão quádrupla.

Observa-se na FIGURA 41 que foi possível determinar a décima ordem de acurácia para ψ_m usando a ordem efetiva p_E ($p_4 = 10 \text{ em } m = 4$). Para m = 5 os resultados já são influenciados pelo erro de arredondamento. Vale lembrar que, sempre é possível obter ordens mais elevadas usando p_E em vez de p_U . Entretanto, é impossível utilizar p_E se a solução analítica for desconhecida. A importância da comparação entre os cálculos das ordens efetiva (p_E) e aparente equivalente (p_U) deve-se ao fato de que é importante desenvolver expertise para que as características de ambos os testes sejam identificadas sem dificuldades, pois, sabe-se que na prática a solução analítica é em geral, desconhecida.

FIGURA 39 – ERRO DE DISCRETIZAÇÃO DA TEMPERATURA MÉDIA USANDO $W_{c,2}^1$ COM RRE (E1-1Dp).



FONTE: O autor (2022).





Os primeiros testes mostram que tanto p_E quanto p_U podem ser utilizadas para calcular as ordens de acurácia das soluções numéricas obtidas com o método SPH. Isso só é possível porque a partícula centrada em x = 1/2 está presente em todos os níveis de discretização adotados.

FIGURA 41 – ORDEM EFETIVA (p_E) DA TEMPERATURA MÉDIA USANDO $W_{c,2}^1$ COM RRE (E1-1Dp).



5.3.2 Exemplo (E2-1Dp)

5.3.2.1 Verificação qualitativa de soluções numéricas para variável local

Reproduzindo-se os testes de coerência do exemplo (E1-1Dp), agora para (E2-1Dp), é possível observar na FIGURA 42 que determinou-se seis extrapolações Richardson (m = 6), porque em m = 7 o módulo do erro de discretização ja está influenciado pelo erro de arredondamento. As outras extrapolações posteriores não foram mostradas porque já têm uma influência significativa do erro de arredondamento.

FIGURA 42 – ERRO DE DISCRETIZAÇÃO DA TEMPERATURA NO PONTO MÉDIO USANDO $W_{c,2}^1$ COM RRE (E2-1Dp).



FONTE: O autor (2022).

Como consequência do número de extrapolações confiáveis (m = 5) mostrado na FIGURA 42, foi possível determinar apenas um único resultado com décima ordem de acurácia para o conjunto p_U , ou seja, $p_4 = 10 \in p_U$, conforme observa-se na FIGURA 43. Entende-se que a diferença entre o número de extrapolações nos exemplos (E1-1Dp) e (E2-1Dp), está relacionada com a função f(x) que define o termo fonte no modelo matemático. Em (E1-1Dp) utilizou-se uma função exponencial, enquanto em (E2-1Dp) a função é trigonométrica. Isso pode afetar o erro numérico, de modo que possa conter outra fonte de erro além do erro de discretização.

Na FIGURA 44 mostrou-se que foi possível determinar a décima segunda ordem de acurácia $p_5 = 12 \in p_E$. Nota-se que, a cada extrapolação de Richardson, perde-se uma solução (sempre observar o lado direito das figuras) e quanto mais refinada for a discretização, mais influente se torna o erro de arredondamento (ver as linhas mostrando a degeneração das ordens do lado esquerdo). Vale a pena lembrar que matematicamente $p_E e p_U$ são dois conjuntos de ordens de acurácia que têm uma sequência de elementos chamados p_m , ou seja, $p_m \,\subset p_E e p_m \subset p_U$. Contudo, os valores de p_m para o conjunto p_E são calculados utilizando |E|, enquanto os valores de p_m para o conjunto p_U utilizam apenas os valores da variável de interesse $\psi(1/2)$ ou ψ_m obtidas com três discretizações consecutivas. Nesse exemplo, obteve-se $p_5 = 12$, ou seja, o conjunto p_E é determinado como $p_E = \{p_0, p_1, p_2, p_3, p_4, p_5\} = \{p_m, m = 0, \dots, 5\} = \{2, 4, 6, 8, 10, 12\}$. O mesmo acontece com o conjunto p_U .

FIGURA 43 – ORDEM APARENTE (p_U) DA TEMPERATURA NO PONTO MÉDIO USANDO $W_{c,2}^1$ COM RRE (E2-1Dp).



FONTE: O autor (2022).

5.3.2.2 Verificação qualitativa de soluções numéricas para variável global

Ao observar o comportamento do módulo de erro de discretização para a variável ψ_m na FIGURA 45, nota-se que as linhas m = 0 e m = 1 são paralelas, isso nos diz que a ordem de acurácia das soluções numéricas sem aplicar a RRE (m = 0) e ao efetuar uma única extrapolação de Richardson (m = 1), comportam-se igualmente. Para compreender esse fenômeno matemático que causou a mudança nos resultados esperados, é necessário desenvolver expertise no que entende-se ser reconhecido como erro de poluição numérica (ε_{np}) com base em (MARCHI; SILVA, 2002; BURDEN; FAIRES; BURDEN, 2016). Nas equações (5.35)-(5.38), introduziu-se os esquemas numéricos com os seus respectivos termos do erro de poluição numérica e erro de truncamento para as variáveis de interesse primária e secundária, respectivamente.





Como a solução analítica é conhecida, o erro de poluição numérica pode ser calculado e incorporado à solução numérica das variáveis de interesse. O acoplamento $(\psi_m + \varepsilon_{np}(\psi_m))$ corrige completamente as soluções numéricas que apresentam o perfil inesperado. Realizou-se esse procedimento apenas para identificar a fonte do erro numérico que alterou os resultados esperados para a variável de interesse ψ_m em m = 0. É importante destacar que o erro de poluição numérica pode aparecer em qualquer esquema numérico (ver equações (5.35)-(5.38)), mas é comum que os seus efeitos sejam negligenciáveis. No exemplo (E2-1Dp), nota-se a predominância do erro de poluição numérica (a princípio desconhecido), devido à metodologia de verificação de soluções numéricas utilizando os testes de coerência (cálculos e gráficos de |E|, $p_U e p_E$). Os resultados corrigidos não foram mostrados, pois é comum resolver problemas em que a solução analítica seja desconhecida. Nesses casos, pode ser difícil calcular o erro de poluição numérica, uma vez que necessita-se determinar uma solução analítica estimada. Sabendo-se como se comporta o erro de poluição numérica, pode-se afirmar que $\varepsilon_{np}(\psi_m)$ é responsável pela alteração inesperada do perfil da solução numérica ψ_m em m = 0, mostrado na FIGURA 45 para o módulo do erro de discretização, na FIGURA 46 para a ordem aparente (p_U) e na FIGURA 47 para a ordem efetiva (p_E) . Os testes de coerência foram reproduzidos utilizando o FDM e obteve-se o mesmo comportamento que as soluções numéricas com o método SPH. O acoplamento $(\psi_m + \varepsilon_{np}(\psi_m))$ com o FDM também corrigiu o perfil das soluções ψ_m em m = 0.

Na TABELA 7 pode-se observar a comparação entre as ordens verdadeiras (p_m) de acurácia deduzidas *a priori* e as ordens aparentes (p_U) e efetivas (p_E) de acurácia

FIGURA 45 – ERRO DE DISCRETIZAÇÃO DA TEMPERATURA MÉDIA USANDO $W_{c,2}^1$ COM RRE (E2-1Dp).



FIGURA 46 – ORDEM APARENTE (p_U) DA TEMPERATURA MÉDIA USANDO $W_{c,2}^1$ COM RRE (E2-1Dp).



calculadas *a posteriori*. É fácil notar que a RRE corrigiu a falha que gerou uma ordem de acurácia sobrestimada para p_E ($p_0 = 4$). Ao analisar cuidadosamente, nota-se que a partir de p_1 os valores encontrados coincidem com os valores de p_m em todos os níveis. Este comportamento não foi repetido com p_U , que manteve os valores sobrestimados. É possível que este comportamento de p_E seja algo aleatório. É importante destacar que no caso em que a solução numérica não apresente incoerência nos testes numéricos, ainda





assim, é possível incorporar o termo do erro de poluição numérica e suas ordens de acurácia calculadas *a posteriori* serão coincidentes com as deduzidas *a priori*, desde que a solução analítica seja conhecida. Essa informação ficou evidente ao incorporar tanto os termos do erro de poluição numérica, quanto os do erro de truncamento à solução numérica, conforme descrito nas equações (5.35) e (5.37), respectivamente. Esse procedimento permitiu que o erro de discretização de $\psi(1/2)$ e ψ_m atingissem a ordem de magnitude do erro de arredondamento, ou seja, o erro verdadeiros dessas soluções atingiram o zero de máquina, como o esperado.

m	$p_m (a \ priori)$	$p_U (a \ posteriori)$	$p_E (a \ posteriori)$
0	$p_0 = 2$	$p_0 = 4$	$p_0 = 4$
1	$p_1 = 4$	$p_1 = 6$	$p_1 = 4$
2	$p_2 = 6$	$p_2 = 8$	$p_2 = 6$
3	$p_3 = 8$	$p_3 = 10$	$p_3 = 8$
4	$p_4 = 10$	NA	$p_4 = 10$
5	$p_5 = 12$	NA	$p_5 = 12$

TABELA 7 – COMPARAÇÃO ENTRE AS ORDENS DE ACURÁCIA DA VARIÁ-VEL ψ_m (E2-1Dp).

FONTE: O autor (2022).

5.3.3 Exemplo (E3-1Dp)

5.3.3.1 Verificação qualitativa de soluções numéricas para variável local

Nesse exemplo os testes de coerência são aplicados em um caso em que a condição de contorno em x = 0 é do tipo Neumann. Na FIGURA 48, apresenta-se o módulo do erro de discretização versus h até atingir o máximo de extrapolações de Richardson. A quantidade máxima de extrapolações é atingida quando o módulo do erro de discretização atinge a mesma ordem de magnitude que o erro de arredondamento. Veja que a inclinação das retas é diferente das FIGURAS 36 e 42. Esse fenômeno acontece porque as ordens de acurácia são reduzidas ao aproximar a derivada no contorno esquerdo em x = 0. Para explicar melhor esse fenômeno, utiliza-se as aproximações para a derivada primeira e segunda usando FDM.

$$\Psi^{1}(x_{i}) = \frac{-3\Psi_{i} + 4\Psi_{i+1} - \Psi_{i+2}}{2h} + \frac{\Psi_{i}^{3}}{3}h^{2} + \frac{\Psi_{i}^{4}}{4}h^{3} + \frac{7\Psi_{i}^{5}}{60}h^{4} + \dots$$
(5.39)

$$\Psi^{2}(x_{i}) = \frac{\Psi_{i-1} - 2\Psi_{i} + \Psi_{i+1}}{h^{2}} - \frac{\Psi_{i}^{4}}{12}h^{2} - \frac{\Psi_{i}^{6}}{360}h^{4} - \frac{\Psi_{i}^{8}}{20160}h^{6} + \dots$$
(5.40)

Aplicando a equação (5.39) no ponto x_0 e a equação (5.40) no ponto $x_1 = x_0 + h$, obtém-se

$$\Psi^{1}(x_{0}) = \frac{-3\Psi_{0} + 4\Psi_{1} - \Psi_{2}}{2h} + \frac{\Psi_{0}^{3}}{3}h^{2} + \frac{\Psi_{0}^{4}}{4}h^{3} + \frac{7\Psi_{0}^{5}}{60}h^{4} + \dots$$
(5.41)

$$\Psi^{2}(x_{1}) = \frac{\Psi_{0} - 2\Psi_{1} + \Psi_{2}}{h^{2}} - \frac{\Psi_{1}^{4}}{12}h^{2} - \frac{\Psi_{1}^{6}}{360}h^{4} - \frac{\Psi_{1}^{8}}{20160}h^{6} + \dots$$
(5.42)

Fazendo $\Psi^1(x_0) = g(x_0)$ e isolando a equação (5.41) para Ψ_0 , obtém-se

$$\Psi_0 = \frac{4\Psi_1 - \Psi_2 - 2hg(x_0)}{3} + \frac{2\Psi_0^3}{9}h^3 + \frac{\Psi_0^4}{6}h^4 + \frac{7\Psi_0^5}{90}h^5 + \dots \dots$$
(5.43)

Substituindo Ψ_0 da equação (5.43) na equação (5.42), determina-se a aproximação

$$\Psi_1^2 = \frac{-2\Psi_1 + 2\Psi_2}{3h^2} - \frac{2g(x_0)}{3h} + \frac{2\Psi_0^3}{9}h + \left[\frac{\Psi_0^4}{6} - \frac{\Psi_1^4}{12}\right]h^2 + \frac{7\Psi_0^5}{90}h^3 - \frac{\Psi^6}{360}h^4 + \dots$$
(5.44)

Substituindo Ψ_1^2 por $f(x_1)$ e passando do termo que depende de $g(x_0)$ para o lado esquerdo da equação (5.44), obtém-se

$$f(x_1) + \frac{2g(x_0)}{3h} = \frac{-2\Psi_1 + 2\Psi_2}{3h^2} + \frac{2\Psi_0^3}{9}h + \left[\frac{\Psi_0^4}{6} - \frac{\Psi_1^4}{12}\right]h^2 + \frac{7\Psi_0^5}{90}h^3 - \frac{\Psi^6}{360}h^4 + \dots \quad (5.45)$$

$$\frac{3hf(x_1) + 2g(x_0)}{3h} = \frac{-2\Psi_1 + 2\Psi_2}{3h^2} + \frac{2\Psi_0^3}{9}h + \left[\frac{\Psi_0^4}{6} - \frac{\Psi_1^4}{12}\right]h^2 + \frac{7\Psi_0^5}{90}h^3 - \frac{\Psi^6}{360}h^4 + \dots$$
(5.46)

$$3hf(x_1) + 2g(x_0) = \frac{-2\Psi_1 + 2\Psi_2}{h} + \frac{2\Psi_0^3}{3}h^2 + \left[\frac{\Psi_0^4}{3} - \frac{\Psi_1^4}{4}\right]h^3 + \frac{21\Psi_0^5}{90}h^4 - \frac{\Psi^6}{120}h^5 + \dots$$
(5.47)

As ordens verdadeiras mostradas na equação (5.47) justificam o comportamento mostrado na FIGURA 48, onde as discretizações com o método SPH se comportam igualmente ao FDM. Consequentemente, as ordens aparentes (p_U) mostradas na FIGURA 49 e efetivas (p_E) , FIGURA 50, confirmam os resultados deduzidos na equação (5.47). É importante destacar que a redução do incremento está diretamente relacionada com a aproximação da derivada no contorno x = 0, sendo que, se a aproximação mostrada na equação (5.47) tivesse $\Delta m = 2$, não haveria tal redução. Nesse caso, a aproximação local do contorno afeta as aproximações, em todos os nós da malha FDM e também sobre as partículas \mathbf{x}_i do método SPH. A redução do incremento em todo o domínio acontece porque a derivada em $\Psi^2(\mathbf{x}_2)$ depende de $\Psi(\mathbf{x}_1)$ (tanto em FDM quanto no SPH), de modo que no último nó (e/ou partícula) interno, a dependência da derivada nas coordenadas de \mathbf{x}_1 terá sido carregada.

FIGURA 48 – ERRO DE DISCRETIZAÇÃO DA TEMPERATURA NO PONTO MÉDIO USANDO $W_{c,2}^1$ COM RRE (E3-1Dp).



FONTE: O autor (2022).

A redução do incremento das ordens de acurácia não acontece na aproximação do contorno x = 0, onde sabe-se, por meio da equação (5.39), que seu valor é unitário





FIGURA 50 – ORDEM EFETIVA DA TEMPERATURA NO PONTO MÉDIO USANDO $W_{c,2}^1$ COM RRE (E3-1Dp).



 $(\Delta m = 1)$, conforme observa-se na FIGURA 51 para o módulo do erro de discretização; na FIGURA 52 para as ordens aparentes; e FIGURA 53 para ordens efetivas. Entende-se que a aproximação SPH para a derivada no contorno (equação (5.17)) é uma adaptação da aproximação mostrada na equação (5.39), por isso, suas ordens verdadeiras são as mesmas.

FIGURA 51 – ERRO DE DISCRETIZAÇÃO DA DERIVADA DA TEMPERATURA NO CONTORNO x = 0 USANDO $W_{c,2}^1$ COM RRE (E3-1Dp).



FONTE: O autor (2022).

FIGURA 52 – ORDEM APARENTE DA DERIVADA DA TEMPERATURA NO CONTORNO x = 0 USANDO $W_{c,2}^1$ COM RRE (E3-1Dp).



