UNIVERSIDADE FEDERAL DO PARANÁ

CARLOS ALBERTO REZENDE DE CARVALHO JUNIOR

EXPERIMENTOS COM EXTRAPOLAÇÃO DE RICHARDSON COMPLETA E ALGUMAS VARIAÇÕES

CURITIBA

2020

CARLOS ALBERTO REZENDE DE CARVALHO JUNIOR

EXPERIMENTOS COM EXTRAPOLAÇÃO DE RICHARDSON COMPLETA E ALGUMAS VARIAÇÕES

Tese apresentada ao Programa de Pós-Graduação em Métodos Numéricos em Engenharia, do Setor de Ciências e Tecnologia da Universidade Federal do Paraná, como requisito parcial para à obtenção do título de Doutor em Ciências.

Orientador: Prof. Dr. Carlos Henrique Marchi

CURITIBA

2020

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

C331e Carvalho Junior, Carlos Alberto Rezende de
Experimentos com extrapolação de Richardson completa e algumas variações [recurso eletrônico] / Carlos Alberto Rezende de Carvalho Junior. – Curitiba, 2020.

Tese - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-Graduação em Métodos Numéricos em Engenharia, 2020.

Orientador: Carlos Henrique Marchi.

1. Diferenças Finitas. 2. Equações diferenciais. I. Universidade Federal do Paraná. II. Marchi, Carlos Henrique. III. Título.

CDD: 515.35

Bibliotecária: Vanusa Maciel CRB- 9/1928

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 em MÉTODOS NUMÉRICOS EM ENGENHARIA da Universidade Federal do Paraná foram convocados para realizar a arguição da tese de Doutorado de **CARLOS ALBERTO REZENDE DE CARVALHO JUNIOR** intitulada: **EXPERIMENTOS COM EXTRAPOLAÇÃO DE RICHARDSON COMPLETA E ALGUMAS VARIAÇÕES**, sob orientação do Prof. Dr. CARLOS HENRIQUE MARCHI, 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, 15 de Julho de 2020.

CARLOS HENRIQUE MARCHI
Presidente da Banca Examinadora

FRANCISCO MARCONDES
Avaliador Externo (UNIVERSIDADE FEDERAL DO CEARÁ)

GEOVANI NUNES GRAPIGLIA
Avaliador Externo (UNIVERSIDADE FEDERAL DO PARANÁ)

DIOGO BERTA PITZ
Avaliador Externo (UNIVERSIDADE FEDERAL DO PARANÁ)

GUILHERME BERTOLDO
Avaliador Externo (UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ)

AGRADECIMENTOS

Agradeço aos meus pais, irmãos, família e amigos, pelo incentivo e apoio incondicional. Em especial para o meu pai (in memorian), por não poder estar presente neste momento especial, sem ele nada disso seria possível.

Agradeço ao meu orientador, Prof. Dr. Carlos Henrique Marchi pela dedicação, sabedoria compartilhada, amizade, paciência e, acima de tudo confiança.

Agradeço aos membros da banca examinadora, Prof. Dr. Luciano Kiyoshi Araki, Prof. Dr Geovani Nunes Grapiglia, Prof. Dr. Guilherme Bertoldo, Prof. Dr. Diogo B. Pitz e Prof. Dr. Francisco Marcondes pelo tempo dispensado à leitura deste trabalho.

Agradeço aos meus colegas e amigos do laboratório de experimentação numérica da UFPR (LENA).

Agradeço ao Programa de Pós-Graduação em Métodos Numéricos em Engenharia (PPGMNE) da Universidade Federal do Paraná (UFPR), à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela oportunidade de cursar o doutorado.

"No xadrez, como na vida,
o adversário mais perigoso é você mesmo"
Vasily Vasiliyevich Smyslov

RESUMO

Este trabalho tem como base a Extrapolação de Richardson (ER) e sua versão aprimorada a Multiextrapolação de Richardson (MER), cujo objetivo é aperfeiçoar as soluções numéricas sem a necessidade de novas simulações, funcionando como um pós-processamento de dados. Estas técnicas são excelentes aliadas para melhoria das simulações, porque possuem um baixíssimo custo computacional. Entretanto, não é possível realizar extrapolações de forma efetiva em todo os pontos do domínio do problema, também conhecido por campos. Neste cenário, surgiu a Extrapolação de Richardson Completa (CRE) e algumas variações, técnicas voltadas para realização de extrapolações no campo. Desde então, este conceito é estudado, aprimorado, ou simplesmente utilizado. Neste trabalho são realizadas extrapolações no campo utilizando algumas técnicas entre elas CRE e introduzindo o conceito de submalhas, baseados em MER e ER, tendo como intuito estender este conceito. Para os experimentos são utilizados diversos modelos matemáticos com soluções analíticas: Poisson 1D com dois termos fontes, Burgers 1D e 2D, Laplace 2D e Navier-Stokes 2D. As discretizações são realizadas através da técnica de Diferenças Finitas. Para os modelos unidimensionais e Laplace 2D foram utilizadas malhas uniformes e não uniformes. As simulações foram realizadas em linguagem Fortran 90 com precisão quádrupla, sendo essa a precisão máxima de armazenamento disponível da linguagem. Através do conceito de submalha foram obtidos resultados expressivos atingindo em alguns casos erros em módulo inferiores a 10^{-20}, obtendo expressiva redução em relação aos erros iniciais. Além disso, houve considerável aumento nas ordens de acurácia, nos modelos matemáticos empregados. Em alguns casos se mostraram consideravelmente superiores em comparação com as técnicas já existentes. Em geral, apresentaram estabilidade em todos os modelos utilizados, apresentando redução do erro e aumento das ordens de acurácia em todos os casos.

Palavras-chave: Erro numérico. Multiextrapolação de Richardson. Diferenças Finitas. SubMalha. Extrapolação de Richardson Completa.

ABSTRACT

This work is based on Richardson extrapolation (RE) and its improved version the Repetead Richardson Extrapolation (RRE), whose objective is to improve numerical solutions without the need for new simulations, working as a post-processing of data. These techniques are excellent allies for improving the simulations, because of their very low computational cost. However, it is not possible to carry out extrapolations effectively in all points of the problem domain, also known as fields. In this scenario, Completed Richardson Extrapolation (CRE) appeared and some variations, techniques aimed at performing extrapolations in the field. Since then, this concept has been studied, improved, or simply used. In this work, extrapolations are performed in the field using some techniques, among them CRE and introducing the concept of sub-meshes, based on RRE and RE, with the intention of extending this concept. For the experiments, several mathematical models with analytical solutions are used: Poisson 1D with two source terms, Burgers 1D and 2D, Laplace 2D and Navier-Stokes 2D. Discretization is performed using the Finite Difference technique. For the one-dimensional and Laplace 2D models, uniforme and non-uniform meshes were used. The simulations were performed in Fortran 90 language with quadruple precision, this being the maximum precision available for storage in the language. Through the sub-mesh concept, expressive results were obtained, reaching in some cases module errors of less than 10^{-20}, obtaining a significant reduction in relation to the initial errors. In addition, there was a considerable increase in orders of accuracy, in the mathematical models employed. In some cases, they have shown to be considerably superior compared to existing techniques. In general, they showed stability in all models used, with a reduction in error and an increase in accuracy orders in all cases.

Keywords: Numerical error. Richardson extrapolation. Repetead Richardson Extrapolation. Finite Difference. Completed Richardson Extrapolation. Submeshes.

LISTA DE ALGORITMOS

1 Algoritmo base - MER. 45

2 Algoritmo base: CRE 1D, adaptado de Roache e Knupp (1993). 51

3 CRE 2D, adaptado de Roache e Knupp (1993). 52

4 Algoritmo base - SMER. 60

5 Algoritmo SER 1D, para diferenças finitas. 63

6 Algoritmo base SMER Interpolado. 64

7 Algoritmo alternativo de interpolação das correções 2D, adaptado de Moro (2018). 66

8 Resolução numérica da Equação de Laplace. 145

9 Resolução numérica da Equação de Burgers 1D. 147

10 Resolução numérica da Equação de Poisson 1D. 149

11 Algoritmo SUBER 2D, para diferenças finitas. 150

12 Algoritmo SMER 1D, para diferenças finitas. 151

13 Algoritmo SMER 2D, para diferenças finitas. 152

14 CRE clássico com Interpolação e FRE 1D (adaptado de Roache e Knupp (1993)). 153

15 Algoritmo cálculo da correção (C) por interpolação, para os nós não coincidentes, caso unidimensional. 154

LISTA DE ILUSTRAÇÕES

Figura 1 – Métodos de solução de problemas de engenharia 22
Figura 2 – Exemplos de malhas para discretização de um mesmo domínio unidimensional 25
Figura 3 – Exemplos de malhas para discretização de um mesmo domínio bidimensional 26
Figura 4 – Exemplo de malha não uniforme estruturada aplicada 26
Figura 5 – Exemplo de desempenho de MER 27
Figura 6 – Efeito da precisão de armazenamento utilizada no desempenho das variáveis $Em1$ e Eh 33
Figura 7 – Malha unidimensional, discretizada em diferenças finitas 38
Figura 8 – Exemplos de discretização em diferenças finitas para problemas bidimensionais 38
Figura 9 – Tipos de variáveis de MER 47
Figura 10 – Malhas unidimensionais discretizadas para Diferenças finitas 49
Figura 11 – Esquema de obtenção de soluções numéricas unidimensional com CRE 50
Figura 12 – Esquema de obtenção de soluções numéricas bidimensional com CRE . 50
Figura 13 – Malhas genéricas discretizadas em diferenças finitas 53
Figura 14 – Exemplo de Aplicação de CRE e algumas variações 56
Figura 15 – Passo 1, SMER 1D 59
Figura 16 – SMER: Configuração final da malha **M1** com seus pontos e coincidentes nas malhas **M2**, **M3** e **M4** já extrapolados 60
Figura 17 – SER, Malha **M1** 61
Figura 18 – SER, Malha **M2** 61
Figura 19 – SER, Malha **M3** 62
Figura 20 – Exemplo de aplicações: SMER Interpolado 64
Figura 21 – Distribuição dos pontos para interpolação 2D de grau máximo 10 . . . 67
Figura 22 – Distribuição dos pontos para interpolação 1D de grau máximo 10 . . . 67
Figura 23 – Exemplos de nós vizinhos para interpolação em torno do nó P, malha cartesiana 68
Figura 24 – Malha discretizada inicial (g), intervalo [a,b] 68
Figura 25 – Exemplo do esquema de refino uniforme com razão 2, aplicado em malhas não uniformes unidimensionais para Diferenças Finitas 71
Figura 26 – Malhas iniciais bidimensionais, utilizadas nos experimentos 72
Figura 27 – $T(x)$: Numérica vs Analítica, malha uniforme 80
Figura 28 – Resultados: SMER, Malhas Base 1, Variável l_1, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha uniforme 81

Figura 29 – Resultados: SMER, Malha Base 2, Variável l_1, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha uniforme 81
Figura 30 – Resultados: SMER, Malhas Base 7, Variável l_1, Poisson 1D com solução $T = \text{sen}(\pi x)$, Malha uniforme 82
Figura 31 – Resultados: SMER, Malha Base 8, Variável l_1, Poisson 1D com solução $T = \text{sen}(\pi x)$, Malha uniforme 82
Figura 32 – Resultados: SMER, Malhas $g = 1{,}2 \cdots ,14$, Variável l_1, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme 83
Figura 33 – Resultados: SMER INTERPOLADO (SMER-I), Poisson 1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$, Variável l_1, Malha uniforme, conjunto β_1 88
Figura 34 – Resultados: Normas l_1 e p_E^*, CRE com FRE (FRE), Poisson 1D, Malha uniforme 89
Figura 35 – Resultados: CRE clássico (CRE), Poisson 1D, Variável l_1, $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme 90
Figura 36 – Resultados: CRE com interpolação (CRE-I), Variável l_1, Burgers 1D,$Re = 1$, $T = \frac{e^{xRe} - 1}{e^{Re} - 1}$, Malha uniforme 90
Figura 37 – Resultados: SER, Variável l_1, Burgers 1D,$Re = 1$, $T = \frac{e^{xRe} - 1}{e^{Re} - 1}$, Malha uniforme 91
Figura 38 – $T(x)$: Numérica vs Analítica, malha não uniforme, Burgers 1D, $Re = 1$ 94
Figura 39 – $T(x)$: Numérica vs Analítica, Poisson 1D, malha não uniforme 95
Figura 40 – Resultados: SMER, Malhas 1-2, Variável l_1, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme 96
Figura 41 – Resultados: SMER, Malhas 7-8, Variável l_1, Poisson 1D com solução $T(x) = \text{sen}(\pi x)$, Malha não uniforme 97
Figura 42 – Resultados: SMER, Malhas $g = 1{,}2 \cdots ,14$, Variável l_1, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha não uniforme 98
Figura 43 – Resultados: CRE-I, Variável l_1, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme 99
Figura 44 – Resultados: FRE, Variável l_1, Poisson 1D com solução $T(x) = \text{sen}(\pi x)$, Malha não uniforme 99
Figura 45 – Resultados: SER, Variável l_1, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme 101
Figura 46 – $T(x,y)$: Numérica vs Analítica, Laplace 2D, malha uniforme. 102
Figura 47 – Resultados: SMER, Malhas $g = 1{,}2, \cdots ,9$, Variável l_1, Laplace 2D, Malha uniforme 105
Figura 48 – Resultados: SMER, Malhas 1 e 6, Variável l_1, Laplace 2D, Malha uniforme 106

Figura 49 – Resultados: SMER-I, Conjunto β_3, Variável l_1, Laplace 2D, Malha uniforme . . . 107
Figura 50 – Resultados: CRE, Variável l_1, Laplace 2D, Malha uniforme . . . 107
Figura 51 – Resultados: CRE-I, Variável l_1, Laplace 2D, Malha uniforme . . . 108
Figura 52 – Resultados: SER, Variável l_1, Laplace 2D, Malha uniforme . . . 108
Figura 53 – Resultados: SMER, Malhas 1 e 4, Variável l_1, Laplace 2D, Malha não uniforme . . . 111
Figura 54 – Resultados: SMER, Malhas $g = 1,2,\cdots,9$, Variável l_1, Laplace 2D, Malha não uniforme . . . 112
Figura 55 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Laplace 2D, Malha não uniforme . . . 112
Figura 56 – Resultados: CRE, Variável l_1, Laplace 2D, Malha não uniforme . . . 113
Figura 57 – Resultados: CRE-I, Variável l_1, Laplace 2D, Malha não uniforme . . . 113
Figura 58 – Resultados: SER, Variável l_1, Laplace 2D, Malha não uniforme . . . 114
Figura 59 – $u(x,y)$, $v(x,y)$: Numérica vs Analítica, Burgers 2D, $Re = 1$ . . . 115
Figura 60 – Resultados: SMER, Variável ł$_1$, Malhas $g = 1,2,\cdots,9$, Burgers 2D, $Re = 1$ . . . 116
Figura 61 – Resultados: SMER, Malha base 1, Variável l_1, Burgers 2D, $Re = 1$, velocidade u . . . 117
Figura 62 – Resultados: SMER, Malha base 1, Variável l_1, Burgers 2D, $Re = 1$, velocidade v . . . 117
Figura 63 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Burgers 2D, $Re = 1$, velocidade u . . . 118
Figura 64 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Burgers 2D, $Re = 1$, velocidade v . . . 118
Figura 65 – Resultados: CRE, Variável l_1, Burgers 2D, $Re = 1$, velocidade u . . . 119
Figura 66 – Resultados: CRE, Variável l_1, Burgers 2D, $Re = 1$, velocidade v . . . 120
Figura 67 – Resultados: CRE-I, Variável l_1, Burgers 2D, $Re = 1$, velocidade u . . . 120
Figura 68 – Resultados: CRE-I, Variável l_1, Burgers 2D, $Re = 1$, velocidade v . . . 121
Figura 69 – Resultados: SER, Variável l_1, Burgers 2D, $Re = 1$, velocidade u . . . 121
Figura 70 – Resultados: SER, Variável l_1, Burgers 2D, $Re = 1$, velocidade v . . . 122
Figura 71 – $u(x,y)$, $v(x,y)$: Numérica vs Analítica, Navier-Stokes 2D, $Re = 1$ . . . 123
Figura 72 – Resultados: SMER, Variável ł$_1$, Malhas $g = 1,2,\cdots,8$, Navier-Stokes 2D 126
Figura 73 – Resultados: SMER, Malha base 1, Variável l_1, Navier-Stokes 2D, velocidade u . . . 126
Figura 74 – Resultados: SMER, Malha base 1, Variável l_1, Navier-Stokes 2D, velocidade v . . . 127
Figura 75 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Navier-Stokes 2D, velocidade u . . . 127

Figura 76 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Navier-Stokes 2D, velocidade v . . . 128
Figura 77 – Resultados: CRE-I, Variável l_1, Navier-Stokes 2D, velocidade u . . . 128
Figura 78 – Resultados: CRE-I, Variável l_1, Navier-Stokes 2D, velocidade v . . . 129
Figura 79 – Resultados: CRE, Variável l_1, Navier-Stokes 2D, velocidade u . . . 129
Figura 80 – Resultados: CRE, Variável l_1, Navier-Stokes 2D, velocidade v . . . 130
Figura 81 – Resultados: SER, Variável l_1, Navier-Stokes 2D, velocidade u . . . 130
Figura 82 – Resultados: SER, Variável l_1, Navier-Stokes 2D, velocidade v . . . 131
Figura 83 – SER, excluindo a malha mais fina em cada extrapolação, Navier-Stokes, velocidade u, variável l_1 . . . 134
Figura 84 – Malha não uniforme, discretizada para diferenças finitas. . . . 142
Figura 85 – Resultados: SMER, Malhas 7-8, Variável l_∞, Poisson 1D com solução $T(x) = \text{sen}(\pi x)$, Malha não uniforme . . . 156
Figura 86 – Resultados: SMER, Malhas $g = 1{,}2\cdots{,}14$, Variável l_∞, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha não uniforme . . . 157
Figura 87 – Resultados: FRE, Variável l_∞, Poisson 1D com solução $T(x) = \text{sen}(\pi x)$, Malha não uniforme . . . 157
Figura 88 – Resultados: SMER, Malhas 1-2, Variável l_∞, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme . . . 158
Figura 89 – Resultados: CRE-I, Variável l_∞, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme . . . 159
Figura 90 – Resultados: SER, Variável l_∞, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme . . . 159
Figura 91 – Resultados: SMER, Malhas 7-8, Variável l_∞, Poisson 1D com solução $T = \text{sen}(\pi x)$, Malha uniforme . . . 160
Figura 92 – Resultados: SMER, Malhas $g = 1{,}2\cdots{,}14$, Variável l_∞, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme. . . . 161
Figura 93 – Resultados: SMER INTERPOLADO (SMER-I), Poisson 1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$, Variável l_∞, Malha uniforme, conjunto β_1 . . . 162
Figura 94 – Resultados: CRE clássico (CRE), Variável l_∞, Poisson 1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme. . . . 162
Figura 95 – Resultados: Normas l_∞ e p_E^*, CRE com FRE (FRE), Poisson 1D, Malha uniforme . . . 163
Figura 96 – Resultados: SMER, Malhas 1-2, Variável l_∞, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha uniforme . . . 164
Figura 97 – Resultados: CRE com interpolação (CRE-I), Variável l_∞, Burgers 1D,$Re = 1$, $T = \frac{e^{xRe} - 1}{e^{Re} - 1}$, Malha uniforme . . . 165

Figura 98 – Resultados: SER, Variável l_∞, Burgers 1D,$Re = 1$, $T = \frac{e^{xRe} - 1}{e^{Re} - 1}$, Malha uniforme 165
Figura 99 – Resultados: SMER, Malhas 1 e 6, Variável l_∞, Laplace 2D, Malha uniforme 166
Figura 100 – Resultados: SMER, Malhas $g = 1,2,\cdots,9$, Variável l_∞, Laplace 2D, Malha uniforme 167
Figura 101 – Resultados: SMER-I, Conjunto β_3, Variável l_∞, Laplace 2D, Malha uniforme 167
Figura 102 – Resultados: CRE, Variável l_∞, Laplace 2D, Malha uniforme 168
Figura 103 – Resultados: CRE-I, Variável l_∞, Laplace 2D, Malha uniforme 168
Figura 104 – Resultados: SER, Variável l_∞, Laplace 2D, Malha uniforme 169
Figura 105 – Resultados: SMER, Malhas $g = 1,2,\cdots,9$, Variável l_∞, Laplace 2D, Malha não uniforme 169
Figura 106 – Resultados: SMER, Malhas 1 e 4, Variável l_∞, Laplace 2D, Malha não uniforme 170
Figura 107 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Laplace 2D, Malha não uniforme 171
Figura 108 – Resultados: CRE, Variável l_∞, Laplace 2D, Malha não uniforme 171
Figura 109 – Resultados: CRE-I, Variável l_∞, Laplace 2D, Malha não uniforme . . . 172
Figura 110 – Resultados: SER, Variável l_∞, Laplace 2D, Malha não uniforme 172
Figura 111 – Resultados: SMER, Malhas $g = 1,2,\cdots,9$, Burgers 2D, Variável ł$_\infty$, $Re = 1$ 173
Figura 112 – Resultados: SMER, Malha base 1, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u 173
Figura 113 – Resultados: SMER, Malha base 1, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v 174
Figura 114 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u 174
Figura 115 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v 175
Figura 116 – Resultados: CRE, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u. . . . 175
Figura 117 – Resultados: CRE, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v . . . 176
Figura 118 – Resultados: CRE-I, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u . . 176
Figura 119 – Resultados: CRE-I, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v . . 177
Figura 120 – Resultados: SER, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u . . . 177
Figura 121 – Resultados: SER, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v . . . 178
Figura 122 – Resultados: SMER, Variável ł$_\infty$, Malhas $g = 1,2,\cdots,8$, Navier-Stokes 2D 178

Figura 123 – Resultados: SMER, Malha base 1, Variável l_∞, Navier-Stokes 2D, velocidade u 179
Figura 124 – Resultados: SMER, Malha base 1, Variável l_∞, Navier-Stokes 2D, velocidade v 179
Figura 125 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Navier-Stokes 2D, velocidade u 180
Figura 126 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Navier-Stokes 2D, velocidade v 180
Figura 127 – Resultados: CRE-I, Variável l_∞, Navier-Stokes 2D, velocidade u 181
Figura 128 – Resultados: CRE-I, Variável l_∞, Navier-Stokes 2D, velocidade v 181
Figura 129 – Resultados: CRE, Variável l_∞, Navier-Stokes 2D, velocidade u 182
Figura 130 – Resultados: CRE, Variável l_∞, Navier-Stokes 2D, velocidade v 182
Figura 131 – Resultados: SER, Variável l_∞, Navier-Stokes 2D, velocidade u. 183
Figura 132 – Resultados: SER, Variável l_∞, Navier-Stokes 2D, velocidade v 183

LISTA DE TABELAS

Tabela 1 – Esquema de obtenção de resultados com MER, variável de interesse ϕ, ordem de acurácia máxima p_G, onde $0 < p_0 < p_1 < p_2 < \cdots < p_G$, para G malhas com $m = G - 1$ extrapolações 45

Tabela 2 – Metodologia para o emprego de MER 48

Tabela 3 – Exemplo de aplicação de ER sobre o campo unidimensional 49

Tabela 4 – Exemplo de aplicação de CRE sobre o campo unidimensional 51

Tabela 5 – Esquema de obtenção de resultados com CRE e algumas variações, para $G = 4$ malhas com $m = 3$ extrapolações e ordem de acurácia máxima p_3, onde $0 < p_0 < p_1 < p_2 < p_3$; $p_i \in \mathbb{Z}, i = 0,1,2,3$ 56

Tabela 6 – Esquema de obtenção de resultados com **CRE** e algumas variações, para G malhas com $m = G - 1$ extrapolações, e ordem de acurácia máxima p_{G-1}, onde $0 < p_0 < p_1 < p_2 < \cdots < p_{G-1}$; $p_i \in \mathbb{Z}, i = 0, 1, \cdots G - 1$. . 57

Tabela 7 – Esquema de obtenção de resultados com **SER** e **SMER**, para G malhas, $m = G - 1$ extrapolações com ordem de acurácia máxima p_{G-1}, em que $0 < p_0 < p_1 < p_2 < \cdots < p_G$; $p_i \in \mathbb{Z}, i = 0, 1, \cdots G - 1$ 57

Tabela 8 – Esquema de obtenção de resultados com SMER, variável σ 58

Tabela 9 – Identificação de variáveis da Equação de Burgers 2D 75

Tabela 10 – Resultados: SMER, Malha base 13, Variável l_1, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme 79

Tabela 11 – Resultados: SMER, Malha base 14, Variável l_1, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme 79

Tabela 12 – SER com uma extrapolação ($m = 1$), Poisson 1D, $T = \text{sen}(\pi x)$, malha uniforme 88

Tabela 13 – SER com duas extrapolações ($m = 2$), Variável l_1, Poisson 1D, $T = \text{sen}(\pi x)$, Malha uniforme 91

Tabela 14 – Resultados: SMER INTERPOLADO, conjunto β_2, Variável l_1, Malha não uniforme, Poisson1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$ 94

Tabela 15 – Resultados: SMER INTERPOLADO, conjunto β_2, Variável l_1, Malha uniforme, Poisson1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$ 95

Tabela 16 – Resultados: SMER INTERPOLADO, conjunto β_2, Variável l_∞, Malha não uniforme, Poisson1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$ 155

Tabela 17 – Resultados: SMER, Malha base 13, Variável l_∞, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme 161

Tabela 18 – Resultados: SMER, Malha base 14, Variável l_∞, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme 161

LISTA DE ABREVIATURAS E SIGLAS

CDS	*Central Differencing Scheme*
CRE	*Completed* Richardson *Extrapolation*
CRE-I	CRE com interpolação
DDN	Diferenças Divididas de Newton
ER	Extrapolação de Richardson
FRE	*Full* Richardson *Extrapolation*
MER	Multiextrapolação de Richarson
MFD	Método de Diferenças Finitas
MFV	Método de Volumes Finitos
MSI	*Modified Strongly Implicity*
SER	SUBMALHA com ER
SMER	SUBMALHA com MER
SMER-I	SMER interpolado
SM	SUBMALHA sem MER
SM-I	SUBMALHA interpolada sem MER
TDMA	*TriDiagonal Matrix Algorithm*
V&V	Verificação e Validação

LISTA DE SÍMBOLOS

Símbolo – **Descrição**

a_P – Termo de discretização central

a_E – Termo de discretização leste

a_W – Termo de discretização oeste

a_S – Termo de discretização sul

a_N – Termo de discretização norte

b_P – Termo de discretização independente (Termo fonte)

C – Termo de correção

$C^m_{g,p}$ – Termo de correção na malha g, posição p e nível de extrapolação m

$\left(\frac{dT}{dx}\right)_j$ – Derivada da função T avaliada no ponto x_j

$\left(\frac{d^2T}{dx^2}\right)_j$ – Derivada de segunda ordem da função T avaliada no ponto x_j

E – Erro numérico

Eh – Erro de discretização

Em – Erro de discretização com MER

h – Espaçamento entre os pontos

l_∞ – Norma infinito

l_1 – Norma l_1

m – Nível de extrapolação m

$M_{g,m}$ – Malha g com m extrapolações

$N_{inicial}$ – Quantidade de nós $N_{inicial} + 1$ da malha inicial

p_U – Ordem aparente

p_E – Ordem efetiva

p^*_E – Ordem efetiva equivalente

p^*_E: l_1, SM – Ordem efetiva equivalente da norma l_1 de SM

p_E^*: l_1, SMER – Ordem efetiva equivalente da norma l_1 de SMER

p_E^*: l_∞, SM – Ordem efetiva equivalente da norma l_∞ de SM

p_E^*: l_∞, SMER – Ordem efetiva equivalente da norma l_∞ de SMER

p_V – Ordem verdadeira

p_0 – Ordem assintótica

P_i – Pontos nodais com exceção dos extremos da malha inicial

r – Razão de refino

$T_{\texttt{CDS}}^{(i)}$ – Aproximação do tipo CDS para a derivada de primeira ordem

$T_{\texttt{CDS}}^{(ii)}$ – Aproximação do tipo CDS para a derivada de segunda ordem

T_j – Função T avaliada no ponto x_j

Símbolos gregos

β – Conjunto de pontos para aplicação da técnica SMER-I

ε_T – Erro de truncamento

ε_n – Erro de iteração

ε_π – Erro de arredondamento

ε_p – Erro de programação

φ_p – Polinômio de ordem p

ϕ – Solução numérica

Φ – Solução analítica exata

Φ_∞ – Solução analítica estimada

$\phi_{g,p}^m$ – Variável de interesse ϕ na malha g, posição p e nível de extrapolação m

$\phi_{g,m}$ – Variável de interesse ϕ na malha g com nível de extrapolação m

SUMÁRIO

1 INTRODUÇÃO 22
1.1 MOTIVAÇÃO E DELIMITAÇÃO DO PROBLEMA 24
1.2 REVISÃO BIBLIOGRÁFICA 27
1.3 OBJETIVOS 29
1.4 ORGANIZAÇÃO DO TEXTO 30

2 FUNDAMENTAÇÃO TEÓRICA 31
2.1 ERRO NUMÉRICO 31
2.1.1 Erro de truncamento e ordens verdadeiras 31
2.1.2 Erro de iteração 32
2.1.3 Erro de arredondamento 32
2.1.4 Erro de programação 34
2.1.5 Erro de discretização e ordem efetiva 34
2.2 DIFERENÇAS FINITAS 36
2.2.1 Resolução de sistemas lineares 38
2.3 INTERPOLAÇÃO POLINOMIAL 39
2.3.1 Abordagem Unidimensional 40
2.3.2 Abordagem Bidimensional 41
2.3.3 Erro de Interpolação 42
2.4 VERIFICAÇÃO E VALIDAÇÃO 42
2.5 EXTRAPOLAÇÃO DE RICHARDSON E MER 43
2.5.1 Metodologia para o emprego de MER 46
2.5.2 Extrapolação de Richardson Completa 48

3 METODOLOGIA 53
3.1 CRE clássico com interpolação e FRE 54
3.2 SMER 57
3.3 SER 61
3.4 SMER INTERPOLADO 62
3.5 INTERPOLAÇÕES 65
3.5.1 SMER INTERPOLADO 65
3.5.2 CRE com interpolação 66
3.6 MODELOS MATEMÁTICOS E NUMÉRICOS 68
3.6.1 Poisson 1D 72
3.6.2 Laplace 2D 73
3.6.3 Burgers 74
3.6.4 Navier-Stokes (Problema de Shih) 75

4 RESULTADOS - MALHA UNIFORME 1D 77
4.1 SMER 77
4.2 SMER INTERPOLADO 83
4.3 CRE clássico com interpolação e FRE 84
4.4 SER 86
4.5 Resumo do capítulo e conclusões 87

5 RESULTADOS - MALHA NÃO UNIFORME 1D 92
5.1 SMER 92
5.2 SMER INTERPOLADO 93
5.3 CRE clássico com interpolação e FRE 98
5.4 SER 100
5.5 Resumo do capítulo e conclusões 100

6 RESULTADOS - LAPLACE 2D 102
6.1 MALHA UNIFORME 102
6.1.1 SMER 103
6.1.2 SMER INTERPOLADO 103
6.1.3 CRE clássico e com interpolação 104
6.1.4 SER 105
6.2 MALHA NÃO UNIFORME 109
6.2.1 SMER 109
6.2.2 SMER INTERPOLADO 109
6.2.3 CRE clássico e com interpolação 110
6.2.4 SER 110
6.3 Resumo do capítulo e conclusões 114

7 RESULTADOS: EQUAÇÃO DE BURGERS 2D 115
7.1 SMER 116
7.2 SMER INTERPOLADO 116
7.3 CRE clássico e com interpolação 119
7.4 SER 119
7.5 Resumo do capítulo e conclusões 122

8 RESULTADOS - NAVIER-STOKES 123
8.1 SMER 124
8.2 SMER INTERPOLADO 124
8.3 CRE clássico e com interpolação 124
8.4 SER 124
8.5 Resumo do capítulo e conclusões 125

9 CONCLUSÃO 132
9.1 CONTRIBUIÇÕES PARA LITERATURA 135
9.2 TRABALHOS FUTUROS 135

REFERÊNCIAS 137

APÊNDICE A – DISCRETIZAÇÕES 142
A.1 Laplace 2D 144
A.2 Burgers 1D 146
A.3 Poisson 1D 147

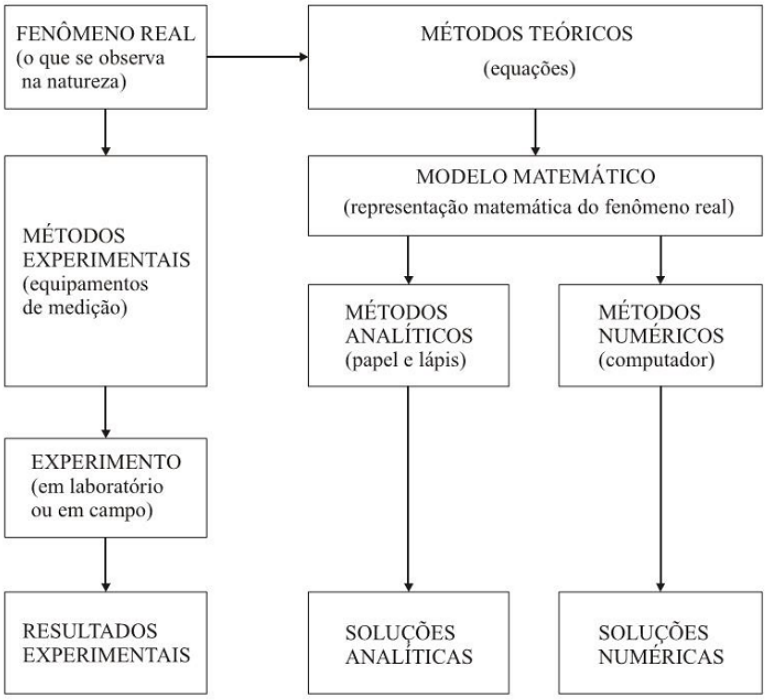
APÊNDICE B – ALGORITMOS GERAIS 150

APÊNDICE C – RESULTADOS ADICIONAIS 155
C.1 MALHA NÃO UNIFORME 1D 155
C.1.1 Poisson 155
C.1.2 Burgers 158
C.2 MALHA UNIFORME 1D 160
C.2.1 Poisson 160
C.2.2 Burgers 164
C.3 LAPLACE 2D 166
C.3.1 Malha uniforme 166
C.3.2 Malha não uniforme 169
C.4 BURGERS 2D 173
C.5 NAVIER-STOKES 178

1 INTRODUÇÃO

Existem três métodos que podem ser empregados na solução de um problema de engenharia: experimentais, analíticos e numéricos. Marchi (2001) apresenta o diagrama (Figura 1) que exemplifica o sistema de funcionamento destes três métodos.

Figura 1 – Métodos de solução de problemas de engenharia



FONTE: Adaptado de Marchi (2001).

Cada método apresenta suas vantagens e desvantagens sobre os demais, dependendo do problema e dos recursos disponíveis para resolvê-lo. Neste contexto, surgiu a área de Dinâmica dos Fluidos Computacional (em inglês, *Computational Fluid Dynamics* (CFD)). Segundo Fortuna (2000), **CFD** consiste basicamente na aplicação de métodos numéricos para resolver equações que representam problemas envolvendo fluidos em movimento, com ou sem troca de calor. Segundo Maliska (2004) a relativa facilidade de aplicação dos métodos numéricos, mesmo em problemas complexos, e a grande disseminação do computador foram os responsáveis pelo avanço desta área.

Atualmente, as demandas em CFD requerem o uso de métodos que forneçam soluções numéricas cada vez mais acuradas. Para atingir este objetivo, é necessário controlar o erro numérico (E), definido como a diferença entre a solução analítica exata (Φ) da

variável de interesse para sua numérica (ϕ), isto é:

$$E(\phi) = \Phi - \phi. \tag{1.1}$$

Muitas vezes a solução Φ é difícil de ser encontrada. Neste caso, pode-se aproximá-la, obtendo uma solução analítica estimada (Φ_∞), e diante disso é calculado a estimativa (U) do erro numérico através de

$$U(\phi) = \Phi_\infty - \phi. \tag{1.2}$$

No entanto, neste trabalho utiliza-se somente modelos matemáticos que possuem solução analítica, porque é possível mensurar exatamente qual será o erro numérico, objeto de estudo dos experimentos.

O interesse principal dos experimentos deste trabalho é reduzir E ao mínimo possível. Segundo Marchi (2001), Schneider (2007), Martins (2013) entre outros autores, o erro numérico (E) é causado por quatro fontes, são elas: **erro de truncamento** (ε_T), **erro de iteração** (ε_n), **erro de arrendondamento** (ε_π) e **erro de programação** (ε_p); simbolicamente

$$E(\phi) = E(\varepsilon_T, \varepsilon_n, \varepsilon_\pi, \varepsilon_p). \tag{1.3}$$

Quando ε_n, ε_p e ε_π são minimizados ou até mesmo inexistentes, o erro de truncamento (ε_T) passa a ser o principal causador do erro numérico. Diante disso, ε_T passa a ser denominado "erro de discretização" (Eh); neste âmbito Eh pode ser calculado diretamente pela Equação (1.1) ou estimado pela Equação (1.2), caso não se conheça Φ. Agora, buscam-se alternativas para diminuir Eh. Conforme mencionado em Martins (2013), é possível resolver o problema de três formas:

- **Refinamento da malha**;
- **Métodos de alta ordem**;
- **Técnicas de extrapolação**.

A opção empregada no trabalho são as técnicas de extrapolação. Dentre as várias extrapolações existentes, conforme algumas mencionadas e estudadas em Anunciação (2013), a Extrapolação de Richardson (ER) é uma possibilidade. Quando se utiliza ER de forma recursiva, isto é, em que cada aplicação representa um nível de extrapolação, este processo é chamado de Multiextrapolação de Richardson (MER) ou em inglês *Repeated Richardson Extrapolation* (RRE).

Neste trabalho, utiliza-se MER em problemas uni e bidimensionais, empregando tanto malhas uniformes quanto não uniformes, visando a redução de Eh.

1.1 MOTIVAÇÃO E DELIMITAÇÃO DO PROBLEMA

Segundo Ferziger, Perić e Street (2020), o primeiro passo para obter uma solução numérica é discretizar a geometria do domínio, ou seja, uma malha numérica deve ser definida. Uma malha pode ser definida como um conjunto finito de pontos utilizado para descrever de forma discreta o domínio de um problema. As figuras 2 e 3 são exemplos de malhas utilizadas em Diferenças Finitas, para os casos uni e bidimensionais. O desenvolvimento de malhas é um tema recorrente e importante em CFD. Yang e Hayes (2008), Soni, Shih e Ito (2010) e Rane e Kovacevic (2017) são exemplos de trabalhos que abordam o assunto.