FONTE: O autor (2022).

5.3.3.2 Verificação qualitativa de soluções numéricas para variável global

Os efeitos da redução de incremento mostrados na Seção 5.3.3.1, FIGURAS 48-50, são estendidos à variável global ψ_m . Dessa forma, os resultados mostrados na FIGURA 54 para o módulo do erro de discretização; na FIGURA 55 para as ordens aparentes; e FIGURA 56 para ordens efetivas, possuem os mesmos perfis. A desvantagem que recai

FIGURA 53 – ORDEM EFETIVA DA DERIVADA DA TEMPERATURA NO CONTORNO x = 0 USANDO $W_{c,2}^1$ COM RRE (E3-1Dp).



sobre a redução do incremento, deve-se ao foto de que torna-se mais difícil atingir ordens mais elevadas. Por exemplo, para determinar oitava ordem de acurácia com um método de segunda ordem e incremento $\Delta m = 2$, são necessárias 3 extrapolações de Richardson. Enquanto isso, com um método também de segunda ordem de acurácia e incremento $\Delta m = 1$, precisa-se de 7 extrapolações.

FIGURA 54 – ERRO DE DISCRETIZAÇÃO DA TEMPERATURA MÉDIA USANDO $W_{c,2}^1$ COM RRE (E3-1Dp).



FONTE: O autor (2022).





FIGURA 56 – ORDEM EFETIVA DA TEMPERATURA MÉDIA USANDO $W^1_{c,2}$ COM RRE (E3-1Dp).



5.4 Equação da difusão de calor em regime transiente 1D

5.4.1 Exemplo (E1-1Dt)

5.4.1.1 Verificação qualitativa de soluções numéricas para variável local

Nessa seção analisa-se os resultados obtidos para o modelo matemático da difusão de calor em regime transiente. Na FIGURA 57 pode-se notar que o módulo de erro de discretização diminui mais lentamente do que em casos da difusão em regime permanente. No entanto, pode-se determinar soluções ainda mais acuradas.

FIGURA 57 – ERRO DE DISCRETIZAÇÃO DA TEMPERATURA NO PONTO MÉDIO EM t = 1.0s USANDO $W_{c,2}^1$ COM RRE (E1-1Dt).



Nota-se na FIGURA 58 três soluções numéricas com ordem de acurácia superior ao valor esperado $p_4 = 10$. Nestes casos, a solução já está sofrendo influências do erro de arredondamento e por esse motivo, a ordem de acurácia relativa ao nível m = 4 já não apresenta comportamento de convergência assintótica.

Analisando-se as ordens efetivas encontradas para a variável $\psi(1/2)$, percebeu-se que foi possível reduzir significativamente o módulo do erro de discretização, de modo que determinou-se uma solução numérica com décima sexta ordem de acurácia ($p_7 = 16$) e com sete extrapolações de Richardson (m = 7), conforme observa-se na FIGURA 59. Em termos do módulo de erro de discretização, foi possível encontrar em m = 0 o valor de 8,87E–10 com 2049 partículas e além disso, atingir no nível m = 7 o valor de 3,39E–32 associado também à discretização com 2049 partículas. Essa redução foi cerca de 22 ordens de magnitude. Nesse exemplo, utilizou-se 13 níveis de discretizações por partículas,

FIGURA 58 – ORDEM APARENTE (p_U) DA TEMPERATURA NO PONTO MÉDIO EM t = 1,0s USANDO $W_{c,2}^1$ COM RRE (E1-1Dt).



sempre com q = 2, variando os espaços médios uniformes de $N_p = 8$ até $N_p = 32768$ mas percebeu-se que não é necessário utilizar as discretizações com $N_p > 4096$, uma vez que de $N_p = 8$ até $N_p = 2048$ já foi possível determinar o menor erro verdadeiro para a variável.

FIGURA 59 – ORDEM EFETIVA (p_E) DA TEMPERATURA NO PONTO MÉDIO EM t = 1,0s USANDO $W_{c,2}^1$ COM RRE (E1-1Dt).



FONTE: O autor (2022).

5.4.1.2 Verificação qualitativa de soluções numéricas para variável global

Na FIGURA 60 pode-se observar que foi possível aplicar a RRE até ao nível m = 7, onde o erro de discretização atingiu a magnitude do erro de arredondamento. Nota-se também que no nível m = 7 determinou-se duas soluções não influenciadas pelo erro de arredondamento. Essas soluções estão associadas a $N_{\bar{t}} = 1025$ e $N_{\bar{t}} = 2049$ partículas, respectivamente. Os resultados apresentados mostram que não há necessidade de resolver o modelo numérico utilizando muitos valores diferentes para N_p . Ao utilizar $N_{\bar{t}} = 2049$ partículas para discretizar o domínio, conseguiu-se o número máximo de extrapolações (m = 7), de modo que o erro de discretização não tenha influência do erro de arredondamento. Resolver sistemas para $N_{\bar{t}} >= 4095$ partículas, é um esforço computacional desnecessário, nesse caso.

FIGURA 60 – ERRO DE DISCRETIZAÇÃO DA TEMPERATURA MÉDIA EM t = 1,0sUSANDO $W_{c,2}^1$ COM RRE (E1-1Dt).



FONTE: O autor (2022).

Os resultados da FIGURA 61 mostram que foi possível obter soluções de oitava ordem de acurácia com a RRE usando p_U . Nota-se que para p_4 e p_5 não há comportamento assintótico, por isso acredita-se que nesse caso a ordem mais alta de acurácia obtida foi a oitava.

Sabe-se que é sempre possível obter ordens de maior acurácia usando p_E do que com p_U . Isso acontece porque são necessárias apenas duas soluções numéricas e a solução analítica para obter p_E . Por outro lado, para calcular p_U , são necessárias três soluções numéricas. Esses requisitos associados a RRE tornam possível a obtenção de ordens mais





elevadas de acurácia ao utilizar p_E no cálculo. É fácil ver isso ao comparar a FIGURAS 61 e 62. Nota-se que em $p_7 = 16$ há uma solução com décima sexta ordem de acurácia. Isso acontece porque as soluções numéricas obtidas com o maior número de partículas são as primeiras a serem influenciadas pelo erro de arredondamento. Em outras palavras, os efeitos do erro de arredondamento degeneram a ordem de acurácia e aparecem sempre no lado esquerdo do gráfico. Entretanto, a perda de soluções ao aplicar a RRE aparece no lado direito do gráfico.

Na FIGURA 63 apresenta-se uma comparação entre as verdadeiras ordens deduzidas a priori (p_m) e as ordens efetivas (p_E) das variáveis de interesse $\psi(1/2)$ e ψ_m determinadas a posteriori. As ordens apresentadas são as mais altas obtidas em cada uma das extrapolações até atingir a décima sexta ordem, que foi o melhor resultado obtido entre todos os exemplos. Na prática, selecionou-se a solução numérica mais precisa para as variáveis de interesse em cada extrapolação e associou-se a seu valor correspondente h. Esse gráfico tem a finalidade de facilitar a compreensão de que não é preciso usar valores muito pequenos para h com a finalidade de determinar soluções numéricas com alta ordem de acurácia. Escolheu-se p_E porque é o método mais realista, uma vez que a solução analítica é conhecida em todos os exemplos. É importante notar que p_m é um conjunto infinito, pois é uma forma analítica de determinar as ordens de acurácia. Por outro lado, tanto p_E como p_U são conjuntos finitos de ordens de acurácia, uma vez que são obtidos numericamente e com limitações computacionais.

Para encerrar as análises dos exemplos unidimensionais, apresenta-se na TABELA





FIGURA 63 – COMPARAÇÃO DAS ORDENS EFETIVAS (p_E) DOS EXEMPLOS (E3-1Dp) e (E1-1Dt).



FONTE: O autor (2022).

8 os valores numéricos obtidos para as integrais que compõem os termos do erro de truncamento mostrados na equação (5.15). Em suma, as integrais correspondem aos

valores de $-h^2$, $-h^4$, $-h^6$, ..., $-h^{\gamma-2}$ devido ao fato de que a equação da difusão de calor em regime permanente é dada na forma $-\nabla^2 \Psi(\mathbf{x}_i) = f(\mathbf{x}_i)$. Com isso, ao observar os resultados da tabela, pode-se concluir que as integrais calculadas numericamente determinaram aproximações exatas para os valores de $-h^{\gamma-2}$, conforme dedução *a priori*. Se a aproximação fosse por CDS-2 usando o FDM, por exemplo, os valores de *h* seriam explícitos na expansão por série de Taylor e não seria necessário calcular as integrais para determiná-los.

h	$O(\mathbf{x}_i - \mathbf{x}_j)^2$	$O(\mathbf{x}_i - \mathbf{x}_j)^4$	$O(\mathbf{x}_i - \mathbf{x}_j)^6$
1,25E - 01	-1,56E - 02	-2,44E - 04	-3,81E - 06
6,25E - 02	-3,90E - 03	-1,52E - 05	-5,96E - 08
3,12E - 02	-9,76E - 04	-9,53E - 07	-9,31E - 10
1,56E - 02	-2,44E - 04	-5,96E - 08	-1,45E - 11
7,81E - 03	-6,10E - 05	-3,72E - 09	-2,27E - 13
3,90E - 03	-1,52E - 05	-2,32E - 10	-3,55E - 15
1,95E - 03	-3,81E - 06	-1,45E - 11	-5,55E - 17
9,76E - 04	-9,53E - 07	-9,09E - 13	-8,67E - 19
4,88E - 04	-2,38E - 07	$-5,\!68E - 14$	-1,35E - 20
2,44E - 04	-5,96E - 08	-3,55E - 15	-2,11E - 22
1,22E - 04	-1,49E - 08	-2,22E - 16	-3,30E - 24
6,10E - 05	-3,72E - 09	-1,38E - 17	-5,16E - 26
3,05E - 05	-9,31E - 10	-8,67E - 19	-8,07E - 28
1,52E - 05	-2,32E - 10	-5,42E - 20	-1,26E - 29
		()	

TABELA 8 – TERMOS DO ERRO DE TRUNCA-MENTO (E1-1Dp)

FONTE: O autor (2022).

Na TABELA 9 apresentam-se as soluções *benchmark* obtidas com o método SPH para a variável primária $\psi(1/2)$. A variável secundária ψ_m foi resolvida com FDM, sendo assim, não é apropriado afirmar que produzimos soluções de referência para um método tão clássico. Pode-se observar que usando o SPH e a RRE, determinou-se o erro de discretização na ordem de magnitude do erro de arredondamento, o que torna os resultados mais satisfatórios que os mostrados em (MERGA; CHEMEDA, 2021), quando resolveu-se a mesma equação da difusão de calor em regime transiente usando FDM.

TABELA 9 – SOLUÇÕES *BENCHMARK* OBTIDAS COM SPH PARA A VARIÁVEL $\psi(1/2)$.

Exemplo Solução analítica Solução numérica Erro numéri	co verdadeiro
$\hline \texttt{E1-1Dp} 1,64872127070012814684865078781416E + 00 1,64872127070012814684865078781416E + 00 3,4666738998970245356667389989702453566673899897024535667389989702453566673899897024556667389989702455666789970907666678997970700766678979707007667667897970707070707070707070707070707070707$	50076029665286E - 33
$\label{eq:2.1} {\rm E2-1} {\rm Dp} 1,00000000000000000000000000000000000$	67383574067442E - 32
= E3-1Dp = 1,64872127070012814684865078781416E + 00 = 1.64872127070012814684865078781416E + 00 = 1,925929944387235853 = 1,92592994438723585 = 1,92592994438723585 = 1,92592994438723585 = 1,92592994438723585 = 1,92592994438723585 = 1,92592994438723585 = 1,92592994438723585 = 1,92592994438723585 = 1,92592994438723585 = 1,9259294438723585 = 1,9259294438723585 = 1,92592994438723585 = 1,9259294438723585 = 1,92592945 = 1,92592945 = 1,9259294438725585 = 1,9259294438725585 = 1,92592945 = 1,9259294438725585 = 1,9259294438725585 = 1,92592945 = 1,92592945 = 1,92592945 = 1,92592945 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259295 = 1,9259505 = 1,9259505 = 1,92595 = 1,92595 = 1,925950000 = 1,925950	05597794258492E - 34
$\label{eq:constraint} \begin{array}{ c c c c c c c c c c c c c c c c c c c$	83180429840280E - 32

FONTE: O autor (2022).

5.5 Equação da difusão de calor em regime permanente 2D

Para iniciar as análises dos exemplos bidimensionais, primeiramente define-se alguns parâmetros para as simulações, conforme mostrado na TABELA 10. Esses parâmetros referem-se à tolerância (critério de parada do *solver*) das iterações internas e ciclos externos usadas no AML e os fatores de mudança de níveis. Os valores ótimos paras esses parâmetros foram apontados em (SUERO, 2010).

Parâmetro	AML	SL
Tolerância	1,0E - 08	1,0E - 08
Iterações internas	2	—
Ciclos externos	7	_
Fator de redução de nível	0,25	_
Fator de forte dependência no nível inconsistente	$0,\!35$	—
FONTE: O autor (2022)	•	

TABELA 10 – PARÂMETROS DAS SIMULAÇÕES.

Na TABELA 11, apresenta-se o Grau de Esparsidade (GE) das matrizes associadas ao sistema de equações lineares, obtidos ao discretizar o modelo matemático da difusão de calor em regime permanente com o método SPH. Essa informação é importante, pois não deve-se cometer o erro de resolver os sistemas na forma matricial. Nessa tese, utilizou-se o método de compactação CSR. Para exemplificar, caso fosse utilizado o Matlab[®] para resolver o sistema na forma matricial, em alguns casos, não seria possível calcular o sistema de ordem $127^2 \times 127^2$, pois o Matlab[®] apresentaria erro por falta de memória que o próprio *software* gerencia automaticamente (essa informação pode variar de acordo com a quantidade de memória RAM do computador utilizado). Além disso, a compactação torna-se vantajosa por que é capaz de acelerar a convergência da solução dos sistemas em relação a forma matricial.

TABELA 11 – GRAU DE ESPARSIDADE DAS MATRIZES.

Espaços médios uniformes entre partículas	Ordem da Matriz	GE
8×8	$7^2 imes 7^2$	$64,\!9729\%$
16 imes 16	$15^2 \times 15^2$	$90,\!5956\%$
32×32	$31^2 \times 31^2$	$97,\!5960\%$
64×64	$63^2 \times 63^2$	$99,\!3939\%$
128×128	$127^2 \times 127^2$	$99,\!8479\%$
256×256	$255^2 \times 255^2$	$99,\!9619\%$
512×512	$511^2 \times 511^2$	$99,\!9905\%$
1024×1024	$1023^2\times1023^2$	$99,\!9976\%$
2048×2048	$2047^2 \times 2047^2$	$99,\!9994\%$
4096×4096	$4095^2 \times 4095^2$	99.9999%

FONTE: O autor (2022).

5.5.1 Exemplo (E1-2Dp)

5.5.1.1 Verificação qualitativa de soluções numéricas para variável local

Na FIGURA 64 pode-se notar que não é possível reduzir significativamente a ordem do erro de discretização usando a RRE, conforme nos exemplos 1D. O fato do erro de discretização não ter sido reduzido conforme o esperado, não significa que a RRE tenha sido ineficiente. Na verdade, o problema aqui é outro. Nos exemplos 1D, as derivadas das funções núcleo Spline cúbica (W_c^1) e new quártico (W_d^1) foram capazes de cumprir os momentos do núcleo, ou seja, não apresentaram o erro de suavização (ε_W) mostrado na Seção 5.1.7. Ao utilizar a função núcleo Spline quíntica (apropriada para os modelos 2D), o momento $\overrightarrow{M_1^1} = \mathbf{1}$ não foi cumprido (equação (5.34)), pois o valor obtido numericamente foi $\approx 1,17$ (O momento é um vetor, ou seja, o valor 1,17 é obtido para cada uma das partículas se houver tratamento de fronteira, como feito nesse trabalho), sendo que esse resultado é constante em todos os níveis de discretização e depende apenas da expressão da derivada do núcleo escolhido. Quando o momento não é cumprido, estamos diante do erro de suavização e como consequência natural, a representação do fenômeno matemático denominado consistência não pode ser observada, ou seja, a diminuição do erro de discretização versus refinamento deixa de ocorrer conforme o esperado. Na prática, substituindo-se a solução analítica $\Psi(1/4,1/4) \in \Psi(1/8,1/8)$ no esquema numérico SPH com CSPM, é possível determinar 3 extrapolações ($p_E = \{2,4,6,8\}$ e $p_U = \{2,4\}$), em ambos os exemplos 2D, até atingir as operações impossíveis, que são ocasionadas por indeterminações durante o cálculo das ordens de acurácia. Isso significa que se for possível determinar soluções numéricas consistentes, a RRE vai funcionar corretamente. Os teste usando a variável $\Psi(1/2,1/2)$ falharam. Outro efeito que o erro de suavização causa, além da perda de consistência, é a convergência para um ponto de acumulação que não coincide com a solução analítica (mas na vizinhança), isto é, em alguns casos, pode fazer p_U coincidir com algumas ordens de p_V mas o mesmo não acontece com p_E .