Uma das motivações da realização desta pesquisa se deve pela influência ocasionada no erro numérico ao utilizar uma simples malha não uniforme no lugar da tradicional (uniforme) para discretizar o domínio de um problema, já para o caso mais simples unidimensional (Figura 2).

Define-se como uma malha não uniforme, aquela em que os tamanhos dos elementos que a compõem variam aleatoriamente (Figura 2b, 3b e 3d). Define-se como malha não estruturada quando não existe uma lei de formação para ordenação dos seus pontos (Figura 3d), ela é estruturada caso contrário.

Segundo Maliska (2004), em CFD existe a necessidade de discretizar problemas que envolvam grande complexidade geométrica, onde as malhas uniformes se tornam ineficazes em adaptar com êxito a geometria do problema. Portanto, melhorar as soluções obtidas para estas situações se torna fundamental. Na figura 4 é apresentado um exemplo de malha não uniforme aplicada.

Neste trabalho serão utilizadas as malhas exemplificadas nas figuras 2, 3a e 3b. Deve ser ressaltado que caso uma malha estruturada e uniforme consiga adaptar perfeitamente a geometria do problema, ela deve ser priorizada. Para este caso em geral não se justifica a utilização de outro tipo de malha, em virtude do custo computacional e erro numérico.

Dando continuidade, para obter uma maior acurácia das soluções numéricas a Multiextrapolação de Richardson (MER) é uma importante ferramenta para atingir este objetivo. O estímulo da pesquisa se deve aos seus resultados significativos na redução do Eh e aumento da ordem de acurácia com um custo computacional extremamente baixo. Por exemplo, na Figura 5 é mostrado o gráfico bilogarítmico entre "h: distância entre os nós da malha X $|E|$: Eh em módulo" de uma variável de interesse.

Através deste exemplo é possível verificar a notória diferença de resultados conforme h é reduzido desde a sua malha inicial, usualmente denotada como "malha grossa", até suas malhas finais originadas de seu refino, usualmente denotadas por "malha fina", "super fina", etc. Para obter estes resultados não é necessário realizar novas simulações, são utilizadas somente as soluções numéricas já obtidas, funcionando como um pós-processamento de dados. Neste exemplo é adaptado um dos resultados obtidos em Carvalho Junior (2015).

Entretanto, não é possível aplicar ER para reduzir o erro em todos os pontos de um domínio, conforme é mostrado no trabalho de Roache e Knupp (1993), e consequentemente MER também não se aplica. Denota-se por "CAMPO" todos os pontos do domínio do problema.

Para aplicar ER de forma efetiva para campos, Roache e Knupp (1993) propõem como solução a "Extrapolação de Richardson Completa" (CRE), obtendo ótimos resultados em seus experimentos.

Desde então, este conceito de extrapolação no campo tem sido estudado e aprimorado. Neste trabalho, através de CRE e algumas variações, são realizadas multiextrapolações no campo em problemas envolvendo malhas uniformes e não uniformes, para os casos uni e bidimensionais. São comparados desempenho e custo computacional, posteriormente serão propostos novos métodos para realização deste procedimento.

Figura 2 – Exemplos de malhas para discretização de um mesmo domínio unidimensional

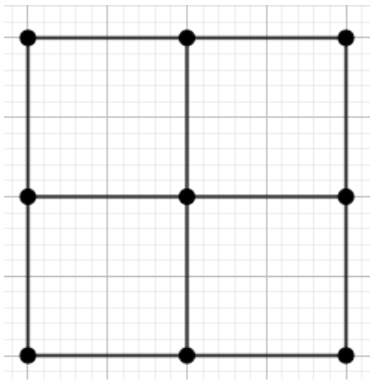
(a) Malha uniforme

(b) Malha não uniforme

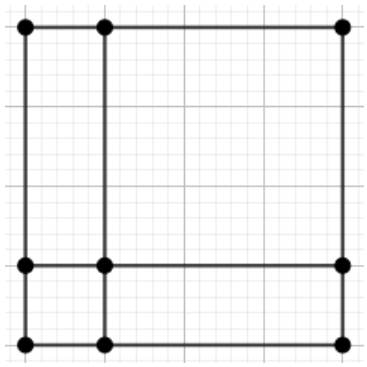
FONTE: Autor(2020).

Figura 3 – Exemplos de malhas para discretização de um mesmo domínio bidimensional

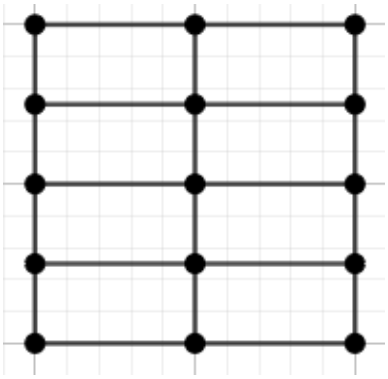
(a) Malha estruturada uniforme



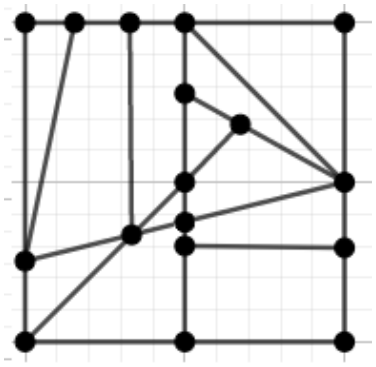
(b) Malha estruturada não uniforme



(c) Malha estruturada uniforme por direção

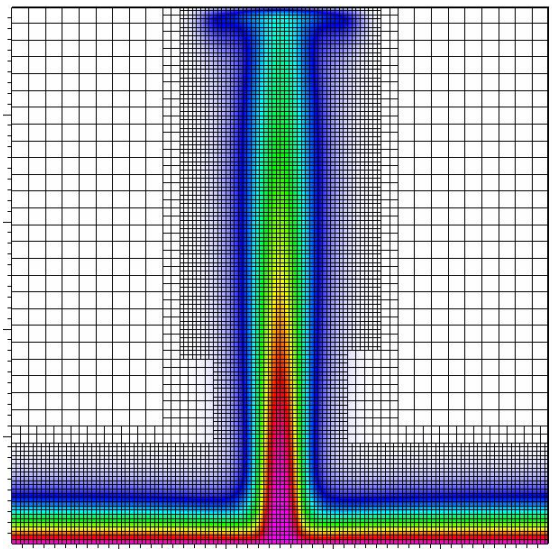


(d) Malha não estruturada e não uniforme



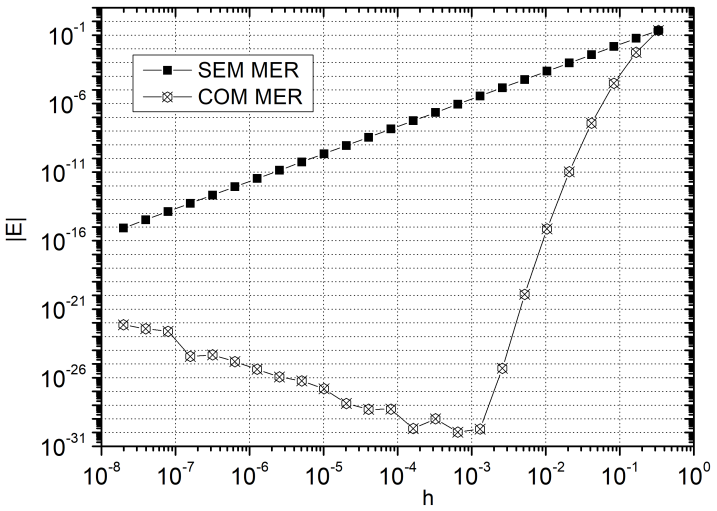
FONTE: Autor(2020).

Figura 4 – Exemplo de malha não uniforme estruturada aplicada



FONTE: Adaptado de Oliveira (2015).

Figura 5 – Exemplo de desempenho de MER



FONTE: Adaptado de Carvalho Junior (2015).

1.2 REVISÃO BIBLIOGRÁFICA

Este trabalho é baseado na Extrapolação de Richardson (ER), criada e desenvolvida inicialmente em Richardson (1910) e Richardson e Gaunt (1927), tendo como objetivo aperfeiçoar as aproximações numéricas, isto é, aumentar sua ordem de acurácia e reduzir o erro numérico envolvido na resolução de equações diferenciais pelo método de Diferenças Finitas. As necessidades da época já demandavam métodos rápidos, fáceis de serem entendidos e aplicáveis às equações estudadas. Richardson e Gaunt (1927) já consideravam a aplicação de ER de maneira repetida (MER), com dois níveis de extrapolação.

A vantagem do uso de MER se dá pelo fato de ser um pós-processamento simples, pois não interfere diretamente na obtenção da solução. Seu custo computacional é muito baixo em termos de memória e tempo de CPU. Pode ser aplicada a códigos computacionais já existentes ou a resultados já obtidos. Aplica-se a diversas aproximações numéricas e variáveis de interesse. Independe de análises a priori e conhecimento da solução analítica do problema (VARGAS, 2013).

Existem vários trabalhos na literatura que empregam ER ou MER, seja como base ou simplesmente aplicada, listaremos brevemente alguns trabalhos que abordam o assunto nos últimos anos.

Celik e Karatekin (1997) utilizaram ER para reduzir o erro de discretização em malhas não uniformes. Andreev e Popov (1999) utilizaram ER para desenvolver malhas adaptativas com alta ordem de acurácia. Munyakazi e Patidar (2008) realizaram um estudo teórico sobre a acurácia de ER e Franke e Frank (2008) utilizaram ER para estimar o erro numérico.

Celik et al. (2005) adaptaram ER para problemas oscilatórios, o que já indica que ER na teoria não é indicado para este modelo. Germer e Marchi (2009) avaliaram o efeito das aproximações numéricas sobre a redução do erro obtidas com MER. Alves (2010) e Vargas (2013) utilizaram MER para verificação de soluções numéricas.

Martins (2013) propôs uma metodologia para o emprego de MER, onde são classificadas as variáveis de interesse em cinco tipos e para cada tipo de variável é sugerido uma forma de aplicação. Marchi e Giacomini (2014) analisaram o efeito de MER ao variar o refino da malha.

Fator em comum na maioria destes trabalhos são as análises ou a forma de aplicar MER (ou ER), feitas para algumas ou apenas uma variável de interesse específica. No entanto, este trabalho tem como objetivo principal aplicar ER e posteriormente MER em todos os pontos do domínio do problema (CAMPO). Neste cenário, a "Extrapolação de Richardson Completa" (CRE) é uma das técnicas pioneiras, porque adapta ER para este objetivo.

Considerada versão estendida de ER para campos, a "Extrapolação de Richardson Completa" (CRE) foi proposta por Roache e Knupp (1993), desenvolvida para diferenças finitas, onde foram abordadas as equações de Poisson 1D com termos fontes diferentes, Advecção-Difusão 1D com número de Peclet 1,16 e 100, Poisson 2D em um domínio unitário. Neste trabalho os autores descrevem a técnica e a aplicam em modelos testes já comentados anteriormente.

Roache e Knupp (1993) em seus experimentos aplicaram apenas uma extrapolação, embora previssem a possibilidade da realização de mais extrapolações. A conclusão obtida pelos autores é que o método funciona e pode ser aplicado em trabalhos futuros. Inicialmente a técnica foi desenvolvida para aproximações de segunda ordem e nos seus experimentos foi possível progredir uma ordem verdadeira, isto é, de dois para quatro.

Existem alguns trabalhos na literatura que utilizam a ideia de Roache e Knupp (1993) ou que abordam ER para campos, listaremos brevemente os trabalhos encontrados nesta revisão bibliográfica.

Richards (1997) aplicou CRE na equação advecção de difusão 1D transiente. Wang e Zhang (2009) utilizaram ER combinado com multigrid para obter uma aproximação compacta de ordem seis para a equação de Poisson 2D, cuja discretização foi realizada com aproximação compacta de ordem quatro. Dai, Zhang e Wang (2013) desenvolveram de forma similar a CRE uma aproximação compacta de ordem seis, em seu experimento foi utilizada a equação de Poisson 2D também com uma aproximação compacta de ordem quatro.

Giacomini (2013) desenvolveu uma variação de CRE: FRE (Extrapolação de Richardson Total). Através desta variação foi possível realizar multiextrapolações no

campo em seus experimentos foram utilizados problemas uni e bidimensionais. Giacomini (2013) também verificou que é possível aplicar CRE em uma aproximação compacta de ordem quatro para obter uma aproximação de ordem seis, este experimento foi realizado para a equação de Poisson 1D.

Zhang e Zhang (2016) utilizaram interpolações com ER para realização de uma extrapolação no campo, porém não alcançaram resultados significativos nas malhas mais finas. Os experimentos foram realizados para a equação de Poisson 1D. Dai, Zhang e Wang (2016) propuseram o uso de CRE com o método ADI (Alternating Direction Implicit) de alta ordem para resolver problemas de convecção-difusão bidimensional.

Ferreira (2017) adaptou CRE para volumes finitos. Dai e Lin (2018) obtiveram aproximações de sexta ordem para a equação de Poisson 2D com aproximação compacta de ordem quatro, utilizando malhas uniformes por direção, porém não realizam multiextrapolações no campo.

Moro (2018) e Silva (2019) desenvolveram variações para CRE com interpolações unidimensionais, conseguindo realizar multiextrapolações no campo. Ambos utilizam Diferenças Finitas e Volumes Finitos aplicados em problemas uni e bidimensionais.

Comparativos serão feitos posteriormente com alguns trabalhos, assim como mais detalhes sobre a teoria de CRE, ER e MER. Porém, na pesquisa realizada não foram encontrados estudos referentes a abordagens utilizando malhas diferentes das usuais (uniformes). Além dos trabalhos de Celik e Karatekin (1997) e Carvalho Junior, Marchi e Martins (2015) abordam o assunto com resultados promissores, porém não abordou-se extrapolações no campo, muito menos as multiextrapolações no campo.

Neste cenário, este trabalho tem caráter introdutório, porque além das malhas não uniformes, naturalmente existem outras, por exemplo: malhas não ortogonais e triangulares. É importante destacar a necessidade constante de aprimorar as soluções numéricas e neste âmbito, MER pode ser uma ótima ferramenta para este objetivo, sendo essencial aprimorá-la para diversas situações.

1.3 OBJETIVOS

O **objetivo geral** deste trabalho é contribuir na aplicação de MER para CAMPOS. Pode-se estabelecer como **objetivos específicos**:

1. Reduzir o erro de discretização para as variáveis de interesse estudadas, abordando malhas uni e bidimensionais uniformes e não uniformes.

2. Estudar e investigar os efeitos ocasionados em MER pelo tipo de malha empregado no campo.

3. Desenvolver novas técnicas para complementar ou simplesmente melhorar resultados já existentes na literatura.

4. Estabelecer, se possível novos conceitos sobre o tema.

1.4 ORGANIZAÇÃO DO TEXTO

A continuidade do texto consiste da seguinte forma:

- Capítulo 2: conceitos fundamentais para o desenvolvimento do trabalho, como erro numérico, interpolação polinomial, método de diferenças finitas, Extrapolação de Richardson, entre outros.
- Capítulo 3: metodologia utilizada, técnicas empregadas, modelos matemáticos e numéricos e variáveis de interesse.
- Capítulo 4: exposição dos resultados obtidos para equações unidimensionais discretizadas com malhas uniformes.
- Capítulo 5: exposição dos resultados obtidos para equações unidimensionais discretizadas com malhas não uniformes.
- Capítulo 6 : exposição dos resultados obtidos para equação de Laplace 2D, para este modelo foram utilizados malhas uniformes e não uniformes.
- Capítulo 7: exposição dos resultados obtidos para equação de Burgers 2D.
- Capítulo 8: exposição dos resultados obtidos para equação de Navier-Stokes 2D.
- Capítulo 9: conclusão, análises gerais dos testes, trabalhos futuros e contribuições do trabalho para a literatura.
- Apêndices A, B e C: discretizações, algoritmos e alguns resultados adicionais, respectivamente.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados resumidamente alguns fundamentos importantes que constituem a base teórica do trabalho: Erro Numérico, Diferenças Finitas, Resolução de Equações Algébricas, Interpolação Polinomial, Verificação e Validação, Extrapolação e Multiextrapolação de Richardson e Extrapolação de Richardson Completa.

2.1 ERRO NUMÉRICO

Define-se como erro numérico (E) a diferença entre a solução analítica exata (Φ) de uma variável de interesse e sua solução numérica (ϕ):

$$E(\phi) = \Phi - \phi. \tag{2.1}$$

Segundo Marchi (2001), Schneider (2007), Martins (2013), Vargas (2013) entre outros autores, o erro numérico pode ser gerado por quatro fontes: erro de truncamento (ε_T), erro de iteração (ε_n), erro de arredondamento (ε_π) e erro de programação (ε_p). Neste cenário, pode-se reescrever a Equação (2.1) de forma simbólica:

$$E(\phi) = E(\varepsilon_T, \varepsilon_n, \varepsilon_\pi, \varepsilon_p). \tag{2.2}$$

Conforme estas fontes de erro são diminuídas, tornando algumas até inexistentes, a solução numérica (ϕ) estará cada vez mais acurada. Se conhecida a solução analítica exata (Φ), ϕ tenderá para Φ, isto é

$$\begin{array}{ll} \lim \phi & = \Phi \\ (\varepsilon_T, \varepsilon_n, \varepsilon_\pi, \varepsilon_p) \rightarrow (0,0,0,0) & \end{array}. \tag{2.3}$$

2.1.1 Erro de truncamento e ordens verdadeiras

Segundo Marchi (2001), o erro de truncamento (ε_T) de uma aproximação numérica é o resíduo da substituição da solução analítica exata de uma variável dependente (T) na equação discretizada do modelo matemático.

Considerando uma aproximação numérica realizada através da Série de Taylor, e admitindo-se inicialmente um espaçamento uniforme (h) entre os nós da malha, pode-se representar a equação geral do erro de truncamento por:

$$\varepsilon_T = c_0 h^{p_0} + c_1 h^{p_1} + c_2 h^{p_2} + \cdots, \tag{2.4}$$

em que os coeficientes c_i são constantes, não necessariamente positivas. Por definição as ordens verdadeiras são os expoentes de h dos termos não nulos da Equação (2.4), sendo números inteiros positivos, que seguem a relação $1 \leq p_0 < p_1 < p_2 < p_3 < \cdots < \cdots$.

Esta sequência constitui geralmente uma série aritmética, isto é, a diferença entre ordens subsequentes é constante. Em geral é uma sequência infinita, porque é constituída por uma quantidade infinita de termos. O menor expoente de h (p_0), por definição, é chamado de ordem assintótica (p_0). Este inteiro positivo tem a propriedade $p_0 \geq 1$.

Quando o tamanho (h) dos elementos da malha tende a zero, ou seja, $h \to 0$, o primeiro termo da Equação (2.4) é o principal componente, isto é, ele domina o valor de ε_T. Neste caso, em gráficos logarítmicos de ε_T versus h, a inclinação em relação ao eixo das abscissas da curva ε_T para $h \to 0$ tende ao valor da ordem assintótica (MARCHI, 2001).

A ordem assintótica pode ser obtida através de uma estimava de erro a priori, sendo muito importante para o desenvolvimento deste trabalho. Mais detalhes sobre o assunto, veja Roache (2009).

2.1.2 Erro de iteração

O erro de iteração (ε_n) ocorre quando se utilizam métodos iterativos para a obtenção da solução numérica (ϕ). Métodos como Gauss-Seidel e Jacobi-Richardson são exemplos de métodos numéricos utilizados para resolução de sistemas lineares, em que se obtém a solução numérica (ϕ_n) por iterações. Em geral, quanto maior o número de iterações (n), melhor a solução numérica e consequentemente menor o seu erro de iteração (ε_n).

Segundo Ferziger, Perić e Street (2020), o erro de iteração (ε_n) é definido como a diferença entre a solução exata da equação discretizada e a solução numérica em uma determinada iteração. Matematicamente, tem-se

$$\varepsilon_n(\phi) = \phi_{k\to\infty} - \phi_k, \tag{2.5}$$

onde k representa o número de iteração corrente no processo e o símbolo "$k \to \infty$" é utilizado para representar a solução exata das equações discretizadas.

Na área de CFD, o trabalho de Martins e Marchi (2008) é um exemplo de trabalho que aborda o assunto.

2.1.3 Erro de arredondamento

Os erros de arredondamento (ε_π) são os erros que ocorrem principalmente devido à representação finita dos números reais na computação, ou seja, também são erros de truncamento, porém oriundos da necessidade de se limitar o número de dígitos usados para armazenar os valores das variáveis. Eles dependem do compilador (*software*) usado para gerar o código computacional e do computador (*hardware*) empregado em sua execução (MARCHI, 2001).

Este erro tende a crescer conforme é diminuído o espaçamento entre os nós da malha discretizada ou seja, o erro de arredondamento (ε_π) cresce com o refino da malha. ε_π pode até afetar o desempenho de MER; por exemplo na Figura 6 é apresentada a influência da precisão utilizada de armazenamento, em que Eh representa o erro de discretização sem MER e $Em1$ corresponde por Eh com MER.

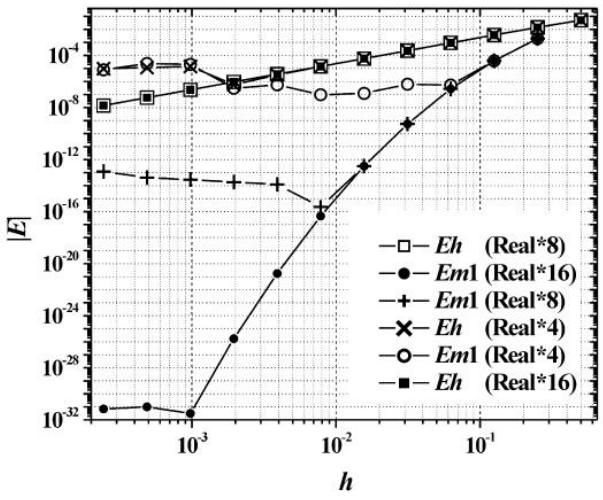
Neste exemplo, um gráfico bilogarítmico entre o módulo do erro ($|E|$) pela distância entre os nós da malha (h). A variável $Em1$ tem seu desempenho afetado conforme o armazenamento é alterado para precisão simples (Real*4), dupla (Real*8) e quádrupla (Real*16).

Estas precisões mencionadas anteriormente, estão relacionadas ao número de algarismos (bytes) usados para representar as variáveis nos computadores. Particularmente na linguagem Fortran onde foi realizado o experimento da Figura 6, temos:

- Precisão simples (Real*4): quatro bytes;
- Precisão dupla (Real*8): oito bytes;
- Precisão quádrupla (Real*16): 16 bytes.

Portanto, para diminuir o máximo possível este erro recomenda-se utilizar a precisão de armazenamento máxima disponível.

Figura 6 – Efeito da precisão de armazenamento utilizada no desempenho das variáveis $Em1$ e Eh



FONTE: Adaptado de Marchi et al. (2013).

2.1.4 Erro de programação

O erro de programação (ε_p) é ocasionado pelo programador na implementação do código computacional e na execução/configuração da simulação. Existem várias explicações que levam a esse erro, por exemplo: falta de conhecimento teórico do algoritmo, distração e inexperiência com a linguagem de programação utilizada.

2.1.5 Erro de discretização e ordem efetiva

Quando ε_n, ε_p e ε_π são minimizadas ou até mesmo inexistentes, o erro de truncamento (ε_T) passa a ser o principal causador do erro numérico. Neste caso, ε_T passa a ser denominado erro de discretização (Eh). Geralmente, para que o erro numérico seja gerado apenas pelo erro de discretização, deve-se verificar segundo Marchi (2001):

- os modelos matemáticos abordados sejam simples o suficiente para evitar os erros de programação;
- a precisão computacional adotada para armazenamento das variáveis seja adequada para minimizar erros de arredondamento;
- no caso de soluções iterativas, o número de iterações seja suficiente para se atingir o erro de máquina ou para reduzir ao mínimo os erros de iteração.

Assumindo que as fontes de erros estão controladas, a Equação (2.4), pode ser reescrita da forma:

$$Eh = c_0 h^{p_0} + c_1 h^{p_1} + c_2 h^{p_2} + \cdots \tag{2.6}$$

$$1 \leq p_0 < p_1 < p_2 < p_3 < \cdots . \tag{2.7}$$

Se a solução analítica exata (Φ) é conhecida, podemos calcular Eh pela Equação (2.1)

$$Eh = E(\phi) = \Phi - \phi. \tag{2.8}$$

Além disso, pode-se calcular a ordem efetiva (p_E), definida como a inclinação local do erro de discretização (Eh) da solução numérica (ϕ) versus o tamanho (h) dos elementos da malha em um gráfico bilogarítmico. Seu cálculo permite verificar na prática, isto é, a *posteriori* das soluções numéricas, se p_E tende à ordem assintótica dos erros de truncamento, ordem esta que é um resultado teórico, obtido a *priori* das soluções numéricas.

Para obter a ordem efetiva é necessário ao menos duas soluções numéricas ϕ_1 e ϕ_2 obtidas nas malhas Ω^{h_1} (grossa) e Ω^{h_2} (fina), respectivamente, em que a relação entre as duas é a razão de refino (r):

$$r = \frac{h_1}{h_2} \quad (r > 1). \tag{2.9}$$

As constantes h_1 e h_2 são os espaçamentos uniformes entre os nós das malhas Ω^{h_1} e Ω^{h_2}. A expressão de p_E é dada por

$$p_E = \frac{\log\left[\dfrac{\Phi - \phi_1}{\Phi - \phi_2}\right]}{\log(r)}. \tag{2.10}$$

No caso de malhas não uniformes, se o refino é uniforme, a variável r também é dada pela Equação (2.9), em que as variáveis h_1 e h_2 representam o espaçamento médio necessário entre os nós das malhas Ω^{h_1} e Ω^{h_2}. Existem outras formas de se refinar uma malha não uniforme, como pode ser visto em Schneider (2007) e consequentemente r sofre alteração. Neste trabalho, utiliza-se somente refino uniforme; esta escolha é justificada mais adiante.

Dando continuidade ao assunto, conforme h é reduzido espera-se que p_E tenda à ordem assintótica (p_0); em outras palavras

$$\lim_{h \to 0} p_E = p_0. \tag{2.11}$$

A Equação (2.11) diz que a ordem efetiva pode ser interpretada como a ordem de acurácia do método escolhido. Ela serve para medir a taxa em que o método converge. Neste aspecto, a ordem assintótica (p_0) seria a ordem de acurácia estimada a priori do método. Em geral, a Equação (2.11) é satisfeita; quando isto ocorre é um indicativo de que a aproximação utilizada está funcionando adequadamente. Vale ressaltar que esta análise a posteriori só é possível quando o erro numérico está apenas dominado pelo erro de truncamento.

Um exemplo de quando a Equação (2.11) não é válida pode ser vista no trabalho de Marchi (2001), ao se utilizar malhas não uniformes, com refino uniforme. Caso a Equação (2.11) não seja satisfeita, primeiramente deve-se verificar se Eh não é afetado por algum outro tipo de erro (Ex: Erro de programação). Caso não seja esta opção, deve-se verificar na literatura a ocorrência de casos semelhantes e em seguida investigar, procurando justificativas matemáticas para explicar o ocorrido.

No entanto, caso não seja conhecida a solução analítica Φ, é possível medir a acurácia de Eh através de análises semelhantes ao que foi mostrado anteriormente. Para

tanto, utilizam-se estimadores do erro discretização, também conhecido na literatura por estimativa (U). Além disso, pode-se obter a ordem aparente (p_U) da estimativa, dada pela equação

$$p_U = \frac{\log\left[\dfrac{\phi_2 - \phi_1}{\phi_3 - \phi_2}\right]}{\log(r)}, \tag{2.12}$$

em que ϕ_1, ϕ_2 e ϕ_3 correspondem, respectivamente, às soluções numéricas obtidas nas malhas Ω^{h_1} (grossa), Ω^{h_2} (fina), Ω^{h_3} (superfina); com $r = h_1/h_2 = h_2/h_3$ (razão de refino constante).

A ordem aparente formalmente é definida como a inclinação local da curva da estimativa (U) da solução numérica (ϕ) versus o tamanho h dos elementos da malha num gráfico bilogarítmico. Em geral vale

$$\lim_{h\to 0} p_U = p_0, \tag{2.13}$$

isto é, a análise da acurácia do método pode ser realizada através do cálculo de p_U. Portanto, em outras palavras, p_U tem função similar a p_E, tornando-se fundamental para eventuais análises, quando Φ não é conhecida.

Roache (2009), AIAA (2002) e Ferziger, Perić e Street (2020) são exemplos de trabalhos que abordam com mais detalhes o tema do erro numérico. No presente trabalho não são calculados U e p_U porque todos os modelos matemáticos utilizados possuem solução analítica. Na próxima seção é apresentado o método numérico utilizado nas discretizações.

2.2 DIFERENÇAS FINITAS

O método numérico utilizado neste trabalho, para obtenção das soluções numéricas, chama-se Método de Diferenças Finitas (MDF). Seu princípio básico consiste em aproximar através de expressões algébricas cada termo do modelo matemático em cada nó da malha discretizada. As discretizações podem ser feitas em domínios com várias dimensões, conforme é ilustrado pelas Figuras 7 e 8, em que são apresentados exemplos de malhas discretizadas para problemas uni e bidimensionais, abordados neste trabalho.

As aproximações utilizadas são do tipo CDS (*Central Difference Scheme*), para aproximar as derivadas de primeira e segunda ordens avaliadas no nó (ou ponto) j, que envolvem a variável dependente T e independentes x e y (caso bidimensional).

Para obtê-las, considere sem perda de generalidade a malha discretizada ilustrada na Figura 7, considerando o nó j com coordenada x_j. As aproximações são feitas, através de expansões da série de Taylor em torno do ponto j, para os seus nós vizinhos:

$$T_{j-1} = T_j - \left(\frac{dT}{dx}\right)_j h_j + \left(\frac{d^2T}{dx^2}\right)_j \frac{h_j^2}{2} - \left(\frac{d^3T}{dx^3}\right)_j \frac{h_j^3}{6} + \cdots, \tag{2.14}$$

$$T_{j+1} = T_j + \left(\frac{dT}{dx}\right)_j h_{j+1} + \left(\frac{d^2T}{dx^2}\right)_j \frac{h_{j+1}^2}{2} + \left(\frac{d^3T}{dx^3}\right)_j \frac{h_{j+1}^3}{6} + \cdots. \tag{2.15}$$

As constantes h_{j+1} e h_j são os espaçamentos, não necessariamente uniformes em relação ao nó j, ilustrados pela Figura 7. Utiliza-se a notação: $T_j = T(x_j)$ e $\left(\frac{dT}{dx}\right)_j = T^{(i)}(x_j)$.

Para aproximar as derivadas de primeira e segunda ordens, avaliadas no nó j, são subtraídas as equações (2.14) e (2.15), para cada caso, isolando os termos $\left(\frac{d^2T}{dx^2}\right)_j$ e $\left(\frac{dT}{dx}\right)_j$, isto é:

$$\left(\frac{dT}{dx}\right)_j = \frac{T_{j+1} - T_{j-1}}{h_j + h_{j+1}} - \left(\frac{d^2T}{dx^2}\right)_j \frac{(h_{j+1}^2 - h_j^2)}{2(h_j + h_{j+1})} - \left(\frac{d^3T}{dx^3}\right)_j \frac{h_{j+1}^3 + h_{j+1}^3}{6(h_j + h_{j+1})} \tag{2.16}$$
$$- \cdots = (T^i_{\texttt{CDS}})_j + \varepsilon(T^i_{\texttt{CDS}})_j$$

e

$$\left(\frac{d^2T}{dx^2}\right)_j = 2\frac{[h_{j+1}T_{j-1} + h_jT_{j+1} - (h_j + h_{j+1})T_j]}{(h_j + h_{j+1})h_jh_{j+1}} + \left(\frac{d^3T}{dx^3}\right)_j \frac{h_j - h_{j+1}}{3} \tag{2.17}$$
$$- \left(\frac{d^4T}{dx^4}\right)_j \frac{h_j^3 + h_{j+1}^3}{12(h_j + h_{j+1})} + \cdots = (T^{ii}_{\texttt{CDS}})_j + \varepsilon(T^{ii}_{\texttt{CDS}})_j.$$

Portanto, obtêm-se as aproximações das derivadas de primeira ($T^i_{\texttt{CDS}}$) e segunda ordens ($T^{ii}_{\texttt{CDS}}$), avaliadas no nó j, do tipo CDS, dadas pelas equações

$$\left(\frac{d^2T}{dx^2}\right)_j \approx (T^{ii}_{\texttt{CDS}})_j = 2\frac{[h_{j+1}T_{j-1} + h_jT_{j+1} - (h_j + h_{j+1})T_j]}{(h_j + h_{j+1})h_jh_{j+1}} \tag{2.18}$$

$$\left(\frac{dT}{dx}\right)_j \approx (T^i_{\texttt{CDS}})_j = \frac{T_{j+1} - T_{j-1}}{h_j + h_{j+1}} \tag{2.19}$$

e seus erros de truncamento $\varepsilon(T^i_{\texttt{CDS}})_j$, $\varepsilon(T^{ii}_{\texttt{CDS}})_j$, são respectivamente

$$\varepsilon(T^i_{\texttt{CDS}})_j = -\left(\frac{d^2T}{dx^2}\right)_j \frac{(h_{j+1}^2 - h_j^2)}{2(h_j + h_{j+1})} - \left(\frac{d^3T}{dx^3}\right)_j \frac{h_{j+1}^3 + h_{j+1}^3}{6(h_j + h_{j+1})} - \cdots \tag{2.20}$$

$$\varepsilon(T^{ii}_{\texttt{CDS}})_j = \left(\frac{d^3T}{dx^3}\right)_j \frac{h_j - h_{j+1}}{3} - \left(\frac{d^4T}{dx^4}\right)_j \frac{h_j^3 + h_{j+1}^3}{12(h_j + h_{j+1})} + \cdots. \tag{2.21}$$

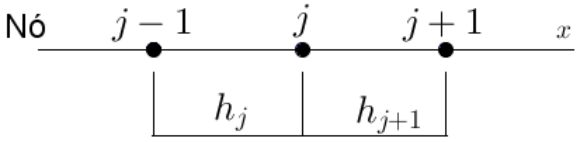
No entanto, não é possível determinar a priori as ordens verdadeiras neste caso. Portanto é considerado o padrão $p_0 = 1$,$p_1 = 2$,$p_2 = 3, \cdots$. Porém, com base em vários testes foi verificado a posteriori que as ordens verdadeiras são $p_0 = 2$, $p_1 = 4$, $p_2 = 6$,$\cdots$, ao se utilizar refino uniforme, isto é, são idênticas para o caso de malhas uniformes, em que

$$h_j = h_{j+1} = h. \tag{2.22}$$

Justificando-se a escolha da utilização do refino uniforme porque neste cenário é possível obter análises efetivas a posteriori do erro.

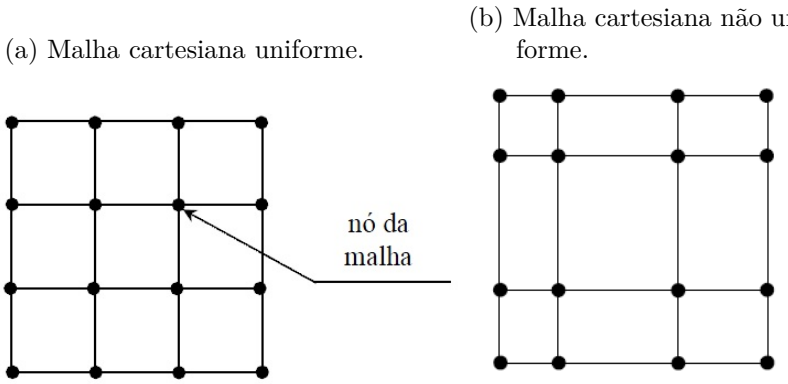
Portanto, graças ao refino uniforme é esperado obter a posteriori as mesmas ordens de acurácia para ambos os casos: malha uniforme e não uniforme. Recomenda-se a leitura de Schneider (2007), caso o leitor deseje obter mais informações sobre este resultado.

Figura 7 – Malha unidimensional, discretizada em diferenças finitas



FONTE: Autor(2020).

Figura 8 – Exemplos de discretização em diferenças finitas para problemas bidimensionais



FONTE: Adaptado de Schneider (2007).

2.2.1 Resolução de sistemas lineares

Após a discretização, seja por MDF ou algum outro método numérico similar, por exemplo Volumes Finitos, é possível obter um sistema linear

$$Ax = b, \tag{2.23}$$

em que A é a matriz dos coeficientes, b é o vetor independente e x é um vetor de incógnitas (solução numérica). A estrutura da matriz A vai depender do método usado para discretizar o modelo matemático. Segundo Maliska (2004), a estrutura da matriz de coeficientes obtida

na aproximação é de fundamental importância na escolha do método de solução do sistema linear. Por exemplo, as matrizes tridiagonais com coeficientes (a_P,a_E,a_W,b)

$$\begin{bmatrix} -a_W & a_P & -a_E & & & \\ & -a_W & a_P & -a_E & & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & -a_W & a_P & -a_E & \\ & & & -a_W & a_P & -a_E \end{bmatrix}_{N\text{x}N} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ \vdots \\ x_N \end{bmatrix}_{N\text{x}1} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ \vdots \\ \vdots \\ b_N \end{bmatrix}_{N\text{x}1} \tag{2.24}$$

são frequentes em CFD na discretização de modelos unidimensionais, em que a_E, a_P, e a_W representam os termos discretizados leste, central e oeste, respectivamente, b: termo independente, N: dimensão do problema e x são as incógnitas. Já para os casos bidimensionais e tridimensionais é comum trabalhar com matrizes pentadiagonal e heptadiagonal, respectivamente.

Neste trabalho, aborda-se os casos uni e bidimensionais. Para resolver o modelo unidimensional utiliza-se o método direto TDMA (TriDiagonal Matrix Algorithm). Para o caso bidimensional, utiliza-se o método iterativo Gauss-Seidel e MSI (Modified Strongly Implicit Method). As discretizações utilizadas e os respectivos sistemas lineares, podem ser vistos no Apêndice A.

Para mais detalhes sobre métodos de discretização e formas de resolver o sistema resultante recomenda-se a leitura das obras Maliska (2004) e Fortuna (2000).

2.3 INTERPOLAÇÃO POLINOMIAL

Com a resolução de um modelo matemático através do emprego de um método numérico são obtidos valores para ϕ em pontos nodais determinados pela malha adotada. Entretanto, em alguns casos, pode ocorrer a necessidade de se obter ϕ em locais específicos que não coincidem com os pontos dessa malha. Nessa situação, o emprego de interpolação polinomial representa uma ferramenta que pode ser empregada com sucesso (MARTINS, 2013).

Segundo Franco (2006), aproximar funções por polinômios é uma das ideias mais antigas da análise numérica, e até hoje é umas das técnicas mais usadas na área. Isto ocorre, pelo fato dos polinômios serem facilmente computáveis, integrais e derivadas de fácil obtenção e, por fim suas raízes podem ser obtidas, quando possível, de forma razoavelmente simples. Neste trabalho são realizadas duas abordagens, são os casos uni e bidimensionais.