As ordens aparentes mostradas na FIGURA 65 indicam que os valores encontrados *a posteriori* não são os mesmos obtidos na dedução *a priori*. Isso apenas confirma que a função núcleo possui propriedades capazes de reduzir a ordem de acurácia das soluções numéricas obtidas com o SPH. Não há erro de inconsistência, pois realizou-se o tratamento de fronteira usando partículas *dummy*, isso implica que todos os suportes compactos (V_i) associados às partículas \mathbf{x}_i possuem o mesmo número de partículas \mathbf{x}_j , de modo que a simetria seja garantida. Em outras palavras, o efeito que causa a redução das ordens aparentes, recai sobre a expressão do núcleo e suas derivadas.

Ao observar a FIGURA 66, nota-se algumas curiosidades. A primeira delas é que p_0 converge para ordem 2 até o nível 129 × 129; a segunda nos diz que nesse mesmo nível 129 × 129, determinou-se soluções com ordens $p_0 = 2$, $p_1 = 3$ e $p_2 = 4$. Analisando-se

FIGURA 64 – ERRO DE DISCRETIZAÇÃO DA TEMPERATURA NO PONTO MÉDIO USANDO $W_{e,24}^1$ COM RRE (E1-2Dp).



FIGURA 65 – ORDEM APARENTE DA TEMPERATURA NO PONTO MÉDIO USANDO $W^1_{e,24}$ COM RRE (E1-2Dp).



novamente a FIGURA 64, pode-se perceber que nesse mesmo nível de discretização, o módulo do erro mostrou-se com maior redução (exceto para m = 0). É possível que alguma
investigação detalhada envolvendo as aproximações até o nível de discretização 257×257 , indique alguma característica de melhor funcionamento das propriedades do núcleo.

FIGURA 66 – ORDEM EFETIVA DA TEMPERATURA NO PONTO MÉDIO USANDO $W_{e,24}^1$ COM RRE (E1-2Dp).



FONTE: O autor (2022).

5.5.1.2 Verificação qualitativa de soluções numéricas para variável global

Analisando-se a FIGURA 67, onde mostra-se o módulo do erro de discretização versus h para a variável ψ_m , observou-se o mesmo comportamento daquele mostrado na FIGURA 64, porém, a maior redução do erro encontra-se no nível 257 × 257. Exatamente o nível de menor erro verdadeiro da variável $\psi(1/2)$, onde a ordem efetiva p_0 calculada, ultrapassou seu valor analítico que é $p_L = 2$.

Na FIGURA 68 novamente tem-se a informação de que o núcleo reduziu em uma ordem de acurácia *a posteriori*, os valores deduzidos *a priori*. Tal afirmação decorre dos resultados obtidos nos exemplos unidimensionais, onde os momentos do núcleo foram cumpridos e tanto p_U quanto p_E calculadas *a posteriori* confirmaram a teoria. Deve-se tomar o cuidado de não afirmar que o método é de primeira ordem, pois seria uma afirmação falsa, uma vez que a dedução analítica comprova que a aproximação é de segunda ordem.

Curiosamente, na FIGURA 69, especificamente no nível 257×257 , as ordens efetivas determinadas ($p_E = \{2,4,6,8,10\}$) coincidem com a dedução *a priori*. É importante ressaltar que a temperatura média é calculada usando-se a regra do trapézio para o FDM.





FIGURA 68 – ORDEM APARENTE DA TEMPERATURA MÉDIA USANDO $W_{e,24}^1$ COM RRE (E1-2Dp).



FONTE: O autor (2022).

Diferentemente dos resultados mostrados na TABELA 8, a TABELA 12 confirma o erro de suavização que a função núcleo provoca sobre as aproximações. As integrais





contidas na equação geral do erro de truncamento (equação (5.27)) foram calculadas e apresentadas de modo que seja possível repetir as analises feitas sobre a TABELA 8. Com isso, pode-se perceber que os valores numéricos se aproximam dos analíticos, porém não atingem a exatidão. Como esses valores dependem prioritariamente da expressão da derivada do núcleo e o esquema numérico para aproximar o laplaciano bidimensional também, ambos estão contaminados com o erro de suavização. Eliminar o erro de suavização na aproximação bidimensional do operador laplaciano, requer a continuidade desse estudo.

h	$O[(x_i - x_j)^2, (y_i - y_j)^2]$	$O[(x_i - x_j)^4, (y_i - y_j)^4]$	$O[(x_i - x_j)^6, (y_i - y_j)^6]$
1,25E - 01	-2,26E - 02	-7,94E - 04	-3,99E - 05
6,25E - 02	-5,66E - 03	-4,96E - 05	-6,23E - 07
3,12E - 02	-1,41E - 03	-3,10E - 06	-9,74E - 09
1,56E - 02	-3,54E - 04	-1,93E - 07	-1,52E - 10
7,81E - 03	-8,85E - 05	-1,21E - 08	-2,37E - 12
3,90E - 03	-2,21E - 05	-7,57E - 10	-3,71E - 14
1,95E - 03	-5,53E - 06	-4,73E - 11	-5,80E - 16
9,76E - 04	-1,38E - 06	-2,95E - 12	-9,07E - 18
4,88E - 04	-3,45E - 07	-1,84E - 13	-1,41E - 19
2,44E - 04	-8,64E - 08	-1,15E - 14	-2,21E - 21

TABELA 12 – TERMOS DO ERRO DE TRUNCAMENTO(E1-2Dp).

FONTE: O autor (2022).

5.5.1.3 Verificação quantitativa de soluções numéricas

Conforme mencionado anteriormente, a verificação quantitativa de soluções numéricas aborda aspectos sobre tempo de processamento, coerência entre resultados obtidos com paralelização e teorias. Essa metodologia é recomendável apenas para modelos matemáticos 2D ou 3D, afinal, o *solver* utilizado para resolver o sistema associado ao modelo 1D possui, em geral, ordem de complexidade unitária (STÜBEN, 1983; RUGE; STÜBEN, 1987; WESSELING, 2004). Aliado a essa informação, tem-se que a programação dos métodos *multigrid/multilevel* tornam-se desnecessárias devido sua complexidade para atingir desempenho do *solver single level*.

Na FIGURA 70 apresenta-se o tempo de CPU em segundos versus número de níveis (N_L) do AML em escala semilog-y. Nos três níveis de discretização do domínio usando partículas, o AML mostrou-se mais eficiente quando utilizado o número máximo de níveis (do consistente, passando pelos intermediários até atingir o nível inconsistente, ver FIGURA 14). Essa informação confirma a teoria de robustez do método. O tempo de processamento para alguns níveis foram desconsiderados, pois, por exemplo, utilizando-se apenas o nível $N_L = 1$ para $N_t = 4097 \times 4097$, a simulação demoraria algo superior a dezenas de anos.





Para verificar a eficiência do AML, realizou-se também a comparação com a para-

lelização usando a interface de programação de aplicativo chamada Multi-processamento Aberto (do inglês, Open Multi-Processing, OpenMP). Dessa forma, usando a lei de Amdahl (AMDAHL, 1967), encontrou-se a fração paralelizável de 90% para o solver Gauss-Seidel-S CSR descrito no ALGORITMO 4. A fração paralelizável de 90% foi obtida com a média aritmética de um número superior a 4 simulações por CPU (τ) e descartando-se os tempos do solver serial quando apresentavam um tempo muito elevado em relação aos demais. Em suma, a razão entre o tempo do solver serial e paralelizado usando-se 1 CPU deve estar muito próximo da unidade. Esse parâmetro deve ser usado como base para eliminar soluções com tempo de processamento muito oscilatórios. Na FIGURA 71 essa informação pode ser constatada. Além disso, verifica-se que a fração paralelizada de 90% usando-se 28 CPU's, produz um speed-up $S_{28} = 8$.





FONTE: O autor (2022).

A equação que mede a ordem de complexidade sobre operações computacionais escolhida para determinar essa métrica é dada por

$$t_{cpu}(N_t) = c(N_t)^{\overline{p}},\tag{5.48}$$

onde $t_{cpu}(N_t)$ é o tempo de processamento de cada nível de discretização, N_t é o número de variáveis, c uma constante de proporcionalidade e \overline{p} a ordem de complexidade (TROT-TENBERG; OOSTERLEE; SCHÜLLER, 2001; WESSELING, 2004; BURDEN; FAIRES; BURDEN, 2016).

Na TABELA 13 apresentam-se os valores da métrica \overline{p} (chamado de $\overline{p_u}$ para diferenciar o caso de discretização desordenada) para o *solver* Gauss-Seidel-S CSR SL serial, paralelizado e AML serial. Os valores obtidos estão de acordo com o previsto na literatura, onde sabe-se que $\overline{p_u} = 2$ para os métodos SL e $\overline{p_u} = 1$ para o AML (BRIGGS; HENSON; MCCORMICK, 2000; WESSELING, 2004). Além disso, calculou-se também a mesma métrica para o recrutamento das partículas, que realizado por força bruta é da ordem $O(N_t^2)$. Como o recrutamento foi realizado por subdomínio (ver FIGURA 10) (PAIVA *et al.*, 2009), a ordem de complexidade passou para $\overline{p_u} = 1$ (equivalente a $O(N_t)$), conforme esperava-se.

Método	с	$\overline{p_u}$		
Gauss-Seidel-S SL serial	1,996E - 06	2,013		
Gauss-Seidel-S SL paralelizado	1,672E - 04	$1,\!492$		
Gauss-Seidel-S AML serial	8,878E - 05	$1,\!005$		
Recrutamento de partículas (Seção 3.3)	1,902E - 04	$1,\!004$		
FONTE: O autor (2022).				

TABELA 13 - ORDEM DE COMPLEXIDADE.

Na FIGURA 72 apresenta-se o tempo de CPU em segundos (escala semilog-y) versus a raiz quadrada do número de variáveis, para os solvers SL, SL paralelizado, AML (serial) e AML com projeção de paralelização ($S_{28} = 8$, baseada na fração paralelizável determinada, $f_p = 90\%$). Os resultados mostram que a paralelização somente torna-se eficiente a partir do nível 65 × 65, pois nos níveis mais inconsistentes de partículas o custo da paralelização é maior do que o tempo necessário para resolver o sistema com o SL. A reta do AML possui inclinação diferente do Gauss-Seidel-S CSR SL e SL paralelizado, de modo que, quanto maior o número de variáveis, maior será o ganho computacional. A projeção do AML é uma reta paralela a do AML, pois significa que a paralelização do AML possibilita um speed-up em torno de 8 vezes sobre o AML serial.

Para evidenciar a eficiência do AML, na FIGURA 73 destaca-se o speed-up obtido com a paralelização, com o AML serial e com a projeção do AML usando $f_p = 90\%$. Novamente verifica-se que para o SL paralelizado o ganho somente acontece a partir do nível 65×65 , ou seja, o speed-up só é maior do que 1 no nível 129×129 . Enquanto isso, o AML e a projeção do AML paralelizado já são mais vantajosos desde o nível 9×9 . No nível 513×513 o Gauss-Seidel-S CSR SL confirma o $S_{28} = 8$, enquanto isso, o AML determina um speed-up de $S_p = 4084$ vezes e sua projeção $S_p = 32670$ vezes. É importante destacar que o solver Gauss-Seidel-S CSR SL serial é o mesmo solver contido no AML e SL paralelizado. A única diferença entre eles é a paralelização. É possível que testes usando outros solver's possam determinar valores superiores para os speed-ups. Entretanto, essa análise não é fundamental para a pesquisa apresentada nessa tese.



FIGURA 72 – TEMPO DE CPU VERSUS NÚMERO DE PARTÍCULAS (E1-2Dp).

FIGURA 73 – SPEED-UP VERSUS NÚMERO DE PARTÍCULAS (E1-2Dp).



5.5.2 Exemplo (E2-2Dp)

O segundo exemplo bidimensional apresentou os mesmos perfis dos resultados mostrados no exemplo (E1-2Dp). Dessa forma, para evitar repetições de comentários,

optou-se por não apresentá-los.

5.6 Aplicação

5.6.1 Uso do AML em discretização com partículas canonicamente desordenadas

De acordo com Chaussonnet *et al.* (2015), a desordem (ou perturbação) canônica é definida como uma medida atribuída às partículas na configuração inicial, ou seja, sobre a discretização uniforme que considera $h = \Delta x = \Delta y = 1/N_{p_z}$. Essa medida é dada por

$$R = \eta \Delta x, \tag{5.49}$$

onde $\eta \in [0; 0, 5[$. A representação da desordem canônica definida pela equação (5.49) pode ser observada na FIGURA 74. A perturbação canônica impõe um deslocamento multidirecional, de modo que, a partícula se movimenta apenas dentro da região limitada pela circunferência de raio R. Em outras palavras, a partícula 41 não invade o espaço limitado de alguma das partículas vizinhas 31, 32, 33, 40, 42, 49, 50 e 51. Essa dinâmica fica evidente na FIGURA 75. Essa configuração impede a colisão entre as partículas, permitindo assim que análises sejam feitas apenas sob o efeito da desordem.

FIGURA 74 – REPRESENTAÇÃO DO RAIO DE INFLUÊNCIA PARA DISCRETIZAÇÃO COM DESORDEM CANÔNICA DE PARTÍCULAS.



5.6.2 Discretização canônica com partícula sensor fixa

Introduz-se como discretização canônica com partícula sensor a forma de discretização de domínio baseada em (CHAUSSONNET *et al.*, 2015). A diferença é caracterizada pela atribuição de uma ou mais partículas sensores (\mathbf{x}_s), de modo que, a desordem seja aplicada em todas as partículas internas ao domínio, exceto sobre \mathbf{x}_s .

Definição 10. Denomina-se por partícula sensor fixa $\mathbf{x}_s(\mathbf{Q})$ ou $\mathbf{x}_s(\mathbf{Q},t)$, onde \mathbf{x}_s é toda e qualquer partícula que possua coordenadas coincidentes \mathbf{Q} em todos os níveis de discretização e passos de tempo, simultaneamente.

A partir da definição 10, é possível descrever a RRE generalizada para o método SPH, incluíndo os casos em que as partículas possuem movimento. Essa descrição encontrase detalhada no apêndice B.

Na FIGURA 75, a partícula sensor foi definida como $\mathbf{x}_s = \mathbf{x}_s(1/2,1/2)$, ocupandose a posição 25 (nesse caso) obtida com ordenação lexicográfica. Pode-se destacar que, em todos os casos com discretização uniforme abordados nessa tese, utiliza-se a partícula sensor para capturar a temperatura da variável local escolhida. Em outras palavras, quando deseja-se determinar a temperatura da variável $\psi(1/2)$ ou $\psi(1/2,1/2)$, utiliza-se a partícula sensor $\mathbf{x}_s(1/2)$ ou $\mathbf{x}_s(1/2,1/2)$. Desta forma, é equivalente dizer que $\psi(1/2) = \psi(\mathbf{x}_s)$ e $\psi(1/2,1/2) = \psi(\mathbf{x}_s)$, para $\mathbf{x}_s(1/2)$ e $\mathbf{x}_s(1/2,1/2)$, respectivamente.

Partindo da configuração inicial que considera as partículas uniformemente distribuídas, escolheu-se o valor $\eta = 0.49$ para impor uma desordem mais acentuada e a partir disso, mostrar que o AML é suficientemente robusto a ponto de resolver o sistema de equações lineares associado à discretização do modelo matemático proposto nos exemplos (E1-2Dp) e (E2-2Dp).