2.3.1 Abordagem Unidimensional

Neste trabalho, o problema a ser resolvido é definido por dados $n+1$ pontos reais distintos

$$x_0 < x_1 < x_2 < x_3 < \cdots < x_n, \tag{2.25}$$

e $n+1$ números reais y_0, y_1, .. , y_n, valores de uma função $f : [x_0,x_n] \longrightarrow \mathbb{R}$, podendo ser discreta, em que

$$y_0 = f(x_0), y_1 = f(x_1), y_2 = f(x_2), .., y_n = f(x_n). \tag{2.26}$$

Procura-se um polinômio $\varphi_n(x)$ de grau máximo n, definido no intervalo $[x_0,x_n]$, que possua a propriedade

$$\varphi_n(x_0) = y_0; \quad \varphi_n(x_1) = y_1; \quad ... \quad \varphi_n(x_n) = y_n. \tag{2.27}$$

Este polinômio existe, sendo único, desde que satisfaça as condições do teorema a seguir:

Teorema 2.3.1 *Dados $n+1$ pontos distintos x_0, x_1, ..., x_n (reais ou complexos) e $n+1$ valores y_0, y_1, ..., y_n, existe um e só um polinômio $\varphi_n(x)$, de grau menor ou igual a n, tal que*

$$\varphi_n(x_k) = y_k, \quad k = 0,1,...,n. \tag{2.28}$$

Uma versão da demonstração deste importante resultado clássico pode ser visto em Franco (2006). Este teorema diz que independentemente do método utilizado, desconsiderando o erro de interpolação e outras fontes de erros possíveis, qualquer método que for utilizado para obter o polinômio (φ_n) da equação (2.28), obtém-se a mesma solução. Neste trabalho utiliza-se a técnica de Diferenças Divididas de Newton, para $n+1$ pontos distintos, definida pelas funções:

$$\left[\begin{array}{ll} (1) & f[x_0,x_1] = \frac{f(x_1)-f(x_0)}{x_1-x_0} \\ (2) & f[x_0,x_1,x_2] = \frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0} \\ \vdots & \\ (n) & f[x_0,x_1,...,x_n] = \frac{f[x_1,x_2,...,x_n]-f[x_0,x_1,...,x_{n-1}]}{x_n-x_0} \end{array}\right. . \tag{2.29}$$

Utilizando as funções acima, obtém-se o polinômio (φ_n) de Newton de grau menor ou igual a n, representado pela equação:

$$\begin{aligned} \varphi_n(x) = f(x_0) + (x-x_0)f[x_0,x_1] + (x-x_0)\cdot(x-x_1)f[x_0,x_1,x_2] + \\ \cdots + (x-x_0)\cdot(x-x_1)\cdots(x-x_{n-1})f[x_0,x_1,...,x_n]. \end{aligned} \tag{2.30}$$

Uma segunda alternativa possível é dada pela fórmula de Lagrange, definida pela equação

$$\varphi_n(x) = \sum_{i=0}^{n} f(x_i) l_i(x) \tag{2.31}$$

em que

$$l_i(x) = \prod_{\substack{k=0 \\ k \neq i}}^{n} \frac{(x - x_i)}{(x_k - x_i)}. \tag{2.32}$$

2.3.2 Abordagem Bidimensional

Neste cenário mais complexo, dados $p_x + 1$ e $p_y + 1$ pontos reais distintos

$$x_0 < x_1 < x_2 < x_3 < \cdots < x_{p_x} \text{ e } y_0 < y_1 < y_2 < y_3 < \cdots < y_{p_y}, \tag{2.33}$$

nas direções dos eixos x e y, respectivamente, e $k = (p_x + 1)(p_y + 1)$ números reais z_0, z_1, ... , $z_{(p_x+1)(p_y+1)}$, valores de uma função $f : [x_0,x_{p_x}]\mathtt{X}[y_0,y_{p_y}] \longrightarrow \mathbb{R}$, podendo ser discreta, do conjunto

$$\{f(x_i,y_j)/i = 0,1,2,..,p_x;\ j = 0,1,2,...,p_y\}. \tag{2.34}$$

O objetivo é determinar um polinômio $\varphi_p^*(x,y)$ de grau máximo $p^* = \mathtt{Max}\{p_x,p_y\}$, definido no domínio $[x_0,x_{p_x}]\mathtt{X}[y_0, y_{p_y}]$, que possua a propriedade

$$\{\varphi_p^*(x_i,y_i) = f(x_i,y_j)/\ i = 0,1,2,..,p_x;\ j = 0,1,2,...,p_y\}. \tag{2.35}$$

A existência deste polinômio $\varphi_p^*(x,y)$ pode ser mostrada através da fórmula de Lagrange bidimensional, técnica utilizada no trabalho, definida pela equação

$$\varphi_{p^*}(x,y) = \sum_{i=0}^{p_x} \sum_{j=0}^{p_y} f(x_i,y_j) l_i(x) l_j(y) \tag{2.36}$$

onde

$$l_i(x) = \prod_{\substack{k=0 \\ k \neq i}}^{p_x} \frac{(x - x_i)}{(x_k - x_i)},\ l_j(y) = \prod_{\substack{k=0 \\ k \neq j}}^{p_y} \frac{(y - y_j)}{(y_k - y_j)}. \tag{2.37}$$

Porém, para garantir a unicidade, é necessário acrescentar a hipótese

$$p_x = p_y. \tag{2.38}$$

Uma versão da demonstração deste importante resultado clássico pode ser visto em Dahlquist e Bjorck (2008) ou Martins (2013).

2.3.3 Erro de Interpolação

Conforme apresentado anteriormente, teoricamente o polinômio de interpolação (φ_n) apresenta a propriedade básica para os pontos escolhidos x_k. Seja x_k com dimensão uni ou bidimensional, a função φ_n é uma aproximação exata da função $f(x)$ nestes pontos, porém para $\vee\overline{x} \neq x_k$, em geral não é verdade que $\varphi_n(\overline{x}) = f(\overline{x})$.

Neste trabalho avaliou-se o erro de interpolação $\xi_n(x)$ definido pela equação

$$\xi_n(x) = f(x) - \varphi_n(x) \quad \forall x \in \mathbb{R}^k, k = 1{,}2 \tag{2.39}$$

nos pontos interpolados, para cada caso uni e bidimensional. Neste cenário a função f corresponde à solução numérica no ponto interpolado. Ambas as técnicas Lagrange e Diferenças Divididas de Newton (DDN), apresentaram resultados similares.

Para o caso unidimensional optou-se pela DDN, porque pelos testes realizados obteve menor erro de arredondamento. Porém, no caso bidimensional foi utilizado apenas Lagrange. Para trabalhos futuros é interessante testar a versão de DDN bidimensional.

Recomenda-se a leitura da obra de Steffensen (1950), caso o leitor deseja mais informações sobre a teoria da interpolação.

2.4 VERIFICAÇÃO E VALIDAÇÃO

Antes de abordar as extrapolações de Richardson (ER), é necessário destacar o tema de **Verificação e Validação numérica** (V&V), uma importante área de pesquisa em CFD que aborda a confiabilidade de uma simulação numérica.

Usualmente as palavras "Verificação" e "Validação" são tratadas como sinônimos, no entanto, no âmbito da engenharia atual, elas têm significados distintos. Segundo Roy (2005), **Verificação** é um processo puramente matemático, que tem como ocupação avaliar o quão bem o método numérico resolve um modelo matemático — avalia a resolução numérica de equações. **Validação**, por sua vez, trata da adequação do modelo matemático em reproduzir a realidade — avalia se as equações descrevem corretamente o fenômeno físico estudado.

Como este trabalho aborda a parte de pós-processamento, são utilizados dois procedimentos básicos de **Verificação**, porque não é objetivo do trabalho validar o modelo matemático. Os dois procedimentos básicos utilizados são as estimativas a **priori** e a **posteriori**.

Segundo Germer (2009), na estimativa a *priori*, procura-se avaliar de forma quantitativa e qualitativa a ordem do erro gerado pelas diferentes aproximações numéricas disponíveis para utilização. Já no caso da estimativa a *posteriori*, faz-se uma análise do erro de forma quantitativa, determinando a diferença entre a solução exata. Caso não

seja conhecida, e necessita-se então de estimativas de erro com base em soluções exatas estimadas ou também conhecidas como incertezas da solução.

Para mais detalhes sobre os procedimentos de V&V recomenda-se a leitura do manual AIAA (2002). Verificada a confiança do código, a acurácia das soluções numéricas deve ser investigada e consequentemente técnicas de pós processamento de dados como ER e MER podem ser aplicadas, no intuito de melhorar as soluções obtidas sem a necessidade da realização de novas simulações.

2.5 EXTRAPOLAÇÃO DE RICHARDSON E MER

A metodologia denominada **Extrapolação de Richardson** (ER) e sua versão aprimorada Multiextrapolação de Richardson (MER), podem ser aplicadas quando a técnica de aproximação utilizada possui um termo indicativo de erro previsível, isto é, uma forma que dependa apenas de um parâmetro real h.

Para empregar ER ou MER efetivamente em problemas de CFD, espera-se que o erro numérico seja dominado apenas pelo erro de discretização (Eh), isto é, neste cenário Eh corresponde ao parâmetro real h, comentado anteriormente. O objetivo de ER e MER é aumentar a acurácia das soluções já obtidas e consequentemente reduzir Eh. Portanto, pode-se pensar em ER e MER como técnicas de pós-processamentos de dados.

Para aplicar ER são necessárias ao menos duas soluções numéricas $\phi(h)$ e $\phi(rh)$, em que $\phi(h)$ e $\phi(rh)$ correspondem aos valores obtidos nas malhas Ω^h (Malha fina) e Ω^{rh} (Malha grossa) ($rh > h$), que tem a mesma ordem assintótica p_0. Lembrando do conceito de ordem verdadeira (Seção 2.1.1):

$$1 \leq p_0 < p_1 < p_2 < p_3 < \cdots . \tag{2.40}$$

A relação entre estas duas malhas está na razão de refino (r), preferencialmente uniforme. De Richardson (1910), a expressão de ER para a solução analítica exata (Φ) é dada pela expressão:

$$\Phi = \phi(h) + \frac{\phi(h) - \phi(rh)}{r^{p_0} - 1} + O(h^{p_1}), \tag{2.41}$$

onde $O(h^{p_1})$ representa o termo indicativo da ordem do erro. Pela definição de ordem verdadeira $p_1 > p_0$, portanto a aproximação obtida por ER, tem ordem assintótica mais elevada do que $\phi(h)$ e $\phi(rh)$ e, consequentemente, proporcionando a obtenção de um resultado mais acurado. Finalmente, a aproximação obtida por ER (ϕ_∞) extrapolada é dada pela expressão:

$$\phi_\infty = \phi(h) + \frac{\phi(h) - \phi(rh)}{r^{p_0} - 1}. \tag{2.42}$$

Segundo Vargas (2013), ER é um método simples que contribui para a taxa de convergência, como também para um resultado mais acurado.

Aplicando a equação (2.41) de forma recursiva em G malhas distintas Ω^{h^1}, Ω^{h^2}, Ω^{h^3}, Ω^{h^2},...,Ω^{h^g},...,Ω^{h^G}, geradas com razão de refino $r = h^{g-1}/h^g$ (g = 2,..,G), obtém-se a **Multiextrapolação de Richardson**:

$$\phi_{g,m} = \phi_{g,m-1} + \frac{\phi_{g,m-1} - \phi_{g-1,m-1}}{r^{p_{m-1}} - 1};\ g = 2,..,G;\ m = 1,...,g-1, \tag{2.43}$$

em que m representa o nível da extrapolação e g o número da malha. Portanto as variáveis $\phi_{1,0}$, $\phi_{2,0}$, $\phi_{3,0}$,.., $\phi_{G,0}$ equivalem às soluções numéricas ϕ_1, ϕ_2, ϕ_3,.., ϕ_G obtidas nas malhas Ω^{h^1}, Ω^{h^2}, Ω^{h^3}, Ω^{h^2},...,Ω^{h^g},...,Ω^{h^G}, respectivamente.

Na Tabela 1 é apresentado o esquema de obtenção dos resultados, com MER. Para uma variável de interesse ϕ, é perceptível a evolução da ordem de acurácia do método, teoricamente, conforme são feitas as extrapolações. Portanto, na teoria, utilizando MER a convergência do método é acelerada significativamente em cada extrapolação, proporcionando uma solução mais acurada, sem a necessidade de realizar mais simulações. No Algoritmo 1, apresenta-se a técnica de MER, baseado no que é exposto pela Tabela 1.

O cálculo da ordem efetiva (acurácia) com MER é realizada através da equação:

$$(p_E)_{g,m} = \frac{\log\left[\dfrac{\Phi - \phi_{g-1,m}}{\Phi - \phi_{g,m}}\right]}{\log(r)}, \tag{2.44}$$

se conhecida a solução analítica exata. Para calcular o erro de discretização com MER (Em) em cada extrapolação, basta calcular a diferença da solução analítica exata (Φ) com a solução extrapolada ($\phi_{g,m}$):

$$Em = \Phi - \phi_{g,m}. \tag{2.45}$$

Além disso, é possível estimar p_U, caso Φ não seja conhecida, através da equação

$$(p_U)_{g,m} = \frac{\log\left[\dfrac{\phi_{g-1,m} - \phi_{g-2,m}}{\phi_{g,m} - \phi_{g-1,m}}\right]}{\log(r)}, \tag{2.46}$$

e para Em, existem várias possibilidades, entre elas o estimador Delta (Δ), proposto por Marchi, Suero e Araki (2009)

$$U_\Delta = |\phi_{G,m-1} - \phi_{G-1,m-1}|. \tag{2.47}$$

Lembrando que neste trabalho não são abordadas estimativas porque todos os modelos matemáticos utilizados possuem solução analítica e deste modo, é possível mensurar exatamente o erro numérico, objeto de estudo do trabalho. Recomenda-se o trabalho de Martins (2013), caso o leitor deseje mais detalhes sobre estimativas para MER.

Teoricamente, à medida que $h \to 0$, os valores das ordens de acurácia obtidas em cada extrapolação $(p_E)_{g,m}$ devem tender às ordens verdadeiras do método empregado, estimadas a priori, do respectivo nível de extrapolação (m). Assim, a convergência do método empregado é acelerada consideravelmente.

Os trabalhos de Vargas (2013) e Martins (2013) são recomendados, caso o leitor deseje mais informações sobre as técnicas de ER e MER.

Tabela 1 – Esquema de obtenção de resultados com MER, variável de interesse ϕ, ordem de acurácia máxima p_G, onde $0 < p_0 < p_1 < p_2 < \cdots < p_G$, para G malhas com $m = G - 1$ extrapolações

p_v	p_0	p_1	p_2	p_3	$\cdots$	p_{G-1}
$\downarrow g \ / \ \overrightarrow{m}$	$m=0$	$m=1$	$m=2$	$m=3$	$\cdots$	$m=G-1$
1	$\phi_{1,0}$					
2	$\phi_{2,0}$	$\phi_{2,1}$				
3	$\phi_{3,0}$	$\phi_{3,1}$	$\phi_{3,2}$			
4	$\phi_{4,0}$	$\phi_{4,1}$	$\phi_{4,2}$	$\phi_{4,3}$		
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	
G	$\phi_{G,0}$	$\phi_{G,1}$	$\phi_{G,2}$	$\phi_{G,3}$	$\cdots$	$\phi_{G,G-1}$

FONTE: Autor(2020).

Algoritmo 1: Algoritmo base - MER.

Entrada: Obter soluções numéricas para a variável de interesse ϕ em G malhas distintas: ϕ_1, ϕ_2, $\cdots$, ϕ_G.
Fornecer o valor de r, p_0 e dv (razão de refino entre as malhas, ordem de acurácia das soluções numéricas e variação das ordens verdadeiras p_0,p_1,p_2,.., respectivamente).

Saída: $\phi_{g,m}$ (Tabela 1).

1 Faça: $\phi_{1,0} = \phi_1, \phi_{2,0} = \phi_2, \cdots, \phi_{G,0} = \phi_G$
para $m = 1, \cdots, G-1$ **faça**
2 $\quad p_{m-1} = p_0 + (m-1)dv$
$\quad$ **para** $g = m+1, m+2, \cdots, G$ **faça**
3 $\quad\quad \phi_{g,m} = \phi_{g,m-1} + \dfrac{\phi_{g,m-1} - \phi_{g-1,m-1}}{r^{p_{m-1}} - 1}$

2.5.1 Metodologia para o emprego de MER

Baseado nos trabalhos de Martins (2013) e Martins et al. (2013), para melhor empregar MER na variável de interesse, deve-se primeiramente identificar qual o seu tipo, de acordo com a classificação:

Tipo I: são variáveis locais, cuja localização da coordenada (c) é mantida em todas as malhas (g) consideradas e coincide com um ponto nodal (Figura 9I), ou ainda, é caracterizada por uma variável global.

Tipo II: são variáveis locais em que c situa-se no ponto médio entre pontos nodais de (g), ou seja, suas coordenadas coincidem com a média aritmética entre as coordenadas nodais (Figura 9II).

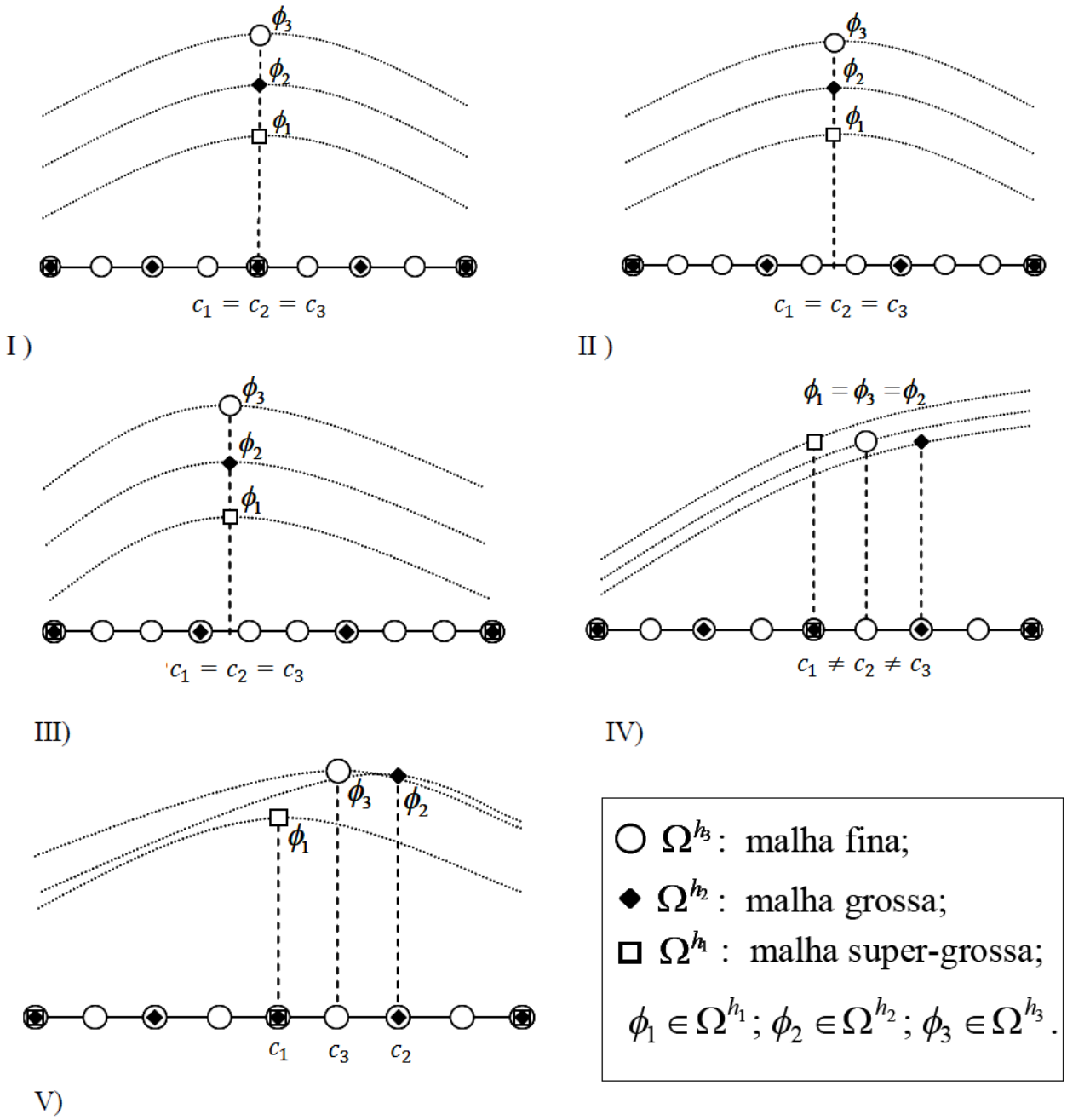
Tipo III: são variáveis locais com c não coincidente com um ponto nodal de g, nem com o ponto médio entre coordenadas nodais, entretanto, possui uma localização fixa (Figura 9III).

Tipo IV: são variáveis locais cujo valor é conhecido previamente. Nesse caso, busca-se identificar c (Figura 9IV).

Tipo V: são variáveis locais onde c é desconhecida previamente, isto é, são variáveis que podem apresentar mudança de coordenada em malhas distintas (Figura 9V).

Para cada variável de interesse ϕ são determinadas formas para sua obtenção, de acordo com a sua classificação, resumidamente apresentado pela Tabela 2.

Figura 9 – Tipos de variáveis de MER



FONTE: Adaptado de Martins (2013), Martins et al. (2013).

Tabela 2 – Metodologia para o emprego de MER

Variável	**Obtenção de ϕ**
Tipo I	Obtém-se ϕ em cada malha e emprega-se MER diretamente com a Eq. (2.43).
Tipo II	Emprega-se interpolação linear, para obtenção da ϕ, e em seguida considera-se a Eq. (2.43).
Tipo III	Emprega-se interpolação polinomial, com grau máximo possível, para obtenção da ϕ, em seguida considera-se a Eq. (2.43).
Tipo IV	Emprega-se interpolação polinomial, com grau máximo possível, e resolve-se a equação polinomial obtida considerando-se a igualdade entre o polinômio obtido e o valor estabelecido a priori, para a obtenção de ϕ em cada malha e, então, considera-se a Eq. (2.43).
Tipo V	Emprega-se interpolação polinomial com o grau máximo possível e, busca-se o seu ponto ótimo (máximo ou mínimo) para determinação de ϕ em cada malha, e ao final considera-se a Eq. (2.43).

FONTE: Adaptado de (MARTINS et al., 2013).

2.5.2 Extrapolação de Richardson Completa

A extrapolação de Richardson completa (CRE) é uma extensão da Extrapolação Richardson. Criada por Roache e Knupp (1993), considera-se completa quando uma solução de alta ordem de acurácia é produzida em todos os pontos da malha fina, isto é, em todo o campo do domínio de cálculo. Da mesma forma que ER e MER, CRE é usada como pós-processamento de dados, não tendo nenhuma relação com a geração das soluções numéricas.

Para entender seu funcionamento, considere por exemplo, três malhas unidimensionais, discretizadas para Diferenças Finitas com dois, três e cinco nós respectivamente, conforme é mostrado na Figura 10. Cada ponto $\bullet_i$ corresponde a uma solução numérica, obtida na malha através da resolução da equação algébrica originada da discretização. Neste exemplo, através de ER é possível apenas obter soluções extrapoladas nos pontos ímpares (Malhas 2 e 3), conforme é mostrado na Tabela 3. Utiliza-se a notação $\phi^m_{g,p}$, onde

- m: representa o nível da extrapolação,
- ϕ: variável de interesse,

- g: número da malha,
- p: posição do ponto.

Figura 10 – Malhas unidimensionais discretizadas para Diferenças finitas

Malha 1	$\bullet_1$				$\bullet_2$
Malha 2	$\bullet_1$		$\bullet_2$		$\bullet_3$
Malha 3	$\bullet_1$	$\bullet_2$	$\bullet_3$	$\bullet_4$	$\bullet_5$

FONTE: Autor (2020).

Tabela 3 – Exemplo de aplicação de ER sobre o campo unidimensional

pV	p_0					p_1		
$\downarrow g$ / $\overrightarrow{m}$	$m=0$					$m=1$		
1	$\phi^0_{1,1}$				$\phi^0_{1,2}$			
2	$\phi^0_{2,1}$		$\phi^0_{2,2}$		$\phi^0_{2,3}$	$\phi^1_{2,1}$		$\phi^1_{2,3}$
3	$\phi^0_{3,1}$	$\phi^0_{3,2}$	$\phi^0_{3,3}$	$\phi^0_{3,4}$	$\phi^0_{3,5}$	$\phi^1_{3,1}$	$\phi^1_{3,3}$	$\phi^1_{3,5}$

FONTE: Autor (2020).

No trabalho de Roache e Knupp (1993), os autores mostram teoricamente que ER é aplicável somente nos nós ímpares. Para obter as soluções acuradas nos restantes dos nós, é proposto o acréscimo da correção C, obtido através da realização de média simples ou linear dos termos

$$\frac{\phi(h) - \phi(rh)}{r^{p_0} - 1}, \tag{2.48}$$

da equação original da Extrapolação de Richardson (Equação 2.42), obtidos nos nós vizinhos. Portanto, conclui-se que a Extrapolação de Richardson Completa pode ser resumida pela equação

$$P \longleftarrow P + C, \tag{2.49}$$

em que P representa o conjunto de pontos com soluções numéricas da malha, que recebe ($\longleftarrow$) as originais com o acréscimo da correção C, para cada caso. Por exemplo, na Figura 11 é apresentado o esquema de obtenção das soluções numéricas com CRE. Os pontos representados por $\blacktriangle_i$ são obtidos com o termo de correção obtido através de ER e $\triangle_i$ pela média das correções.

Utilizando esta estratégia é possível realizar uma extrapolação em todos os pontos do domínio e na teoria com maior acurácia, fato que pode ser observado pela Tabela 4. Nela é ilustrada o exemplo abordado anteriormente para ER (Tabela 3), mas agora com CRE.

Para o caso bidimensional, é necessário para os nós centrais empregar interpolações além da média simples para cada direção, podendo ser bilinear, desde que a interpolação escolhida respeite a ordem do esquema da aproximação numérica. Na Figura 12 é apresentado um exemplo de aplicação. Utiliza-se a notação lexicográfica e cada ponto $\bullet_i$, $\triangle_i$ e $\blacktriangle_i$, corresponde a um ponto com coordenadas (x,y). Nos algoritmos 2 e 3 são apresentados os pseudocódigos de ambos os casos.

Note que é possível repetir o uso de CRE. Por exemplo, observa-se na Figura 11 que é possível continuar aplicando CRE para a malha 3, já com as soluções extrapoladas obtidas nas malhas 2 e 3. Neste cenário, seria equivalente a versão "completa" de MER, embora os autores de CRE previssem este acontecimento, não chegaram a aplicar este conceito. No entanto, na literatura já é possível encontrar trabalhos que abordam o tema. Outra sugestão dos autores seria utilizar submalhas, assunto abordado no Capítulo 3, como CRE novamente.

Figura 11 – Esquema de obtenção de soluções numéricas unidimensional com CRE

	$m=0$					$m=1$				
Malha 1	$\bullet_1$				$\bullet_2$					
Malha 2	$\bullet_1$		$\bullet_2$		$\bullet_3$	$\blacktriangle_1$		$\triangle_2$		$\blacktriangle_3$
Malha 3	$\bullet_1$	$\bullet_2$	$\bullet_3$	$\bullet_4$	$\bullet_5$	$\blacktriangle_1$	$\triangle_2$	$\blacktriangle_3$	$\triangle_4$	$\blacktriangle_5$

FONTE: Autor (2020).

Figura 12 – Esquema de obtenção de soluções numéricas bidimensional com CRE

$m=0$					$m=1$				
$\bullet_7$		$\bullet_8$		$\bullet_9$					
$\bullet_4$		$\bullet_5$		$\bullet_6$					
$\bullet_1$		$\bullet_2$		$\bullet_3$					

$\bullet_{21}$	$\bullet_{22}$	$\bullet_{23}$	$\bullet_{24}$	$\bullet_{25}$	$\blacktriangle_{21}$	$\triangle_{22}$	$\blacktriangle_{23}$	$\triangle_{24}$	$\blacktriangle_{25}$
$\bullet_{16}$	$\bullet_{17}$	$\bullet_{18}$	$\bullet_{19}$	$\bullet_{20}$	$\triangle_{16}$	$\triangle_{17}$	$\triangle_{18}$	$\triangle_{19}$	$\triangle_{20}$
$\bullet_{11}$	$\bullet_{12}$	$\bullet_{13}$	$\bullet_{14}$	$\bullet_{15}$	$\blacktriangle_{11}$	$\triangle_{12}$	$\blacktriangle_{13}$	$\triangle_{14}$	$\blacktriangle_{15}$
$\bullet_6$	$\bullet_7$	$\bullet_8$	$\bullet_9$	$\bullet_{10}$	$\triangle_6$	$\triangle_7$	$\triangle_8$	$\triangle_9$	$\triangle_{10}$
$\bullet_1$	$\bullet_2$	$\bullet_3$	$\bullet_4$	$\bullet_5$	$\blacktriangle_1$	$\triangle_2$	$\blacktriangle_3$	$\triangle_4$	$\blacktriangle_5$

FONTE: Autor (2020).

Tabela 4 – Exemplo de aplicação de CRE sobre o campo unidimensional

pV	p_0					p_1				
$\downarrow g$ / $\overrightarrow{m}$	$m=0$					$m=1$				
1	$\phi^0_{1,1}$				$\phi^0_{1,2}$					
2	$\phi^0_{2,1}$		$\phi^0_{2,2}$		$\phi^0_{2,3}$	$\phi^1_{2,1}$		$\phi^1_{2,2}$		$\phi^1_{2,3}$
3	$\phi^0_{3,1}$	$\phi^0_{3,2}$	$\phi^0_{3,3}$	$\phi^0_{3,4}$	$\phi^0_{3,5}$	$\phi^1_{3,1}$	$\phi^1_{3,2}$	$\phi^1_{3,3}$	$\phi^1_{3,4}$	$\phi^1_{3,5}$

FONTE: Autor (2020).

Algoritmo 2: Algoritmo base: CRE 1D, adaptado de Roache e Knupp (1993).

Entrada: Obter G soluções numéricas distintas: $\phi^0_{1,p}$ com nós $p = 0,1,\cdots,N_{inicial}$; $\phi^0_{2,p}$ com nós $p = 0,1,2,\cdots,2N_{inicial}$; $\cdots$; $\phi^0_{G,p}$ com nós $p = 0,1,2,\cdots,2^{G-1}N_{inicial}$.

Fornecer o valor de p_0 (ordem de acurácia das soluções numéricas).

Saída: $\phi^1_{g,p}$, para $g = 2,3,4,\cdots,G$ e $p = 0,1,2,\cdots,2^{g-1}N_{inicial}$.

para $g = 2,3,\cdots G$ **faça**
- $N = 2^{g-1}N_{inicial}$
- **para** $i = 0,2,4,6,..,N$ **faça**
  - $C^1_{g,i} = \dfrac{\phi^0_{g,i} - \phi^0_{g-1,i/2}}{2^{p_0} - 1}$
  - $\phi^1_{g,i} = \phi^0_{g,i} + C^1_{g,i}$
- **para** $i = 0,2,4,6,..,N-2$ **faça**
  - $C^1_{g,i+1} = \dfrac{C^1_{g,i} + C^1_{g,i+2}}{2}$
  - $\phi^1_{g,i+1} = \phi^0_{g,i+1} + C^1_{g,i+1}$

Algoritmo 3: CRE 2D, adaptado de Roache e Knupp (1993).

Entrada: Obter G soluções numéricas distintas: $\phi^0_{1,p}$ com $Nx+1$ e $Ny+1$ ($Nx>0$ e $Ny>0$ pares) nós na direção x e y respectivamente; $\phi^0_{2,p}$ com $2Nx+1$ e $2Ny+1$ nós na direção x e y respectivamente; $\cdots$,$\phi^0_{G,p}$ com $2^{G-1}Nx+1$ e $2^{G-1}Ny+1$ nós na direção x e y respectivamente.
Fornecer o valor de p_L (ordem de acurácia das soluções numéricas).

Saída: $\phi^1_{g,p}$, para $g=2,3,4,\cdots,G$ e $2^{g-1}Nx+1$, $2^{g-1}Ny+1$ nós na direção x e y, respectivamente.

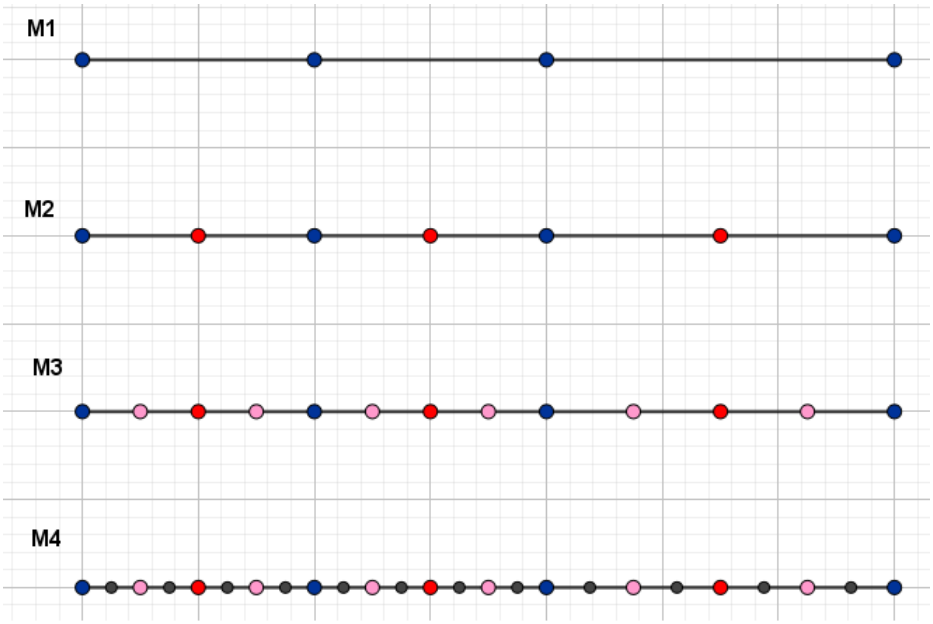
para $g=2,3,\cdots,G$ **faça**
- $NxI = 2^{g-1}Nx+1$
- $NyI = 2^{g-1}Ny+1$
- $k=1$
- **para** $j=1,3,5,\cdots,NyI$ **faça**
  - **para** $i=1,3,5,\cdots,NxI$ **faça**
    - $p = i+NxI(j-1)$
    - $C^1_{g,p} = \dfrac{\phi^0_{g,p}-\phi^0_{g-1,k}}{2^{p_L}-1}$
    - $\phi^1_{g,p} = \phi^0_{g,p}+C^1_{g,p}$
    - $k=k+1$
- **para** $j=1,3,5,\cdots,NyI$ **faça**
  - **para** $i=1,3,5,\cdots,NxI-2$ **faça**
    - $p = i+1+NxI(j-1)$
    - $C^1_{g,p} = \dfrac{C^1_{g,p-1}+C^1_{g,p+1}}{2}$
    - $\phi^1_{g,p} = \phi^0_{g,p}+C^1_{g,p}$
- **para** $j=1,3,5,\cdots,NyI-2$ **faça**
  - **para** $i=1,3,5,\cdots,NxI$ **faça**
    - $p = i+NxI\cdot j$
    - $C^1_{g,p} = \dfrac{C^1_{g,p-NxI}+C^1_{g,p+NxI}}{2}$
    - $\phi^1_{g,p} = \phi^0_{g,p}+C^1_{g,p}$
- **para** $j=1,3,5,\cdots,NyI-2$ **faça**
  - **para** $i=1,3,5,\cdots,NxI-2$ **faça**
    - $p = i+1+NxI(j+1)$
    - $C^1_{g,p} = \dfrac{C^1_{g,p-1-NxI}+C^1_{g,p+1-NxI}+C^1_{g,p-1+NxI}+C^1_{g,p+1+NxI}}{4}$
    - $\phi^1_{g,p} = \phi^0_{g,p}+C^1_{g,p}$

3 METODOLOGIA

Neste capítulo é apresentada a metodologia para aplicação de extrapolações no campo, para os casos uni e bidimensionais.

Para exemplificar a metodologia empregada, considere as malhas genéricas **M1**, **M2**, **M3** e **M4**, representadas na ilustração a seguir, discretizadas em diferenças finitas.

Figura 13 – Malhas genéricas discretizadas em diferenças finitas



FONTE: Autor(2020).

As malhas **M2**, **M3** e **M4** são geradas a partir da malha **M1**, obtidas com refino uniforme de razão 2. Pode-se notar a existência de pontos coincidentes nas malhas, os quais são destacados com a mesma cor. Neste trabalho, procura-se melhorar as soluções numéricas em todos os nós de cada malha (CAMPO), variáveis de interesse e suas ordens de acurácia. Para tanto propõe-se as técnicas:

- CRE clássico (**CRE**) com interpolação (**CRE-I**) e **FRE**;
- SUBMALHA com MER (**SMER**) e **SMER** INTERPOLADO (**SMER-I**);
- SUBMALHA com ER (**SER**).

Nas próximas seções elas são apresentadas com análises prévias. Em seguida, abordam-se os modelos matemáticos e numéricos e as variáveis de interesse utilizadas nos experimentos.

3.1 CRE clássico com interpolação e FRE

Denota-se por CRE clássico (**CRE**) a primeira versão da extrapolação de Richardson Completa, desenvolvida por Roache e Knupp (1993), servindo de motivação para o desenvolvimento de variações. Esta técnica utiliza duas malhas por vez, sendo uma fina e outra grossa, onde a malha grossa e fina são relacionadas pelo refino uniforme de razão dois.

O objetivo da técnica é melhorar as soluções numéricas em todos os nós na malha mais fina, consistindo basicamente em dois passos:

1. Separa-se os nós coincidentes das duas malhas;
2. Na malha mais fina acrescenta-se o fator de correção C em cada nó, para cada caso.

Este fator de correção C, conforme mencionando anteriormente, varia para os casos em que o ponto é coincidente da malha grossa ou não. Para o primeiro caso, C é dado pela Extrapolação de Richardson e caso contrário neste trabalho são empregadas algumas variações além da clássica, exemplificadas no algoritmo a seguir, para Diferenças Finitas $1D$.

Seja $g = 1,2,3,\cdots,G$ malhas, em que para $g = 1$ possui $N + 1$ pontos, as malhas seguintes são originadas do seu refino uniforme de razão dois. Portanto, para $g = 2,\cdots,G$ malhas com $i = 0, 2, 4,6,...,2^{g-1}N$ pontos, adaptado de Roache e Knupp (1993):

$$\phi_{g,i}^{m} = \phi_{g,i}^{m-1} + C_{g,i}^{m} \tag{3.1}$$

em que

$$C_{g,i}^{m} = \frac{\phi_{g,i}^{m-1} - \phi_{g-1,i/2}^{m-1}}{2^{p_0} - 1}. \tag{3.2}$$

Para os restantes dos nós ($i = 0, 2, 4, 6, ..,2^{g-1}N - 2$):

$$\phi_{g,i+1}^{m} = \phi_{g,i+1}^{m-1} + C_{g,i+1}^{m} \tag{3.3}$$

onde

- Clássico (**CRE**), adaptado de Roache e Knupp (1993):

$$C_{g,i+1}^{m} = \frac{C_{g,i}^{m} + C_{g,i+2}^{m}}{2} \tag{3.4}$$

- Interpolado (**CRE-I**): Se possível, são interpoladas as correções obtidas em até $p+1$ nós coincidentes em torno do nó $i+1$, obtendo um polinômio φ_p de grau máximo p e

$$C^m_{g,i+1} = \varphi_p(x_{i+1}). \tag{3.5}$$

  Caso contrário, utiliza-se a técnica anterior (Clássico).

- Extrapolação de Richardson Total (**FRE**), adaptado de Giacomini (2013):

$$C^m_{g,i+1} = C^m_{g,i} + k(C^m_{g,i+2} - C^m_{g,i}) \tag{3.6}$$

  onde k é o fator de ponderação dado pela equação

$$k = \frac{\phi^{m-1}_{g,i+1} - \phi^{m-1}_{g,i}}{\phi^{m-1}_{g,i+2} - \phi^{m-1}_{g,i}}. \tag{3.7}$$

Através da Figura 14 é possível verificar os procedimentos listados anteriormente. Neste exemplo, utiliza-se as malhas **M2** e **M3** (Figura 14a) com $N = 3$ da malha **M1** e $G = 4$ que corresponde ao total de malhas disponíveis. Nesta figura são listados os pontos x_p e suas soluções numéricas associadas $\phi^m_{g,p}$, lembrando que g representa o número da malha, p o número de nó e m é o nível de extrapolação.

Para obter a malha **M3** extrapolada (Figura 14b), é necessário aplicar CRE através das equações (3.1) e (3.2), que correspondem aos pontos coincidentes das malhas **M2** e **M3**, destacados com a mesma cor (Figura 14a). Os restantes dos pontos não coincidentes (pontos em rosa), utiliza-se a Equação (3.3), podendo variar o termo de correção C: versão clássica (Equação (3.4)), interpolada (Equação (3.5)) ou FRE (Equação (3.6)).

No caso 2D a ideia é similar, já exposta na Seção 2.5.2, com exceção da interpolação que exige mais atenção, devido ao seu alto custo computacional, foram utilizadas interpolações uni e bidimensionais, cuja a forma de utilização pode ser vista na Seção 3.5. No apêndice B, o Algoritmo 14 aborda o caso unidimensional.

Note que é possível continuar o algoritmo com as malhas extrapoladas, porém a cada extrapolação m perde-se a malha mais grossa, conforme é mostrado na Tabela 5, para as quatro malhas usadas como exemplo. Utiliza-se a notação $M_{g,m}$ em que g corresponde ao número da malha e m ao nível de extrapolação da malha.

Pode ser observado que a ordem máxima teórica é p_3 para o exemplo utilizando quatro malhas **M1**, **M2**, **M3** e **M4**, isto é, teoricamente a malha mais fina pode **aumentar** $G-1$ unidades a acurácia da ordem assintótica do método, onde G é a quantidade de malhas disponíveis. Por exemplo, na Tabela 5, tem-se $G = 4$, nota-se que a acurácia da solução da malha 4 em $m = 0$ é p_0, mas em $m = G - 1 = 3$ é p_3, portanto ela "caminhou"

três unidades em sua ordem de acurácia para $m = 3$, comparando com a mesma malha sem nenhuma extrapolação ($m = 0$). Generalizando este resultado teórico, obtém-se a Tabela 6 com ordem de acurácia máxima teórica p_{G-1}. Estas técnicas trabalham com variáveis do Tipo I e II de MER, abordados na Seção 2.5.1.

Nas próximas seções são apresentadas técnicas que a priori seguem caminho inverso, com exceção da técnica SMER-I. Aplicando SER e SMER a cada extrapolação m perde-se a malha mais fina, ao contrário da apresentada anteriormente. Este resultado foi obtido a posteriori e seguiram o comportamento apresentado na Tabela 7 e serão mais discutidos nas próximas seções e capítulos.

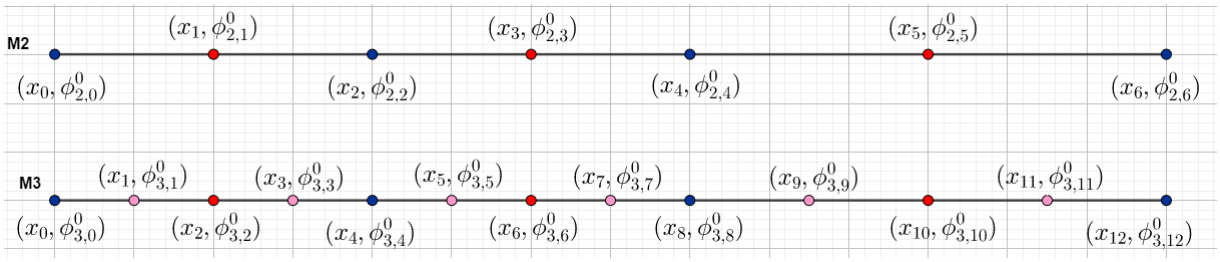
Tabela 5 – Esquema de obtenção de resultados com CRE e algumas variações, para $G = 4$ malhas com $m = 3$ extrapolações e ordem de acurácia máxima p_3, onde $0 < p_0 < p_1 < p_2 < p_3$; $p_i \in \mathbb{Z}, i = 0, 1, 2, 3$

p_v	p_0	p_1	p_2	p_3
$\downarrow g \ / \ \overrightarrow{m}$	$m = 0$	$m = 1$	$m = 2$	$m = 3$
1	$M_{1,0}$			
2	$M_{2,0}$	$M_{2,1}$		
3	$M_{3,0}$	$M_{3,1}$	$M_{3,2}$	
4	$M_{4,0}$	$M_{4,1}$	$M_{4,2}$	$M_{4,3}$

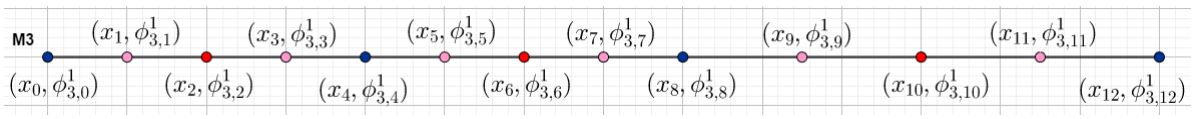
FONTE: Autor(2020).

Figura 14 – Exemplo de Aplicação de CRE e algumas variações

(a) Malhas **M2** e **M3**, $m = 0$



(b) Malha **M3**, $m = 1$



FONTE: Autor(2020).

Tabela 6 – Esquema de obtenção de resultados com **CRE** e algumas variações, para G malhas com $m = G-1$ extrapolações, e ordem de acurácia máxima p_{G-1}, onde $0 < p_0 < p_1 < p_2 < \cdots < p_{G-1};\ p_i \in \mathbb{Z}, i = 0, 1, \cdots G-1$.

p_v	p_0	p_1	p_2	p_3	$\cdots$	p_{G-1}
$\downarrow g\ /\ \overrightarrow{m}$	$m=0$	$m=1$	$m=2$	$m=3$	$\cdots$	$m=G-1$
1	$M_{1,0}$					
2	$M_{2,0}$	$M_{2,1}$				
3	$M_{3,0}$	$M_{3,1}$	$M_{3,2}$			
4	$M_{4,0}$	$M_{4,1}$	$M_{4,2}$	$M_{4,3}$		
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	
G	$M_{G,0}$	$M_{G,1}$	$M_{G,2}$	$M_{G,3}$	$\cdots$	$M_{G,G-1}$

FONTE: Autor(2020).

Tabela 7 – Esquema de obtenção de resultados com **SER** e **SMER**, para G malhas, $m = G-1$ extrapolações com ordem de acurácia máxima p_{G-1}, em que $0 < p_0 < p_1 < p_2 < \cdots < p_G;\ p_i \in \mathbb{Z}, i = 0, 1, \cdots G-1$

p_v	p_0	p_1	p_2	p_3	$\cdots$	p_{G-1}
$\downarrow g\ /\ \overrightarrow{m}$	$m=0$	$m=1$	$m=2$	$m=3$	$\cdots$	$m=G-1$
1	$M_{1,0}$	$M_{1,1}$	$M_{1,2}$	$M_{1,3}$	$\cdots$	$M_{1,G-1}$
2	$M_{2,0}$	$M_{2,1}$	$M_{2,2}$	$\vdots$	$\iddots$	
3	$M_{3,0}$	$M_{3,1}$	$M_{3,2}$	$\vdots$	$\iddots$	
4	$M_{4,0}$	$\vdots$	$\vdots$	$\vdots$	$\iddots$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$M_{G-3,3}$		
$G-2$	$M_{G-2,0}$	$M_{G-2,1}$	$M_{G-2,2}$			
$G-1$	$M_{G-1,0}$	$M_{G-1,1}$				
G	$M_{G,0}$					

FONTE: Autor(2020).

3.2 SMER

Esta técnica tem como objetivo realizar extrapolações em apenas uma única malha, usando todos os nós coincidentes disponíveis nas malhas posteriores a ela, originadas do seu refino. Por exemplo, para aplicar a técnica na malha **M1** separa-se apenas os nós coincidentes e suas respectivas soluções numéricas nas malhas posteriores **M2**, **M3** e **M4**, conforme é apresentado na Figura 15a. Este seria o passo 1 (Figura 15). Repete-se o processo para as malhas **M2** e **M3**, representados pelas figuras 15b e 15c respectivamente.

Feito isso, aplica-se a Multiextrapolação de Richardson (MER) nos pontos da malha de cada caso: Malha **M1**, Malha **M2** e Malha **M3**. Para obter MER nos pontos. Nestes experimentos utiliza-se o Algoritmo 4, para obter as variáveis

$$\sigma_i,\ \texttt{para}\ i = 1,2,\cdots G\ \texttt{malhas}, \tag{3.8}$$

de cada ponto. Cada σ_i corresponde a um nível m de extrapolação de MER, como pode ser visto na Tabela 8 do ponto extrapolado. Teoricamente, quanto maior o índice i, mais acurada a solução do ponto, porque possui uma ordem verdadeira p_v superior a anterior, conforme mostra a Tabela 8.

Por exemplo, para a malha **M1**, após aplicado o Algoritmo 4 em cada ponto, obtém-se σ_i, γ_i, β_i e ξ_i, que representam as extrapolações obtidas nos quatro pontos da malha. Por fim, cada malha possui uma variável σ_i, γ_i, β_i e ξ_i, conforme a sua ordem. Por exemplo, a malha **M1** possui os pontos σ_1, γ_1, β_1 e ξ_1 e assim por diante, conforme mostra a Figura 16. Para efeito de estudo, basta calcular as variáveis de interesse em cada malha com seus nós extrapolados (Figura 16) e não extrapolados (Figura 15a) e compará-las.

Note que o caso 2D é análogo. No Apêndice B os algoritmos 12 e 13 trabalham os dois casos para diferenças finitas. Perceba que nesta técnica é possível variar a razão de refino r quaisquer, desde que sejam inteiros positivos maiores que um. Esta técnica é uma aplicação direta de variáveis do Tipo I de MER, abordados na Seção 2.5.1.

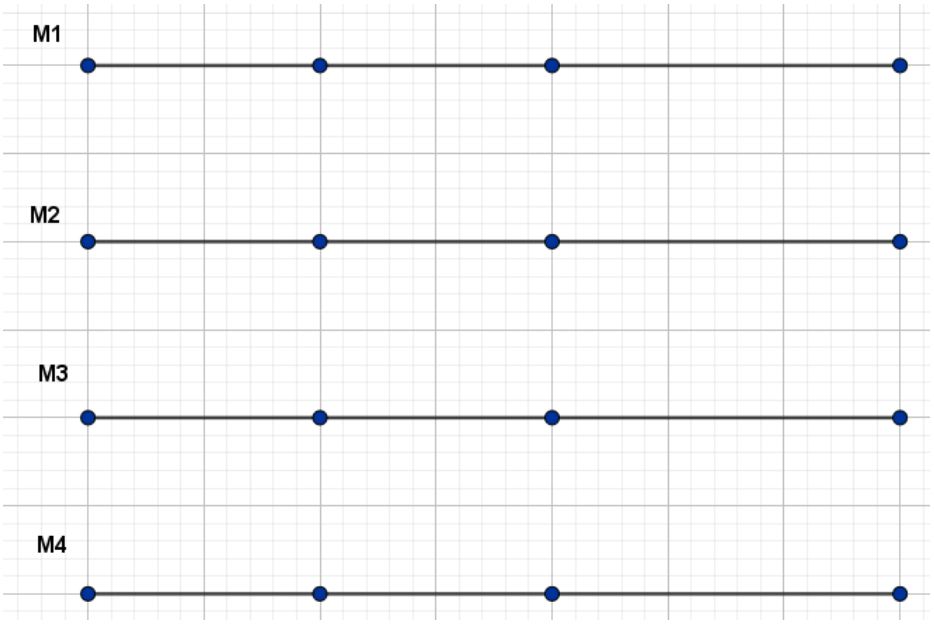
Tabela 8 – Esquema de obtenção de resultados com SMER, variável σ

p_v	p_0	p_1	p_2	p_3	$\cdots$	p_{G-1}
$\downarrow g \ / \ \overrightarrow{m}$	$m=0$	$m=1$	$m=2$	$m=3$	$\cdots$	$m=G-1$
1	$\sigma_1 = \phi_{1,0}$					
2	$\phi_{2,0}$	$\sigma_2 = \phi_{2,1}$				
3	$\phi_{3,0}$	$\phi_{3,1}$	$\sigma_3 = \phi_{3,2}$			
4	$\phi_{4,0}$	$\phi_{4,1}$	$\phi_{4,2}$	$\sigma_4 = \phi_{4,3}$		
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	
G	$\phi_{G,0}$	$\phi_{G,1}$	$\phi_{G,2}$	$\phi_{G,3}$	$\cdots$	$\sigma_G = \phi_{G,G-1}$

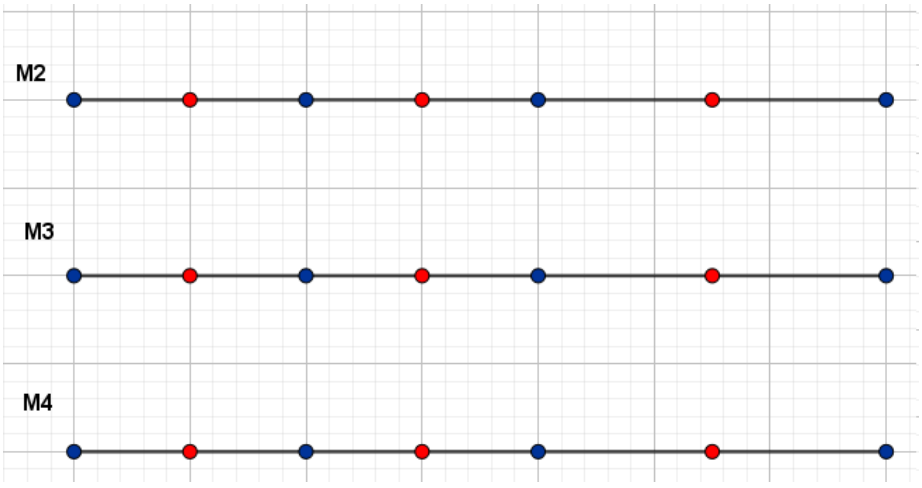
FONTE: Autor(2020).

Figura 15 – Passo 1, SMER 1D

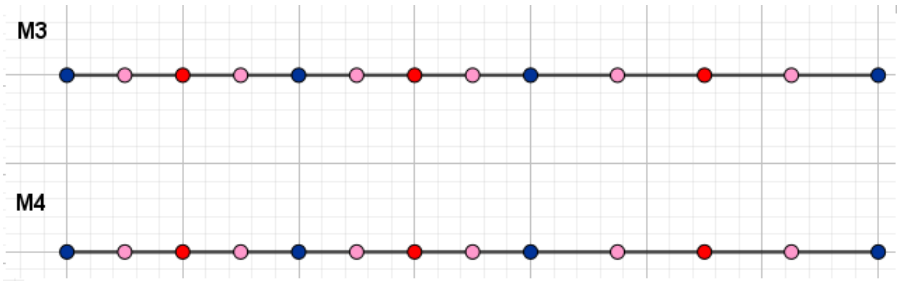
(a) Malha **M1**



(b) Malha **M2**



(c) Malha **M3**



FONTE: Autor(2020).

Algoritmo 4: Algoritmo base - SMER.

Entrada: Obter soluções numéricas para a variável de interesse ϕ em G malhas distintas: ϕ_1, ϕ_2, $\cdots$, ϕ_G.
Fornecer o valor de r, p_0 e dv (razão de refino entre as malhas, ordem de acurácia das soluções numéricas e variação das ordens verdadeiras p_0,p_1,p_2,.., respectivamente).

Saída: σ_i, para $i = 1, 2, \cdots, G$

1 Faça: $\phi_{1,0} = \phi_1, \phi_{2,0} = \phi_2, \cdots, \phi_{G,0} = \phi_G$

para $m = 1, \cdots, G-1$ **faça**

2 $\quad p_{m-1} = p_0 + (m-1)dv$

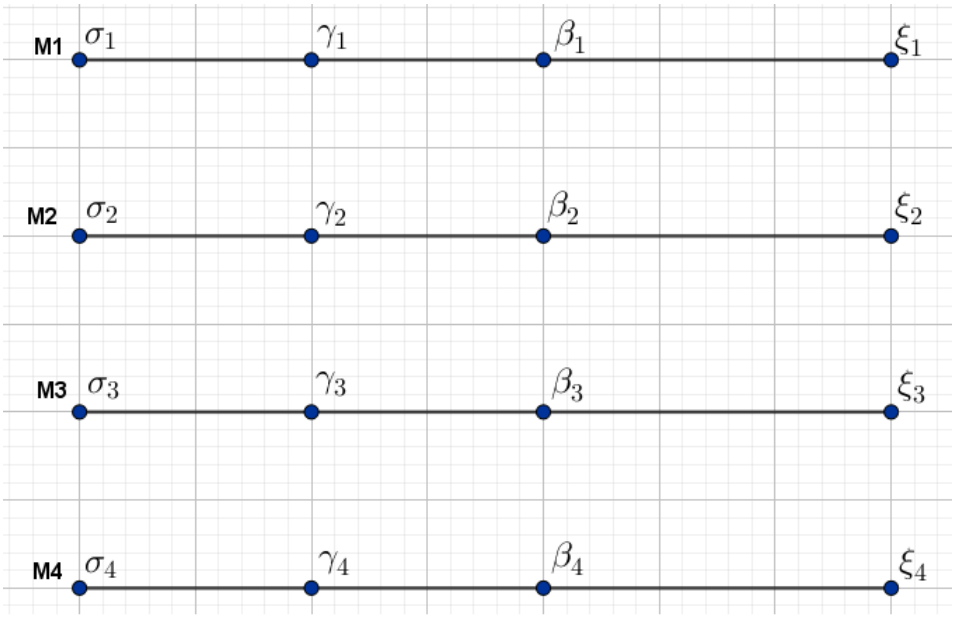
$\quad$ **para** $g = m+1, m+2, \cdots, G$ **faça**

3 $\quad\quad \phi_{g,m} = \phi_{g,m-1} + \dfrac{\phi_{g,m-1} - \phi_{g-1,m-1}}{r^{p_{m-1}} - 1}$

4 **para** $it = 1,2, \cdots, G$ **faça**

5 $\quad \sigma_{it} = \phi_{it,it-1}$

Figura 16 – SMER: Configuração final da malha **M1** com seus pontos e coincidentes nas malhas **M2**, **M3** e **M4** já extrapolados



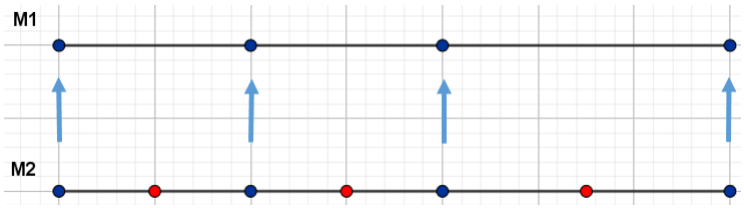
FONTE: Autor(2020).

3.3 SER

Esta técnica, ao contrário da SMER procura realizar extrapolações com todas as malhas simultaneamente. Sua ideia base consiste em utilizar os nós coincidentes com suas respectivas soluções numéricas da malha posterior a que está sendo trabalhada, para melhorar os seus nós, substituindo eles por suas versões extrapoladas, através da extrapolação de Richardson (ER). Por exemplo, para a malha **M1** utiliza-se os nós coincidentes da malha **M2** e suas respectivas soluções numéricas; **M2** utiliza-se **M3**; **M3** utiliza-se **M4** e pôr fim a malha **M4** por não possuir malha posterior, aplica-se CRE clássico ou alguma outra variação.

Para efeito de ilustração da técnica, para extrapolar a malha **M1** aplica-se a extrapolação de Richardson (ER) em todos os seus nós, utilizando apenas os nós coincidentes da malha **M2**. Basicamente os pontos coincidentes da malha mais fina são "injetados" em seus pontos equivalentes na malha mais grossa, através de ER, conforme é ilustrado na Figura 17;

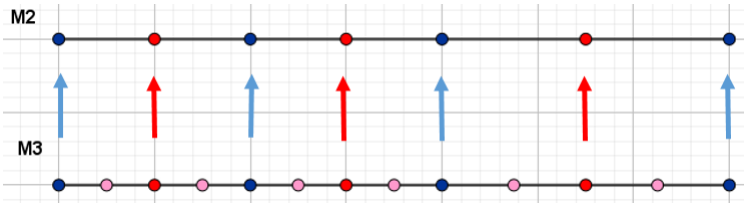
Figura 17 – SER, Malha **M1**



FONTE: Autor(2020).

- Para **M2** aplica-se ER em todos os seus nós, utilizando apenas os nós coincidentes da malha **M3** (Figura 18);

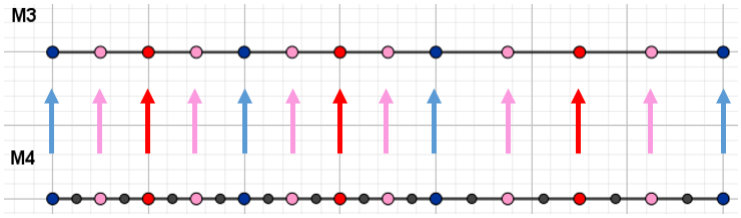
Figura 18 – SER, Malha **M2**



FONTE: Autor(2020).

- **M3** aplica-se ER em todos os seus nós, utilizando apenas os nós coincidentes da malha **M4** (Figura 19);

Figura 19 – SER, Malha **M3**



FONTE: Autor(2020).

- **M4** aplica-se CRE clássico ou alguma outra variação.

Obtidas as novas malhas, repete-se o processo. Nota-se que as novas soluções obtidas teoricamente possuem ordem de acurácia superior em comparação com as anteriores. A metodologia para o caso 2D é análoga ao caso 1D. O algoritmo 5 apresenta o caso 1D. No Apêndice B, o Algoritmo 11 trabalha o caso 2D. Esta técnica é uma aplicação direta de variáveis do Tipo I e II de MER, abordados na Seção 2.5.1.

3.4 SMER INTERPOLADO

Versão estendida de SMER, sua ideia base consiste em determinar um conjunto β de pontos iniciais. Naturalmente eles devem pertencer ao domínio do problema, não é necessário escolher somente pontos com solução numérica associada em todas as malhas.

Selecionado o conjunto β, em cada malha disponível é separada a solução numérica para cada ponto determinado. Caso exista algum ponto que não possua solução numérica associada, ele é interpolado com o grau máximo possível utilizando apenas as soluções numéricas disponíveis na malha. Deste modo, obtém-se o Passo 1 da técnica SMER, para o conjunto β.

Este raciocínio é ilustrado na Figura 20, onde são apresentados dois exemplos de aplicação. O primeiro apresentado pela Figura 20a, β são os pontos da malha **M3**, em "$\times$" correspondem as soluções numéricas obtidas com interpolação em cada malha. Para efeito de ilustração da técnica, os pontos já foram separados para aplicar SMER, correspondendo ao primeiro passo desta técnica.

Através de SMER-I é possível escolher pontos que não possuem solução numérica associada, este caso é ilustrado pela Figura 20b, em que β corresponde apenas por pontos "$\times$" e não possuem solução numérica associada em nenhuma das malhas disponíveis. As soluções numéricas utilizadas para β são obtidas somente através de interpolações, utilizando apenas as soluções numéricas disponíveis em cada malha; deste modo, é possível aplicar o primeiro passo de SMER para este conjunto β.

Claramente, observa-se a vantagem de utilizar um número maior de malhas. Outro benefício é a possibilidade de escolher um conjunto β com pontos que não possuem solução numérica em nenhuma malha ou somente em uma, abrindo-se então a possibilidade de aplicar esta técnica para outros métodos numéricos além de Diferenças Finitas, por exemplo: Volumes Finitos e Elementos de Finitos.

No Algoritmo 6 é apresentado o resumo da técnica. Observe que é possível utilizar malhas com soluções já extrapoladas. Na próxima seção são apresentadas as estratégias para realizar as interpolações. Esta técnica é uma aplicação direta de variáveis do Tipo I, II e III de MER, abordadas na Seção 2.5.1.

Algoritmo 5: Algoritmo SER 1D, para diferenças finitas.

Entrada: Obter G soluções numéricas distintas: $\phi^0_{1,p}$ com nós $p = 0,1,\cdots,N_{inicial}$; $\phi^0_{2,p}$ com nós $p = 0,1,2,\cdots,2N_{inicial}$; $\cdots$; $\phi^0_{G,p}$ com nós $p = 0,1,2,\cdots,2^{G-1}N_{inicial}$.
Fornecer o valor de p_0 (ordem de acurácia das soluções numéricas).

Saída: $\phi^1_{g,p}$, para $g = 1,2,3,\cdots,G$ e $p = 0,1,2,\cdots,2^{g-1}N_{inicial}$.

para $g = 1,2,3,\cdots,G-1$ **faça**
  $N = 2^g N_{inicial}$
  para $i = 0,2,4,6,..,N$ **faça**
    $C^1_{g,i} = \dfrac{\phi^0_{g+1,i} - \phi^0_{g,i/2}}{2^{p_0} - 1}$
    $\phi^1_{g,i/2} = \phi^0_{g+1,i} + C^1_{g,i}$
para $g = G$ **faça**
  $N = 2^{g-1} N_{inicial}$
  para $i = 0,2,4,6,..,N$ **faça**
    $C^1_{g,i} = \dfrac{\phi^0_{g,i} - \phi^0_{g-1,i/2}}{2^{p_0} - 1}$
    $\phi^1_{g,i} = \phi^0_{g,i} + C^1_{g,i}$
  para $i = 0,2,4,6,..,N-2$ **faça**
    $C^1_{g,i+1} = \dfrac{C^1_{g,i} + C^1_{g,i+2}}{2}$
    $\phi^1_{g,i+1} = \phi^0_{g,i+1} + C^1_{g,i+1}$

Algoritmo 6: Algoritmo base SMER Interpolado.

1 Determine o conjunto finito de pontos $\beta \subset \mathbb{R}^n$ que deseja aplicar a técnica, ele naturalmente deve estar no domínio do problema;

2 Em cada malha disponível, utilizando apenas as soluções numéricas $\phi^m_{g,p}$ disponíveis com seus pontos associados $x \in \mathbb{R}^n$, interpole com o grau máximo possível os pontos do conjunto β que não tenham solução numérica associada;

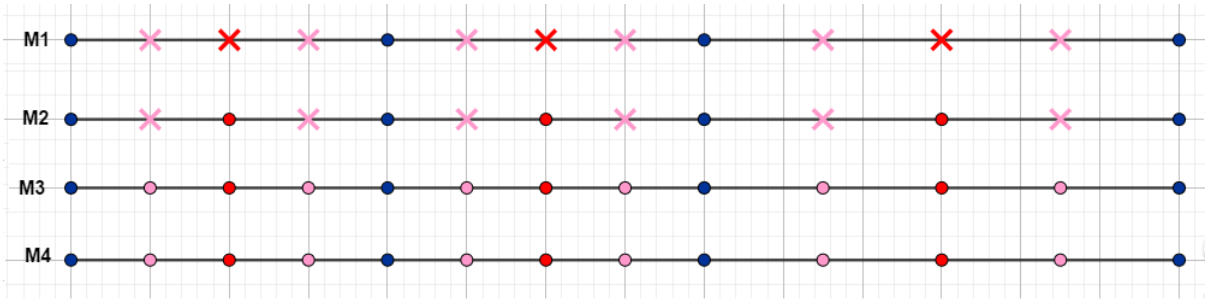
3 Para todas as malhas disponíveis, calcule em cada malha as variáveis de interesse no conjunto β, utilizando as soluções numéricas associadas de cada ponto, obtidas com ou sem interpolação ;

4 Aplique SMER no conjunto β utilizando todas as malhas disponíveis;

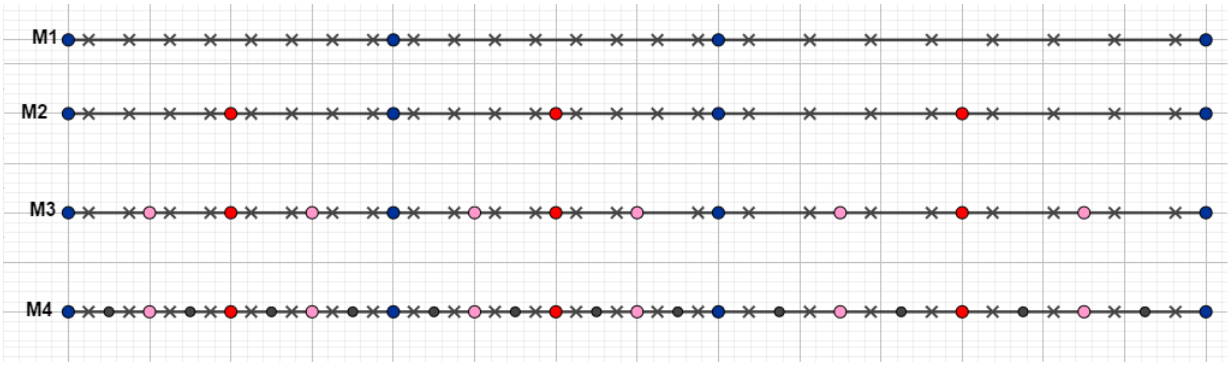
5 Calcule as variáveis de interesse no conjunto β em cada malha disponível e compare com os resultados obtidos no item 3.

Figura 20 – Exemplo de aplicações: SMER Interpolado

(a) SMER Interpolado, Passo 1: SMER, β= "Malha M3". Os pontos representados por $\times$ foram obtidos com interpolações, os restantes já possuíam soluções numéricas associadas. Para obter as soluções com interpolação foram utilizados somente as soluções numéricas disponíveis em cada malha.



(b) Conjunto β, composto somente por pontos não coincidentes, representados por $\times$. Neste caso as soluções numéricas dos pontos foram obtidas com interpolações, utilizando somente as soluções numéricas disponíveis em cada malha. Agora é possível aplicar o passo 1 de SMER, para β.



FONTE: Autor(2020).

3.5 INTERPOLAÇÕES

Nesta seção é apresentada a metodologia utilizada para a escolha dos pontos para realizar as interpolações de SMER interpolado e CRE com interpolação. As técnicas utilizadas para a obtenção dos polinômios foram Diferenças Divididas de Newton (DDN) 1D e Lagrange 2D já comentadas na Seção 2.3.

3.5.1 SMER INTERPOLADO

Naturalmente o primeiro passo deve ser a escolha do ponto base p, que consiste no ponto inicial, cujos pontos vizinhos são utilizados para completar a interpolação desejada. A escolha do ponto base é fundamental para o funcionamento da técnica corretamente. Para cada ponto do conjunto β que necessite da interpolação, trabalha-se com duas situações:

1. Em cada malha disponível, para cada nó k (x_k) do conjunto β, utiliza-se o ponto mais próximo p (x_p) disponível tal que

$$||x_p - x_k|| = \texttt{mínimo}. \tag{3.9}$$

2. Caso a busca demore muito (Caso 2D principalmente), utiliza-se os pontos coincidentes encontrados nas malhas anteriores, isto é, por exemplo: para 10 malhas, realiza-se o passo anterior até a malha 7. As malhas 8, 9 e 10 utilizam os nós p coincidentes determinados na malha 7.

Determinados os pontos, procura-se interpolar com o grau máximo possível, sendo $p_x + 1$ e $p_y + 1$ pontos na direção dos eixos x e y respectivamente, conforme é mostrado nas Figuras 21 e 22. Neste trabalho utiliza-se a relação

$$p_x \leq 10 \texttt{ e } p_y \leq 10. \tag{3.10}$$

Deste modo, o grau máximo possível das interpolações uni e bidimensionais é dez. Pelas características dos pontos buscados nos experimentos sempre haverá no mínimo um vizinho do ponto base em cada direção, conforme é mostrado nas figuras.

Importante destacar que é necessário escolher uma interpolação que possua ordem igual ou superior em relação as soluções numéricas interpoladas, de modo a não prejudicar a acurácia obtida em todos os pontos do domínio. Outra justificativa, tanto SMER-I e CRE-I lembram variáveis do tipo II e III de MER, e neste caso é sugerida a utilização de interpolação linear e com ordem máxima possível, respectivamente. Optou-se pela interpolação de grau máximo 10 pelo fato de ser esperado soluções numéricas com ordens entre 2 e 10.

Para ambos os casos são utilizadas malhas em que os nós não são necessariamente equidistantes em torno do nó p. Por outro lado, em todos os casos as malhas são cartesianas (Figura 23). Neste trabalho não foram abordados por exemplo, o caso do plano transformado, para problemas envolvendo malhas não ortogonais, sugestão para trabalhos futuros.

3.5.2 CRE com interpolação

Neste caso, o ponto base p já é dado, visto que ele é o próprio nó a ser interpolado da malha. Porém, ele não pode ser usado na interpolação porque não possui uma correção C associada.

A primeira estratégia utilizada e mais natural, tanto para o caso unidimensional quanto para o caso bidimensional, consiste em usar a distribuição dos pontos das Figuras 22 e 21 respectivamente. No entanto, todos os nós são somente os que possuem correção C associada, isto é, são os pontos que possuem nós coincidentes na malha anterior a ela, vizinhos do ponto base.

No entanto, para o caso 2D, utilizou-se também outra estratégia, proposta em Moro (2018), onde emprega-se interpolações unidimensionais a partir do ponto base p. O Algoritmo 7 apresenta a sua base.

Algoritmo 7: Algoritmo alternativo de interpolação das correções 2D, adaptado de Moro (2018).

1. Para os pontos bases no eixo x: aplica-se interpolação 1D na direção do eixo x, utilizam-se apenas os nós vizinhos que possuem correção C associada.
2. Para os pontos bases no eixo y: aplica-se interpolação 1D na direção do eixo y, utilizam-se apenas os nós vizinhos que possuem correção C associada.
3. Para os pontos base internos: aplica-se interpolação 1D na direção do eixo x ou y, utilizam-se apenas os nós vizinhos que possuem correção C associada, podendo utilizar os pontos obtidos dos itens anteriores.

Por questões de notação e comparação de resultados, denota-se esta técnica por "Interpolação alternativa". A distribuição dos pontos também segue a lógica apresentada pela Figura 21.

Está técnica alternativa é menos custosa em comparação com a tradicional. Por outro lado, ela utiliza pontos com correções interpoladas que podem estar contaminadas pelo erro de interpolação, sendo aplicada em problemas específicos onde os vizinhos do ponto base deve estar no mesmo eixo, não podendo ser utilizada em malhas não ortogonais, por exemplo.

A técnica clássica, além de ser mais robusta, é possível aplicar em pontos vizinhos,

não precisando estar no mesmo eixo, e pode ser aplicada em malhas não ortogonais. Note que a unicidade da interpolação para o caso 2D não é garantida, quando

$$p_x \neq p_y, \tag{3.11}$$

conforme comentado na Seção 2.3.1, mas em análises a posteriori não foi constatada nenhuma influência nos resultados.

Figura 21 – Distribuição dos pontos para interpolação 2D de grau máximo 10

$\bullet$		$\bullet$	$\bullet$	$\bullet$	$\bullet$
$(p_x - l_w, p_y + l_n)$	$\cdots$	$(p_x - 1, p_y + l_n)$	$(p_x, p_y + l_n)$	$(p_x + 1, p_y + l_n)$	$\cdots$	$(p_x + l_e, p_y + l_n)$
$\vdots$		$\vdots$	$\vdots$	$\vdots$		$\vdots$
$\bullet$		$\bullet$	$\bullet$	$\bullet$		$\bullet$
$(p_x - l_w, p_y + 1)$	$\cdots$	$(p_x - 1, p_y^* + 1)$	$(p_x, p_y + 1)$	$(p_x + 1, p_y + 1)$	$\cdots$	$(p_x + l_e, p_y + 1)$
$\bullet$		$\bullet$	$\bullet$	$\bullet$		$\bullet$
$(p_x - l_w, p_y)$	$\cdots$	$(p_x - 1, p_y)$	(p_x, p_y)	$(p_x + 1, p_y)$	$\cdots$	$(p_x + l_e, p_y)$
$\bullet$		$\bullet$	$\bullet$	$\bullet$		$\bullet$
$(p_x - l_w, p_y - 1)$	$\cdots$	$(p_x - 1, p_y - 1)$	$(p_x, p_y - 1)$	$(p_x + 1, p_y - 1)$	$\cdots$	$(p_x + l_e, p_y - 1)$
$\vdots$		$\vdots$	$\vdots$	$\vdots$		$\vdots$
$\bullet$		$\bullet$	$\bullet$	$\bullet$		$\bullet$
$(p_x - l_w, p_y - l_s)$	$\cdots$	$(p_x - 1, p_y - l_s)$	$(p_x, p_y - l_s)$	$(p_x + 1, p_y - l_s)$	$\cdots$	$(p_x + l_e, p_y - l_s)$

p_x e p_y: coordenadas x e y do ponto base p;
$l_w + l_e \leq 10$; $l_s + l_n \leq 10$; $l_w, l_e, l_s, l_n \in \mathbb{N}$.
Notação: $(i,j) = (x_i, y_j)$.

FONTE: Adaptado de Steffensen (1950).