5.6.3 Equação da difusão de calor em regime permanente considerando a discretização canônica com partícula sensor fixa

É muito comum na literatura SPH (FRAGA FILHO, 2019) a resolução dos modelos matemáticos utilizando-se o método de Euler explícito. Nessa tese, tanto nos modelos em regime permanente, quanto transiente, resolve-se sistemas de equações lineares. No modelo matemático da difusão de calor em regime permanente, tem-se garantido a diagonal dominância das matrizes e no caso do regime transiente, a formulação implícita por meio do método de Crank-Nicolson garante a estabilidade incondicional. Dessa forma, para contribuir com trabalhos futuros, propõe-se a metodologia de desordenação canônica usando a discretização com partícula sensor fixa. Com isso, pretende-se reafirmar a robustez do AML e permitir que, futuramente, análises com os mesmos testes de coerência possam ser utilizados e melhorados. Dessa forma, além de analisar o perfil de soluções numéricas

FIGURA 75 – REPRESENTAÇÃO DA DISCRETIZAÇÃO CANÔNICA COM PARTÍCULA SENSOR FIXA.



obtidas com qualquer tipo de discretização por partículas, tem-se a possibilidade de aumentar a ordem de acurácia das soluções obtidas com o SPH.

As FIGURAS 76 e 77 mostram as soluções analítica e numérica do exemplo (E1-2Dp) no nível 65×65 partículas usando-se a discretização canônica com partícula sensor fixa para $\eta = 0,499999$.

As FIGURAS 78 e 79 mostram as soluções analítica e numérica do exemplo (E2-2Dp) no nível 65×65 partículas usando-se a mesma discretização canônica com partícula sensor fixa para $\eta = 0,499999$.

As FIGURAS 80 e 81 mostram o desempenho do AML em tempo de CPU variando o número de níveis. Nos dois casos foram utilizados a discretização proposta pela metodologia que considera a desordem canônica com partícula sensor fixa e $\eta = 0,499999$. Nos mesmos dois exemplos, mas usando discretização uniforme, o tempo de CPU dos dois últimos níveis foram muito próximos. No caso da desordem, a decaimento do tempo computacional é mantido até que se atinja o número máximo de níveis. O nível 1 foi desconsiderado nas duas figuras porque o AML encerrou a execução da simulação por motivo de erro (que pode acontecer quando algum critério de execução do programa não é atendido). Além disso, é possível perceber que as soluções numéricas convergem com tempo maior que nos casos de discretização uniforme.



FIGURA 76 – SOLUÇÃO ANALÍTICA DO EXEMPLO (E1-2Dp).



FIGURA 77 – SOLUÇÃO NUMÉRICA DO EXEMPLO (E1-2Dp).



FONTE: O autor (2022).

Na FIGURA 82 repetiu-se a comparação do tempo de CPU versus raiz quadrada do número de partículas. Novamente o Gauss-Seidel-S CSR SL mostrou-se vantajoso a partir do nível 65×65 e o AML continua apresentando-se com maior eficiência à medida que o número de partículas aumenta e a projeção do AML paralelizado considera a mesma



FIGURA 78 – SOLUÇÃO ANALÍTICA DO EXEMPLO (E2-2Dp).



FIGURA 79 - SOLUÇÃO NUMÉRICA DO EXEMPLO (E2-2Dp).



FONTE: O autor (2022).

fração paralelizada $f_p=90\%$ que determina o $speed\mathchar`up\ S_{28}=8.$

Para facilitar as análises da eficiência do AML, na FIGURA 83 destaca-se os valores do *speed-up* sobre as curvas de cada *solver*. Os resultados para o SL paralelizado



FIGURA 80 – TEMPO DE CPU VERSUS NÚMERO DE NÍVEIS DO AML (E1-2Dp).

FIGURA 81 – TEMPO DE CPU VERSUS NÚMERO DE NÍVEIS DO AML (E2-2Dp).



foram propositalmente mostrados sem efetuar a médida aritmética e isso fica evidente ao observar a oscilação dos *speed-ups*. Entretanto, se for considerado o nível 1025×1025 e sucessores, a curva deve tender ao valor $S_{28} = 8$. O AML mostrou-se ainda mais eficiente ao considerar a desordem canônica, isso significa que o fato de apresentar convergência



FIGURA 82 – TEMPO DE CPU VERSUS NÚMERO DE PARTÍCULAS (E1-2Dp).

mais lenta para o Gauss-Seidel-S CSR SL, o AML determinou um $S_p = 5136$, superior ao $S_p = 4084$ mostrado na FIGURA 73.

FIGURA 83 – SPEED-UP VERSUS NÚMERO DE PARTÍCULAS (E1-2Dp).



Devido ao fato da convergência das soluções numéricas com a desordem canônica determinarem maior tempo computacional, a métrica \overline{p} escolhida para medir a ordem de complexidade das operações, demonstra seus efeitos. Nos três casos os valores são maiores do que os mostrados na TABELA 13. Em suma, se a ordem de complexidade do Gauss-Seidel-S SL serial é $\overline{p_u} = 2$ usando discretização uniforme, com a discretização desordenada o tempo é maior. Entende-se que o valor de $\overline{p_d}$ para esse caso será maior que 2, porém nunca será 3. Isso acontece porque não está sendo aumentado o número de partículas. O valor $\overline{p} = 2$ é determinado analiticamente, sendo que numericamente os valores são aproximados. Dessa forma, o valor $\overline{p_d} > \overline{p_u}$, onde $\overline{p_d}$ é a ordem de complexidade usando discretização desordenada e $\overline{p_u}$ uniforme. A ordem de complexidade para o recrutamento de partículas não varia de acordo com o tipo de discretização mostradas nessa tese, pois o número de operações permanece constante.

TABELA 14 –	ORDEM	DE	COMPLEXID.	ADE
	USANDO	DISC	CRETIZAÇÃO	DE-
	SORDENA	DA (I	E1-2Dp).	

Método	С	$\overline{p_d}$		
Gauss-Seidel-S SL serial	2,230E - 06	2,023		
Gauss-Seidel-S SL paralelizado	7,883E - 06	1,729		
Gauss-Seidel-S AML serial	7,414E - 05	1,227		
FONTE(O autor (2022))				

FONTE: O autor (2022).

Com isso, mostrou-se que é possível utilizar o AML para determinar soluções numéricas usando o método SPH, mesmo que a discretização seja desordenada. A importância desse resultado é verdadeiramente significativa, pois a partir dessa certeza, pode-se desenvolver novas tecnologias que sejam capazes de melhorar a ordem de acurácia das soluções numéricas. Vale reforçar que todas os esquemas numéricos adotados não possuem limitação por ausência de diagonal dominância, tampouco por critério de estabilidade. Isso pode mudar o estado da arte, onde simulações usando dezenas de milhões de partículas são necessárias. Além disso, a verificação quantitativa das soluções numéricas apontou a incomparável vantagem do AML sobre SL paralelizado. Dessa forma, agora sabe-se como se comportam as soluções numéricas obtidas com o SPH e também o erro de truncamento. Então, a metodologia de verificação qualitativa e quantitativa de soluções numéricas mostradas nessa tese, podem determinar a confiabilidade das soluções obtidas e estimar a capacidade de geração dessas soluções.

6 CONSIDERAÇÕES FINAIS

Na TABELA 15 há um resumo das variáveis de interesse e as ordens de acurácia obtidas *a priori* e *a posteriori* em todos os exemplos apresentados nessa tese. Nota-se que a variável primária $\psi(1/2)$ foi obtida com o método SPH e a variável secundária ψ_m com a regra do trapézio usando FDM. Isso é possível porque as partículas foram posicionadas propositalmente, de modo a coincidir com os nós das malhas do FDM. Ambos os métodos possuem segunda ordem de acurácia em m = 0. As duas variáveis de interesse determinam *a priori* as mesmas ordens de acurácia. Enquanto isso, as ordens de acurácia obtidas *a posteriori* seguem os mesmos perfis, exceto a variável ψ_m no exemplo (E2-1Dp) e das variáveis $\psi(1/2)$ e ψ_m no exemplo (E3-1Dp) devido ao erro de poluição numérica e aproximação do contorno com $\Delta m = 1$, respectivamente. Essas ordens de acurácia obtidas para $\psi(1/2)$ e ψ_m podem indicar que atingiu-e com o método SPH o mesmo perfil de excelência (para casos 1D) dos resultados já conhecidos ao utilizar FDM e FVM.

TABELA 15 – RESUMO DAS ORDENS DE ACURÁCIA ENCONTRADAS A PRIORI E A POSTERIORI.

Exemplos	Variáveis	p_m (a priori)	p_U (a posteriori)	p_E (a posteriori)	Figuras
E1-1Dp	$\psi(1/2)$	$\{2,4,6,8,\ldots\}$	{2,4,6,8}	$\{2,4,6,8,10\}$	37 e 38
E2-1Dp	$\psi(1/2)$	$\{2,4,6,8,\ldots\}$	$\{2,4,6,8,10\}$	$\{2,4,6,8,10,12\}$	$43 \mathrm{~e~} 44$
E3-1Dp	$\psi(1/2)$	$\{2,4,6,8,\ldots\}$	$\{2,3,4,5,6,7\}$	$\{2,3,4,5,6,7,8,9\}$	$49 \ \mathrm{e} \ 50$
E1-1Dt	$\psi(1/2,1)$	$\{2,4,6,8,\ldots\}$	$\{2,4,6,8\}$	$\{2,4,6,8,10,12,14,16\}$	$58 \ge 59$
E1-2Dp	$\psi(1/2,1/2)$	$\{2,4,6,8,\ldots\}$	$\{1,2,3\}$	$\{2,3,4\}$	$65 \ \mathrm{e} \ 66$
E1-1Dp	ψ_m	$\{2,4,6,8,\ldots\}$	$\{2,4,6,8\}$	$\{2,4,6,8,10\}$	$40~{\rm e}~41$
E2-1Dp	ψ_m	$\{2,4,6,8,\ldots\}$	$\{4,6,8,10\}$	$\{4,4,6,8,10,12\}$	$46 \ \mathrm{e} \ 47$
E3-1Dp	ψ_m	$\{2,4,6,8,\ldots\}$	$\{2,3,4,5,6\}$	$\{2,3,4,5,6,7,8,9\}$	$55 \ \mathrm{e} \ 56$
E1-1Dt	$\psi_m _{t=1,0s}$	$\{2,4,6,8,\ldots\}$	$\{2,4,6,8\}$	$\{2,4,6,8,10,12,14,16\}$	$61 \ \mathrm{e} \ 62$
E3-1Dp	$\psi(0)$	$\{2,3,4,5,\ldots\}$	$\{2,3,4,5,6,7\}$	$\{2,3,4,5,6,7,8,9\}$	$52 \ \mathrm{e} \ 53$
E1-2Dp	ψ_m	$\{2,4,6,8,\ldots\}$	$\{1,2,3\}$	$\{2,4,6,8,10\}$	68 e 69

FONTE: O autor (2022).

6.1 Conclusão

Descreveu-se como aplicar a RRE às soluções SPH a fim de diminuir o erro de discretização. Calculou-se as soluções numéricas variando o número de partículas de $N_{\bar{t}} = 9, \ldots, 65537$ para o modelo de difusão de calor em regime permanente 1D e $N_{\bar{t}} = 9, \ldots, 32769$ para o modelo da difusão de calor em regime transiente. O modelo matemático da difusão de calor em regime permanente 2D foi resolvido variando o número de partículas $N_t = 9 \times 9, \ldots, 4097 \times 4097$. Diminuiu-se o erro de discretização até atingir a magnitude do erro do arredondamento nos casos unidimensionais, verificando-se que não é necessário utilizar milhões de partículas para determinar soluções acuradas. Utilizou-se refinamento constante q = 2 na discretização do domínio. Para o modelo da difusão de calor em regime transiente, considerou-se $\Delta x = \Delta t$, pois a formulação é incondicionalmente estável e por meio dela foi possível diminuir o erro de discretização cerca de 22 ordens de magnitude que, por consequência, determinou décima sexta ordem de acurácia. Definiu-se fontes de erro numérico específicas para o método SPH. Mostrou-se os efeitos do erro de poluição numérica nas soluções obtidas com tal método. A demonstração do operador laplaciano 1D e 2D foi apresentada na íntegra mostrando tanto analítica quanto numericamente os termos do erro de truncamento e sua equação geral. Definiu-se a aproximação assimétrica para aproximar a derivada no contorno x = 0 e apresentou-se a dedução que mostra a redução das ordens de acurácia provocada por aproximações envolvendo diferentes tipos de Δm . Provou-se que o AML é capaz de resolver os problemas usando o método SPH, tanto com discretização uniforme quanto desordenada. Criou-se a metodologia da desordem canônica com partícula sensor fixa. Aplicou-se os testes de coerência e métricas para realizar a verificação qualitativa e quantitativa das soluções numéricas com o SPH, respectivamente. As métricas da verificação quantitativa mostram a eficiência do AML sobre o SL serial e paralelizado, de modo a garantir sem dúvidas que o AML é progressivamente mais eficiente, tornando-o atrativo. Identificou-se que a paralelização do AML é vantajosa. Descobriu-se que o erro de suavização reduziu as ordens de acurácia da aproximação usada no operador laplaciano 2D.

6.2 Contribuições

Esta seção aparesenta-se separada em dois grupos: a Seção 6.2.1 mostra os resultados orginais da tese que influenciam diretamente o estado da arte e a Seção 6.2.2 mostra os resultados determinados a partir da aplicação original de métodos já conhecidos na literatura e também a contribuição computacional por meio da disponibilização dos códigos após o registro no Instituto Nacional da Propriedade Industrial (INPI).

6.2.1 Contribuições originais

As contribuições da presente tese para o estado da arte são as seguintes:

- Deduziu-se os operadores laplaciano 1D e 2D com as devidas equações gerais do erro de truncamento usando o método SPH;
- Calculou-se os termos do erro de truncamento dos operadores laplaciano 1D e 2D usando o método SPH;
- Apresentou-se o Grau de Esparsidade das matrizes associadas à discretização com o método SPH;
- Definiu-se fontes de erro numérico específicas do método SPH;

- Determinou-se a dinâmica de recrutamento das partículas de modo que seja fácil a implementação usando paralelização;
- Deduziu-se a redução das ordens de acurácia por efeito da aproximação no contorno esquerdo x = 0;
- Mostrou-se como aproximar a derivada primeira no contorno usando um esquema SPH;
- Criou-se a definição da discretização canônica com partícula sensor fixa. A determinação da partícula sensor fixa permite que mesmo nos casos de movimentação das partículas, a RRE pode ser aplicada;
- Mostrou-se que a multiextrapolação de Richardson funciona para reduzir o erro de discretização e aumentar a ordem de acurácia das soluções numéricas obtidas com o método SPH;
- Determinou-se soluções numéricas de alta ordem de acurácia e com erro de discretização na ordem de magnitude do erro de arredondamento (usando precisão quádrupla). Isso significa que gerou-se soluções *benchmark* para as variáveis de interesse primárias com o SPH nos modelos unidimensionais apresentados;
- Aplicou-se o AML em discretização canônica com partícula sensor fixa, ou seja, mostrou-se que é possível resolver com o AML os sistemas lineares associados à discretização desordenada e ainda aplicar a RRE;
- Definiu-se os níveis de discretização: consistente, intermediário e inconsistente para padronizar as nomenclaturas ao agrupar o SPH com o AML.

6.2.2 Aplicações originais

As aplicações que formam a base fundamental das contribuições destacadas na Seção 6.2.1 são as seguintes:

- Mostrou-se como aplicar os testes de coerência para realizar a verificação qualitativa das soluções numéricas obtidas com o SPH;
- Aplicou-se as métricas para realizar a verificação quantitativa das soluções numéricas obtidas com o SPH;
- Apresentou-se o comportamento dos erros numéricos obtidos com SPH, originados de fontes já conhecidas para outros métodos clássicos;
- Deduziu-se a equação da difusão de calor em regime transiente usando a formulação implícita de Crank-Nicolson para obter soluções incondicionalmente estáveis;

- Mostrou-se como determinar a compactação das matrizes associadas à discretização com SPH para aplicação do *solver* Gauss-Seidel-S CSR SL paralelizado;
- Adaptou-se o *Algebraic Multilevel* para resolver os sistemas compactados com o método CSR e originados da discretização com o método SPH;
- Determinou-se a fração paralelizável do *solver* Gauss-Seidel-S CSR adaptado ao SPH;
- Calculou-se as ordens de complexidade para operações computacionais com discretização SPH uniforme e desordenada;
- Implementou-se os códigos computacionais em linguagem Fortran 95 com precisão quádrupla (Real*16 ou estendida), RRE e geração de relatórios completos. Isso foi pensado para facilitar a utilização dos códigos por cientistas interessados em comparar resultados e/ou dar continuidade nos estudos (trabalhos futuros). Os códigos podem ser disponibilizados à comunidade acadêmica logo após o registro no INPI.