Figura 22 – Distribuição dos pontos para interpolação 1D de grau máximo 10

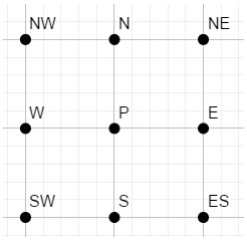
$\bullet$		$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\bullet$	$\bullet$
$p - l_w$	$\cdots$	$p - 2$	$p - 1$	p	$p + 1$	$p + 2$	$\cdots$	$p + l_e$

$l_w + l_e \leq 10; l_w, l_e \in \mathbb{N}$;
p: ponto base;
Notação: $i = x_i$.

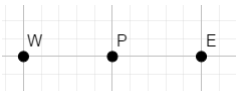
FONTE: Adaptado de Steffensen (1950).

Figura 23 – Exemplos de nós vizinhos para interpolação em torno do nó P, malha cartesiana

(a) Caso 2D



(b) Caso 1D



FONTE: Autor(2020).

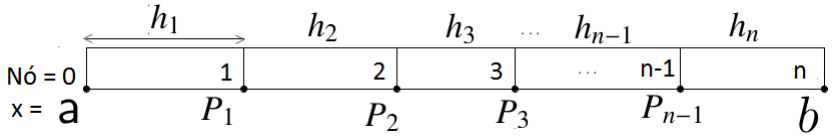
3.6 MODELOS MATEMÁTICOS E NUMÉRICOS

Para os experimentos utiliza-se a técnica de Diferenças Finitas, cuja discretização do intervalo [a,b] é realizada escolhendo-se o vetor h inicial

$$h = (h_1, h_2, ..., h_n)^T, \tag{3.12}$$

da malha discretizada inicial (g), conforme é apresentado na Figura 24.

Figura 24 – Malha discretizada inicial (g), intervalo [a,b]



FONTE: Autor(2020).

Este vetor possui a propriedade:

$$\sum_{i=1}^{n} h_i = |b - a| \quad h_i \in \mathbb{R}^+ - \{0\}; \quad i = 1,...,n. \tag{3.13}$$

Os pontos P_i, para $i = 1,2,..,n-1$, correspondem aos pontos nodais da malha inicial, com exceção dos extremos ($x = a$ e $x = b$). As outras malhas são obtidas a partir da malha inicial, utilizando refino uniforme de razão 2, conforme é mostrado na Figura 25, por exemplo. No caso bidimensional realiza-se o mesmo processo, no entanto são dois vetores h_x e h_y para as direções x e y respectivamente, podendo ser distintos.

Foram geradas de 8 a 15 malhas, variando para cada caso. Os vetores iniciais escolhidos para as discretização uni e bidimensionais são os seguintes:

- Caso 1D:
  - Malha não uniforme

$$\begin{aligned} h_1 &\approx 6.848E-08 \\ h_2 &\approx 4.450E-03 \\ h_3 &\approx 6.157E-02 \\ h_4 &\approx 1.165E-01 \\ h_5 &\approx 1.682E-01 \\ h_6 &\approx 1.464E-01 \\ h_7 &\approx 5.857E-02 \\ h_8 &\approx 1.599E-01 \\ h_9 &\approx 1.390E-01 \\ h_{10} &\approx 1.454E-01 \end{aligned} \tag{3.14}$$

  - Malha uniforme

$$h_i = 0.1 \quad i = 1,2,..,10 \tag{3.15}$$

- Caso 2D:
  - Malha não uniforme (Figura 26b)

$$h_x = \begin{pmatrix} 3.7523448429E-07 \\ 2.4385264497E-02 \\ 3.3736461778E-01 \\ 6.3824974249E-01 \end{pmatrix}, \; h_y = \begin{pmatrix} 3.1554630840E-01 \\ 2.7466614253E-01 \\ 1.0987948516E-01 \\ 2.9990806391E-01 \end{pmatrix} \tag{3.16}$$

  - Malha uniforme (Figura 26a)

$$h_y = h_x = \begin{pmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{pmatrix} \tag{3.17}$$

As escolhas das malhas iniciais foram realizadas para que MER convergisse rapidamente, deste modo foram escolhidos os vetores h com dimensões mínimas possíveis para alcançar este objetivo. Em trabalhos futuros é interessante trabalhar com casos mais específicos, voltados para aplicações físicas.

Outro detalhe importante para ser destacado, foi utilizado a subrotina "random_number" da linguagem Fortran 90 da biblioteca PORTLIB, também utilizada em Carvalho Junior (2015). Sua utilização consiste na obtenção das entradas dos vetores h não uniformes de forma **aleatória**. A escolha da aleatoriedade se deve pelas infinitas

possibilidades e a ideia de não influenciar de forma inconsciente nos resultados. Mas para trabalhos futuros é interessante testar outros casos utilizando valores específicos.

Embora a subrotina "random_number" seja aleatória, para o caso 1D foi obtida uma das malhas empregadas em Carvalho Junior (2015) para Poisson 1D. Porém é interessante esta igualdade para as análises a posteriori, porque este modelo também será abordado nos experimentos.

Trabalha-se com o conjunto β como sendo os pontos obtidos de refinos das malhas iniciais, uniforme e não uniforme, utilizados nas discretizações, em que são excluídos os pontos dos contornos, denotando o conjunto que contém os pontos equidistantes por β_1, caso contrário é assinalado por β_2.

Mais uma alternativa testada, para o caso bidimensional, o conjunto β_3 são pontos gerados a partir da malha inicial, excluindo os contornos, dada pela Figura 26c. A ideia desta alternativa é diferenciar da quantidade de pontos nas direções x e y em relação as malhas iniciais utilizadas anteriormente (figuras 26a e 26b), empregadas nas discretizações dos modelos matemáticos.

Dando continuidade aos dados iniciais, são analisadas as seguintes normas da variável de interesse ϕ:

- Norma l_1 do erro de discretização avaliado em cada nó i (E_i), com exceção dos contornos
$$l_1 = \frac{\sum_{i=1}^{N} |E_i|}{N}; \tag{3.18}$$
- Norma infinito (l_∞) avaliado em cada nó i (E_i), com exceção dos contornos
$$l_\infty = \texttt{max}_{1 \leqslant i \leqslant N} |E_i|, \tag{3.19}$$

e suas ordens de acurácia equivalente (p_E^*)

$$p_E^* = \frac{\log\left[\left|\frac{(\Phi - \phi_1)}{(\Phi - \phi_2)}\right|\right]}{\log(r)}, \tag{3.20}$$

onde

- r: razão de refino;
- Φ: solução analítica exata;
- ϕ_1: solução numérica na malha grossa;
- ϕ_2: solução numérica na malha fina;

- N: quantidade de nós avaliados;
- $|E_i| = |\phi_i - \Phi_i|$, $i = 1,2,\cdots N$.

Por questões visuais e de espaço, em geral fica inviável apresentar um único gráfico contendo as variáveis de interesse l_1 e l_∞ com suas extrapolações. Deste modo, em geral, são apresentados gráficos separados para estas variáveis.

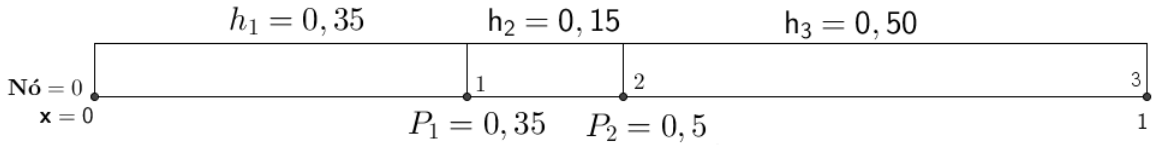
No entanto, salvo algumas exceções, l_1 e l_∞ apresentaram resultados similares e optou-se por apresentar somente os resultados da variável l_1. A variável l_∞ poderá ser vista nos apêndices. Estão publicados todos os resultados e simulações de forma integral neste link:

http://ftp.demec.ufpr.br/CFD/monografias/2020 _Carlos-Carvalho_doutorado/

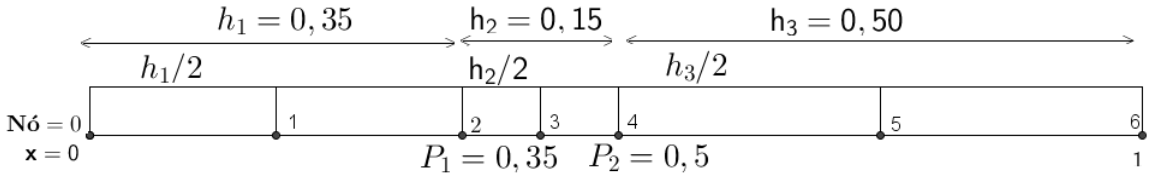
Determinadas as malhas e variáveis de interesse, nas próximas seções são apresentados brevemente os modelos matemáticos utilizados nos experimentos e suas respectivas variáveis ϕ de interesse. Todas as implementações e códigos computacionais utilizados são feitos em Fortran 90 com a precisão máxima disponível (quádrupla), explicada na Seção 2.1.3.

Figura 25 – Exemplo do esquema de refino uniforme com razão 2, aplicado em malhas não uniformes unidimensionais para Diferenças Finitas

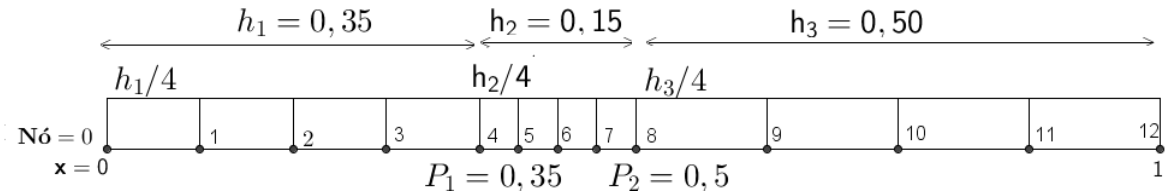
(a) Malha 1 (Malha inicial)



(b) Malha 2



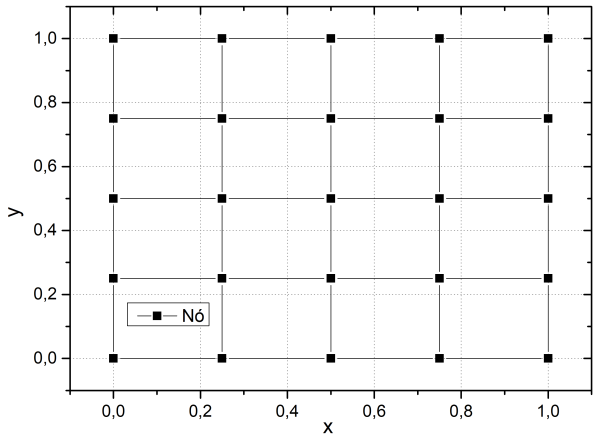
(c) Malha 3



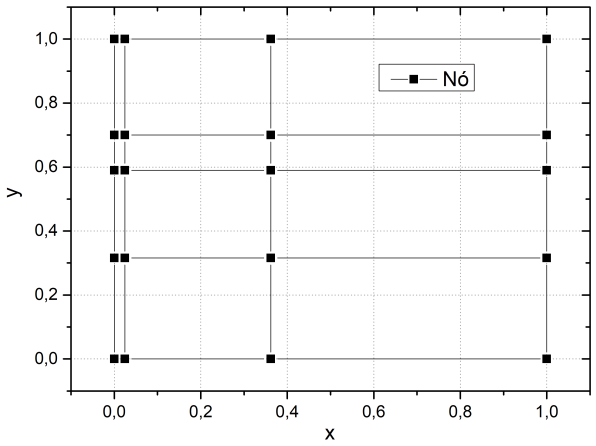
FONTE: Autor(2020).

Figura 26 – Malhas iniciais bidimensionais, utilizadas nos experimentos

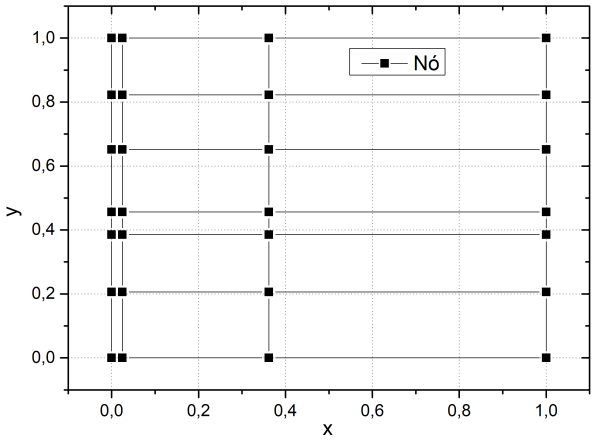
(a) Malha 1, caso uniforme.



(b) Malha 2, caso não uniforme.



(c) Malha 3, caso não uniforme.



FONTE: Autor(2020).

3.6.1 Poisson 1D

A equação de Poisson 1D pode modelar por exemplo, o fenômeno físico de difusão unidimensional com geração de calor é dada pela equação:

$$\frac{d^2T}{dx^2} = S \quad x \in [a,b] \tag{3.21}$$

em que T denota a variável dependente, a coordenada x denota a variável independente no intervalo $[a,b]$ e S denota a conversão de alguma forma de energia em energia térmica. Utilizam-se as condições de contorno de Dirichlet:

$$T(a) = a_1, \quad T(b) = a_2 \quad a_1, a_2 \in \mathbb{R}. \tag{3.22}$$

Neste experimento, consideram-se casos específicos para fins de estudo, adotando

as soluções analíticas T_1 e T_2 no intervalo [0,1]:

$$T_1(x) = \frac{-1}{5}e^{3x} + 3x + 2,\ T_2(x) = \text{sen}(\pi x). \tag{3.23}$$

Portanto, das equações (3.21) e (3.22), obtém-se

$$\begin{cases} \dfrac{d^2T_1}{dx^2} = \dfrac{-9}{5}e^{3x} \quad x \in [0,1] \\ T_1(0) = \dfrac{9}{5}\ ;\ T_1(1) = \dfrac{-e^3}{5} + 5 \end{cases}; \tag{3.24}$$

$$\begin{cases} \dfrac{d^2T_2}{dx^2} = -\pi^2\,\text{sen}(\pi x) \quad x \in [0,1] \\ T_2(0) = T_2(1) = 0 \end{cases}. \tag{3.25}$$

Após feita a discretização do intervalo, empregam-se aproximações do tipo CDS (*Central Difference Scheme*) para as derivadas de segunda ordem e como solver o TDMA. Para mais detalhes veja o Apêndice A.3. As variáveis de interesse são as normas l_1 e l_∞ de todo o domínio do campo, com exceção dos contornos.

3.6.2 Laplace 2D

A equação de Laplace pode modelar problemas físicos como a condução de calor independente do tempo e o escoamento potencial. Neste trabalho, utiliza-se a seguinte variação:

$$\begin{cases} \dfrac{\partial^2T}{\partial x^2} + \dfrac{\partial^2T}{\partial y^2} = 0, x,y \in [0,1] \\ T(x,1) = \text{sen}(\pi x),\ T(0,y) = T(1,y) = T(x,0) = 0 \end{cases} \tag{3.26}$$

em que T representa a temperatura, x e y são coordenadas espaciais.

Para este modelo, a solução analítica é dada pela equação

$$T(x,y) = \text{sen}(\pi x)\frac{senh(\pi y)}{senh(\pi)}. \tag{3.27}$$

A equação foi discretiza utilizando malhas não uniformes e uniformes. Após feita a discretização do intervalo, empregam-se aproximações do tipo CDS (*Central Difference Scheme*), para as derivadas de segunda ordem. Como solver é utilizado o Gauss-Seidel.

Utiliza-se Multigrid, técnica que pode acelerar a convergência do Solver consideravelmente, podendo ser vista com mais detalhes em Briggs, Henson e Mccormick (2000). No entanto, está técnica é utilizada apenas para a versão uniforme.

As simulações deste modelo matemático foram realizadas através de um código cedido pelo grupo de CFD da UFPR para o caso uniforme. A versão não uniforme, discretizações e algoritmos bases podem ser vistos no Apêndice A.1. Para este modelo são analisadas as normas l_1 e l_∞ de todo o domínio do campo, com exceção dos contornos.

3.6.3 Burgers

Este modelo matemático envolve o fenômeno de convecção de quantidade de movimento linear. Foram utilizadas duas variações desta equação: uni e bidimensional.

A equação correspondente ao modelo matemático unidimensional é dada por:

$$T\frac{dT}{dx}Re = \frac{d^2T}{dx^2} + S \tag{3.28}$$

onde Re representa o número de Reynolds, T representa a temperatura (dependente), x é uma coordenada espacial (independente) e S é o termo fonte da equação. Para estes experimentos, utiliza-se a solução analítica

$$T = \frac{e^{xRe} - 1}{e^{Re} - 1} \tag{3.29}$$

com condições de contorno de Dirichlet

$$T(0) = 0,\ T(1) = 1. \tag{3.30}$$

Portanto, das equações (3.28), (3.29) e (3.30) obtém-se o modelo a ser resolvido neste trabalho:

$$\begin{cases} T\dfrac{dT}{dx}Re = \dfrac{d^2T}{dx^2} + \dfrac{Re^2e^{xRe}(e^{xRe} - e^{Re})}{(e^{Re} - 1)^2} & x \in [0,1] \\ T(0) = 0; T(1) = 1 \end{cases}. \tag{3.31}$$

Após feita a discretização do domínio, empregam-se aproximações do tipo CDS para as derivadas de primeira e segunda ordens. Para os termos não lineares é feito uma linearização. Emprega-se o TDMA (*Tridiagonal Matrix Algorithm*) como solver de forma iterativa. Para mais detalhes veja o Apêndice A.2. Para este modelo são analisadas as normas l_1 e l_∞ de todo o domínio do campo, com exceção dos contornos.

Dando continuidade, o modelo bidimensional, que envolve o fenômeno de convecção de quantidade de movimento linear 2D, é representada pela equação

$$u\frac{\partial\phi}{\partial x} + v\frac{\partial\phi}{\partial y} = p_r^\phi + \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} + S^\phi, \tag{3.32}$$

onde u e v são as velocidades nas direções x e y, respectivamente e p_r é a derivada da pressão, na Tabela 9 é identificada a equação a partir da variável de interesse ϕ.

O termo fonte S, solução analítica e campo pressões é dado por Shih, Tan e Hwang (1989). As condições de contorno, para u e v, são dadas respectivamente por

$$u(x,1) = 16(x^4 - 2x^3 + x^2),\ u(0,y) = u(1,y) = u(x,0) = 0 \tag{3.33}$$

e

$$v(0,y) = v(1,y) = v(x,0) = v(x,1) = 0 \quad x, y \in [0,1]. \tag{3.34}$$

Após feita a discretização do intervalo, empregam-se aproximações do tipo CDS para as derivadas de primeira e segunda ordens. Como solver utiliza-se o MSI com Multigrid. As simulações deste modelo matemático foram realizadas através de um código cedido pelo grupo de CFD da UFPR. As variáveis de interesse analisadas nos experimentos são as seguintes: normas l_1 e l_∞ de todo o campo, com exceção dos contornos, para as velocidades u e v.

Tabela 9 – Identificação de variáveis da Equação de Burgers 2D

Equação	ϕ	p_r^ϕ	S^ϕ
Quantidade de movimento linear, eixo x (QMLx)	u	$-\frac{\partial p}{\partial x}$	0
Quantidade de movimento linear, eixo y (QMLy)	v	$-\frac{\partial p}{\partial y}$	$-S$

FONTE: Autor(2020).

3.6.4 Navier-Stokes (Problema de Shih)

Dadas as equações tridimensionais de Navier-Stokes que podem ser vistas com mais detalhes em Pritchard (2010). Considerando as simplificações para o caso bidimensional com viscosidade e densidade unitárias são obtidas as equações:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \tag{3.35}$$

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} - \frac{\partial p}{\partial x} \tag{3.36}$$

$$v\frac{\partial v}{\partial y} + u\frac{\partial v}{\partial x} = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} - \frac{\partial p}{\partial y} - S \tag{3.37}$$

com soluções analíticas

$$u = 8(x^4 - 2x^3 + x^2)(4y^3 - 2y) \tag{3.38}$$

e

$$v = -8(4x^3 - 6x^2 + 2x)(y^4 - y^2) \quad x, y \in [0,1]. \tag{3.39}$$

Este problema é chamado de problema de Shih (SHIH; TAN; HWANG, 1989). Trata-se de um modelo clássico na área de CFD de fluxo laminar dentro de uma cavidade quadrada da

qual a tampa se move com velocidade variável. Em que u e v são as velocidades na direção dos eixos x e y, p é a pressão e S é um termo fonte obtido pela substituição das velocidades u e v nas equações listadas anteriormente. Este problema é amplamente empregado para verificação e validação de códigos que resolvem a equação de Navier-Stokes.

As condições de contorno das velocidades u e v são zero em todos os contornos, exceto no superior para o componente u, cuja equação do contorno para esta região é dada por

$$u(x,1) = 16(x^4 - 2x^3 + x^2)\ x \in [0,1]. \tag{3.40}$$

Após feita a discretização do intervalo, empregam-se aproximações do tipo CDS para as derivadas de segunda ordem. Como solver é utilizado o MSI com Multigrid. As simulações deste modelo matemático foram realizadas através de um código cedido pelo grupo de CFD da UFPR. Neste programa também é utilizado a formulação corrente de vorticidade (ψ,ω), proposta em From (1964). Assim são feitas as seguintes mudanças de variáveis, as velocidades são reescritas em termos da função corrente, isto é

$$u = \frac{\partial \psi}{\partial y}, v = -\frac{\partial \psi}{\partial x}, \tag{3.41}$$

e a vorticidade é definida pela equação

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}. \tag{3.42}$$

As variáveis de interesse analisadas nos experimentos são as seguintes: normas l_1 e l_∞ de todo o campo, para as velocidades u e v, com exceção dos contornos.

O leitor atento, deve ter percebido que este problema e Burgers 2D compartilham da mesma solução analítica e das equações QMLx e QMLy, a equação (3.35) é uma variação da equação da conservação da massa. Ambos também possuem outro fator semelhante: o número de **Reynolds** (Re) para sua resolução. Neste caso, Burgers 1D também é influenciado por Re.

Reynolds (1883) define Re como a razão entre forças inerciais e forças viscosas. É um parâmetro bastante utilizado na engenharia para prever se um fluxo será laminar ou turbulento.

Sem dúvida o número de Reynolds pode ser considerado um dos assuntos mais marcantes da engenharia, quando é estudado o comportamento dos fluidos. Neste trabalho é utilizado $Re = 1$. Para trabalhos futuros é interessante abordar Re com outros valores.

4 RESULTADOS - MALHA UNIFORME 1D

Neste capítulo são apresentados os principais resultados dos problemas unidimensionais, discretizados com malhas uniformes. São expostos apenas os resultados para a norma l_1, gráficos das ordens equivalentes (p_E^*) de l_1 é considerado somente o eixo positivo de y. Resultados adicionais em que é incluído l_∞, podem ser vistos nos apêndices e de forma integral na página:

http://ftp.demec.ufpr.br/CFD/monografias/2020_Carlos-Carvalho_doutorado/

Foram geradas 15 malhas a partir da malha base, com vetor inicial h uniforme de dimensão dez, utilizando refino uniforme de razão dois, exemplificadas na Seção 3.6.

Em relação ao tempo de CPU, por se tratar de modelos relativamente simples, com exceção da equação de Burgers que utiliza um solver iterativo foi necessário menos de um minuto para simular as 15 malhas. Burgers demorou cerca de 4 minutos para concluir o mesmo procedimento, mas é um tempo relativamente baixo também.

Para utilizar as técnicas apresentadas na metodologia, uma condição necessária é que as soluções numéricas estejam sendo afetadas somente pelo erro de discretização, fato que é verificado através de análises das ordens verdadeiras a posteriori. Na figura 27 são apresentados os gráficos para as temperaturas T, envolvendo as soluções numéricas e analíticas.

Verificada as ordens das soluções numéricas, agora é possível aplicar as técnicas de pós-processamento apresentadas no capítulo anterior. Caso não for alcançado a convergência das estimativas a priori, ainda assim seria possível aplicar as técnicas. No entanto, já não é garantido o funcionamento em virtude desta condição básica não ser atingida.

4.1 SMER

Para todos os casos **SMER** apresentou o mesmo desempenho qualitativo, onde foram obtidos os melhores resultados nas malhas iniciais. Confirmando as análises prévias realizadas no capítulo anterior. Porém, foi possível obter redução do erro para todas as malhas em que a técnica foi empregada, mas em menor escala para as malhas mais finas.

Por exemplo, considere os resultados obtidos nas malhas 1 e 2 para Burgers 1D com $Re = 1$, ilustradas pelas figuras 28 e 29. Denota-se por **SM**, de "SUBMALHA", os resultados obtidos sem extrapolação, da malha base escolhida. Cada "$\times$" representa os valores obtidos de l_1 ou sua ordem efetiva equivalente em cada malha, isto é

Malha 1, Malha 2, Malha 3, ... , Malha 15,

para o caso de **SM** aplicado na Malha 1 (Figura 28), estas malhas são representadas por "h", que neste cenário representa a distância média entre os nós da malha. Prosseguindo, na Figura 29 temos a mesma analogia, agora com

Malha 2, Malha 3, Malha 4, ... , Malha 15,

e neste caso, **SM** é aplicado na Malha 2. Todavia, lembrando que **SMER** exclui as malhas antecessoras da "Malha base", isto é, se for a malha base 13, não são utilizadas as malhas 1, 2, 3, .., 12. Isto ocorre porque nenhuma delas possui todos os pontos da Malha 13, sendo está a essência do método, e ao mesmo tempo pode ser apontado como um **defeito**.

Dada esta explicação do significado de **SM**, naturalmente a mesma analogia é feita para **SMER**, representado por "■" nas figuras. Em resumo, **SM** é definido como SUBMALHA sem a Multiextrapolação de Richardson (MER), oposto de **SMER**. Isto explica a **notória** diferença de resultados do erro e ordens de acurácia, observada nas figuras 28 e 29. Portanto, as malhas 1 e 2 para este exemplo, obtiveram resultados significativos na redução do erro e aumento da acurácia de suas ordens.

É importante ressaltar, **SMER** apresenta o comportamento teórico das soluções obtidas com MER, isto é, redução do erro com um concomitante aumento das ordens de acurácia. Este resultado pode ser observado através das figuras mencionadas anteriormente.

Porém, não é preciso muitas malhas para obter resultados similares aos obtidos anteriormente. Por exemplo, para as malhas intermediárias 7 e 8, ilustradas pelas figuras 30 e 31, mas agora para Poisson 1D com solução $T = \text{sen}(\pi x)$.

Além disso, até mesmo as duas últimas malhas possíveis de aplicação 13 e 14, mas agora para Poisson 1D com solução $T = \dfrac{-1}{5}e^{3x} + 3x + 2$, ilustradas pelas tabelas 10 e 11, apresentam resultados significativos. Através destas tabelas é possível verificar as diferenças de **SM** para **SMER**. Nelas são mostradas as malhas usadas, começando pela "Malha Base", tamanho do elemento de malha (h), cálculo de l_1 somente para os pontos da malha base com suas ordens equivalentes (p_E^*).

Observando as tabelas, pode-se notar também que o primeiro valor de l_1 é rigorosamente igual para **SMER** e **SM**. Isto ocorre pelo simples fato da primeira malha de **SMER** corresponder somente por pontos com nenhuma extrapolação ($m = 0$), as seguintes correspondem por $m = 1$,$m = 2$,$\cdots$. Novamente a questão da similaridade das figuras 28, 29 e 30 com as encontradas na literatura para MER, isto é, geralmente quando esta técnica obtém êxito é notado uma redução do erro com concomitante aumento nas ordens de acurácia.

Em geral, SMER obteve resultados significativos até nas malhas mais finas, em que na teoria teria pior desempenho. Na Figura 32 são apresentados somente as curvas

de **SMER** de l_1 em cada Malha (g), para Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$. É possível observar as similaridades nas curvas de l_1, corroborando a tese de que SMER beneficia as malhas mais grosseiras, embora mesmo para g=13 e 14 houve considerável redução do erro. Lembrando que o primeiro valor das curvas correspondem a $m = 0$ da Malha Base g.

De modo igual a MER, técnica que serve de "matéria prima" para este método, SMER possui baixo custo computacional para operar. No entanto, ao contrário de MER ela precisa de alguma forma guardar grandes quantidades de soluções numéricas de cada malha para funcionar. Talvez possa acarretar em problemas de memória, para malhas demasiadamente grandes, principalmente no caso 2D. Porém, nos experimentos realizados até no caso bidimensional, não ocorreu.

Infelizmente, outro defeito desta técnica é não conseguir extrapolar a malha mais fina disponível. Nestes experimentos envolvendo malhas unidimensionais seria a malha 15. No entanto, através da sua versão estendida **SMER INTERPOLADO** (SMER-I), esse problema pode ser contornado, através da utilização de interpolações.

Tabela 10 – Resultados: SMER, Malha base 13, Variável l_1, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme

Malha	h	l_1: SM	l_1: SMER	p_E^*: l_1, SM	p_E^*: l_1, SMER
13	2.44E-05	3.73813E-10	3.73813E-10		
14	1.22E-05	9.34532E-11	2.50661E-20	2.00000E+00	3.37959E+01
15	6.10E-06	2.33633E-11	1.47319E-28	2.00000E+00	2.73422E+01

FONTE: Autor(2020).

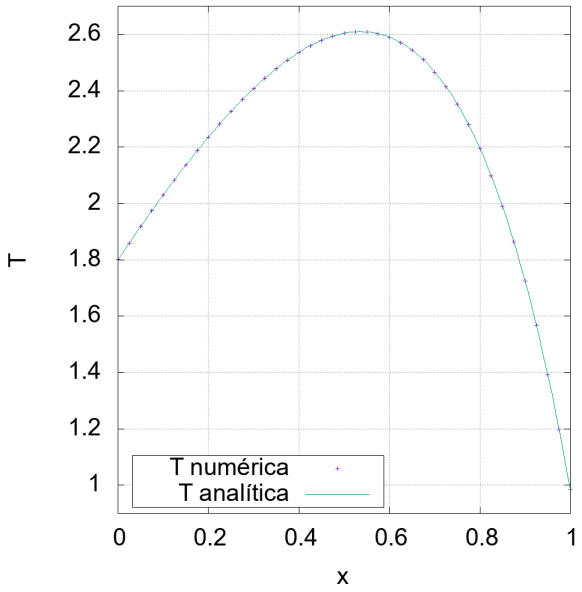
Tabela 11 – Resultados: SMER, Malha base 14, Variável l_1, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme

Malha	h	l_1: SM	l_1: SMER	p_E^*: l_1, SM	p_E^*: l_1, SMER
14	1.22E-05	9.34520E-11	9.34520E-11		
15	6.10E-06	2.33630E-11	1.56661E-21	2.00000E+00	3.57959E+01

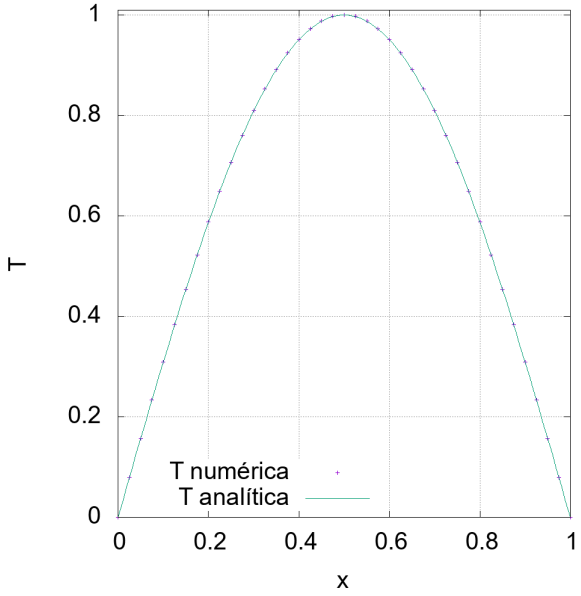
FONTE: Autor(2020).

Figura 27 – $T(x)$: Numérica vs Analítica, malha uniforme

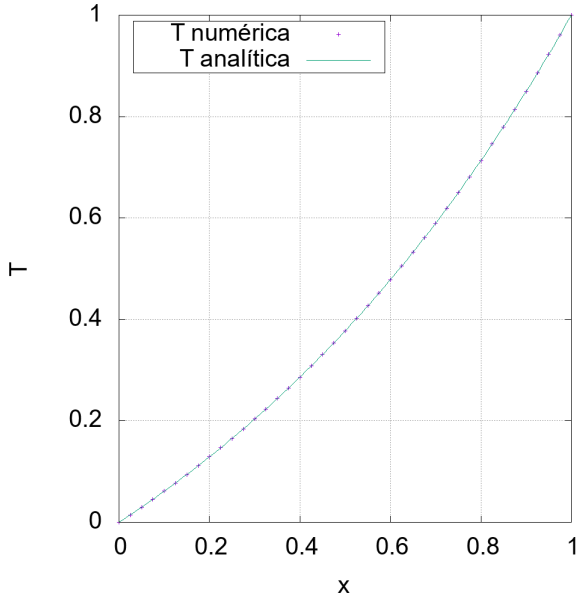
(a) Poisson 1D, $T(x) = \frac{-1}{5}e^{3x} + 3x + 2$.



(b) Poisson 1D, $T(x) = \text{sen}(\pi x)$.



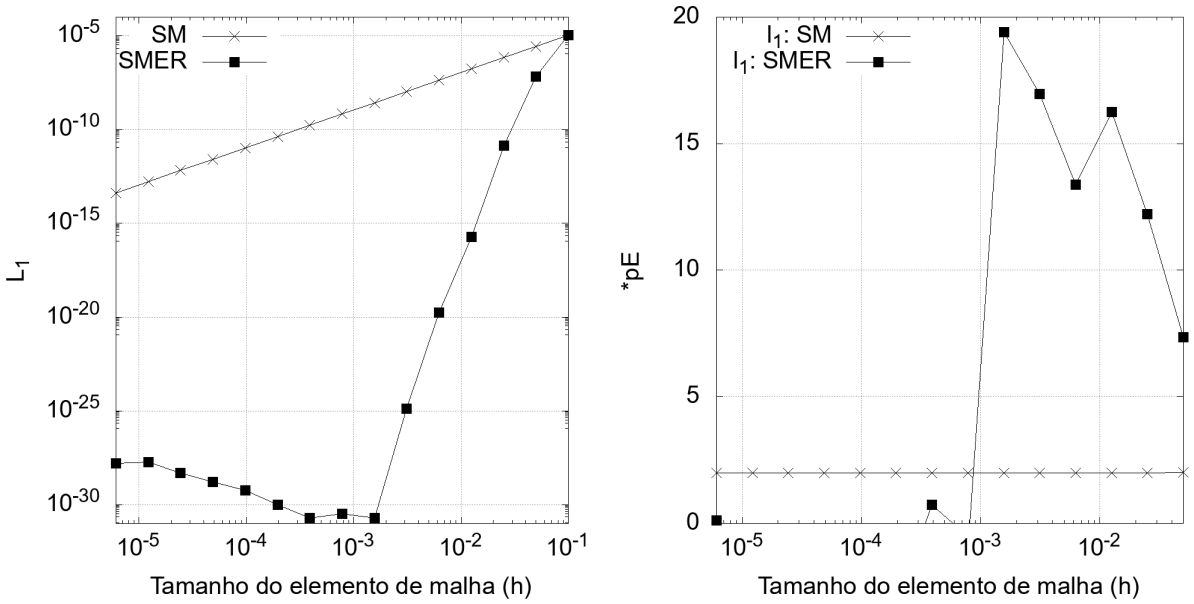
(c) Burgers 1D, $T(x) = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$.



FONTE: Autor(2020).

Figura 28 – Resultados: SMER, Malhas Base 1, Variável l_1, Burgers 1D com solução $T(x) = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha uniforme

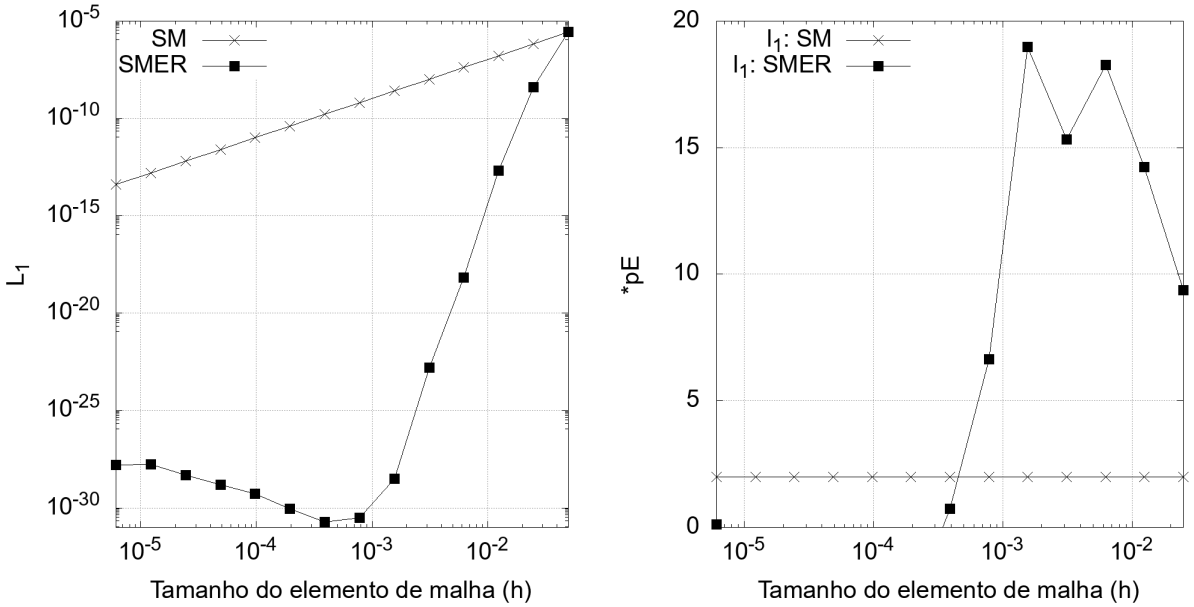
(a) l_1, Malha 1 (b) p_E^*,Malha 1



FONTE: Autor(2020).

Figura 29 – Resultados: SMER, Malha Base 2, Variável l_1, Burgers 1D com solução $T(x) = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha uniforme

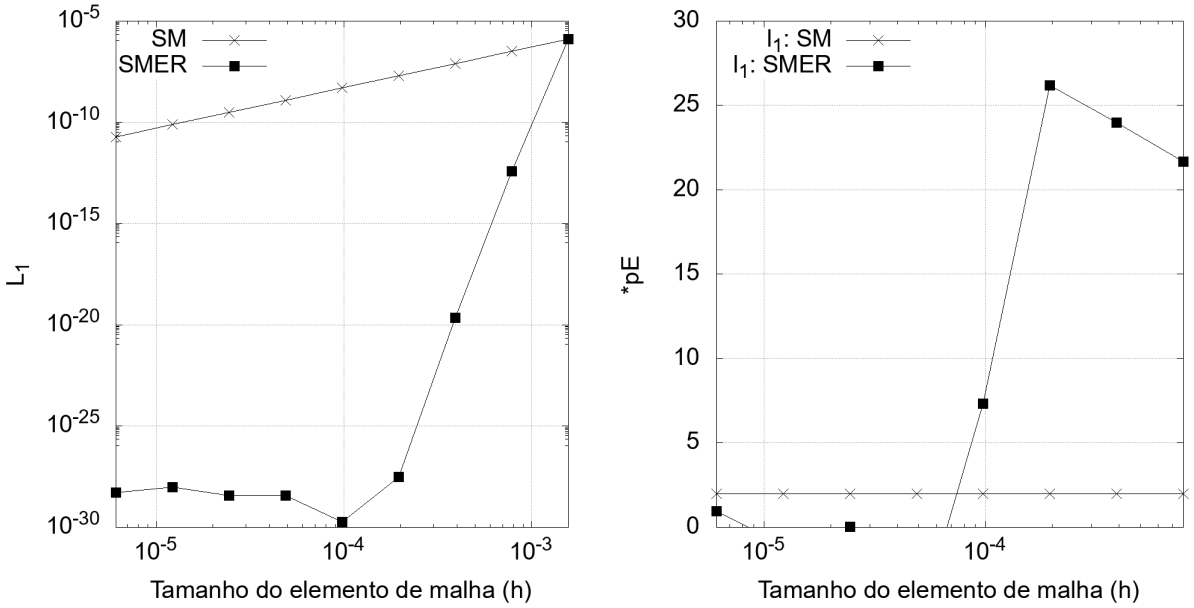
(a) l_1, Malha 2 (b) p_E^*, Malha 2



FONTE: Autor(2020).

Figura 30 – Resultados: SMER, Malhas Base 7, Variável l_1, Poisson 1D com solução $T = \text{sen}(\pi x)$, Malha uniforme

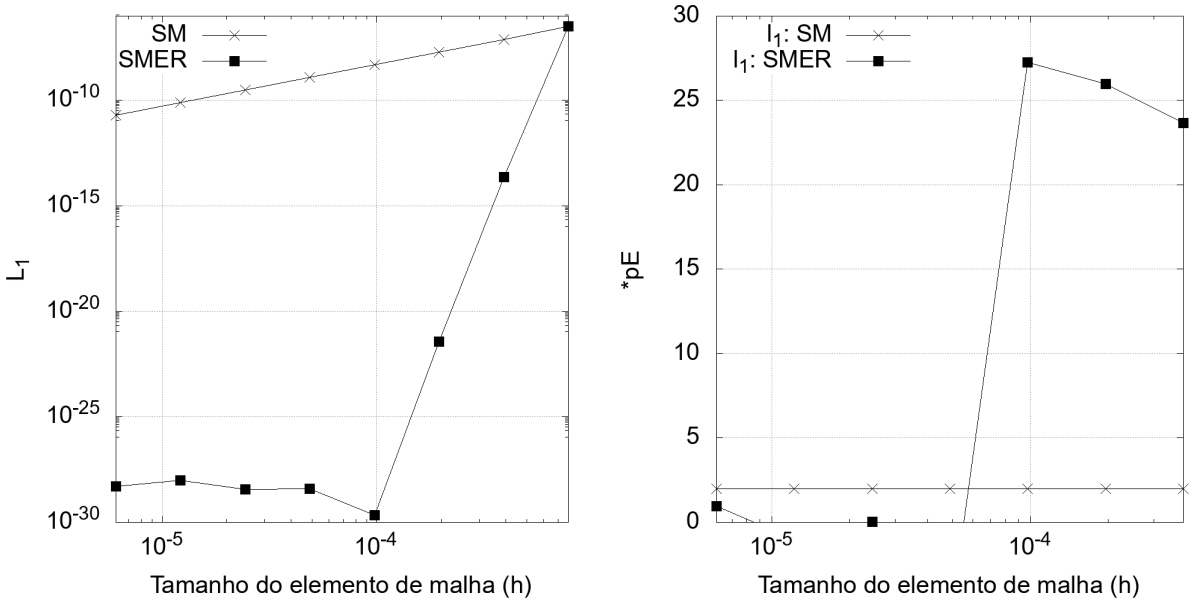
(a) l_1, Malha 7 (b) p_E^*, Malha 7



FONTE: Autor(2020).

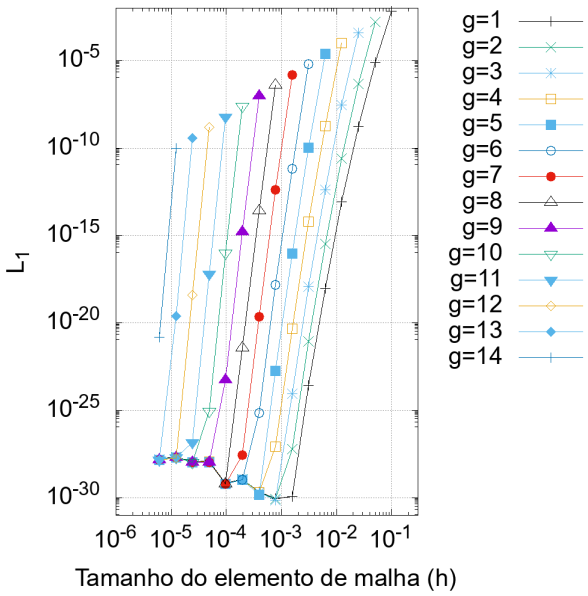
Figura 31 – Resultados: SMER, Malha Base 8, Variável l_1, Poisson 1D com solução $T = \text{sen}(\pi x)$, Malha uniforme

(a) l_1, Malha 8 (b) p_E^*, Malha 8



FONTE: Autor(2020).

Figura 32 – Resultados: SMER, Malhas $g = 1{,}2\cdots{,}14$, Variável l_1, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme



FONTE: Autor(2020).

4.2 SMER INTERPOLADO

Os conjuntos β_1 e β_2 testados correspondem aos pontos equivalentes da malha 15, com exceção dos contornos, para cada caso: uniforme e não uniforme, respectivamente. As malhas foram geradas a partir da malha inicial de cada cenário, utilizando refino de razão dois.

As análises são feitas de forma análoga a realizada na seção anterior, com a diferença nos conjuntos analisados, são denotados por **SM-I** de "SUBMALHA INTERPOLADA", o conjunto β de cada malha (Malha1, Malha2, ...) sem aplicação da técnica **MER**, e **SMER-I** (SMER INTERPOLADO) representa os valores obtidos com **MER** aplicado no conjunto β em cada malha. Lembrando que β possui vários pontos obtidos por interpolação, o que difere dos utilizados na seção anterior.

Feita esta breve explicação dos significados de cada sigla, a Figura 33 apresenta os resultados de **SMER-I** para o conjunto β_1 para Poisson 1D com $T = \frac{-1}{5}e^{3x} + 3x + 2$. Os restantes dos modelos matemáticos obtiveram resultados qualitativamente equivalentes, independente do conjunto β escolhido, isto é, foram obtidos redução do erro significativas com aumento concomitante das ordens.

Algumas peculiaridades de **SMER-I**, naturalmente ela leva esse nome por ser um complemento de **SMER**, porque utiliza interpolações e pelo fato de não serem iguais, havendo então a necessidade de utilizar outra notação, podendo acontecer em algum

momento apresentarem resultados diferentes. No entanto, para os modelos unidimensionais com malhas uniformes, isto não ocorreu, ambas as técnicas apresentaram ótimos resultados e qualitativamente semelhantes, porque ambas possuem a Multiextrapolação de Richardson (MER) como base.

Infelizmente, do mesmo modo que **SMER**, esta técnica necessita guardar grandes quantidades de soluções numéricas de cada malha. Além disso, existe outro problema acrescido das interpolações, porque é necessário em geral realizar várias, mas não é conhecida a região que deve ser interpolada de imediato, havendo a necessidade de uma busca pela região onde será realizada a interpolação.

O processo de busca é custoso, embora no caso unidimensional dependendo do tamanho da malha a busca é realizada em alguns minutos ou até mesmo segundos. Mas, para o caso bidimensional pode levar dias, dependendo do tamanho da malha por este motivo foi acrescido outra forma de encontrar a região interesse, utilizando a ideia de pontos coincidentes, neste caso. Lembrando, as estratégias utilizadas para as interpolações já foram abordadas no capítulo anterior, em particular na Seção 3.5.

Será discutido com mais riqueza de detalhes o problema das "buscas" bidimensionais nos capítulos que apresentam estes resultados, mas para concluir as análises apresentadas nesta seção, **SMER-I** se mostrou uma excelente ferramenta para a redução do erro e o aumento de acurácia dos campos. Deve ser ressaltado que ela consegue otimizar qualquer malha desejada, incluindo a malha mais fina dos experimentos (Malha 15), em que **SMER** não é aplicável, por exemplo.

4.3 CRE clássico com interpolação e FRE

As análises são feitas de forma análoga as realizadas nas seções anteriores, com a diferença que desta vez cada elemento do gráfico corresponde por uma malha, não um conjunto específico de pontos. Conforme h vai diminuindo, significa que se trata das malhas mais finas. Outra alteração corresponde à notação "m", significando a quantidade de extrapolações utilizadas na malha, em que $m = 0$ equivale a zero extrapolação, $m = 1$: uma extrapolação, $m = 2$: duas extrapolações etc.

Feita esta introdução sobre as notações utilizadas, as exposições dos resultados será iniciada por FRE (CRE com FRE). A ideia de alterar o termo de correção C de CRE para FRE surgiu devido aos significativos resultados obtidos nos modelos matemáticos que foram aplicados esta técnica em Giacomini (2013).

Utilizando a solução analítica $T = \text{sen}(\pi x)$ para Poisson 1D foram obtidos resultados excelentes e qualitativos com Giacomini (2013), ilustrados pelas Figuras 34a e 34b. Nota-se a redução significativa do erro e o aumento das ordens de acurácia, conforme aumentam as extrapolações. Porém, para o mesmo modelo, trocando para o caso

$T = \frac{-1}{5}e^{3x} + 3x + 2$, FRE reduz o erro significativamente, mas não apresenta evolução após a primeira extrapolação ($m = 1$). Também não fica claro qual a sua ordem de acurácia, em nenhuma extrapolação ($m \geq 1$), acontecimento que não ocorreu em análises a priori do método. Estes resultados são ilustrados pelas Figuras 34c e 34d.

Para Burgers 1D os resultados com esta técnica foram qualitativos em relação a Giacomini (2013): as ordens de acurácia se mantiveram em quatro, não apresentando evolução na redução do erro após a primeira extrapolação ($m = 1$). No entanto, houve significativa redução do erro, seguindo o comportamento qualitativo com o resultado apresentado na Figura 35, que apresenta os resultados obtidos com o termo de correção clássico (CRE) para Poisson 1D com $T = \frac{-1}{5}e^{3x} + 3x + 2$.

Já a técnica clássica, que consiste na utilização do termo de correção, definido como a média das correções vizinhas, em todos os casos, as ordens de acurácia se mantiveram em quatro, com redução do erro, mesmo extrapolando mais vezes ($m > 1$). Este resultado pode ser visto na Figura 35.

Embora CRE tenha sido desenvolvida para apenas uma extrapolação em Roache e Knupp (1993), era interessante verificar o que aconteceria para $m > 1$ a posteriori, porque são utilizados modelos diferentes do proposto pelo autor. Mas em resumo, ela atendeu às expectativas, conseguindo progredir ao menos uma ordem verdadeira de acurácia, isto é, de $p_0 = 2$, foram obtidas soluções com acurácia $p_1 = 4$ com redução de erro significativa em todos os modelos empregados. Lembrando que foram utilizadas aproximações do tipo CDS em todos os modelos matemáticos.

O leitor atento deve ter notado a menção da mesma ilustração para comentar o resultado de CRE e FRE, a Figura 35. Isto significa que as técnicas alcançaram desempenho semelhante e neste caso, alterar o termo de correção original não surtiu efeito algum. O termo de correção com interpolação (CRE-I) foi a técnica que obteve os melhores resultados em todos os casos. Resultados qualitativos como os apresentados pela Figura 36, onde foram obtidos aumentos significativos das ordens de acurácia e redução do erro.

Em todas as técnicas expostas nesta seção é necessário guardar de alguma forma várias soluções numéricas em cada malha, fator em comum com as técnicas expostas anteriormente. Mas retirando este "empecilho" de memória, que por sinal possui em todas as técnicas utilizadas nos experimentos, já adiantando que **SER** também segue esse princípio. CRE e suas variações foram facilmente aplicáveis e rodando com custo computacional e operacional baixos, em relação ao tempo de CPU, foram questão de segundos até mesmo no caso das interpolações. Ao contrário de **SMER-I**, em CRE-I é conhecida a região da interpolação, o que faz do método consideravelmente mais rápido.

4.4 SER

As análises são realizadas seguindo a mesma lógica da seção anterior, isto é, cada elemento do gráfico corresponde a uma malha, não a um conjunto específico de pontos. Conforme o tamanho dos elementos (h) é reduzido, significa que se trata uma malha mais fina. O símbolo "m" significa a quantidade de extrapolações utilizadas, em que $m = 0$ equivale a zero extrapolação, $m = 1$: uma extrapolação, $m = 2$: duas extrapolações etc.

Para todos os casos **SER** apresentou qualitativamente o mesmo desempenho ilustrado pela Figura 37. No entanto, ficou constatado que a cada extrapolação perde-se a malha mais fina. As malhas iniciais têm maior chance de obterem uma maior acurácia, seguindo a mesma lógica apresentada para **SMER**: quanto maior o número de malhas disponíveis mais chances da técnica obter êxito.

A afirmação que a cada extrapolação perde-se a malha mais fina, pode ser confirmada pelas tabelas 12 e 13, onde são apresentados os resultados obtidos com $m = 1$ e 2, para Poisson 1D com $T = \text{sen}(\pi x)$.

Na Tabela 12 as duas últimas malhas (14 e 15) possuem o mesmo valor em termos de grandeza e por este motivo afeta o valor de p_E^* e, posteriormente a próxima extrapolação ($m = 2$), porque haverá pouca variação no termo de correção, possivelmente nula. Esta afirmação pode ser confirmada pela Tabela 13, onde ocorre o mesmo "problema" como previsto: as malhas 14 e 15 não progrediram em seus resultados e possivelmente devem afetar a Malha 13, caso continuem as extrapolações.

Em resumo esta técnica apresentou ótimos resultados em todos os modelos utilizados. Como já adiantado anteriormente ela possui o defeito de também precisar guardar de alguma forma várias soluções numéricas DE cada malha. Mas contornado este problema, ela se torna outra importante ferramenta na otimização dos campos.

Outro defeito notado, foi a perda da malha mais fina em cada extrapolação, isto é, para $m = 1$, é perdida a Malha 15, $m = 2$: Malha 14, $m = 3$: Malha 13, $\cdots$. O termo "perder" neste cenário tem significado de não conseguir progredir mais, seja na redução do erro ou no aumento de acurácia, após a extrapolação mencionada, fato possível de ser observado nas figuras e tabelas apresentadas nesta seção.

Porém, deve ser ressaltado que foi possível obter resultados significativos mesmo sem poder realizar mais extrapolações nas malhas mais finas, para cada nível de extrapolação. Em relação ao tempo de CPU o custo é baixo, para operar esta técnica levou apenas alguns segundos.

4.5 Resumo do capítulo e conclusões

Todas as técnicas expostas neste capítulo apresentaram resultados significativos, mas algumas acabaram se sobressaindo. **CRE com Interpolação** (CRE-I) se mostrou mais estável dentre todas por conseguir otimizar as malhas finas mais rapidamente, embora a diferença é muita pouca. Pode-se dizer que ela conseguiu leve vantagem em relação as técnicas **SER**, **SMER** e **SMER-I**.

SMER-I em particular deve ser ressaltado, é a mais custosa operacionalmente dentre todas para funcionar, mas é a mais completa, por conseguir extrapolar qualquer malha e conjunto de pontos do domínio. Pode se dizer que ela é um complemento para as outras, mas não deve ser priorizada, caso alguma das outras técnicas seja aplicável no problema.

FRE apresentou desempenho similar com o termo de correção clássico (CRE), só conseguindo resultados semelhantes com as outras técnicas listadas anteriormente em único cenário: Poisson 1D com $T = \text{sen}(\pi x)$, onde esta técnica foi testada inicialmente ao ser desenvolvida.

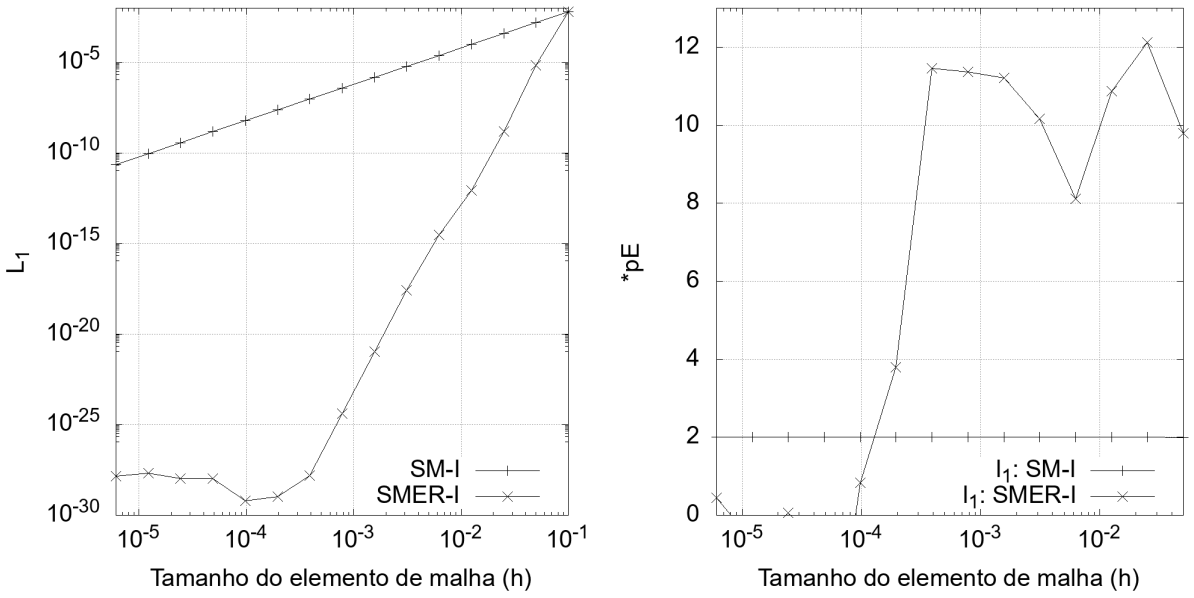
CRE era esperado ser inferior aos demais por ser o mais "defasado" em questão de tempo de criação, mais de dez anos em relação às outras. Mas cumpriu o seu papel, dentro da sua proposta, ela consegue aumentar pelo menos uma ordem verdadeira do método com redução do erro, e deve ser ressaltado sua importância para literatura, por ser uma das precursoras no uso de extrapolações de Richardson sobre os campos.

Sobre a questão de armazenamento das soluções numéricas em cada malha: para operar as técnicas, uma saída possível é criar arquivos de saída para armazenamento porque nos tempos atuais o espaço nos discos rígidos (HD) são demasiadamente grandes, não chegando a ser um problema arquivos que possuem altos valores. No entanto, contornado este problema com exceção de **SMER-I**, que pode tomar um tempo considerável, dependendo do tamanho das malhas, caso não seja utilizada a estratégia de pontos coincidentes, todas as técnicas operam rapidamente.

Lembrando que foram expostos os resultados principais, para evitar repetição e excesso de gráficos, sem haver uma necessidade real da sua exposição. Alguns resultados adicionais podem ser vistos nos apêndices, por exemplo, gráficos de **SER** para variável l_∞. Os restantes podem ser vistos integralmente no site mencionado no início deste capítulo.

Figura 33 – Resultados: SMER INTERPOLADO (SMER-I), Poisson 1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$, Variável l_1, Malha uniforme, conjunto β_1

(a) l_1 (b) p_E^*



FONTE: Autor(2020).

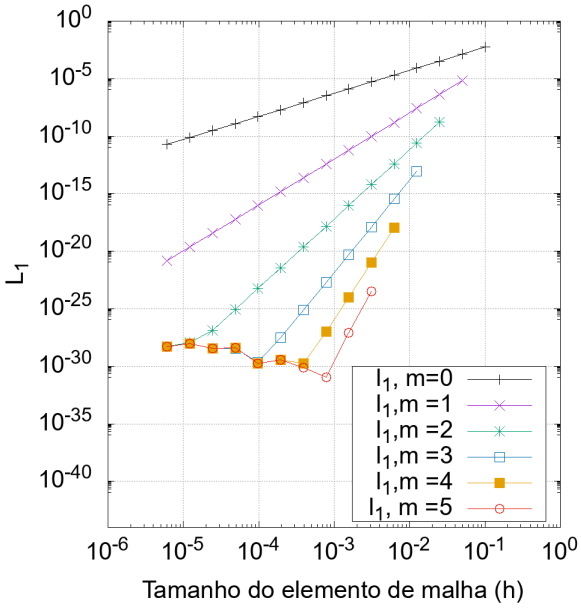
Tabela 12 – SER com uma extrapolação ($m = 1$), Poisson 1D, $T = \text{sen}(\pi x)$, malha uniforme

Malha	l_1	p_E^*: l_1
1	7.1532251319E-06	
2	4.2462173139E-07	4.0743437350E+00
3	2.5874583506E-08	4.0365706624E+00
4	1.5969404946E-09	4.0181531792E+00
5	9.9184923013E-11	4.0090459108E+00
6	6.1796851796E-12	4.0045155870E+00
7	3.8562683469E-13	4.0022559895E+00
8	2.4082847691E-14	4.0011275485E+00
9	1.5045900201E-15	4.0005636633E+00
10	9.4018509671E-17	4.0002818040E+00
11	5.8755830110E-18	4.0001408951E+00
12	3.6720600735E-19	4.0000704457E+00
13	2.2949815058E-20	4.0000352281E+00
14	1.4343459005E-21	4.0000176426E+00
15	1.7929214785E-21	-3.2191932643E-01

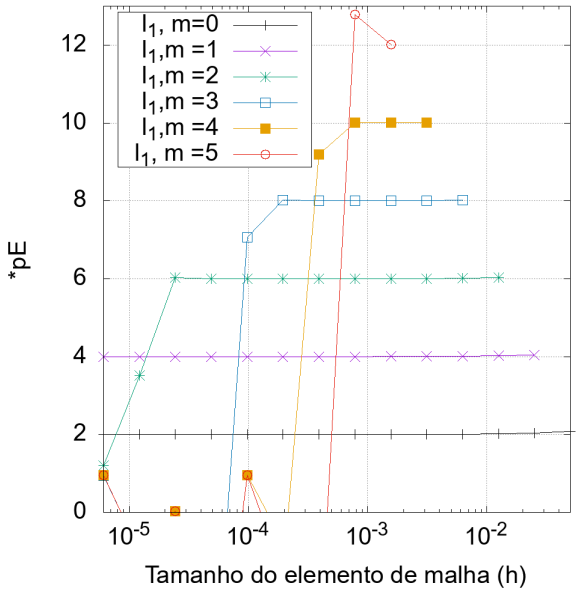
FONTE: Autor(2020).

Figura 34 – Resultados: Normas l_1 e p_E^*, CRE com FRE (FRE), Poisson 1D, Malha uniforme

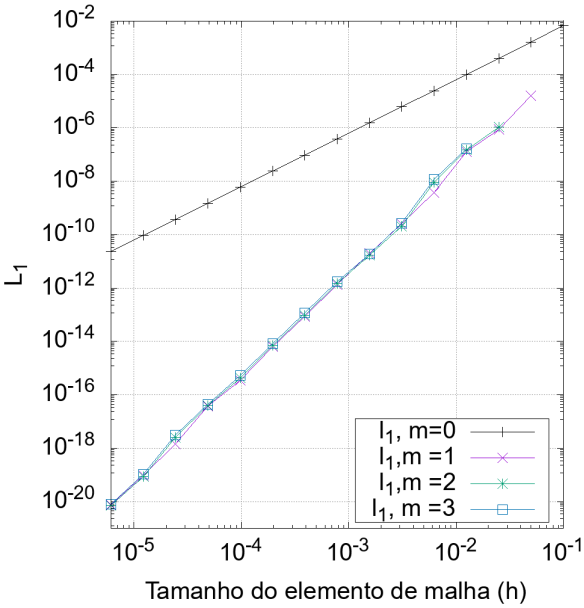
(a) l_1, $T = \text{sen}(\pi x)$



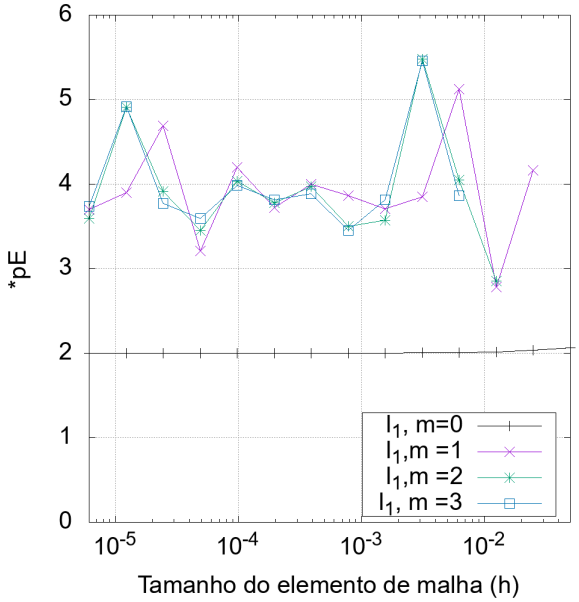
(b) p_E^*, $T = \text{sen}(\pi x)$



(c) l_1, $T = \frac{-1}{5}e^{3x} + 3x + 2$



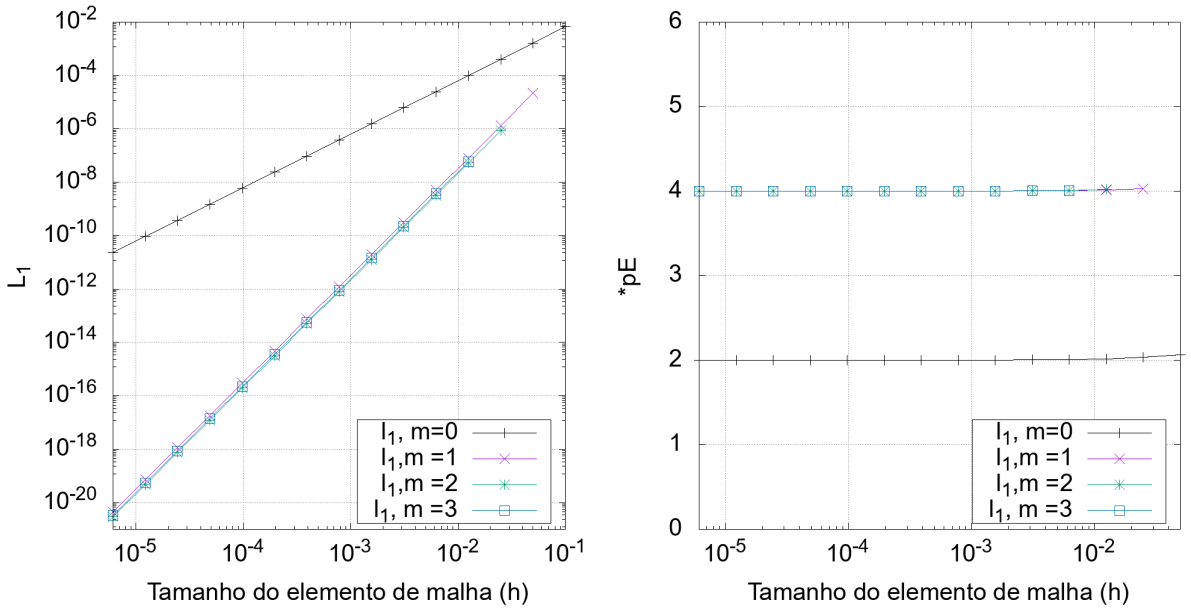
(d) p_E^*, $T = \frac{-1}{5}e^{3x} + 3x + 2$



FONTE: Autor(2020).

Figura 35 – Resultados: CRE clássico (CRE), Poisson 1D, Variável l_1, $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme

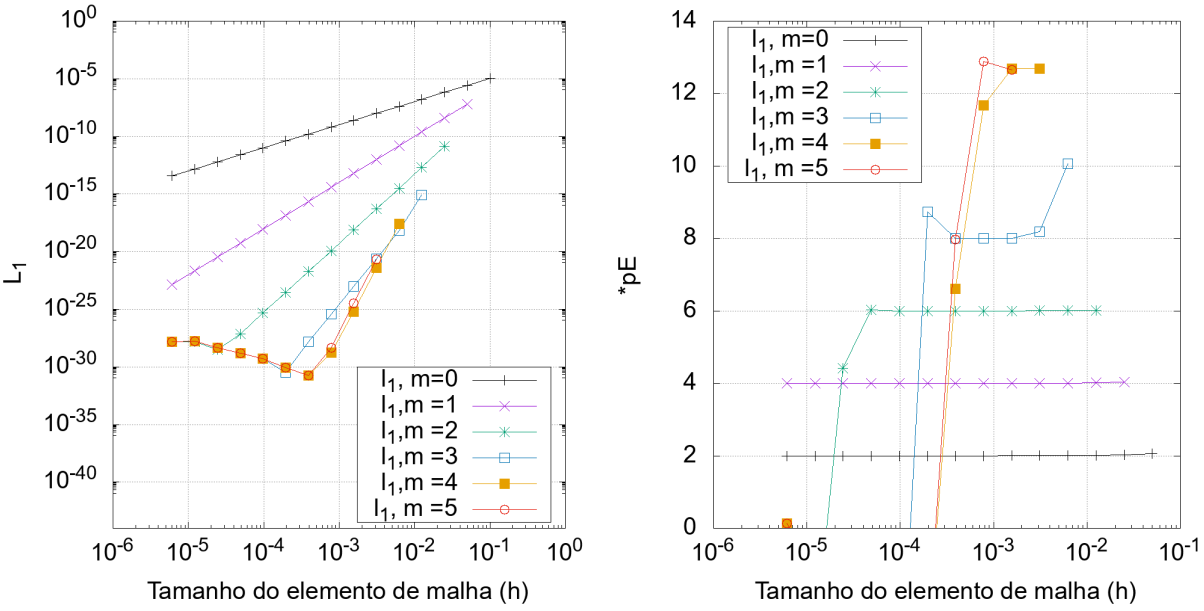
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 36 – Resultados: CRE com interpolação (CRE-I), Variável l_1, Burgers 1D,$Re = 1$, $T = \frac{e^{xRe} - 1}{e^{Re} - 1}$, Malha uniforme

(a) l_1 (b) p_E^*



FONTE: Autor(2020).

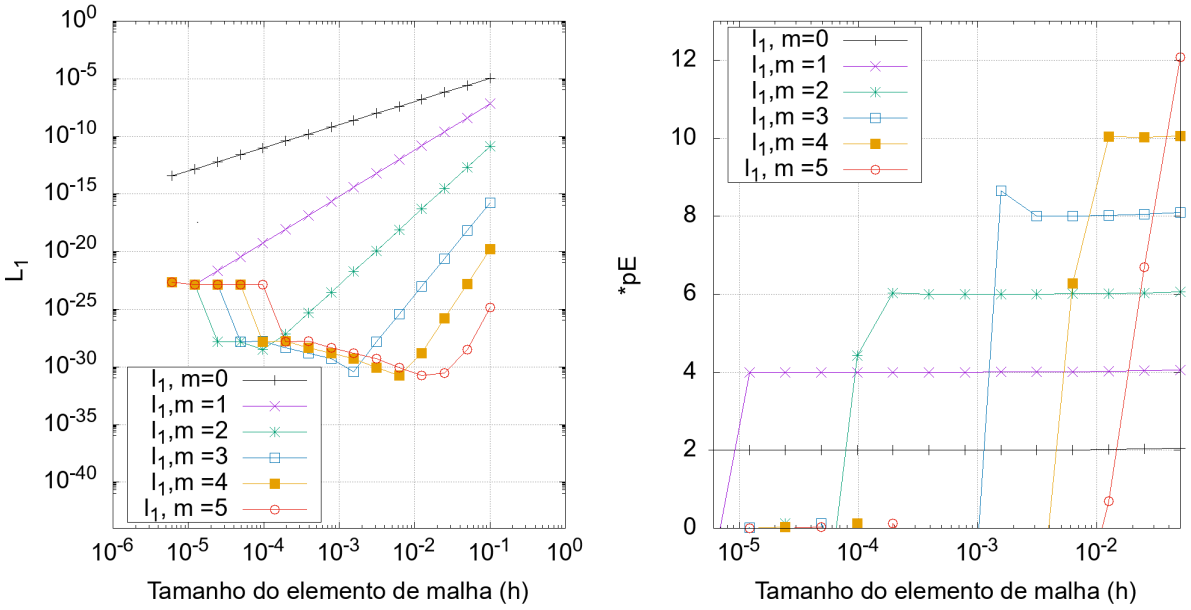
Tabela 13 – SER com duas extrapolações ($m = 2$), Variável l_1, Poisson 1D, $T = \text{sen}(\pi x)$, Malha uniforme

Malha	l_1	p_E^*: l_1
1	1.7503452727E-09	
2	2.5982572642E-11	6.0739514405E+00
3	3.9584283280E-13	6.0364727449E+00
4	6.1078101109E-15	6.0181287096E+00
5	9.4838418554E-17	6.0090397940E+00
6	1.4772209651E-18	6.0045140578E+00
7	2.3045518474E-20	6.0022556071E+00
8	3.5980493549E-22	6.0011274390E+00
9	5.6197559154E-24	6.0005636953E+00
10	8.7754465997E-26	6.0008911117E+00
11	1.3483447004E-27	6.0242112793E+00
12	1.1685710470E-28	3.5283720094E+00
13	5.2410863156E-29	1.1568076759E+00
14	1.4343459005E-21	-2.4705951870E+01
15	1.7929214785E-21	-3.2191932643E-01

FONTE: Autor(2020).

Figura 37 – Resultados: SER, Variável l_1, Burgers 1D,$Re = 1$, $T = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, Malha uniforme

(a) l_1 (b) p_E^*



FONTE: Autor(2020).

5 RESULTADOS - MALHA NÃO UNIFORME 1D

Neste capítulo são apresentados os resultados resumidos dos problemas unidimensionais, discretizados com malhas não uniformes. Resultados adicionais podem ser vistos nos apêndices e de forma integral na página:

http://ftp.demec.ufpr.br/CFD/monografias/2020_Carlos-Carvalho_doutorado/

Foram geradas 15 malhas a partir da malha base, com o vetor inicial h apresentado na Seção 3.6. Os resultados são apresentados de forma análoga ao do capítulo anterior.

Em relação ao tempo de CPU, novamente por se tratar de modelos numéricos relativamente simples, com exceção da equação de Burgers que utiliza um solver iterativo os outros modelos demoraram menos de um minuto para gerar as 15 malhas. Burgers demorou cerca de 5 minutos para concluir o mesmo procedimento, mas é um tempo relativamente baixo também.

Para utilizar as técnicas apresentadas na metodologia, novamente foi verificado se as soluções numéricas estejam sendo afetadas somente pelo erro de discretização, fato foi verificado através de análises das ordens verdadeiras a posteriori. Nas figuras 38 e 39 são apresentados os gráficos para as temperaturas T, envolvendo as soluções numéricas e analíticas. Novamente, verificada as ordens das soluções numéricas, agora é possível aplicar as técnicas de pós-processamento apresentados no capítulo 3.

5.1 SMER

De modo semelhante ao caso uniforme, em todos os experimentos, SMER apresentou o mesmo desempenho qualitativo, seguindo os mesmos padrões observados anteriormente:

1. Malhas iniciais com maior acurácia;
2. Quanto maior a quantidade de malhas disponíveis, mais chances de SMER obter êxito;
3. Soluções numéricas com SMER apresentam comportamento similar aos resultados encontrados com MER na literatura, ou seja:

    a) redução significativa do erro com concomitante aumento nas ordens de acurácia.

É possível observar a terceira observação através das Figuras 40 e 41, que apresentam resultados de Poisson 1D com $T(x) = \text{sen}(\pi x)$ e Burgers 1D, respectivamente. É

fácil ver a rápida convergência do método, quando se possui várias malhas disponíveis e principalmente acuradas (Segunda observação).

A primeira e a segunda observações podem ser novamente comprovadas através da Figura 42 que ilustra as curvas de l_1 com **SMER**, para Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$. Incrível a similaridade com o caso uniforme, desempenho novamente muito satisfatório. É possível visualizar até mesmo nas malhas mais finas uma considerável redução do erro, por exemplo, em $g = 14$ com $m = 0$, l_1 tinha seu erro em módulo aproximadamente em 10^{-10} e com apenas uma malha disponível **SMER** alcançou aproximadamente 10^{-20} , uma **notória** redução do erro.

Em resumo, **SMER** novamente apresentou grande desempenho com resultados qualitativos aos apresentados nesta seção, em todos os modelos unidimensionais utilizados. Para utilizá-la novamente foi exigido pouco tempo de CPU, questão de **segundos**. Seu único problema novamente seria o armazenamento das soluções numéricas.

5.2 SMER INTERPOLADO

Os conjuntos β_1 e β_2 são os mesmos, explicados na Seção 4.2, ou seja, correspondem aos pontos equivalentes da malha 15, com exceção dos contornos, para cada caso: uniforme e não uniforme, respectivamente. As malhas foram geradas a partir da malha inicial de cada cenário, utilizando refino de razão dois.

Para ambos os cenários, houve redução do erro de forma significativa, porém as ordens equivalentes se mantiveram em 4, significando que **SMER** não conseguiu progredir nas extrapolações e só evoluiu uma ordem equivalente, visto que $p_0 = 2$.

Neste cenário, é notória a diferença de desempenho da técnica ao trocar a discretização do mesmo modelo para malha não uniforme. Isso pode ser observado através das tabelas 14 e 15, onde são apresentados os resultados de l_1 do conjunto β_2, para Poisson 1D com $T = \frac{-1}{5}e^{3x} + 3x + 2$, envolvendo as duas discretizações.

As possíveis causas para o aproveitamento inferior são:

1. Interpolação;
2. Erro numérico.

A essência da técnica é MER, e um dos requisitos básicos e fundamentais de seu funcionamento é o erro numérico (E) ser afetado somente por Eh. Porém, é provável que as interpolações interfiram no nível das soluções numéricas, no caso não uniforme. Neste cenário E não estaria mais sendo afetado somente por Eh.