6.3 Know-how

Embora o *know-how* não seja uma contribuição da tese, sem ele, tampouco a tese existiria. Por isso, faz-se necessário explicitar as limitações superadas.

- O sistema de equações lineares associados à discretização com o método SPH em modelos matemáticos 2D (nessa tese) possuem 25 diagonais e não apenas 5, como usando FDM, por exemplo. Essa característica tornou obrigatória a descoberta de um *solver* que fosse capaz de resolver tal sistema. Após a descoberta, primeiramente foi preciso criar a compactação das matrizes usando o método CSR, que por sua vez, foi adaptado; depois adaptou-se também o Gauss-Seidel para usar a nova compactação CSR, passando a ser Gauss-Seidel-S CSR.
- Foi preciso submeter um projeto para utilização do *Cluster* GridUNESP;
- Aprendeu-se a programar dentro do Cluster;
- A licença do Intel[®] Fortran no Cluster estava vencida e demorou cerca de 1 ano para que a desenvolvedora retornasse o pedido de renovação. Dessa forma, precisou-se aprender a utilizar o aplicativo Wine dentro do Cluster para converter o executável criado no Windows[®] para Linux[®] até que a licença fosse renovada;
- Após a renovação da licença, foi indispensável aprender programação paralela OpenMP usando o Windows[®] e só então reescrever o código para o Linux[®] do

Cluster. Isso acontece porque a política de utilização do *cluster* diz que testes devem ser realizados fora do servidor;

- Adaptou-se o AMG usando FDM com precisão dupla para tornar-se o AML com SPH usando precisão quádrupla;
- Paralelizou-se o código (E-2Dp) de ponta-a-ponta, exceto o Gauss-Seidel CSR dentro do AML; entre todas as etapas, a fase de recrutamento das partículas foi a paralelização mais complexa, de modo que, foi preciso subdividir em duas partes e apenas uma delas pode ser paralelizada. Lembrando-se que o recrutamento por força bruta possui ordem de complexidade per 2 e ao adotar a técnica de recrutamento por subdomínio aliada com a paralelização parcial, derrubou-se a ordem de complexidade para per 1;
- implementou-se um código otimizado, capaz de resolver diversos exemplos com ou sem desordem das partículas, mudança de coordenadas da variável de interesse local e criação de inúmeros relatórios de saída automatizados, como por exemplo: ordem efetiva; ordem aparente; módulo do erro de discretização; solução em cada nível do AML; solução em todos os níveis do AML com e sem RRE para todas as variáveis de interesse; mudança de SL serial para paralelizado ou AML, tempo de CPU por etapas, relatório geral (ver exemplo no Apêndice C), entre outros.

6.4 Trabalhos Futuros

- Resolução de modelos matemáticos que caracterizam escoamentos: a pesquisa mostrada nessa tese contribue para que seja feito um estudo de verificação de soluções numéricas obtidas com o SPH usando modelos matemáticos mais sofisticados, uma vez que agora somos capazes de identificar as fontes de erros numéricos (específicas do SPH) com riqueza de detalhes;
- Uso de geometrias complexas: com avanço na pesquisa, pode-se estudar a solução numérica obtida com o SPH em problemas reais, que usam geometrias não triviais, como o problema mostrado na FIGURA 1;
- Estudo aprofundado do erro de suavização (ε_W): é possível que esse item seja um dos mais importantes, pois, reduzir o erro de suavização implica em tornar o método consistente, capaz de diminuir o erro numérico à medida que se refina a discretização do domínio. Como ocorre com FDM e FVM;
- Estudo dos perfis de solução entre os níveis 9 × 9 até 513 × 513: esse item pode ser interessante sob o ponto de vista numérico, pois, pode-se encontrar características essenciais para determinar métodos de correções da inconsistência;

- Aplicação do método *Moving Least Square*: trata-se de verificar os resultados obtidos com a correção do núcleo usando mínimos quadrados;
- Aplicação do FPM e CSPM: trata-se de verificar o funcionamento das técnicas de correção da inconsistência por partículas (caso ideal para alunos de mestrado);
- Estimativa para o erro de poluição numérica (caso ideal para alunos de mestrado);
- Aplicação das funções assimétricas (*skewed*) para evitar criação de partículas *ghosts*: esse item permite investigar o comportamento das soluções numéricas sem o uso do tratamento de fronteira com partículas externas. Do ponto de vista computacional, pode-se otimizar muito a utilização de memória RAM e tempo de CPU com essa estratégia;
- Estimativas da solução numérica em níveis consistentes de discretização com base nas mudanças de níveis do AML: esse item pode ser interessante à luz da aplicação da RRE ou CRRE dentro dos níveis do *multigrid*. Basicamente a ideia é criar uma extrapolação dentro do *multigrid* para acelerar ainda mais a convergência das partículas;
- Uso do Esquema FAS no AML: permite a resolução de sistemas não lineares;
- Paralelização do AML usando Interface de Passagem de Mensagens (do inglês, Message Passing Interface, MPI): essa adaptação permite o uso de múltiplos worker nodes simultaneamente. Com isso, o multigrid teria sua eficência elevada aos níveis extremos em relação ao poder de processamento dos servidores;
- Aplicação a CRRE para extrapolar o domínio completo: ideal para determinar soluções *benchmark* em todo o campo de soluções.

REFERÊNCIAS

ABDELMIGID, T. A. *et al.* Revisiting the lid-driven cavity flow problem: Review and new steady state benchmarking results using GPU accelerated code. *Alexandria Engineering Journal*, v. 56, n. 1, p. 123–135, 2017. Citado na página 36.

ACHIM, C. V.; ROZAS, R. E.; TOLEDO, P. G. Semi-decoupled first-order correction for smoothed particle hydrodynamics. *Applied Mathematical Modelling*, Elsevier, v. 93, p. 314–325, 2021. Citado na página 34.

ADAMCZYK, W. P. *et al.* Modeling of particle transport and combustion phenomena in a large-scale circulating fluidized bed boiler using a hybrid Euler-Lagrange approach. *Elsevier*, v. 16, p. 29–40, 2014. Citado na página 32.

AMDAHL, G. M. Validity of the single processor approach to achieving large scale computing capabilities. In: *Proceedings of the April 18-20, 1967, spring joint computer conference.* [S.l.: s.n.], 1967. p. 483–485. Citado 2 vezes nas páginas 59 e 147.

ANTUONO, M. et al. A measure of spatial disorder in particle methods. Computer Physics Communications, v. 18, n. 10, p. 2609–2621, 2014. Citado na página 37.

ASME. Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer: An American National Standard. [S.l.]: American Society of Mechanical Engineers, 2009. Citado na página 31.

BAUER, F. L. Computational graphs and rounding error. *SIAM Journal on Numerical Analysis*, v. 11, n. 1, p. 87–96, 1974. Citado 2 vezes nas páginas 70 e 99.

BERGMAN, T. L. et al. Heat and Mass Transfer. [S.l.]: John Wiley & Sons, 2011. Citado na página 83.

BERTOLDO, G.; MARCHI, C. H. Verification and validation of the foredrag coefficient for supersonic and hypersonic flow of air over a cone of fineness ratio 3. *Applied Mathematical Modelling*, v. 44, p. 409–424, 2017. Citado na página 27.

BRAESS, D. B. Towards algebraic multigrid for elliptic problems of second order. *Computing*, v. 55, n. 4, p. 379–393, 1995. Citado na página 35.

BRANDT, A. Algebraic multigrid theory: The symmetric case. *Applied mathematics and computation*, v. 19, n. 1-4, p. 23–56, 1986. Citado na página 35.

BRIGGS, W. L.; HENSON, V. E.; MCCORMICK, S. F. A multigrid tutorial. 2. ed. [S.l.]: Siam, 2000. Citado 3 vezes nas páginas 63, 65 e 148.

BROOKSHAW, L. A method of calculating radiative heat diffusion in particle simulations. *Publications of the Astronomical Society of Australia*, Cambridge University Press, v. 6, n. 2, p. 207–210, 1985. Citado na página 51.

BULUÇ, A. *et al.* Parallel sparse matrix-vector and matrix-transpose-vector multiplication using compressed sparse blocks. In: *Proceedings of the twenty-first annual symposium on Parallelism in algorithms and architectures.* [S.l.: s.n.], 2009. p. 233–244. Citado na página 56.

BURDEN, R. L.; FAIRES, J. D. Numerical Analysis. 8. ed. [S.l.: s.n.], 2005. Citado 5 vezes nas páginas 54, 70, 74, 99 e 101.

BURDEN, R. L.; FAIRES, J. D.; BURDEN, A. M. *Numerical Analysis.* 10. ed. [S.l.: s.n.], 2016. Citado 6 vezes nas páginas 66, 72, 90, 113, 124 e 147.

CARMO, F. P. do. A Equação de Poisson e a Decomposição de Helmholtz-Hodge com Operadores SPH. Tese — PUC-Rio, Rio de Janeiro, 2008. Citado na página 31.

ÇENGEL, Y. A.; CIMBALA, J. M. Mecânica dos Fluidos: Fundamentos e Aplicações. [S.l.: s.n.], 2007. Citado na página 31.

CHANG, Q.; WONG, Y. S.; FU, H. On the algebraic multigrid method. *Journal of Computational Physics*, v. 125, n. 2, p. 279–292, 1996. Citado na página 35.

CHAUSSONNET, G. *et al.* Influence of particle disorder and smoothing length on SPH operator accuracy. *Conference Paper - 10th international SPHERIC workshop*, June 2015. Citado 5 vezes nas páginas 32, 37, 38, 150 e 151.

CHEN, D.; HUANG, W.; SLOAN, S. W. An alternative updated Lagrangian formulation for finite particle method. *Computer Methods in Applied Mechanics and Engineering*, Elsevier, v. 343, p. 490–505, 2019. Citado na página 38.

CIARLET, P. G. The finite element method for elliptic problems. [S.l.]: Siam, 2002. v. 40. Citado na página 31.

CUMINATO, J. A.; MENEGUETTE JUNIOR, M. Discretização de Equações Diferenciais Parciais: Técnicas de Diferenças Finitas. 1. ed. [S.l.]: SBM, 2013. Citado na página 53.

DEHNEN, W.; ALY, H. Improving convergence in smoothed particle hydrodynamics simulations without pairing instability. *Monthly Notices of the Royal Astronomical Society*, Blackwell Science Ltd Oxford, v. 425, n. 2, p. 1068–1082, 2012. Citado na página 38.

ENGELN-MÜLLGES, G.; UHLIG, F. *Numerical algorithms with C.* 1. ed. [S.I.]: Springer Science & Business Media, 1996. Citado 3 vezes nas páginas 56, 70 e 99.

FATEHI, R.; FAYAZBAKHSH, M.; MANZARI, M. T. On discretization of second-order derivatives in smoothed particle hydrodynamics. v. 30, p. 243–246, 2008. Citado na página 38.

FATEHI, R.; MANZARI, M. T. Error estimation in smoothed particle hydrodynamics and a new scheme for second derivatives. *Computers and Mathematics with applications*, v. 61, n. 1, p. 482–498, 2011. Citado 2 vezes nas páginas 33 e 38.

FERZIGER, J. H.; PERIĆ, M. Computational methods for fluid dynamics. 3. ed. [S.l.]: Springer Science & Business Media, 2002. Citado 6 vezes nas páginas 31, 37, 69, 70, 85 e 98.

FORTUNA, A. O. *Técnicas Computacionais para Dinâmica dos Fluidos: Conceitos e Aplicações.* 2. ed. [S.l.]: EDUSP, 2012. Citado na página 82.

FRAGA FILHO, C. A. D. *Smoothed Particle Hydrodynamics*. [S.1.]: Springer, 2019. Citado 4 vezes nas páginas 90, 98, 151 e 173.

FUKUCHI, T. Characteristic features of error in high-order difference calculation of 1d poisson equation and unlimited high-accurate calculation under multi-precision calculation. *Mathematics and Computers in Simulation*, Elsevier, v. 190, p. 303–328, 2021. Citado 2 vezes nas páginas 80 e 81.

FULK, D. A. A numerical analysis of smoothed particle hydrodynamics. Tese — Air Force Institute of Technology, 1994. Citado na página 41.

FULK, D. A.; QUINN, D. W. Hybrid formulations of smoothed particle hydrodynamics. *Elsevier Science Ltd*, v. 17, p. 329–1995, 1995. Citado na página 32.

GALANTE, G. Métodos multigrid paralelos em malhas não estruturadas aplicados à simulação de problemas de dinâmica de fluidos computacional e transferência de calor. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Sul, Porto Alegre, 2006. Citado na página 59.

GALLAGHER, P. *et al.* Best practice guidelines for the application of computational fluid dynamics in marine hydrodynamics. *Virtue Project, relatório técnico*, v. 221, p. 16, 2009. Citado na página 31.

GHAZANFARIAN, J.; SAGHATCHI, R.; GORJI-BANDPY, M. SPH simulation of turbulent flow past a high-frequency in-line oscillating cylinder near free-surface. *International Journal of Modern Physics C*, v. 27, n. 12, p. 1650152, 2016. 23 páginas. Citado na página 33.

GINGOLD, R. A.; MONAGHAN, J. J. Smoothed particle hydrodynamics - theory and application to non-spherical stars. *Royal Astronomical Society*, v. 181, p. 375–389, 1977. Citado 7 vezes nas páginas 26, 31, 32, 33, 37, 69 e 84.

GISSLER, C. et al. Approximate air-fluid interactions for SPH. Workshop on Virtual Reality and Physical Simulation VRIPHYS, 2017. Citado na página 33.

GOLDSTEIN, R. J. et al. Heat transfer—a review of 2002 literature. International Journal of Heat and Mass Transfer, v. 48, n. 5, p. 819–927, 2005. Citado na página 31.

GOLDSTEIN, R. J. et al. Heat transfer—a review of 2004 literature. International Journal of Heat and Mass Transfer, v. 53, n. 21, p. 4343–4396, 2010. Citado na página 31.

GRAY, J. P.; MONAGHAN, J. J.; SWIFT, R. P. SPH elastic dynamics. *Computer methods in applied mechanics and engineering*, v. 190, n. 49-50, p. 6641–6662, 2001. Citado na página 32.

GRIEBEL, M.; SCHWEITZER, M. A. Meshfree methods for partial differential equations. [S.l.]: Springer, 2003. Citado na página 31.

HORTMANN, M.; PERIĆ, M.; SCHEUERER, G. Finite volume multigrid prediction of laminar natural convection: bench-mark solutions. *International journal for numerical methods in fluids*, v. 11, n. 2, p. 189–207, 1990. Citado 2 vezes nas páginas 69 e 85.

HU, X. Y.; ADAMS, N. A. A SPH model for incompressible turbulence. *Elsevier*, v. 18, p. 66–75, 2015. Citado na página 33.

HUANG, C. *et al.* Coupled finite particle method with a modified particle shifting technology. *International Journal for Numerical Methods in Engineering*, Wiley Online Library, v. 113, n. 2, p. 179–207, 2018. Citado 3 vezes nas páginas 34, 37 e 38.

JALURIA, Y.; TORRANCE, K. E. *Computational heat transfer.* 2. ed. [S.l.]: Routledge, 2003. Citado na página 31.

JEONG, J. H. *et al.* Smoothed particle hydrodynamics: applications to heat conduction. *Computer Physics Communications*, v. 153, n. 1, p. 71–84, January 2003. Citado na página 33.

JI, Z. *et al.* Numerical simulations of oil flow inside a gearbox by smoothed particle hydrodynamics (SPH) method. *Tribology International*, v. 127, p. 47–58, 2018. Citado 2 vezes nas páginas 26 e 27.

KADALBAJOO, M. K.; KUMAR, A.; TRIPATHI, L. P. A radial basis functions based finite differences method for wave equation with an integral condition. *Applied Mathematics and Computation*, v. 2053, p. 8–16, 2015. Citado na página 31.

KAZEMI, E. *et al.* SPH modelling of depth-limited turbulent open channel flows over rough boundaries. *International journal for numerical methods in fluids*, v. 83, p. 3–27, 2017. Citado na página 33.

KORZANI, M. G. *et al.* Parametric study on smoothed particle hydrodynamics for accurate determination of drag coefficient for a circular cylinder. *Water Science and Engineering*, June 2017. Citado na página 33.

KOSHIZUKA, S.; OKA, Y. Moving-particle semi-implicit method for fragmentation of incompressible fluid. *Nuclear Science and Engineering*, v. 123, p. 421–434, 1996. Citado na página 31.

KOUKOUVINIS, P. K.; ANAGNOSTOPOULOS, J. S.; PAPANTONIS, D. E. SPH method used for flow predictions at a turgo impluse turbine: comparison with fluent. *World Academy of Science, Engineering and Tecnology*, v. 55, 2011. Citado na página 26.