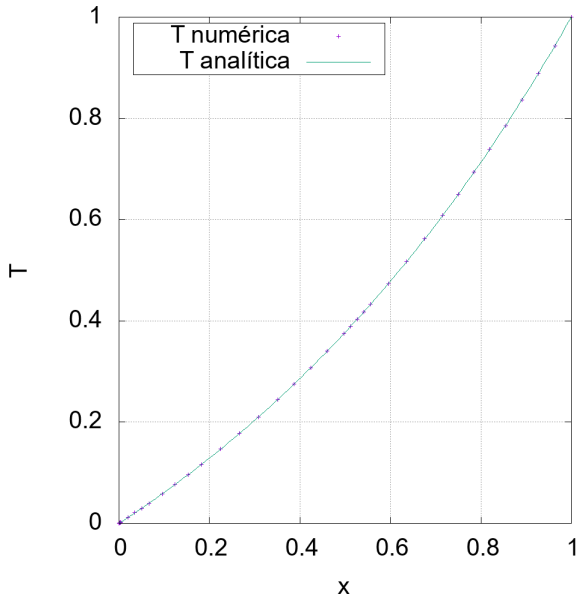
No entanto é só uma suposição, sendo necessária a realização de mais investigações e testes para confirmar estas premissas. Porque SMER funcionou e sua base é puramente MER, ao contrário de SMER-I que são acrescentadas as interpolações.

Tabela 14 – Resultados: SMER INTERPOLADO, conjunto β_2, Variável l_1, Malha não uniforme, Poisson1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$

Malha	h	l_1: SM-I	l_1: SMER-I	p_E^*: l_1, SM-I	p_E^*: l_1, SMER-I
1	1.00E-01	1.45524E-02	1.45524E-02		
2	5.00E-02	2.72234E-03	2.19539E-03	2.41834E+00	2.72871E+00
3	2.50E-02	6.58480E-04	1.98792E-04	2.04764E+00	3.46515E+00
4	1.25E-02	1.64871E-04	7.96199E-06	1.99780E+00	4.64199E+00
5	6.25E-03	4.12217E-05	2.30626E-07	1.99986E+00	5.10950E+00
6	3.13E-03	1.03056E-05	1.17395E-08	1.99998E+00	4.29610E+00
7	1.56E-03	2.57641E-06	7.09886E-10	1.99999E+00	4.04765E+00
8	7.81E-04	6.44103E-07	4.42717E-11	2.00000E+00	4.00313E+00
9	3.91E-04	1.61026E-07	2.76879E-12	2.00000E+00	3.99906E+00
10	1.95E-04	4.02564E-08	1.73270E-13	2.00000E+00	3.99816E+00
11	9.77E-05	1.00641E-08	1.08543E-14	2.00000E+00	3.99668E+00
12	4.88E-05	2.51603E-09	6.81387E-16	2.00000E+00	3.99365E+00
13	2.44E-05	6.29007E-10	4.29299E-17	2.00000E+00	3.98842E+00
14	1.22E-05	1.57252E-10	2.73185E-18	2.00000E+00	3.97403E+00
15	6.10E-06	3.93129E-11	1.74994E-19	2.00000E+00	3.96450E+00

FONTE: Autor(2020).

Figura 38 – $T(x)$: Numérica vs Analítica, malha não uniforme, Burgers 1D, $Re = 1$



FONTE: Autor(2020).

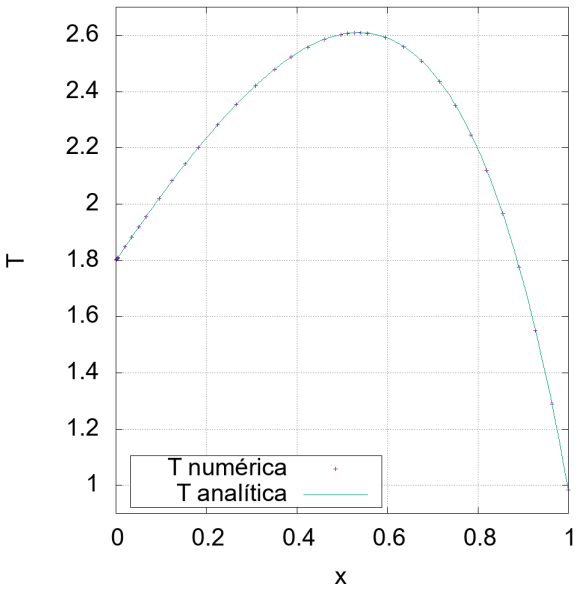
Tabela 15 – Resultados: SMER INTERPOLADO, conjunto β_2, Variável l_1, Malha uniforme, Poisson1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$

Malha	h	l_1: SM-I	l_1: SMER-I	p_E^*: l_1, SM-I	p_E^*: l_1, SMER-I
1	1.00E-01	4.80134E-03	4.80134E-03		
2	5.00E-02	1.20439E-03	5.40196E-06	1.99514E+00	9.79574E+00
3	2.50E-02	3.01351E-04	1.21598E-09	1.99878E+00	1.21171E+01
4	1.25E-02	7.53535E-05	9.29105E-13	1.99970E+00	1.03540E+01
5	6.25E-03	1.88394E-05	3.38711E-15	1.99992E+00	8.09964E+00
6	3.13E-03	4.70991E-06	2.84847E-18	1.99998E+00	1.02156E+01
7	1.56E-03	1.17748E-06	9.23999E-22	2.00000E+00	1.15900E+01
8	7.81E-04	2.94370E-07	3.38573E-25	2.00000E+00	1.14142E+01
9	3.91E-04	7.35926E-08	1.27326E-28	2.00000E+00	1.13767E+01
10	1.95E-04	1.83982E-08	8.46534E-30	2.00000E+00	3.91082E+00
11	9.77E-05	4.59954E-09	4.52605E-30	2.00000E+00	9.03316E-01
12	4.88E-05	1.14988E-09	8.45124E-29	2.00000E+00	-4.22284E+00
13	2.44E-05	2.87471E-10	7.52673E-29	2.00000E+00	1.67140E-01
14	1.22E-05	7.18678E-11	1.57413E-28	2.00000E+00	-1.06446E+00
15	6.10E-06	1.79669E-11	1.05208E-28	2.00000E+00	5.81312E-01

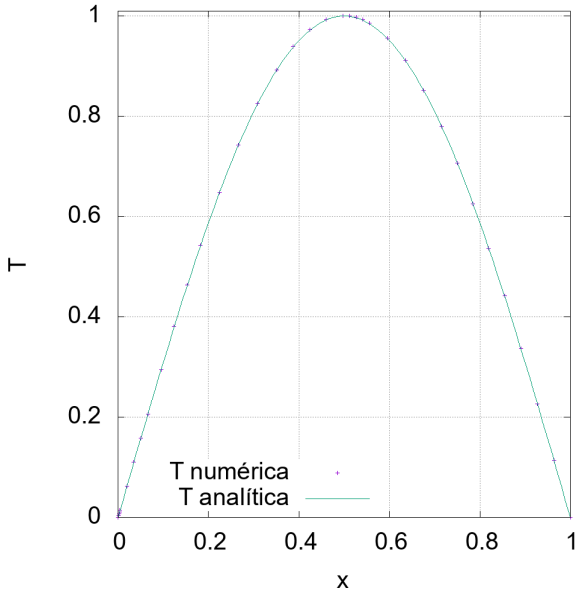
FONTE: Autor(2020).

Figura 39 – $T(x)$: Numérica vs Analítica, Poisson 1D, malha não uniforme

(a) Poisson 1D, $T(x) = \frac{-1}{5}e^{3x} + 3x + 2$.



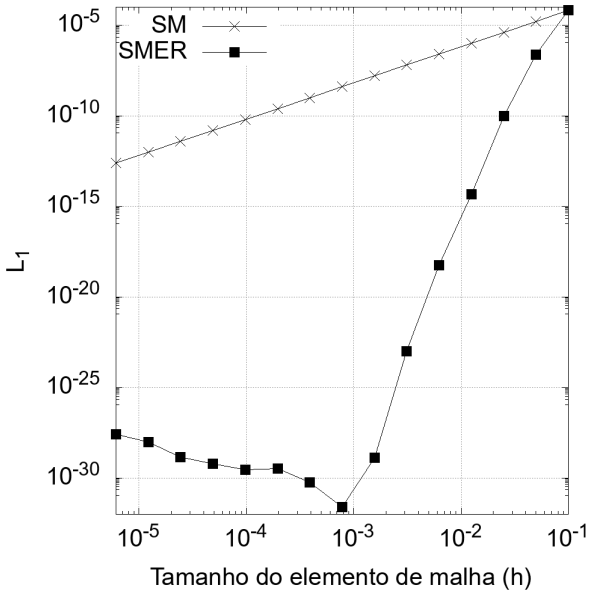
(b) Poisson 1D, $T(x) = \text{sen}(\pi x)$.



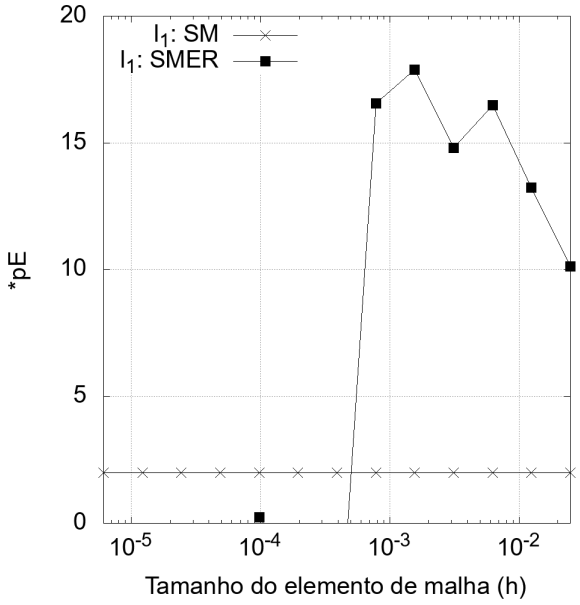
FONTE: Autor(2020).

Figura 40 – Resultados: SMER, Malhas 1-2, Variável l_1, Burgers 1D com solução $T(x) = \frac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme

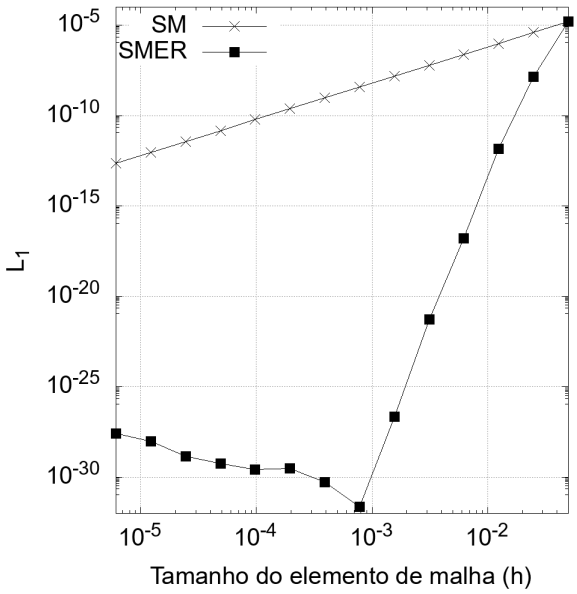
(a) l_1, Malha 1



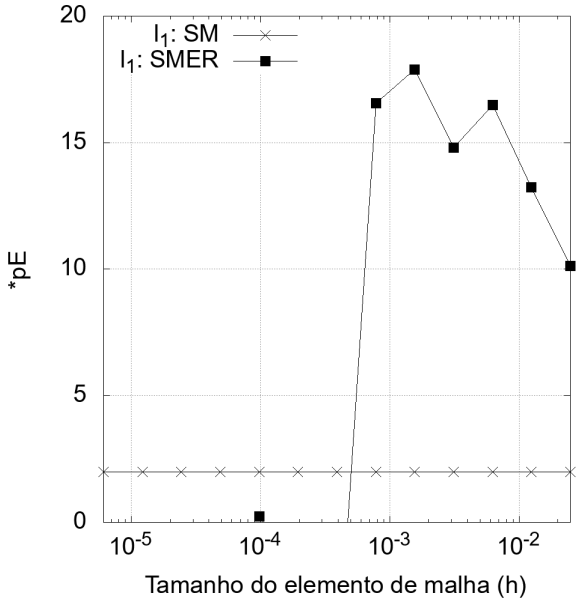
(b) p_E^*, Malha 1



(c) l_1, Malha 2



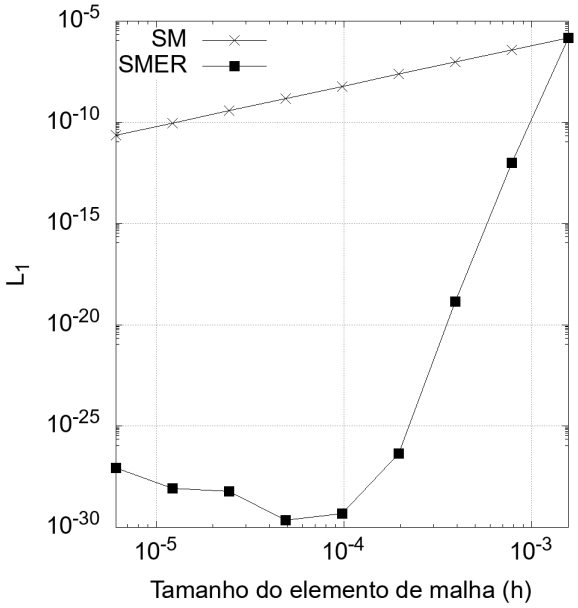
(d) p_E^*, Malha 2



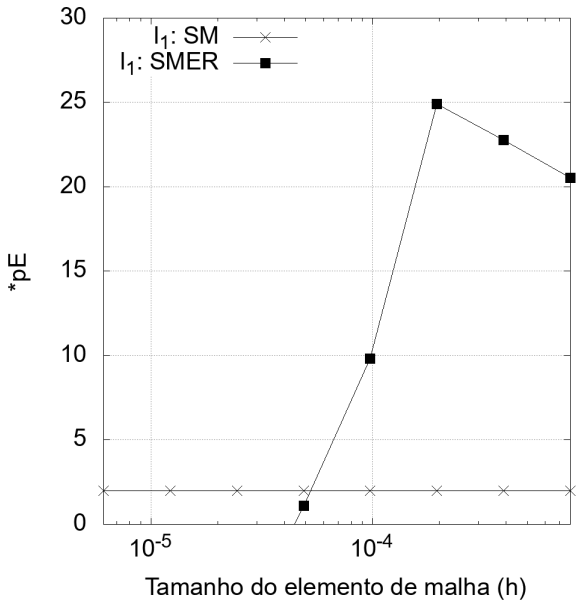
FONTE: Autor(2020).

Figura 41 – Resultados: SMER, Malhas 7-8, Variável l_1, Poisson 1D com solução $T(x) = \text{sen}(\pi x)$, Malha não uniforme

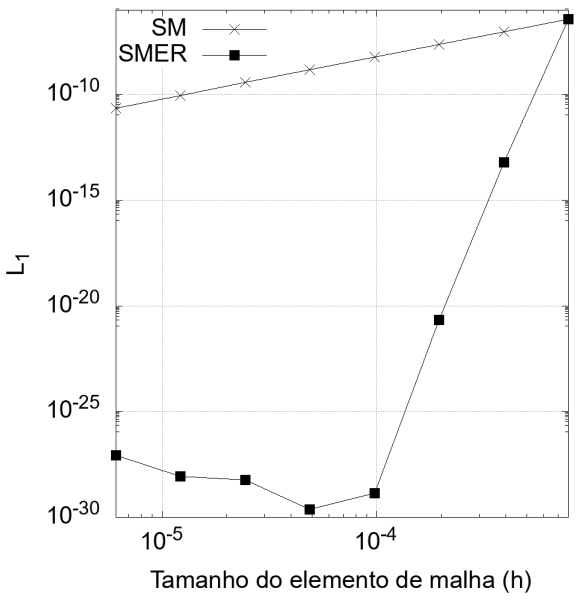
(a) l_1, Malha 7



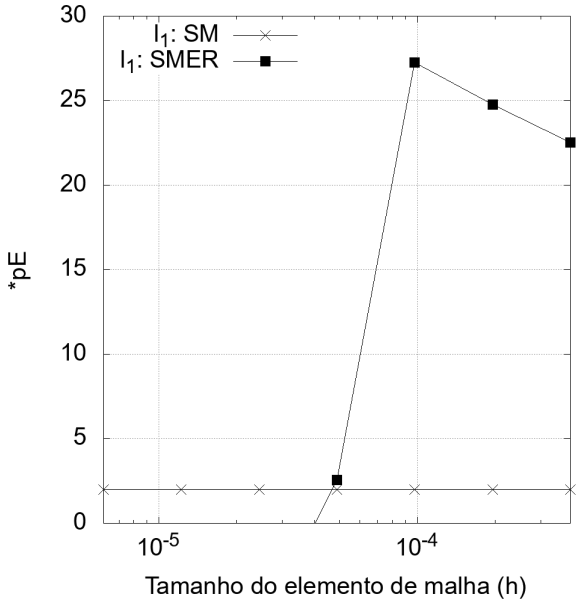
(b) p_E^*, Malha 7



(c) l_1, Malha 8

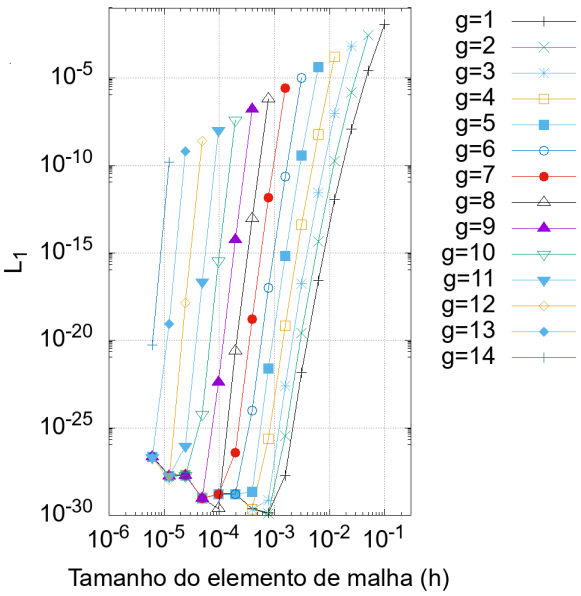


(d) p_E^*, Malha 8



FONTE: Autor(2020).

Figura 42 – Resultados: SMER, Malhas $g = 1{,}2 \cdots ,14$, Variável l_1, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha não uniforme



FONTE: Autor(2020).

5.3 CRE clássico com interpolação e FRE

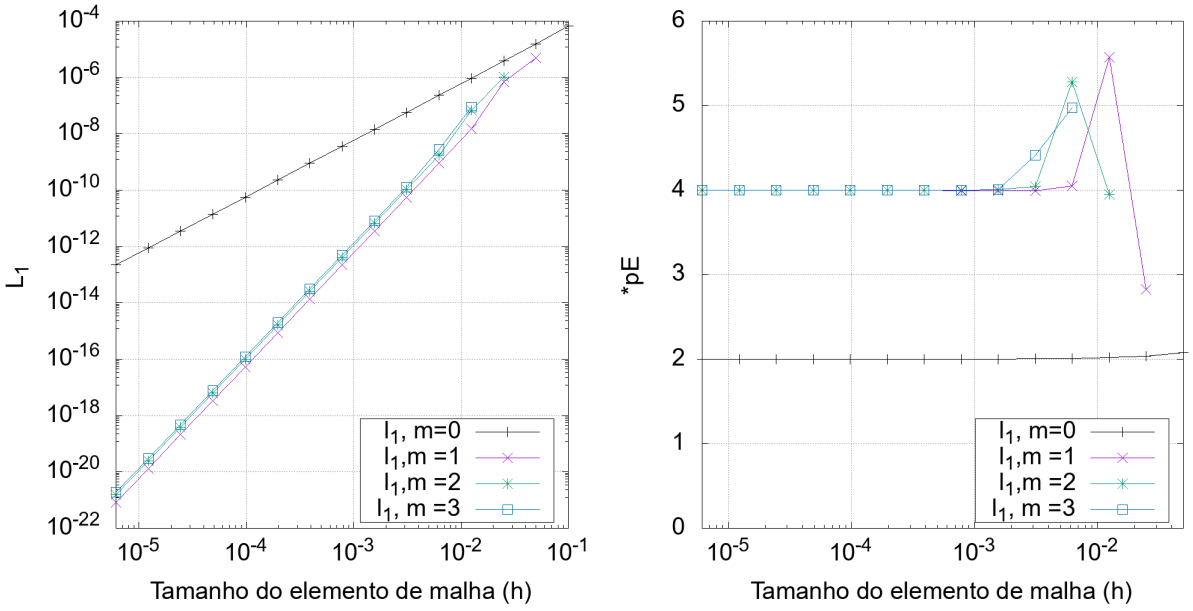
Em todos os casos houve redução do erro, porém as ordens de acurácia alcançaram no máximo 4. Exceção de FRE que não foi possível determinar sua ordem para $Poisson1D$, mostrando a mesma instabilidade em suas ordens de acurácia, apresentadas já no caso uniforme.

Estes resultados comentados anteriormente podem ser observados de forma qualitativa através das Figuras 43 e 44, onde são apresentados os resultados de CRE-I e FRE para Burgers 1D e Poisson 1D com $T(x) = \text{sen}(\pi x)$, respectivamente. Portanto, só foi possível progredir uma ordem verdadeira, visto que $p_0 = 2$. Novamente, pode-se reparar que alterando a estrutura da malha para não uniforme, as interpolações já não conseguem progredir com a mesma eficácia para o caso da malha uniforme. Os motivos provavelmente devem ser os mesmos de SMER-I, pelo fato da sua essência ser também as interpolações.

Neste cenário alterar o termo de correção de CRE não surtiu efeito, porque o termo de correção clássico (CRE) qualitativamente obteve os mesmos resultados.

Figura 43 – Resultados: CRE-I, Variável l_1, Burgers 1D com solução $T(x) = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme

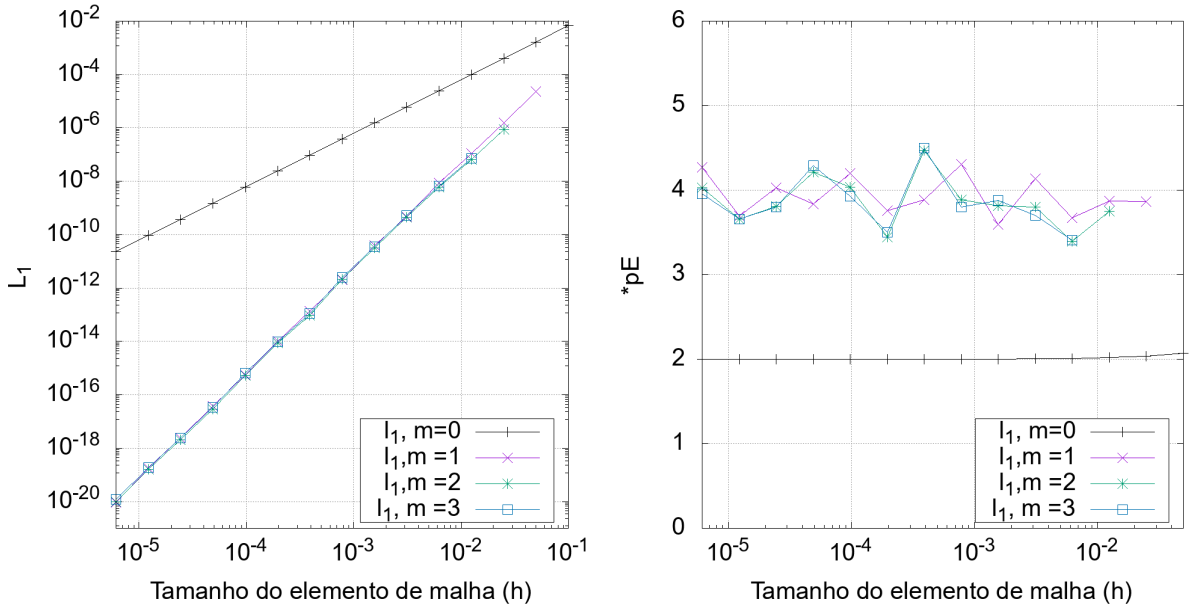
(a) l_1 (b) p^*_E



FONTE: Autor(2020).

Figura 44 – Resultados: FRE, Variável l_1, Poisson 1D com solução $T(x) = \text{sen}(\pi x)$, Malha não uniforme

(a) l_1 (b) p^*_E



FONTE: Autor(2020).

5.4 SER

De modo semelhante ao caso uniforme, em todos os experimentos, SER apresentou o mesmo desempenho qualitativo, seguindo os mesmos padrões observados anteriormente:

1. A cada extrapolação perde-se a malha fina;
2. Quanto maior a quantidade de malhas disponíveis, mais chances de SER obter êxito.

Estas ponderações podem ser observadas através da Figura 45 que apresenta os resultados de l_1 para Burgers 1D com $Re = 1$.

Ressaltando, o termo "perder" mencionado no item 1 neste âmbito, significa que a malha em questão não consegue progredir mais após aquela extrapolação, seja na redução do erro ou no aumento da acurácia. No entanto, isto não significa que seja ruim, como pode ser observado todas as malhas apresentaram significativa melhora, incluindo as malhas mais finas.

Portanto, **SER** novamente apresentou grande desempenho com resultados semelhantes aos apresentados pela Figura 45, em todos os modelos unidimensionais utilizados. Para utilizá-la novamente foi exigido pouco tempo de CPU, questão de **segundos**. Seu único problema novamente seria o armazenamento das soluções numéricas.

Outro dado interessante, uma diferença de SER para CRE-I é justamente as **interpolações**, do mesmo modo que SMER e SMER-I. No caso de CRE-I e SER sua essência é ER, SMER e SMER-I é MER. Claramente as ponderações realizadas para o desempenho inferior de SMER-I podem ser as mesmas para CRE-I, porque MER é uma versão aprimorada de ER.

5.5 Resumo do capítulo e conclusões

Todas as técnicas expostas neste capítulo apresentaram reduções significativas do erro numérico, mas algumas acabaram se sobressaindo consideravelmente. **SER** e **SMER** apresentaram os melhores resultados, mostrando serem mais estáveis que as demais técnicas, porque funcionaram de forma equivalente em comparação com as malhas uniformes, onde também foram obtidos resultados expressivos.

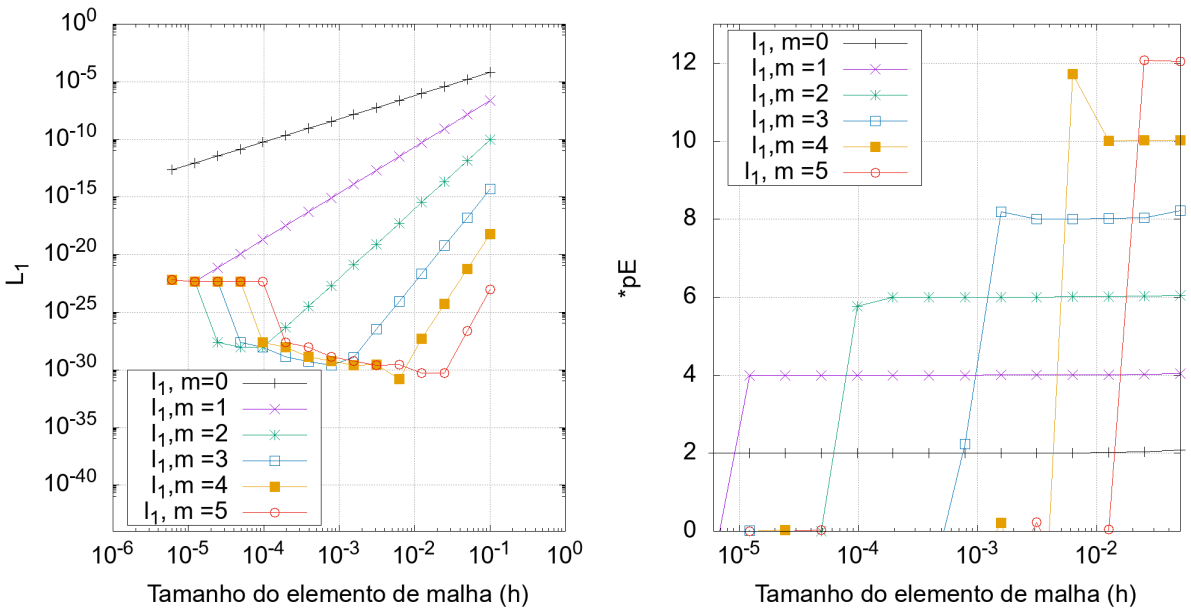
CRE-I, SMER-I e FRE apresentaram desempenho similar com o termo de correção clássico (CRE), obtendo redução de erro significativa, mas estabilizaram suas ordens de acurácia em 4. Neste cenário utilizar estas variações de CRE não apresentou diferença alguma, porque CRE obteve resultados semelhantes. CRE por sinal, novamente cumpriu o seu papel, dentro da sua proposta, ela consegue aumentar pelo menos uma ordem verdadeira do método com redução do erro.

Lembrando que foram expostos os resultados principais, para evitar repetição e excesso de gráficos, sem haver uma necessidade real da sua exposição. A norma l_∞ também apresentou divergências para as técnicas CRE-I e SMER-I em questão de redução do erro e ordens de acurácia, em comparação a l_1. Este resultado não esperado pode ajudar em trabalhos futuros, no entendimento do motivo de CRE-I e SMER-I obterem resultados inferiores neste tipo de malha.

Alguns destes resultados adicionais para l_∞ podem ser vistos nos apêndices, por exemplo, gráficos de **SER** para variável l_∞. Os restantes podem ser vistos integralmente no site mencionado no início deste capítulo. Ressaltando que SMER e SER mantiveram o nível semelhante a l_1, também nesta variável.

Figura 45 – Resultados: SER, Variável l_1, Burgers 1D com solução $T(x) = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme

(a) l_1 (b) p_E^*

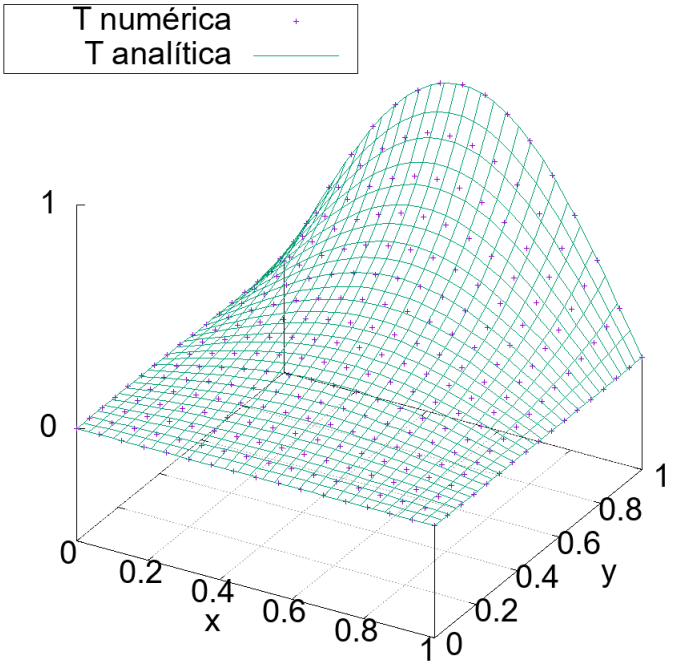


FONTE: Autor(2020).

6 RESULTADOS - LAPLACE 2D

A equação de Laplace foi discretizada com dois tipos de malhas: uniforme e não uniforme. Os resultados obtidos neste capítulo se equiparam de forma qualitativa aos apresentados nos capítulos 4 e 5. Na figura abaixo é apresentado o gráfico "$T(x,y)$: Numérica vs Analítica", para o caso da malha uniforme.

Figura 46 – $T(x,y)$: Numérica vs Analítica, Laplace 2D, malha uniforme.



FONTE: Autor(2020).

São apresentados apenas alguns resultados para a norma l_1. No Apêndice C são apresentados alguns resultados para l_∞. Todos os resultados podem ser vistos no link:

http://ftp.demec.ufpr.br/CFD/monografias/2020_Carlos-Carvalho_doutorado/

6.1 MALHA UNIFORME

Para o caso da malha uniforme foram obtidas 10 malhas, geradas a partir da malha inicial apresentada na Seção 3.6. Em relação ao tempo de CPU, graças ao Multigrid foi necessário apenas aproximadamente 5 minutos para obter as 10 malhas.

Novamente para aplicar as técnicas de pós-processamento, foi verificada a acurácia das soluções através de estimativas a posteriori. As malhas obtidas para os experimentos estão com suas ordens dentro do esperado.

6.1.1 SMER

SMER obteve resultados similares com os relatados anteriormente, isto é, significativa redução do erro com concomitante aumento nas ordens de acurácia.

Novamente as malhas iniciais são as mais beneficiadas, este resultado é ilustrado pela Figura 47. No entanto, conforme é apresentado pela Figura 48, mesmo com número inferior de malhas disponíveis, SMER obteve resultados similares para algumas malhas. Neste exemplo, para a Malha 6 em comparação com a inicial (Malha 1), ambas alcançaram erros em módulo inferiores a 10^{-25} para l_1.

Além disso, até mesmo as malhas mais finas alcançaram significativa redução do erro, mas em menor escala em comparação com as iniciais, como pode ser observado através da Figura 47. Todavia, lembrando que o primeiro valor das curvas correspondem por $m = 0$ da Malha g.

Analisando o custo computacional, a versão bidimensional se equipara ao caso unidimensional qualitativamente, porque necessita de alguns segundos para operar e seu único problema seria a questão de armazenamento das malhas.

6.1.2 SMER INTERPOLADO

SMER-I obteve resultados qualitativos aos apresentados para as discretizações realizadas em malhas uniformes, independente do conjunto escolhido. Para o caso bidimensional foram testados três conjuntos: β_1, β_2 e β_3.

Lembrando que β_1 e β_2 correspondem as malhas finais das discretizações, ou seja

$$\text{“}\beta_1 = \text{Malha 10 uniforme”, “}\beta_2 = \text{Malha 8 não uniforme”}$$

e β_3 que corresponde a Malha 8, originada do refino da malha inicial 3, apresentada na Seção 3.6. A principal diferença deste conjunto para os outros são os vetores iniciais h_x e h_y da malha inicial, porque são não uniformes com dimensões diferentes.

No entanto, conforme comentado anteriormente, independente do conjunto escolhido SMER-I obteve êxito, apresentando resultados qualitativos ao exposto pela Figura 49, onde é ilustrado o caso de β_3. Neste cenário, SMER-I novamente se equipara a SMER, servindo de complemento. Isso porque SMER não consegue extrapolar, por exemplo, a malha mais fina disponível, ao contrário de SMER-I.

Porém, ressalta-se novamente, o processo de busca dos pontos para interpolações pode demorar dias, caso não se utilize a estratégia de pontos coincidentes, para malhas demasiadamente grandes. Este a priori é o principal defeito desta técnica.

As estratégias utilizadas, conforme explicado no Capítulo 3 e especificamente na Seção 3.5.1, variam para o CPU utilizado e a quantidade de pontos do conjunto β. A primeira afirmação se justifica devido ao tempo que cada CPU demora para realizar as buscas pelo ponto base. Já a segunda é uma consequência direta da primeira, quanto maior a quantidade de pontos mais demorado o processo. Outro fator que pode acarretar mais demora seria a quantidade de malhas utilizadas nos experimentos.

Neste experimento em específico, realizou-se a busca até a malha 6, depois empregou-se a estratégia dos pontos coincidentes, no total o processo demorou cerca de uma hora. Tempo considerado alto em comparação com as outras técnicas que demoram segundos.

6.1.3 CRE clássico e com interpolação

Para os problemas bidimensionais não foram aplicados FRE, somente CRE clássico (CRE) e sua variação com o termo de correção com interpolação (CRE-I). Em resumo estes foram os resultados obtidos para o caso uniforme:

1. **CRE** apresentou resultados esperados, novamente conseguiu progredir ao menos uma ordem de acurácia de $p_0 = 2$ para $p_1 = 4$, com redução do erro significativa. Naturalmente, não apresenta evolução para $m > 1$.

2. **CRE-I** apresentou desempenho equivalente com os resultados obtidos nos problemas unidimensionais uniformes, isto é, redução significativa do erro e aumento considerável das ordens de acurácia.

O primeiro item é ilustrado pela Figura 50, CRE cumpriu seu papel dentro da sua proposta inicial. Já o segundo item é ilustrado pela Figura 51, claramente CRE-I foi muito superior se equiparando a SMER e SMER-I. Analisando o custo operacional e tempo CPU, ambas funcionaram rapidamente em questão de segundos.

Em relação as estratégias de interpolações utilizadas para CRE-I, foram alcançados os mesmos resultados de forma qualitativa. Neste cenário, é preferível utilizar a interpolação alternativa unidimensional, pelo baixo custo operacional, caso ela seja aplicável. Todavia, lembrando que estas estratégias foram explicadas na Seção 3.5.

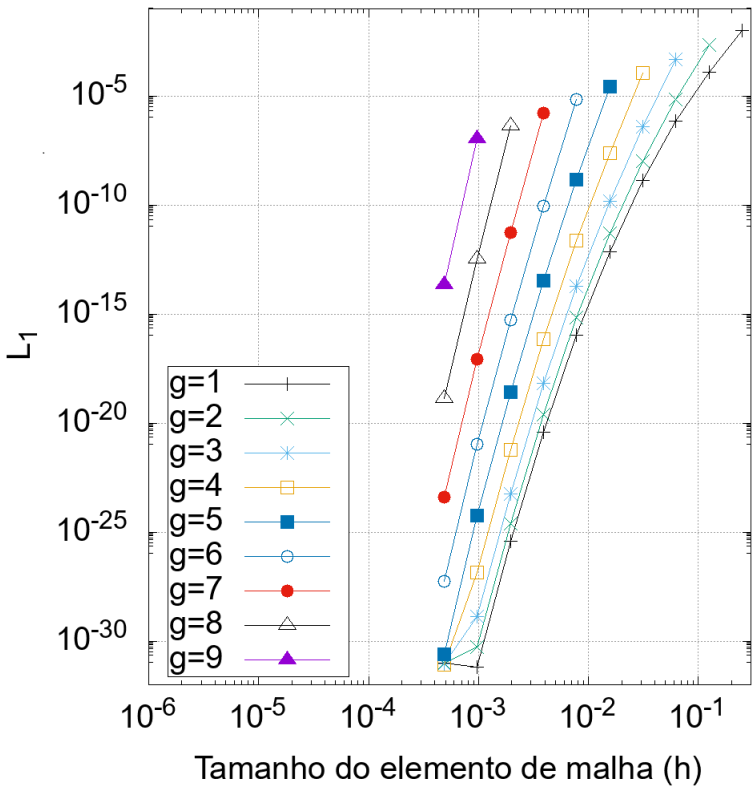
6.1.4 SER

Novamente SER apresentou desempenho qualitativo com os apresentados anteriormente, isto é, redução significativa do erro e aumento das ordens de acurácia. Do mesmo modo, a cada extrapolação m perde-se a malha mais fina. Este resultado é ilustrado pela Figura 52.

Lembrando que o termo "perder" significa que a malha não consegue mais progredir, após aquela extrapolação, seja na redução do erro ou aumento da acurácia. No entanto, não significa que não foi possível obter resultados relevantes, conforme pode ser observado na Figura 52.