KREYSZIG, E. Advanced Engineering Mathematics. [S.l.]: John Wiley & Sons, 1999. Citado na página 74.

LASTIWKA, M.; QUINLAN, N.; BASA, M. Adaptive particle distribution for smoothed particle hydrodynamics. *International Journal for Numerical Methods in Fluids*, v. 47, n. 10-11, p. 1403–1409, 2005. Citado na página 32.

LEVEQUE, R. J. *Finite volume methods for hyperbolic problems*. [S.l.]: Cambridge university press, 2002. v. 31. Citado na página 31.

LEVEQUE, R. J. Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems. [S.l.]: Siam, 2007. v. 98. Citado 3 vezes nas páginas 31, 68 e 84.

LI, D.; LUO, K.; FAN, J. Direct numerical simulation of turbulent flow and heat transfer in a spatially developing turbulent boundary layer laden with particles. *Journal of Fluid Mechanics*, v. 845, p. 417–461, 2018. Citado na página 33.

LI, J. *et al.* Efficient thermomechanical analysis of functionally graded structures using the symmetric SPH method. *Case Studies in Thermal Engineering*, v. 25, p. 100889, 2021. Citado na página 31.

LIND, S. J.; STANSBY, P. K. High-order eulerian incompressible smoothed particle hydrodynamics with transition to lagrangian free-surface motion. *Journal of Computational Physics*, Elsevier, v. 326, p. 290–311, 2016. Citado 2 vezes nas páginas 34 e 37.

LINNAINMAA, S. Taylor expansion of the accumulated rounding error. *BIT Numerical Mathematics*, v. 16, n. 2, p. 146–160, 1976. Citado 2 vezes nas páginas 70 e 99.

LIU, G. R. Mesh free methods: moving beyond the finite element method. [S.1.]: CRC Press, 2002. Citado na página 32.

LIU, G. R.; LIU, M. B. Smoothed Particle Hydrodynamics: a meshfree particle method. 1. ed. [S.l.]: World Scientific Publishing, 2003. Citado 6 vezes nas páginas 31, 37, 39, 41, 45 e 53.

LIU, M. B.; LIU, G. Restoring particle consistency in smoothed particle hydrodynamics. *Applied numerical mathematics*, Elsevier, v. 56, n. 1, p. 19–36, 2006. Citado 3 vezes nas páginas 34, 37 e 38.

LIU, M. B.; LIU, G. R. Smoothed particle hydrodynamics (SPH): an overview and recent developments. *Archives of computational methods in engineering*, Springer, v. 17, n. 1, p. 25–76, 2010. Citado 2 vezes nas páginas 34 e 37.

LIU, M. B.; LIU, G. R.; LAM, K. Construing smoothed functions in smoothed particle hydrodynamics with applications. *Journal of Computational and Applied Mathematics*, v. 155, p. 263–284, 2003. Citado 5 vezes nas páginas 33, 46, 47, 112 e 117.

LUCY, L. B. A numerical approach to the testing of the fission hypothesis. *Astron. Journal*, v. 82, p. 1013–1024, 1977. Citado 8 vezes nas páginas 26, 31, 32, 33, 37, 43, 69 e 84.

MARCHI, C. H. Verificação de soluções numéricas unidimensionais em dinâmica dos fluidos. Tese — UFSC, Florianópolis - SC, 2001. Citado 9 vezes nas páginas 36, 37, 68, 69, 70, 73, 84, 90 e 98.

MARCHI, C. H. *et al.* Repeated Richardson extrapolation applied to the two-dimensional laplace equation using triangular and square grids. *Applied Mathematical Modelling*, v. 37, n. 7, p. 4661–4675, 2013a. Citado 3 vezes nas páginas 35, 75 e 76.

MARCHI, C. H.; GERMER, E. M. Effect of the CFD numerical schemes on Repeated Richardson Extrapolation (RRE). *Applied & Computational Mathematics*, v. 2, p. 128, 2013. Citado 3 vezes nas páginas 35, 36 e 75.

MARCHI, C. H.; GIACOMINI, F. F.; SANTIAGO, C. D. Repeated Richardson extrapolation to reduce the field discretization error in comptational fluid dynamics. *Numerical Heat Transfer, Part B: Fundamentals*, v. 70, n. 4, p. 340–353, 2016. Citado 2 vezes nas páginas 35 e 75. MARCHI, C. H. *et al.* Highly accurate numerical solutions with repeated Richardson extrapolation for 2D Laplace equation. *Applied Mathematical Modelling*, v. 37, n. 12–13, p. 7386 – 7397, 2013b. Citado 3 vezes nas páginas 35, 36 e 75.

MARCHI, C. H.; SILVA, A. F. C. Multi-dimensional discretization error estimation for convergent apparent order. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, SciELO Brasil, v. 27, n. 4, p. 432–439, 2005. Citado na página 81.

MARCHI, C. H.; SILVA, A. F. C. da. Unidimensional numerical solution error estimation for convergent apparent order. *Numerical Heat Transfer: Part B: Fundamentals*, v. 42, n. 2, p. 167–188, 2002. Citado 3 vezes nas páginas 72, 113 e 124.

MARONGIU, J. C.; LEBOEUF, F.; PARKINSON, E. Numerical simulation of the flow in a Pelton turbine using the meshless method smoothed particle hydrodynamics: a new simple solid boundary treatment. *Proceedings of the Institution of Mechanical Engineers*, *Part A: Journal of Power and Energy*, v. 221, 2007. Citado na página 26.

MAUREL, B.; COMBESCURE, A. An SPH shell formulation for plasticity and fracture analysis in explicit dynamics. *International journal for numerical methods in engineering*, v. 76, n. 7, p. 949–971, 2008. Citado na página 32.

MERGA, F. E.; CHEMEDA, H. M. Modified Crank–Nicolson scheme with Richardson Extrapolation for one-dimensional heat equation. *Iranian Journal of Science and Technology*, *Transactions A: Science*, Springer, p. 1–10, 2021. Citado na página 139.

MONAGHAN, J. J. Smoothed particle hydrodynamics. *Anual Review of Astronomical and Astrophysics.*, v. 30, p. 543–574, 1992. Citado 2 vezes nas páginas 41 e 44.

MONAGHAN, J. J. Smoothed particle hydrodynamics. *Reports on progress in physics*, v. 68, n. 8, p. 1703, 2005. Citado na página 32.

MONAGHAN, J. J.; LATTANZIO, J. C. A refined particle method for astrophysical problems. *Astronomy and Astrophysics*, v. 149, p. 135–143, 1985. Citado 4 vezes nas páginas 33, 45, 112 e 117.

MORRIS, J. P. Analysis of smoothed particle hydrodynamics with applications. Tese — Monash University, 1996. Citado 3 vezes nas páginas 33, 47 e 104.

NOBLE, B.; DANIEL, J. W. Álgebra linear aplicada. *Rio de Janeiro, Prenctice-Hall do Brasil*, 1986. Citado na página 74.

OGER, G. et al. An improved SPH method: Towards higher order convergence. Journal of Computational Physics, Elsevier, v. 225, n. 2, p. 1472–1492, 2007. Citado na página 38.

OGER, G. *et al.* On distributed memory MPI-based parallelization of SPH codes in massive hpc context. *Computer Physics Communications*, v. 200, p. 1–14, 2016. Citado na página 27.

ÖZIŞIK, M. N. et al. Finite difference methods in heat transfer. [S.l.]: CRC press, 2017. Citado na página 53.

PAIVA, A. et al. Simulação de Fuidos sem Malha: Uma introdução ao método SPH. [S.l.]: IMPA, 2009. Citado 4 vezes nas páginas 41, 42, 51 e 148.

PATANKAR, S. Numerical heat transfer and fluid flow. [S.l.]: CRC press, 1980. Citado na página 31.

PATANKAR, S. V. Recent Developments in Computational Heat Transfer. *Journal of Heat Transfer*, v. 110, n. 4b, p. 1037–1045, 1988. Citado na página 30.

PLETCHER, R. H.; TANNEHILL, J. C.; ANDERSON JUNIOR, J. D. Computational fluid mechanics and heat transfer. 3. ed. [S.l.]: CRC Press, 2012. Citado na página 37.

PRICE, D. J. Resolving high reynolds numbers in smoothed particle hydrodynamics simulations of subsonic turbulence. *Monthly Notices of the Royal Astronomical Society: Letters*, v. 420, p. 1–5, 2012. Citado na página 33.

PURI, K.; RAMACHANDRAN, P. A comparison of SPH schemes for the compressible Euler equations. *Journal of Computational Physics*, n. 256, p. 308–333, 2014. Citado na página 33.

QUINLAN, N. J.; BASA, M.; LASTIWKA, M. Truncation error in mesh-free particle methods. *International Journal for Numerical Methods in Engineering*, v. 66(13), p. 2064–2085, 2006. Citado na página 85.

RANDLES, P. W.; LIBERSKY, L. D. Smoothed particle hydrodynamics: some recent improvements and applications. *Computer methods in applied mechanics and engineering*, Elsevier, v. 139, n. 1-4, p. 375–408, 1996. Citado na página 32.

REDDY, J. N.; GARTLING, D. K. The finite element method in heat transfer and fluid dynamics. 3. ed. [S.l.]: CRC press, 2010. Citado na página 31.

ROACHE, P. J. Verification and validation in computational science and engineering. [S.l.]: Hermosa, 1998. Citado 4 vezes nas páginas 37, 70, 73 e 98.

ROACHE, P. J.; KNUPP, P. M. Completed Richardson extrapolation. *Communications in Numerical Methods in Engineering*, v. 9, n. 5, p. 365–374, 1993. Citado 2 vezes nas páginas 35 e 75.

RUGE, J. W.; STÜBEN, K. Algebraic multigrid. In: *Multigrid methods*. [S.l.]: SIAM, 1987. p. 73–130. Citado 2 vezes nas páginas 35 e 146.

SHI, Z.; CAO, Y.-y.; CHEN, Q.-j. Solving 2D and 3D Poisson equations and biharmonic equations by the Haar wavelet method. *Applied Mathematical Modelling*, Elsevier, v. 36, n. 11, p. 5143–5161, 2012. Citado na página 82.

SILVA, N. D. P. da *et al.* Completed repeated Richardson extrapolation for compressible fluid flows. *Applied Mathematical Modelling*, v. 77, p. 724–737, 2020. Citado na página 36.

SMITH, G. D. Numerical solution of partial differential equations: finite difference methods. [S.l.]: Oxford university press, 1985. Citado na página 31.

SOLE-MARI, G. *et al.* Numerical equivalence between SPH and probabilistic mass transfer methods for lagrangian simulation of dispersion. *Advances in Water Resources*, v. 126, p. 108–115, 2019. Citado na página 31.

SONG, B.; PAZOUKI, A.; PÖSCHEL, T. Instability of smoothed particle hydrodynamics applied to poiseuille flows. *Computers & Mathematics with Applications*, v. 76, n. 6, p. 1447–1457, 2018. Citado 2 vezes nas páginas 32 e 33.

STEWART, G. W. The effects of rounding error on an algorithm for downdating a cholesky factorization. *IMA Journal of Applied Mathematics*, v. 23, n. 2, p. 203–213, 1979. Citado 2 vezes nas páginas 70 e 99.

STRANEX, T.; WHEATON, S. A new corrective scheme for SPH. *Computer Methods in Applied Mechanics and Engineering*, Elsevier, v. 200, p. 392–402, 2011. Citado na página 38.

STÜBEN, K. Algebraic multigrid (AMG): experiences and comparisons. *Applied mathematics and computation*, Elsevier, v. 13, n. 3-4, p. 419–451, 1983. Citado 2 vezes nas páginas 35 e 146.

SUERO, R. Otimização de parâmetros do método Multigrid Algébrico para problemas difusivos bidimensionais. Tese — UFPR, Curitiba - PR, 2010. Citado 4 vezes nas páginas 35, 62, 66 e 140.

SUERO, R. *et al.* Analysis of algebraic multigrid parameters for two-dimensional steady heat diffusion equations. *Applied Mathematical Modelling*, v. 36, n. 1, p. 2996–3006, 2012. Citado 2 vezes nas páginas 64 e 81.

SUN, X.-H.; CHEN, Y. Reevaluating amdahl's law in the multicore era. *Journal of Parallel and distributed Computing*, Elsevier, v. 70, n. 2, p. 183–188, 2010. Citado na página 60.

SZABÓ, B.; BABUŠKA, I. *Finite element analysis*. [S.l.]: John Wiley & Sons, 1991. Citado na página 68.

SZABÓ, B.; BABUŠKA, I. Introduction to finite element analysis: formulation, verification and validation. [S.l.]: John Wiley & Sons, 2011. v. 35. Citado na página 68.

TARTAKOVSKY, A. M.; MEAKIN, P. Simulation of unsaturated flow in complex fractures using smoothed particle hydrodynamics. *Vadose Zone Journal*, v. 4, n. 3, p. 848–855, 2005. Citado na página 32.

TROTTENBERG, U.; OOSTERLEE, C. W.; SCHULLER, A. *Multigrid.* [S.l.]: Elsevier, 2000. Citado 6 vezes nas páginas 34, 35, 60, 62, 63 e 64.

TROTTENBERG, U.; OOSTERLEE, C. W.; SCHÜLLER, A. *Multigrid.* [S.l.]: Elsevier, 2001. Citado 2 vezes nas páginas 59 e 147.

VIOLEAU, D. Fluid Mechanics and the SPH Method: Theory and Applications. [S.l.]: Oxford University Press, 2012. Citado 4 vezes nas páginas 33, 37, 41 e 43.

VIOLEAU, D.; ISSA, R. Numerical modelling of complex turbulent free-surface flows with the SPH method: an overview. *International Journal for Numerical Methods in Fluids*, v. 53, p. 277–304, 2007. Citado na página 33.

WALTER, W. V. FORTRAN-XSC A Portable Fortran 90 Module Library for Accurate and Reliable Scientific Computing. [S.l.]: Springer, 1993. v. 9. 265-285 p. Citado 2 vezes nas páginas 70 e 99.

WATANABE, K.; IGARASHI, H.; HONMA, T. Comparison of geometric and algebraic multigrid methods in edge-based finite-element analysis. *IEEE Transactions on Magnetics*, v. 41, n. 5, p. 1672–1675, 2005. Citado na página 35.

WESSELING, P. An Introduction to Multigrid Methods. [S.l.]: Book News, 2004. Citado 6 vezes nas páginas 34, 35, 60, 146, 147 e 148.

WU, C. T.; ELMAN, H. C. Analysis and comparison of geometric and algebraic multigrid for convection-diffusion equations. *SIAM Journal on Scientific Computing*, v. 28, n. 6, p. 2208–2228, 2006. Citado na página 35.

XIAO, Y. *et al.* An algebraic multigrid method for isotropic linear elasticity problems on anisotropic meshes. *International Journal for Numerical Methods in Biomedical Engineering*, v. 26, n. 5, p. 534–553, 2010. Citado na página 35.

XU, F. *et al.* Study of numerical and physical fracture with SPH method. *Acta Mechanica Solida Sinica*, v. 23, n. 1, p. 49–56, 2010. Citado na página 32.

YANG, X. F.; PENG, S. L.; LIU, M. B. A new kernel function for SPH with applications to free surface flows. *Journal of Applied Mathematical Modelling*, v. 38, p. 3822–3833, 2014. Citado 2 vezes nas páginas 33 e 48.

ZHU, J. Z.; ZIENKIEWICZ, O. C. Superconvergence recovery technique and a posteriori error estimators. *International Journal for Numerical Methods in Engineering*, v. 30, n. 7, p. 1321–1339, 1990. Citado 2 vezes nas páginas 69 e 85.

ZIENKIEWICZ, O. C. *et al. The finite element method.* [S.l.]: McGraw-hill London, 1977. v. 3. Citado na página 31.

APÊNDICE A – DEDUÇÕES AUXILIARES PARA A DISCRETIZAÇÃO DO OPERADOR LAPLACIANO

Para determinar a aproximação do operdor laplaciano mostrado em (FRAGA FILHO, 2019), são necessárias as resoluções das integrais apresentadas na equação (5.21). Com isso, precisa-se realizar um estudo de sinal dos termos que envolvem as integrais. Determinando-se o produto escalar entre a diferença de posição de duas partículas com a derivada do núcleo, conforme equação (A.1), obtém-se

$$(\mathbf{x}' - \mathbf{x}) \cdot \nabla_{\mathbf{x}'} W = (x' - x, y' - y) \cdot \nabla_{\mathbf{x}'} W$$
$$= (x' - x, y' - y) \cdot \left(\frac{\partial W}{\partial x}, \frac{\partial W}{\partial y}\right)$$
$$(A.1)$$
$$= (x' - x)\frac{\partial W}{\partial x} + (y' - y)\frac{\partial W}{\partial y}$$

A integral do produto escalar definido na equação (A.1) é importante para as deduções e aparecem frequentemente multiplicada por um fator $(\mathbf{x}' - \mathbf{x})/||\mathbf{x}' - \mathbf{x}||^2$, ou uma combinação dos dois. Dessa forma, pretende-se identificar o resultado dessas integrais para conhecer os termos da expressão do erro de truncamento ao aproximar o operador laplaciano usando o método SPH. Primeiramente vamos estudar a integral do produto escalar da equação (A.1) multiplicada pelo fator $(\mathbf{x}' - \mathbf{x})/||\mathbf{x}' - \mathbf{x}||^2$. Dessa forma obtém-se

$$\begin{split} \int_{\Omega} (x'-x) \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' &= \int_{\Omega} (x'-x) \frac{(x'-x,y'-y)}{||\mathbf{x}'-\mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' \\ &= \int_{\Omega} \frac{(x'-x)}{||\mathbf{x}'-\mathbf{x}||^2} \left[(x'-x) \frac{\partial W}{\partial x} + (y'-y) \frac{\partial W}{\partial y} \right] d\mathbf{x}' \\ &= \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2} (x'-x)^2 \frac{\partial W}{\partial x} d\mathbf{x}' + \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2} (x'-x)(y'-y) \frac{\partial W}{\partial y} d\mathbf{x}'}_{2a}. \end{split}$$

$$(A.2)$$

Adotando-se quatro partículas \mathbf{x}' equidistantes de \mathbf{x} , uma em cada quadrante, conforme mostrado na FIGURA 84, e ainda, observando-se a TABELA 16 para verificar o sinal dos termos da integral (1*a*), pode-se estudar apenas o sinal da derivada $\partial W/\partial x$, uma vez que o termo $(x' - x)^2$ sempre será positivo nos quatro quadrantes. Sendo assim, atribuindo-se o valor (1) quando o sinal for positivo e (-1) se negativo nota-se que

$$\underbrace{(\underline{x'-x})^2}_{(1)}\underbrace{\frac{\partial W}{\partial x}}_{(-1)}\Big|_{1^oQ} + \underbrace{(\underline{x'-x})^2}_{(1)}\underbrace{\frac{\partial W}{\partial x}}_{(1)}\Big|_{3^oQ} = 0$$
(A.3)

$$\underbrace{\underbrace{(x'-x)^2}_{(1)}}_{(1)}\underbrace{\frac{\partial W}{\partial x}}_{(1)}\Big|_{2^oQ} + \underbrace{(x'-x)^2}_{(1)}\underbrace{\frac{\partial W}{\partial x}}_{(-1)}\Big|_{4^oQ} = 0,$$

logo, a integral (1a) é nula. Repetindo-se a mesma análise para a integral (2a), observa-se que

$$\underbrace{(x'-x)}_{(1)}\underbrace{(y'-y)}_{(1)}\underbrace{\frac{\partial W}{\partial y}}_{(-1)}\Big|_{1^{o}Q} + \underbrace{(x'-x)}_{(-1)}\underbrace{(y'-y)}_{(-1)}\underbrace{\frac{\partial W}{\partial y}}_{(1)}\Big|_{3^{o}Q} = 0$$
(A.4)

$$\underbrace{(x'-x)}_{(-1)}\underbrace{(y'-y)}_{(1)}\underbrace{\frac{\partial W}{\partial y}}_{(-1)}\Big|_{2^oQ} + \underbrace{(x'-x)}_{(1)}\underbrace{(y'-y)}_{(-1)}\underbrace{\frac{\partial W}{\partial y}}_{(1)}\Big|_{4^oQ} = 0.$$

Assim, a integral da equação (A.2) resulta em

$$\int_{\Omega} (x'-x) \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' = \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2} (x'-x)^2 \frac{\partial W}{\partial x} d\mathbf{x}'}_{1a} + \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2}$$

$$+ \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}' - \mathbf{x}||^2} (x' - x)(y' - y) \frac{\partial W}{\partial y} d\mathbf{x}'}_{2a} \qquad (A.5)$$

$$\int_{\Omega} (x' - x) \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' = 0.$$

Agora repete-se o mesmo procedimento para a integral do produto escalar da equação (A.1) multiplicada pelo fator $(\mathbf{x}' - \mathbf{x})/||\mathbf{x}' - \mathbf{x}||^2$.

$$\begin{split} \int_{\Omega} (y'-y) \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' &= \int_{\Omega} (y'-y) \frac{(x'-x,y'-y)}{||\mathbf{x}'-\mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' \\ &= \int_{\Omega} \frac{(y'-y)}{||\mathbf{x}'-\mathbf{x}||^2} \left[(x'-x) \frac{\partial W}{\partial x} + (y'-y) \frac{\partial W}{\partial y} \right] d\mathbf{x}' \\ &= \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2} (x'-x)(y'-y) \frac{\partial W}{\partial x} d\mathbf{x}' + \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2} (y'-y)^2 \frac{\partial W}{\partial y} d\mathbf{x}'}_{4a} \\ &+ \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2} (y'-y)^2 \frac{\partial W}{\partial y} d\mathbf{x}'}_{4a}. \end{split}$$
(A.6)

Estudando o sinal das integrais (3a) e (4a) têm-se, respectivamente,

$$\underbrace{(x'-x)}_{(1)}\underbrace{(y'-y)}_{(1)}\underbrace{\frac{\partial W}{\partial x}}_{(-1)}\Big|_{1^{o}Q} + \underbrace{(x'-x)}_{(-1)}\underbrace{(y'-y)}_{(-1)}\underbrace{\frac{\partial W}{\partial x}}_{(1)}\Big|_{3^{o}Q} = 0$$
(A.7)

$$\underbrace{(x'-x)}_{(-1)}\underbrace{(y'-y)}_{(1)}\underbrace{\frac{\partial W}{\partial x}}_{(-1)}\Big|_{2^{o}Q} + \underbrace{(x'-x)}_{(1)}\underbrace{(y'-y)}_{(-1)}\underbrace{\frac{\partial W}{\partial x}}_{(1)}\Big|_{4^{o}Q} = 0,$$

$$\underbrace{(y'-y)^{2}}_{(1)}\underbrace{\frac{\partial W}{\partial y}}_{(-1)}\Big|_{1^{o}Q} + \underbrace{(y'-y)^{2}}_{(1)}\underbrace{\frac{\partial W}{\partial y}}_{(1)}\Big|_{3^{o}Q} = 0$$
(A.8)

$$\underbrace{(\underline{y'-y})^2}_{(1)}\underbrace{\frac{\partial W}{\partial y}}_{(1)}\Big|_{2^oQ} + \underbrace{(\underline{y'-y})^2}_{(1)}\underbrace{\frac{\partial W}{\partial y}}_{(-1)}\Big|_{4^oQ} = 0.$$

Assim, a integral da equação (A.6) resulta em

$$\int_{\Omega} (y'-y) \frac{(\mathbf{x}'-\mathbf{x})}{||\mathbf{x}'-\mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' = \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2} (x'-x)(y'-y) \frac{\partial W}{\partial x} d\mathbf{x}' + \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2} (y'-y)^2 \frac{\partial W}{\partial y} d\mathbf{x}'}_{4a}$$

$$+ \underbrace{\int_{\Omega} \frac{1}{||\mathbf{x}'-\mathbf{x}||^2} (y'-y)^2 \frac{\partial W}{\partial y} d\mathbf{x}'}_{4a}$$
(A.9)

$$\int_{\Omega} (y' - y) \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' = 0$$

Generalizando para o resultado dessas integrais, pode-se estabelecer o seguinte resultado

$$\int_{\Omega} (\mathbf{x}' - \mathbf{x})_{\Gamma=1} (\mathbf{x}' - \mathbf{x})_{\Gamma=2} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' = 0, \qquad (A.10)$$

$$\int_{\Omega} (\mathbf{x}' - \mathbf{x})_{\Gamma=2} (\mathbf{x}' - \mathbf{x})_{\Gamma=1} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' = 0, \qquad (A.11)$$

$$\int_{\Omega} (\mathbf{x}' - \mathbf{x})_{\Gamma=1} (\mathbf{x}' - \mathbf{x})_{\Gamma=1} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' = -\delta_{11} = -1, \quad (A.12)$$

$$\int_{\Omega} (\mathbf{x}' - \mathbf{x})_{\Gamma=2} (\mathbf{x}' - \mathbf{x})_{\Gamma=2} \frac{(\mathbf{x}' - \mathbf{x})}{||\mathbf{x}' - \mathbf{x}||^2} \cdot \nabla_{\mathbf{x}'} W d\mathbf{x}' = -\delta_{22} = -1, \quad (A.13)$$

onde $\Gamma = (1,2)$. Sendo assim, para $\Gamma = 1$, tem-se $(\mathbf{x}' - \mathbf{x})_1 = (x' - x)$ e para $\Gamma = 2$, tem-se $(\mathbf{x}' - \mathbf{x})_2 = (y' - y)$.

Com toda essa análise é possível realizar a dedução completa do operador laplaciano 1D e 2D, com seus respectivos termos do erro de truncamento até a derivada de γ -ésima ordem e definir a equação geral do erro de truncamento para ambos os casos.

FIGURA 84 – DISPOSIÇÃO DE PARTÍCULAS EQUIDISTANTES NOS QUADRANTES.



FONTE: O autor (2022).

TABELA 16 – ESTUDO DE SINAL DOS TERMOS DAS INTE-GRAIS CONTIDAS NA EQUAÇÃO (5.21).

Quadrante	$\mathbf{x}' - \mathbf{x}$	(x'-x)	(y'-y)	$\partial W / \partial x$	$\partial W/\partial y$
$1^o Q$	(x'-x,y'-y)	+	+		_
$2^{o}Q$	(x'-x,y'-y)	-	+	+	_
$3^{o}Q$	(x'-x,y'-y)	—	-	+	+
$4^{o}Q$	(x'-x,y'-y)	+	—	—	+

FONTE: O autor (2022).

APÊNDICE B – MULTIEXTRAPOLAÇÃO DE RICHARDSON GENERALIZADA PARA O MÉTODO SPH

A expressão da RRE escrita em termos de soluções numéricas obtidas com o método SPH, usando dicretizações uniformes ou desordenadas, pode ser descrita como:

$$\psi(\mathbf{x}_s)_{g,m} = \psi(\mathbf{x}_s)_{g,m-1} + \frac{\psi(\mathbf{x}_s)_{g,m-1} - \psi(\mathbf{x}_s)_{g-1,m-1}}{q^{p_{m-1}} - 1},$$
(B.1)

onde $\psi(\mathbf{x}_s)$ é a solução numérica da variável de interesse conforme definição 10, g é o nível de discretização em que a solução numérica foi encontrada, m é o número de extrapolações de Richardson e p_m são as ordens verdadeiras da equação do erro de discretização e $q = h_{g-1}/h_g$ é a razão de refino.

Por meio da definição 10, é possível notar que nos casos de discretização desordenada, a partícula \mathbf{x}_s permancece fixa em todos os níveis de discretização. Isso significa que em uma simulação onde há movimento das partículas, também é possível aplicar a RRE para determinar soluções mais acuradas, pois a partícula sensor fixa atende todas as especificidades da multiextrapolação de Richardson. Além disso, vale destacar que é possível determinar partículas sensores em outras coordenadas, como por exemplo $\mathbf{x}_s(1/4,1/4)$, $\mathbf{x}_s(1/8,1/8), \mathbf{x}_s(3/4,3/4) \in \mathbf{x}_s(7/8,7/8)$. Em todos esses exemplos para o caso 2D a definição 10 é satisfeita. O deslocamento da partícula sensor fixa para outra coordenada, desde que atenda a definição 10, é aplicável não somente a modelos 2D como também a 1D e 3D.

Generalizando a equação (B.1) para o caso em que as ordens aparentes são aplicadas, obtém-se as equações (B.2a)-(B.2b). Este procedimento da RRE encontra-se descrito no ALGORITMO 7.

$$(p_U)_{g,m} = \frac{\log\left(\frac{|\psi(\mathbf{x}_s)_{g-1,m} - \psi(\mathbf{x}_s)_{g-2,m}|}{|\psi(\mathbf{x}_s)_{g,m} - \psi(\mathbf{x}_s)_{g-1,m}|}\right)}{\log(q)}$$
(B.2a)

$$\psi(\mathbf{x}_s)_{g,m} = \psi(\mathbf{x}_s)_{g,m-1} + \frac{\psi(\mathbf{x}_s)_{g,m-1} - \psi(\mathbf{x}_s)_{g-1,m-1}}{q^{(p_U)_{g,m-1}} - 1}.$$
 (B.2b)

Para a equação (B.2a), a varição entre o número de discretizações e extrapolações pode ser descrita por $3 \leq g \leq G$ e $0 \leq m \leq$ [parte inteira (g - 3)/2], onde G é o número máximo de discretizações. Enquanto isto, na equação (B.2b) m = 0 não se aplica, resultando em $\psi(\mathbf{x}_s)_{g,0}$ que são as soluções não-extrapoladas. A partir disso, $3 \leq g \leq G$ e $1 \leq m \leq$ [parte inteira (g - 1)/2]. Para aplicar a RRE utilizando as ordens verdadeiras
deduzidas a *priori*, basta substituir p_U por p_V nas equações ((B.2a)) e ((B.2b)) e definir a nova variação $2 \le g \le G$.

Algoritmo 7: RRE GENERALIZADA PARA SPH COM PARTÍCULA SENSOR FIXA.

Entrada: $\psi(\mathbf{x}_s)$ soluções em G discretizações por partículas, q Saída: $\psi(\mathbf{x}_s)$ 1 início **Definir:** $\psi(\mathbf{x}_s)_{1,0} = \psi(\mathbf{x}_s)_1, \dots, \psi(\mathbf{x}_s)_{G,0} = \psi(\mathbf{x}_s)_G$ $\mathbf{2}$ para $g = 3, \ldots, G$ faça 3 para $m = 0, \ldots, [parte inteira (g-1)/2]$ faça $\mathbf{4}$ se $m \ll [parte inteira (g-3)/2]$ então $\mathbf{5}$ $(p_U)_{g,m} = \frac{\log\left(\frac{|\psi(\mathbf{x}_s)_{g-1,m} - \psi(\mathbf{x}_s)_{g-2,m}|}{|\psi(\mathbf{x}_s)_{g,m} - \psi(\mathbf{x}_s)_{g-1,m}|}\right)}{\log(q)}$ 6 fim 7 sem>=1então 8 $\psi(\mathbf{x}_s)_{g,m} = \psi(\mathbf{x}_s)_{g,m-1} + \frac{\psi(\mathbf{x}_s)_{g,m-1} - \psi(\mathbf{x}_s)_{g-1,m-1}}{q^{(p_U)_{g,m-1}} - 1}$ 9 fim 10 fim 11 fim 1213 fim

APÊNDICE C – RELATÓRIO DE SAÍDA DO PROGRAMA E1-2DP

Segue abaixo um dos relatórios gerados pelo programa que resolve a equação da difusão de calor em regime permanente 2D. Esse relatório é um resumo de informações sobre as simulações executadas sequencialmente.

Simulacao iniciada em: 28/01/2022 11:12:52 Solver Numero de threads 28/28 = Numero Niveis (Npi) 10 = Num Esp Med Unif/dir = 8 49 Numero Particulas(i) = 3 Numero de niveis = Numero de threads 28 = Exemplo = 1 = 0.000000000000000000000000000E+00 Desordem Tempo Discretizacao = 3.0158996582E-02 segundos. Tempo Termo Fonte = 4.8089027405E-04 segundos. Tempo Cond. Inicial = 1.2290477753E-03 segundos. Tempo setup 1 = 1.4414100000E-01 segundos. Tempo setup 2 = 7.8761577606E-03 segundos. Tempo Sol Analitica = 6.6995620728E-05 segundos. Coord da particula X = 5.000000000000000000E-01 Gamma = Momento W¹ xey erro = -3.99008759067980546142E-05 Momento W¹ xey = -1.17848597967660048979E+00 Tempo Troca Sinal = 1.9407272339E-04 segundos. 6.43950202257578933088E-10 Erro Iteracao = Numero de Ciclos 5 = Tempo de AMG = 9.4280242920E-03 segundos. A(1)= -1.76337911686914488370E+02Tmeio Analitica = -3.515625000000000000E-02 Tmeio Numerica = -3.56098186494735459726E-02

Erro Tmeio	=	4.53568649473545972595E-04
Densidade(j)	=	24
Tmedia Analitica	=	-1.7777777777777777778E-02
Tmedia Numerica	=	-1.76881396621004144065E-02
Erro Tmedia	=	-8.96381156773633712883E-05
Erro inconsistencia	=	2.41294428362870735918E-05
Tempo Total(N)	=	5.9830904007E-02 segundos.
*****	***	***************************************
Num Esp Med Unif/dir	=	16
Numero Particulas(i)	=	225
Numero de niveis	=	4
Numero de threads	=	28
*****	***	***************************************
Exemplo	=	1
Desordem	=	0.000000000000000000000000000000000000
*****	***	***************************************
Tempo Discretizacao	=	4.7161579132E-03 segundos.
Tempo Termo Fonte	=	6.7949295044E-05 segundos.
Tempo Cond. Inicial	=	2.5415420532E-04 segundos.
Tempo setup 1	=	9.0124000000E-01 segundos.
Tempo setup 2	=	9.6118450165E-03 segundos.
Tempo Sol Analitica	=	7.7009201050E-05 segundos.
Coord da particula X	=	5.000000000000000000000000000000000000
Coord da particula Y	=	5.000000000000000000000000000000000000
Gamma	=	8.000000000000000000E+00
Momento W^1_xey erro	=	-6.23451186043719603348E-07
Momento W^1_xey	=	-1.17848597967660048979E+00
Tempo Troca Sinal	=	1.7094612122E-04 segundos.
Erro Iteracao	=	2.90489781919334246979E-10
Numero de Ciclos	=	6
Tempo de AMG	=	3.6726951599E-02 segundos.
A(1)	=	-7.05351646747657953481E+02
Tmeio Analitica	=	-3.5156250000000000000E-02
Tmeio Numerica	=	-3.54441183460844430523E-02
Erro Tmeio	=	2.87868346084443052298E-04
Densidade(j)	=	24
Tmedia Analitica	=	-1.7777777777777777778E-02
Tmedia Numerica	=	-1.79321920307901134787E-02
Erro Tmedia	=	1.54414253012335700948E-04

Erro inconsistencia	=	9.73167557877870829865E-06	
Tempo Total(N)	=	9.6014976501E-02 segundos.	

Num Esp Med Unif/dir	=	32	
Numero Particulas(i)	=	961	
Numero de niveis	=	5	
Numero de threads	=	28	
*****	* **	*****************	
Exemplo	=	1	
Desordem	=	0.000000000000000000000000000000000000	
******	***	*****************	
Tempo Discretizacao	=	1.9509792328E-03 segundos.	
Tempo Termo Fonte	=	1.1110305786E-04 segundos.	
Tempo Cond. Inicial	=	2.7179718018E-04 segundos.	
Tempo setup 1	=	4.4284450000E+00 segundos.	
Tempo setup 2	=	4.4698715210E-03 segundos.	
Tempo Sol Analitica	=	1.1491775513E-04 segundos.	
Coord da particula X	=	5.000000000000000000E-01	
Coord da particula Y	=	5.000000000000000000E-01	
Gamma	=	8.000000000000000000000000000000000000	
Momento W^1_xey erro	= ·	-9.74142478193311880230E-09	
Momento W^1_xey	= ·	-1.17848597967660048979E+00	
Tempo Troca Sinal	=	1.4533042908E-02 segundos.	
Erro Iteracao	=	1.06854333121045740379E-09	
Numero de Ciclos	=	6	
Tempo de AMG	=	1.5012001991E-01 segundos.	
A(1)	= ·	-2.82140658699063181392E+03	
Tmeio Analitica	= •	-3.51562500000000000000E-02	
Tmeio Numerica	= •	-3.53049356532648672540E-02	
Erro Tmeio	=	1.48685653264867254007E-04	
Densidade(j)	=	24	
Tmedia Analitica	= •	-1.7777777777777777778E-02	
Tmedia Numerica	= •	-1.78987504785151195682E-02	
Erro Tmedia	=	1.20972700737341790408E-04	
Erro inconsistencia	=	1.36712997649544719973E-06	
Tempo Total(N)	=	3.6855792999E-01 segundos.	
******	***	****************	
Num Esp Med Unif/dir	=	64	
Numero Particulas(i)	=	3969	

Numero de niveis	=	6
Numero de threads	=	28
*******	***	*****************
Exemplo	=	1
Desordem	=	0.000000000000000000000000000000000000
******	***	***************************************
Tempo Discretizacao	=	6.9241523743E-03 segundos.
Tempo Termo Fonte	=	1.2292861938E-02 segundos.
Tempo Cond. Inicial	=	3.5851001740E-02 segundos.
Tempo setup 1	=	4.8730310000E+00 segundos.
Tempo setup 2	=	1.1327028275E-02 segundos.
Tempo Sol Analitica	=	5.6979656219E-03 segundos.
Coord da particula X	=	5.000000000000000000000000000000000000
Coord da particula Y	=	5.000000000000000000000000000000000000
Gamma	=	8.000000000000000000000000000000000000
Momento W^1_xey erro	=	-1.52209762217704981286E-10
Momento W^1_xey	=	-1.17848597967660048979E+00
Tempo Troca Sinal	=	5.6791305542E-04 segundos.
Erro Iteracao	=	2.10055208436559807274E-09
Numero de Ciclos	=	6
Tempo de AMG	=	5.7897400856E-01 segundos.
A(1)	=	-1.12856263479625272557E+04
Tmeio Analitica	=	-3.5156250000000000000000000000000000000000
Tmeio Numerica	=	-3.52213675687827626135E-02
Erro Tmeio	=	6.51175687827626134740E-05
Densidade(j)	=	24
Tmedia Analitica	=	-1.7777777777777777778E-02
Tmedia Numerica	=	-1.78438162455185663327E-02
Erro Tmedia	=	6.60384677407885549427E-05
Erro inconsistencia	=	1.59280674253420911476E-07
Tempo Total(N)	=	1.6621520519E+00 segundos.
*****	***	*****************
Num Esp Med Unif/dir	=	128
Numero Particulas(i)	=	16129
Numero de niveis	=	7
Numero de threads	=	28
******	***	****************
Exemplo	=	1
Desordem	=	0.000000000000000000000000000000000000

Tempo Discretizacao = 4.2496919632E-02 segundos. Tempo Termo Fonte = 1.1579036713E-02 segundos. Tempo Cond. Inicial = 4.5704841614E-04 segundos. = 7.3206290000E+00 segundos. Tempo setup 1 Tempo setup 2 = 5.4071903229E-02 segundos. Tempo Sol Analitica = 3.6289691925E-03 segundos. Gamma = Momento W¹_xey erro = -2.37827753465164033259E-12 = -1.17848597967660048979E+00 Momento W¹ xey Tempo Troca Sinal = 1.3664960861E-02 segundos. Erro Iteracao = 3.63124205900503286230E-09 Numero de Ciclos 6 = Tempo de AMG = 2.0497970581E+00 segundos. A(1)= -4.51425053918501090228E+04 Tmeio Analitica = -3.5156250000000000000E-02 Tmeio Numerica = -3.51760368766868821049E-02Erro Tmeio = 1.97868766868821048941E-05 Densidade(j) = 24 Tmedia Analitica = -1.77777777777777777778E-02 Tmedia Numerica = -1.78068756906073314877E-02 = 2.90979128295537099523E-05 Erro Tmedia Erro inconsistencia = 1.77861173000006949848E-08 Tempo Total(N) 5.0564019680E+00 segundos. = Num Esp Med Unif/dir = 256 Numero Particulas(i) = 65025 Numero de niveis 8 = Numero de threads 28 = Exemplo 1 = Desordem Tempo Discretizacao = 1.2804484367E-01 segundos. Tempo Termo Fonte = 5.4609775543E-03 segundos. Tempo Cond. Inicial = 7.6293945312E-04 segundos. Tempo setup 1 = 1.4892319000E+01 segundos.

Tempo setup 2	=	1.0757994652E-01 segundos.
Tempo Sol Analitica	=	2.4628639221E-03 segundos.
Coord da particula X	=	5.000000000000000000E-01
Coord da particula Y	=	5.00000000000000000E-01
Gamma	=	8.000000000000000000000000000000000000
Momento W^1_xey erro	= -	-3.71605864789318801968E-14
Momento W^1_xey	= -	-1.17848597967660048979E+00
Tempo Troca Sinal	=	2.1256923676E-02 segundos.
Erro Iteracao	=	5.80383224207595279210E-09
Numero de Ciclos	=	6
Tempo de AMG	=	8.0247220993E+00 segundos.
A(1)	= -	-1.80570021567400436091E+05
Tmeio Analitica	= -	-3.5156250000000000000E-02
Tmeio Numerica	= -	-3.51524739079640957447E-02
Erro Tmeio	= •	-3.77609203590425530898E-06
Densidade(j)	=	24
Tmedia Analitica	= -	-1.7777777777777777778E-02
Tmedia Numerica	= -	-1.77860463904457045160E-02
Erro Tmedia	=	8.26861266792673817493E-06
Erro inconsistencia	=	2.00136660645433456784E-09
Tempo Total(N)	=	1.9151795149E+01 segundos.
******	***	<*************************************
Num Esp Med Unif/dir	=	512
Numero Particulas(i)	=	261121
Numero de niveis	=	9
Numero de threads	=	28
******	***	<*************************************
Exemplo	=	1
Desordem	=	0.000000000000000000000000000000000000
******	***	<*************************************
Tempo Discretizacao	=	3.8266587257E-01 segundos.
Tempo Termo Fonte	=	1.6688823700E-02 segundos.
Tempo Cond. Inicial	=	8.7130069733E-03 segundos.
Tempo setup 1	=	4.7557402000E+01 segundos.
Tempo setup 2	=	4.0843892097E-01 segundos.
Tempo Sol Analitica	=	1.4642000198E-02 segundos.
Coord da particula X	=	5.000000000000000000E-01
Coord da particula Y	=	5.000000000000000000E-01
Gamma	=	8.0000000000000000000000E+00

Momento W^1_xey erro	=	-5.80634163733310628075E-16
Momento W^1_xey	=	-1.17848597967660048979E+00
Tempo Troca Sinal	=	2.3529052734E-02 segundos.
Erro Iteracao	=	8.87863472018150178733E-09
Numero de Ciclos	=	6
Tempo de AMG	=	3.2717527866E+01 segundos.
A(1)	=	-7.22280086269601744364E+05
Tmeio Analitica	=	-3.5156250000000000000000000000000000000000
Tmeio Numerica	=	-3.51404663651984578401E-02
Erro Tmeio	=	-1.57836348015421598648E-05
Densidade(j)	=	24
Tmedia Analitica	=	-1.7777777777777777778E-02
Tmedia Numerica	=	-1.77750431016999614864E-02
Erro Tmedia	=	-2.73467607781629135440E-06
Erro inconsistencia	=	2.30693062352299771090E-10
Tempo Total(N)	=	7.7510008812E+01 segundos.
*****	**	*****************
Num Esp Med Unif/dir	=	1024
Numero Particulas(i)	=	1046529
Numero de niveis	=	10
Numero de threads	=	28
*****	**	******************
Exemplo	=	1
Desordem	=	0.000000000000000000000000000000000000
*****	**	*****************
Tempo Discretizacao	=	1.5065090656E+00 segundos.
Tempo Termo Fonte	=	6.7816019058E-02 segundos.
Tempo Cond. Inicial	=	1.7634868622E-02 segundos.
Tempo setup 1	=	1.7787365300E+02 segundos.
Tempo setup 2	=	1.2060799599E+00 segundos.
Tempo Sol Analitica	=	5.0187110901E-02 segundos.
Coord da particula X	=	5.000000000000000000000000000000000000
Coord da particula Y	=	5.000000000000000000000000000000000000
Gamma	=	8.000000000000000000000000000000000000
Momento W^1_xey erro	=	-9.07240880833297856367E-18
Momento W^1_xey	=	-1.17848597967660048979E+00
Tempo Troca Sinal	=	5.9637069702E-02 segundos.
Erro Iteracao	=	4.84371625128863951495E-10
Numero de Ciclos	=	7

Tempo de AMG	=	1.4466659403E+02 segundos.
A(1)	=	-2.88912034507840697746E+06
Tmeio Analitica	=	-3.5156250000000000000E-02
Tmeio Numerica	=	-3.51344058579210393918E-02
Erro Tmeio	=	-2.18441420789606081512E-05
Densidade(j)	=	24
Tmedia Analitica	=	-1.7777777777777777778E-02
Tmedia Numerica	=	-1.77693944392227354049E-02
Erro Tmedia	=	-8.38333855504237285244E-06
Erro inconsistencia	=	2.72666644933078259712E-11
Tempo Total(N)	=	3.2445495415E+02 segundos.
******	~**	***************************************
Num Esp Med Unif/dir	=	2048
Numero Particulas(i)	=	4190209
Numero de niveis	=	11
Numero de threads	=	28
******	~**	***************************************
Exemplo	=	1
Desordem	=	0.000000000000000000000000000000000000
******	***	******************
Tempo Discretizacao	=	5.8157799244E+00 segundos.
Tempo Termo Fonte	=	2.3562192917E-01 segundos.
Tempo Cond. Inicial	=	5.6120872498E-02 segundos.
Tempo setup 1	=	6.9390287000E+02 segundos.
Tempo setup 2	=	4.0720930099E+00 segundos.
Tempo Sol Analitica	=	1.3994979858E-01 segundos.
Coord da particula X	=	5.000000000000000000000000000000000000
Coord da particula Y	=	5.000000000000000000000000000000000000
Gamma	=	8.000000000000000000000000000000000000
Momento W^1_xey erro	=	-1.41756387630202790057E-19
Momento W^1_xey	=	-1.17848597967660048979E+00
Tempo Troca Sinal	=	2.1357393265E-01 segundos.
Erro Iteracao	=	7.21987849368179252792E-10
Numero de Ciclos	=	7
Tempo de AMG	=	5.8342130303E+02 segundos.
A(1)	=	-1.15564813803136279098E+07
Tmeio Analitica	=	-3.5156250000000000000E-02
Tmeio Numerica	=	-3.51313613885451973546E-02
Erro Tmeio	=	-2.48886114548026453651E-05

Densidade(j)	=	24
Tmedia Analitica	=	-1.77777777777777777778E-02
Tmedia Numerica	=	-1.77665333664237129508E-02
Erro Tmedia	=	-1.12444113540648270070E-05
Erro inconsistencia	=	3.28826012508926242555E-12
Tempo Total(N)	=	1.2933548729E+03 segundos.
*****	**	***************************************
Num Esp Med Unif/dir	=	4096
Numero Particulas(i)	=	16769025
Numero de niveis	=	12
Numero de threads	=	28
*****	**	***************************************
Exemplo	=	1
Desordem	=	0.000000000000000000000000000000000000
******	<**	***************************************
Tempo Discretizacao	=	2.4630958080E+01 segundos.
Tempo Termo Fonte	=	8.4443593025E-01 segundos.
Tempo Cond. Inicial	=	1.9482588768E-01 segundos.
Tempo setup 1	=	2.8465545400E+03 segundos.
Tempo setup 2	=	1.4989814997E+01 segundos.
Tempo Sol Analitica	=	4.9795007706E-01 segundos.
Coord da particula X	=	5.000000000000000000000000000000000000
Coord da particula Y	=	5.000000000000000000000000000000000000
Gamma	=	8.000000000000000000000000000000000000
Momento W^1_xey erro	=	-2.21494355672191859465E-21
Momento W^1_xey	=	-1.17848597967660048979E+00
Tempo Troca Sinal	=	2.4494400024E+00 segundos.
Erro Iteracao	=	1.06996949119213476563E-09
Numero de Ciclos	=	7
Tempo de AMG	=	2.4028672669E+03 segundos.
A(1)	=	-4.62259255212545116393E+07
Tmeio Analitica	=	-3.5156250000000000000E-02
Tmeio Numerica	=	-3.51298355966517203165E-02
Erro Tmeio	=	-2.64144033482796834881E-05
Densidade(j)	=	24
Tmedia Analitica	=	-1.7777777777777777778E-02
Tmedia Numerica	=	-1.77650936473681549405E-02
Erro Tmedia	=	-1.26841304096228373067E-05
Erro inconsistencia	=	4.02156594107326260150E-13