Analisando o custo operacional e tempo CPU, SER novamente operou rapidamente em questão de segundos. Para concluir esta primeira parte: SER, CRE-I, SMER e SMER-I apresentaram ótimos e similares resultados para este tipo malha. Mas no geral CRE-I levou vantagem, pelo custo operacional e pelo fato de conseguir extrapolar a malhas mais fina.

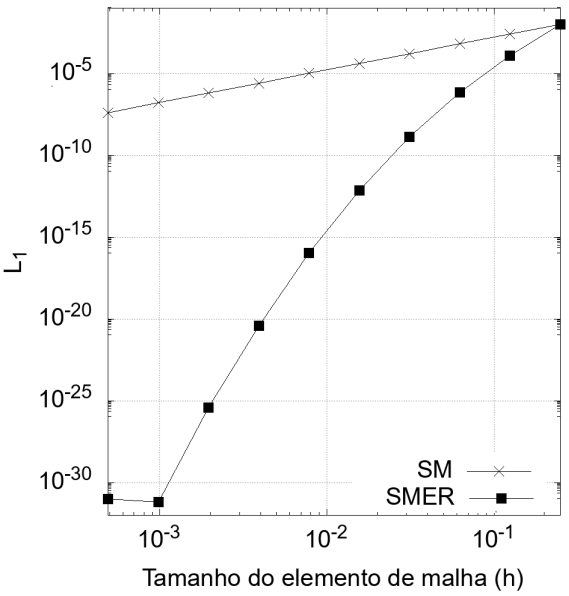
Figura 47 – Resultados: SMER, Malhas $g = 1,2,\cdots,9$, Variável l_1, Laplace 2D, Malha uniforme



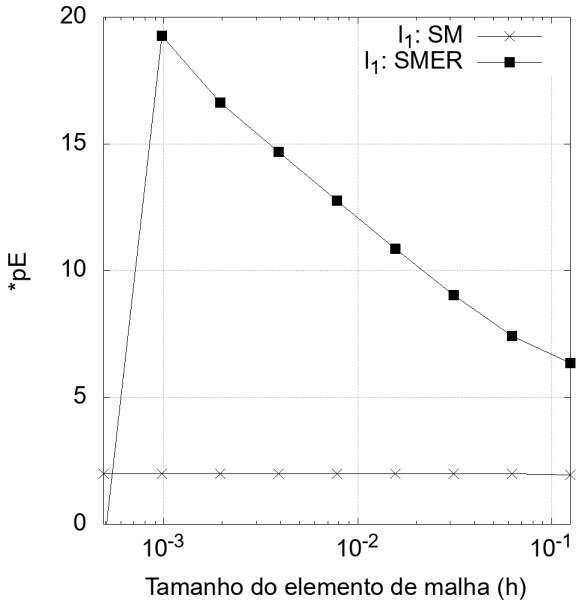
FONTE: Autor(2020).

Figura 48 – Resultados: SMER, Malhas 1 e 6, Variável l_1, Laplace 2D, Malha uniforme

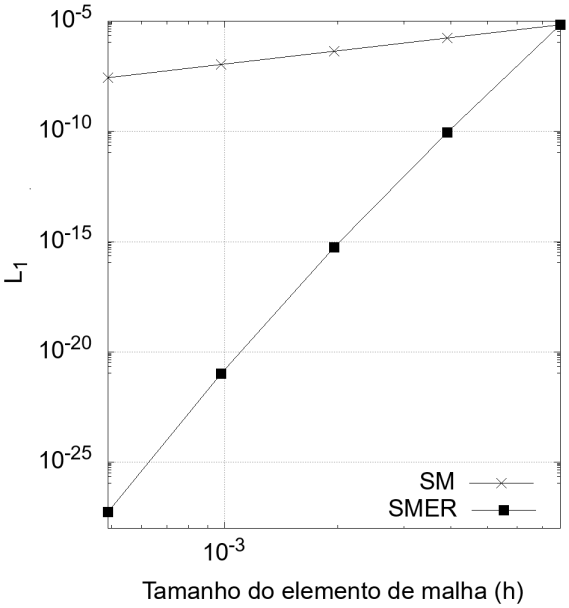
(a) l_1, Malha 1



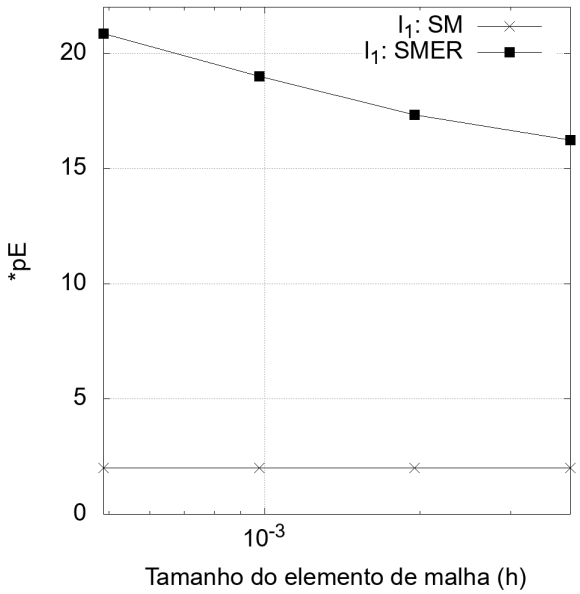
(b) p_E^*, Malha 1



(c) l_1, Malha 6



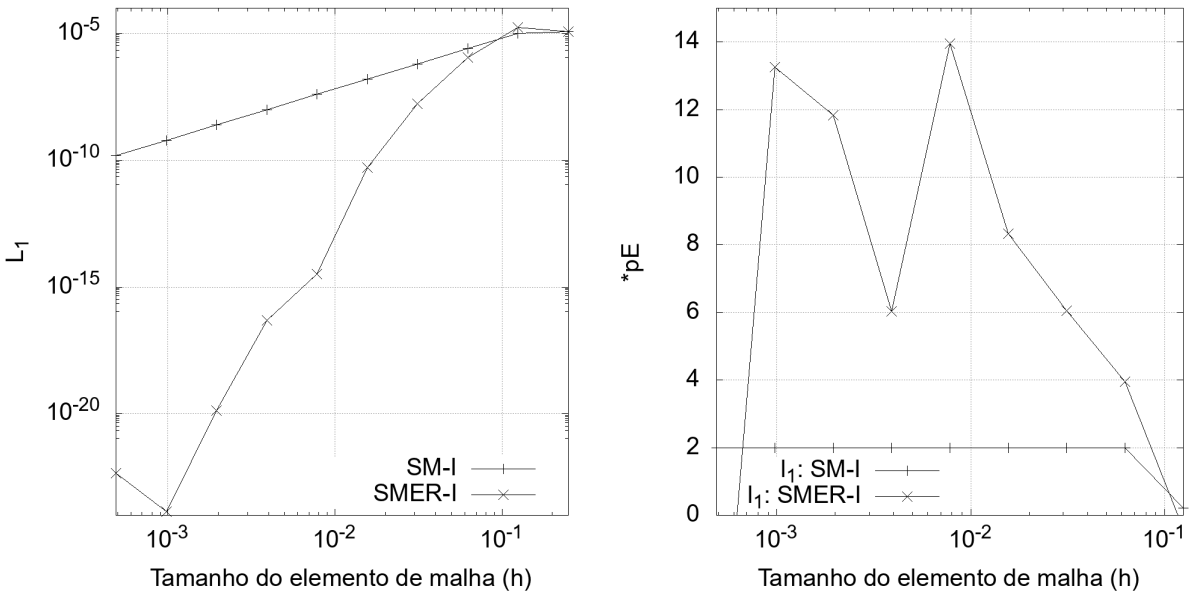
(d) p_E^*, Malha 6



FONTE: Autor(2020).

Figura 49 – Resultados: SMER-I, Conjunto β_3, Variável l_1, Laplace 2D, Malha uniforme

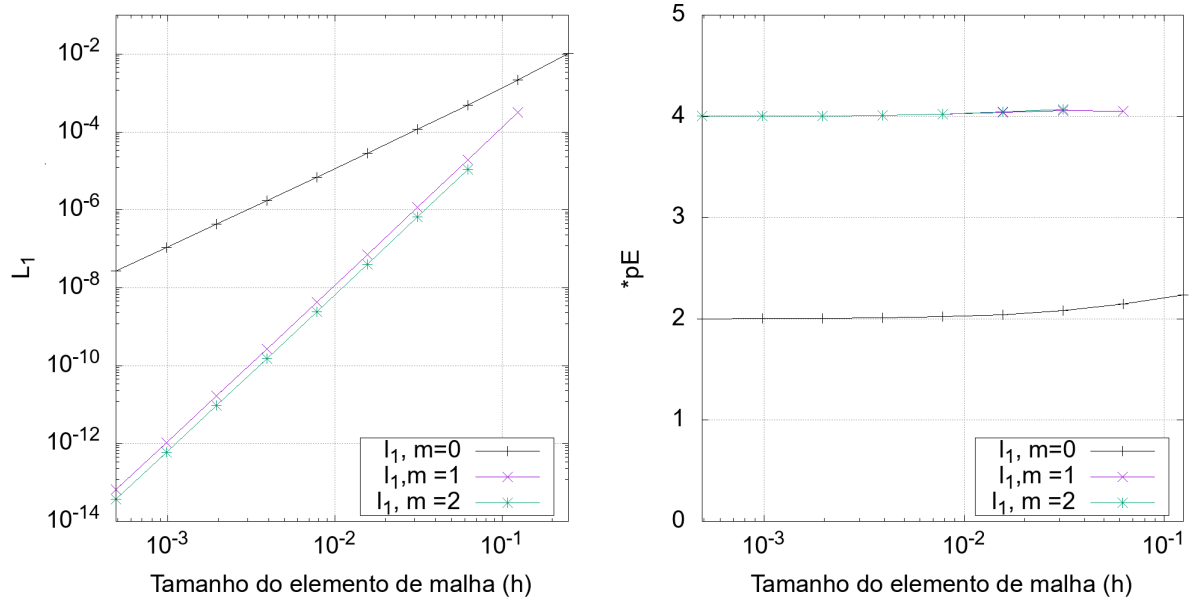
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 50 – Resultados: CRE, Variável l_1, Laplace 2D, Malha uniforme

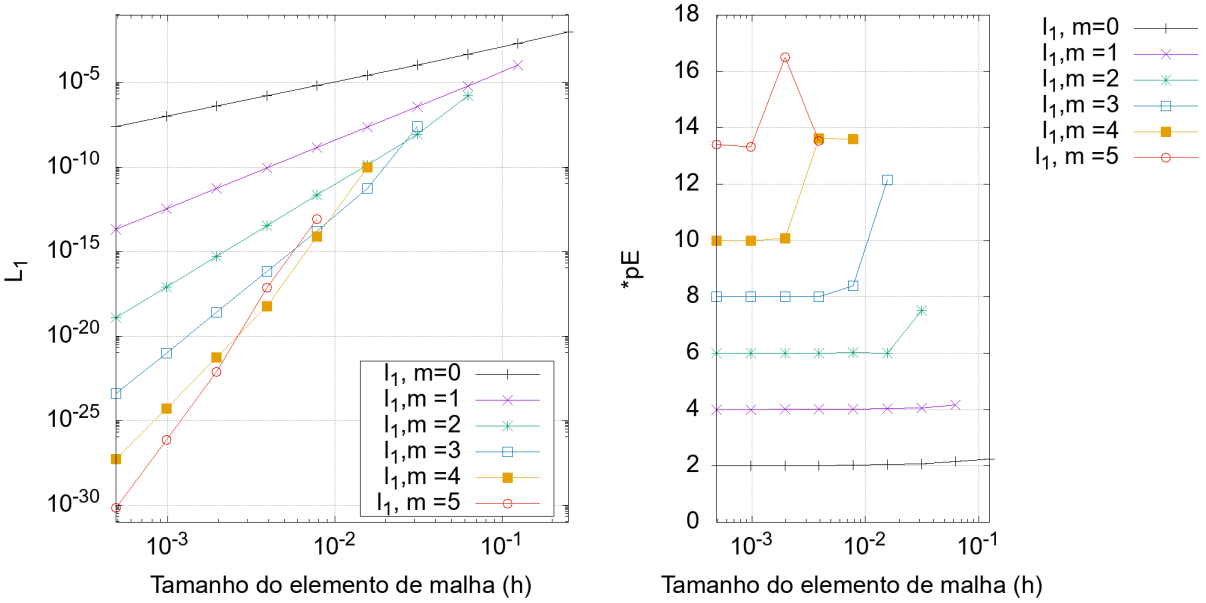
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 51 – Resultados: CRE-I, Variável l_1, Laplace 2D, Malha uniforme

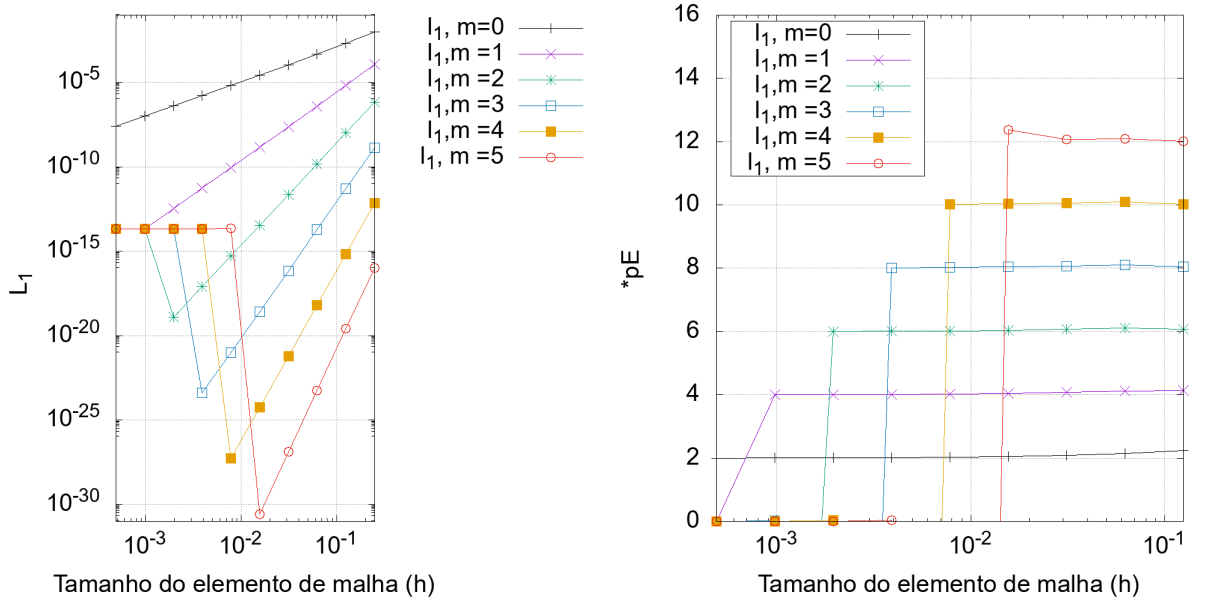
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 52 – Resultados: SER, Variável l_1, Laplace 2D, Malha uniforme

(a) l_1 (b) p_E^*



FONTE: Autor(2020).

6.2 MALHA NÃO UNIFORME

Para o caso da malha não uniforme foram obtidas 8 malhas, geradas com refino uniforme a partir da malha inicial conforme explicado na Seção 3.6. Em relação ao tempo de CPU, para este modelo não foi empregado Multigrid, sendo necessário aproximadamente dois dias para obter as 8 malhas.

Novamente para aplicar as técnicas de pós-processamento, foi verificada a acurácia das soluções através de estimativas a posteriori. As malhas obtidas para os experimentos estão com suas ordens dentro do esperado.

6.2.1 SMER

A quantidade inferior de malhas disponíveis e complexidade do modelo afetaram o desempenho de SMER. No entanto, houve redução do erro novamente, mas com menor escala em comparação para o caso uniforme. As ordens de acurácia apresentam melhora, mas também em menor escala.

Estes resultados são ilustrados qualitativamente pelas figuras 53 e 54. Através da Figura 54 é possível observar que o menor módulo do erro alcançado para l_1 foi aproximadamente 10^{-11}, bem abaixo dos resultados obtidos no caso uniforme que obteve seu erro em módulo valores abaixo de 10^{-20}.

Porém, novamente o tempo computacional para operar SMER foi baixo, questão de segundos. Neste cenário, em particular, foram obtidas soluções numéricas com maior acurácia com grande diferença de tempo em relação ao tempo levado para obter as soluções numéricas utilizadas nos experimentos.

6.2.2 SMER INTERPOLADO

Os conjuntos analisados são os mesmos do caso uniforme. SMER INTERPOLADO (SMER-I) já havia apresentando desempenho inferior no caso unidimensional, e neste cenário bidimensional se mostrou mais instável.

Este resultado é ilustrado pela Figura 55, para o conjunto β_2. Porém, como pode ser observado houve redução do erro para l_1, é possível observar também que SM-I apresenta instabilidade inicialmente, o que certamente afeta SMER-I. Neste cenário, as condições não são ideais para aplicar SMER-I, porque possivelmente l_1 está afetado por alguma outra fonte de erro, diferente de Eh. Neste caso, teoricamente MER não é aplicável, técnica base de SMER-I.

Portanto, em resumo novamente foram constatadas dificuldades com a troca da discretização. De modo semelhante ao caso unidimensional, SMER-I obteve desempenho

inferior na malha não uniforme. Em relação ao tempo e custo computacional, o mesmo raciocínio abordado no caso uniforme se aplica neste caso.

6.2.3 CRE clássico e com interpolação

De forma parecida a SMER-I, CRE com interpolação (CRE-I) apresentou instabilidade. Novamente obtendo desempenho muito inferior ao caso uniforme, não conseguindo progredir após a primeira extrapolação. Suas ordens de acurácia apresentaram instabilidade, porém houve redução do erro.

CRE clássico (CRE) novamente se mostrou estável, cumprindo com a sua proposta de progredir ao menos uma ordem de acurácia do método, com redução do erro.

Neste cenário, CRE obteve desempenho similar a CRE-I, apresentando maior estabilidade em suas ordens de acurácia. Estes resultados são apresentados pelas figuras 56 e 57.

Analisado o tempo e custo computacional, não existe diferença para o caso anterior. Neste âmbito em particular é até mais rápido, porque só são utilizadas 8 malhas.

6.2.4 SER

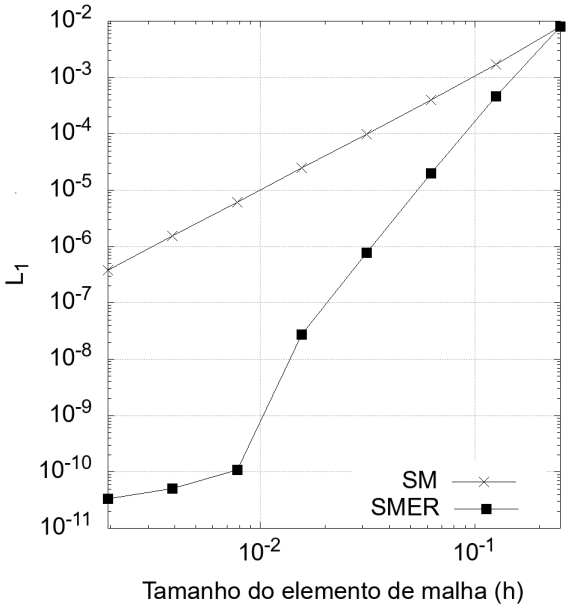
Do mesmo modo que SMER, a quantidade inferior de malhas disponíveis e complexidade do modelo afetaram o desempenho de SER. No entanto, houve redução do erro novamente, mas em menor escala em comparação para o caso uniforme. As ordens de acurácia apresentam melhora, mas apresentaram comportamento atípico.

Neste cenário, o termo "atípico" faz referência as extrapolações (m), isto é, embora para $m > 1$ a princípio o erro de l_1 apresenta evolução. Porém, isto não se reflete nas ordens de acurácia, já apresentando instabilidade para $m = 2$.

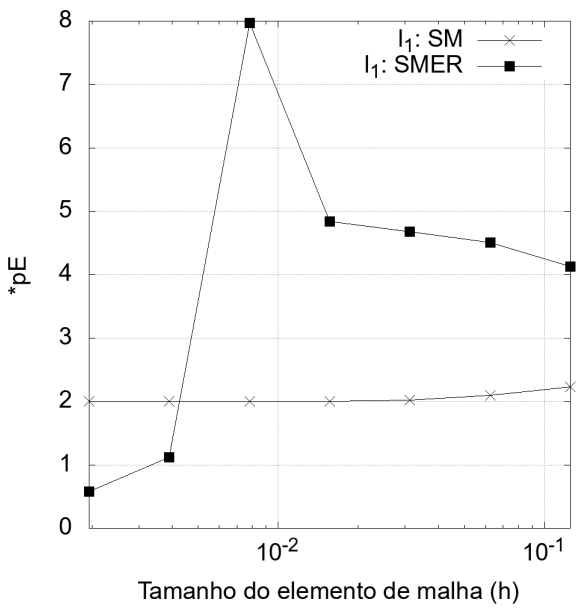
Ressaltando, novamente é constatada a "perda" da malha mais fina a cada extrapolação. Estes resultados comentados anteriormente são ilustrados pela Figura 58. Analisado o tempo e custo computacional é semelhante em relação ao caso anterior, a técnica opera em segundos.

Figura 53 – Resultados: SMER, Malhas 1 e 4, Variável l_1, Laplace 2D, Malha não uniforme

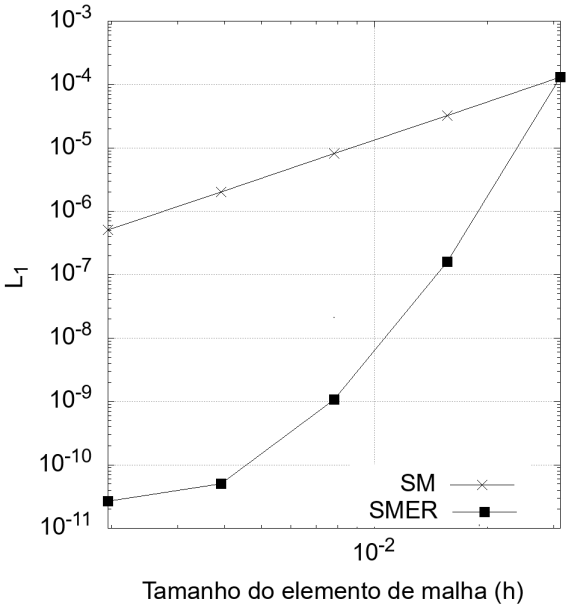
(a) l_1, Malha 1



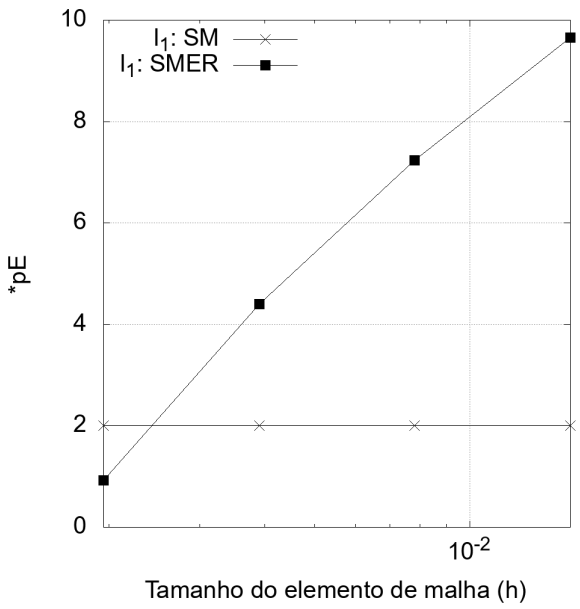
(b) p_E^*, Malha 1



(c) l_1, Malha 4

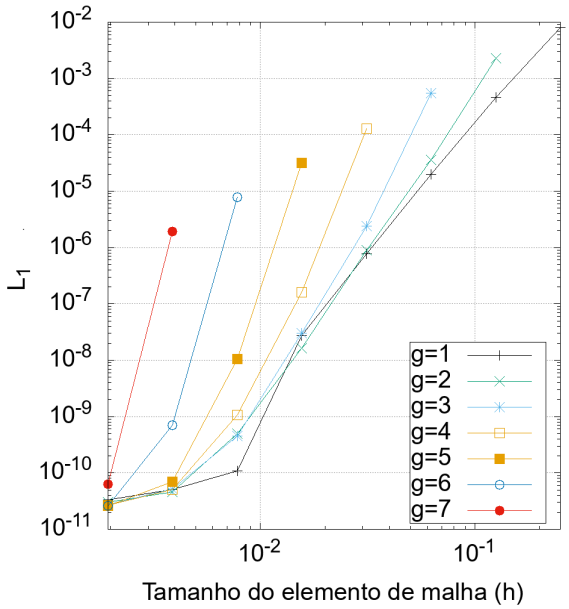


(d) p_E^*, Malha 4



FONTE: Autor(2020).

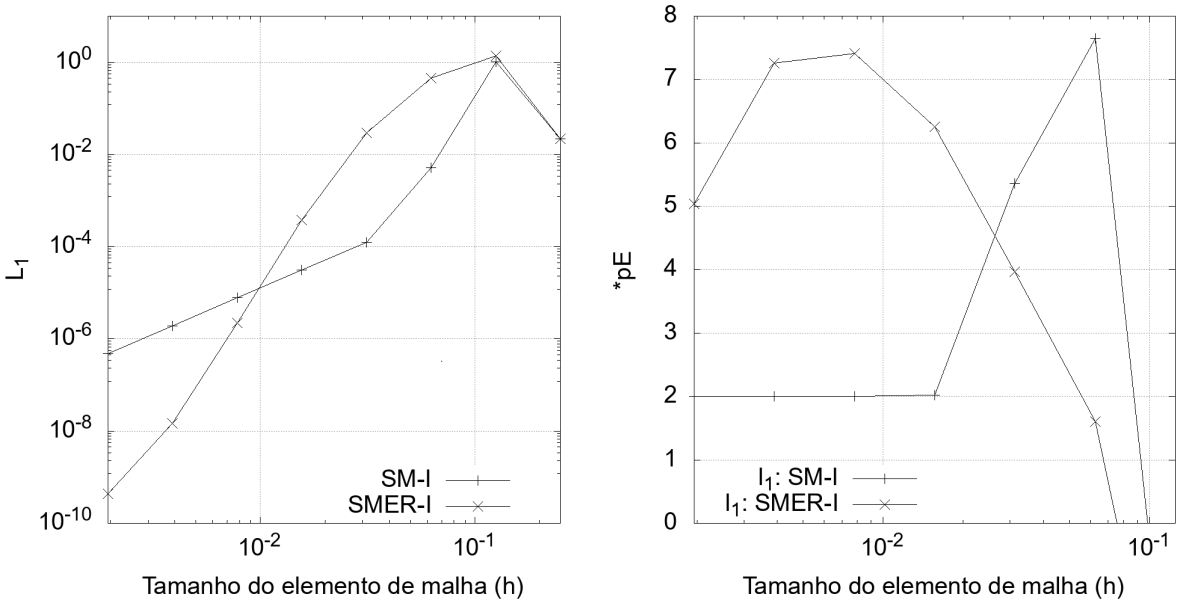
Figura 54 – Resultados: SMER, Malhas $g = 1,2,\cdots,9$, Variável l_1, Laplace 2D, Malha não uniforme



FONTE: Autor(2020).

Figura 55 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Laplace 2D, Malha não uniforme

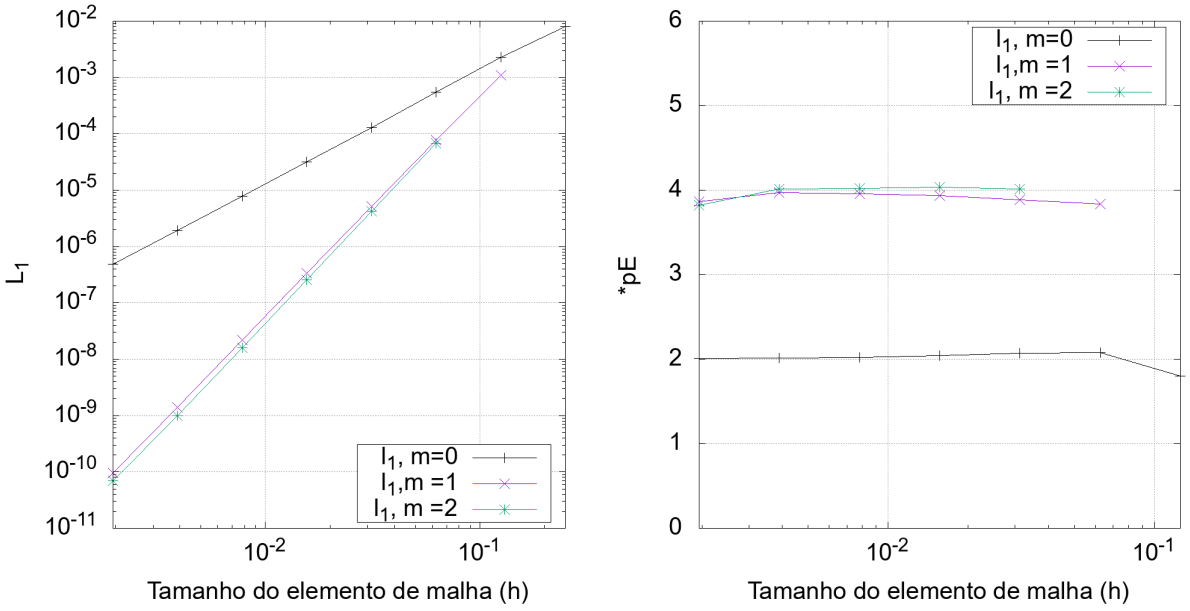
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 56 – Resultados: CRE, Variável l_1, Laplace 2D, Malha não uniforme

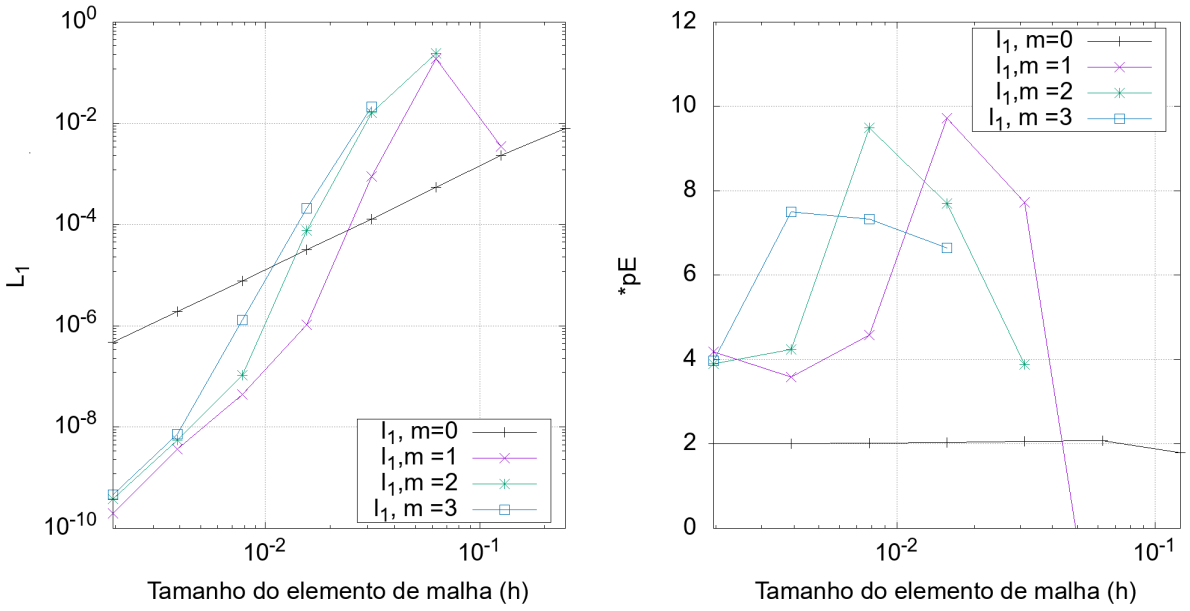
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 57 – Resultados: CRE-I, Variável l_1, Laplace 2D, Malha não uniforme

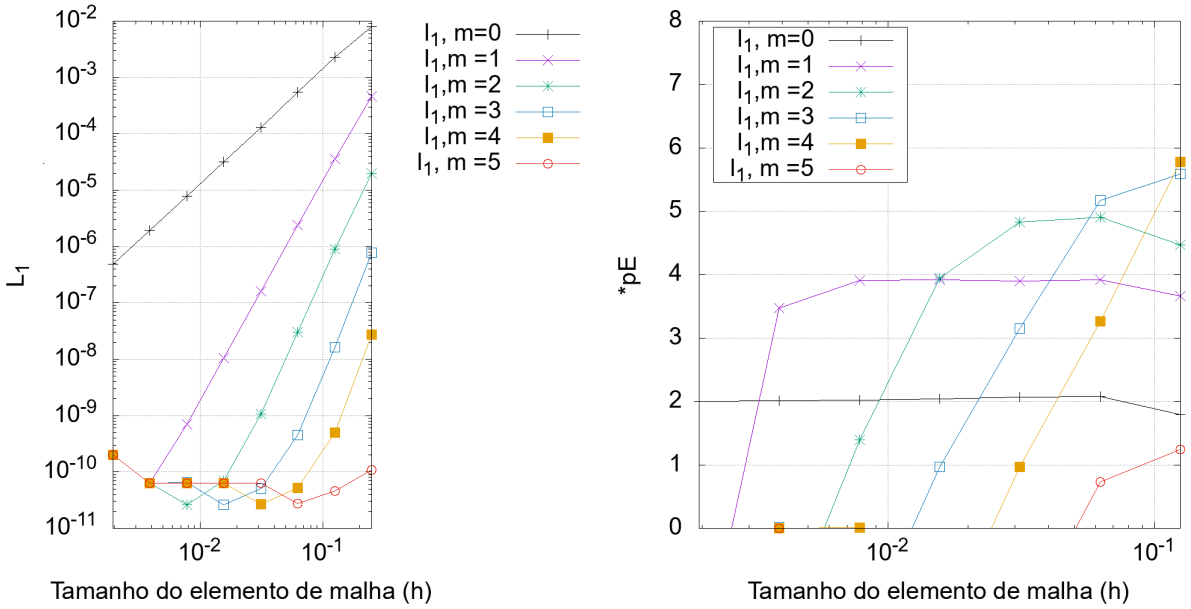
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 58 – Resultados: SER, Variável l_1, Laplace 2D, Malha não uniforme

(a) l_1 (b) p_E^*



FONTE: Autor(2020).

6.3 Resumo do capítulo e conclusões

Ficaram constatadas diferenças notórias entre os resultados obtidos para cada caso. Resultado já observado nos capítulos anteriores.

Nas malhas uniformes CRE-I, SMER, SMER-I e SER alcançaram excelentes resultados. Atingindo erros em módulo inferiores a 10^{-20} para l_1, e aumento considerável nas ordens de acurácia, tudo isso sem a necessidade da realização de novas simulações, somente com pós-processamento de dados.

No entanto, trocando para o caso não uniforme em termos de diminuição do erro de l_1, elas se equiparam a CRE. Isso porque todas as técnicas atingiram o erro em módulo aproximado em 10^{-11} para l_1. CRE se mostrou estável em ambos os cenários, cumprindo com a sua proposta de progredir ao menos uma ordem de acurácia com redução do erro.

As interpolações provavelmente afetaram o desempenho de SMER-I e CRE-I. Possivelmente com o acesso a um número maior de malhas, SER e SMER poderiam ter obtido resultados melhores. Elas se mostraram estáveis, mas não conseguiram reduzir o erro com a mesma eficácia do caso uniforme.

Os resultados de l_∞ são apresentados nos apêndices de forma resumida, para trabalhos futuros podem ajudar no entendimento do que aconteceu no caso não uniforme, porque também apresentaram divergências.

7 RESULTADOS: EQUAÇÃO DE BURGERS 2D

Neste capítulo são apresentados os resultados para a equação de Burgers 2D com $Re = 1$, discretizada com malhas uniformes. Na Figura 59 é apresentado o gráfico das velocidades u e v com suas soluções numéricas e analíticas.

Foram geradas 10 malhas para este modelo, cuja malha mais grossa foi apresentada na Seção 3.6 e as malhas restantes foram obtidas através do seu refinamento. Em relação ao tempo de CPU, graças ao Multigrid foi necessário apenas aproximadamente 18 minutos, para obter as 10 malhas.

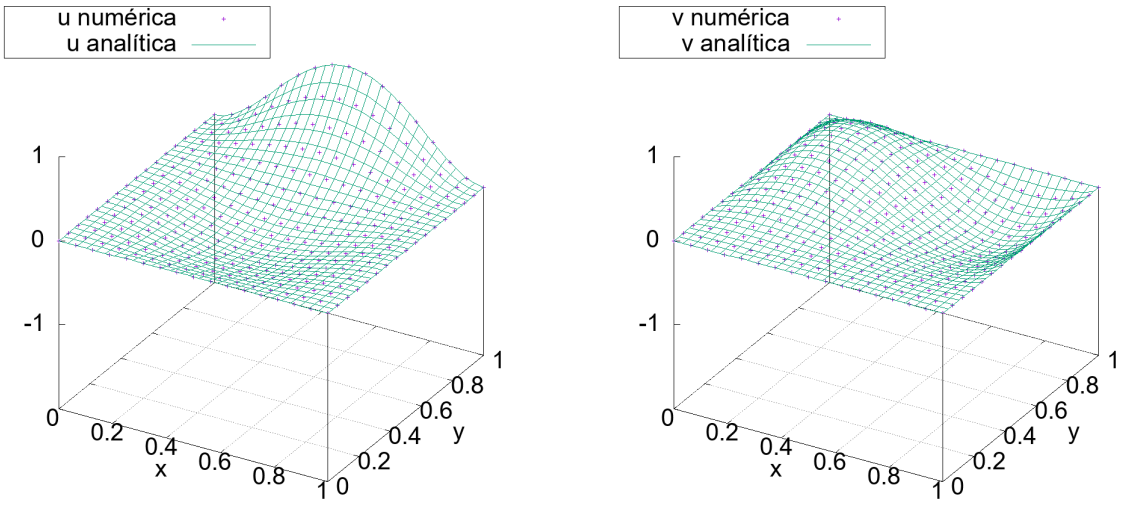
Novamente para aplicar as técnicas de pós-processamento, foi verificada a acurácia das soluções através de estimativas a posteriori. As malhas obtidas para os experimentos estão com suas ordens dentro do esperado.

Mais uma vez são apresentados apenas alguns resultados de l_1, para ilustrar as soluções obtidas. No Apêndice C, são apresentados alguns resultados para l_∞. Todos os resultados podem ser vistos no link:

http://ftp.demec.ufpr.br/CFD/monografias/2020_Carlos-Carvalho_doutorado/

Figura 59 – $u(x,y)$, $v(x,y)$: Numérica vs Analítica, Burgers 2D, $Re = 1$

(a) Velocidade u (b) Velocidade v



FONTE: Autor(2020).

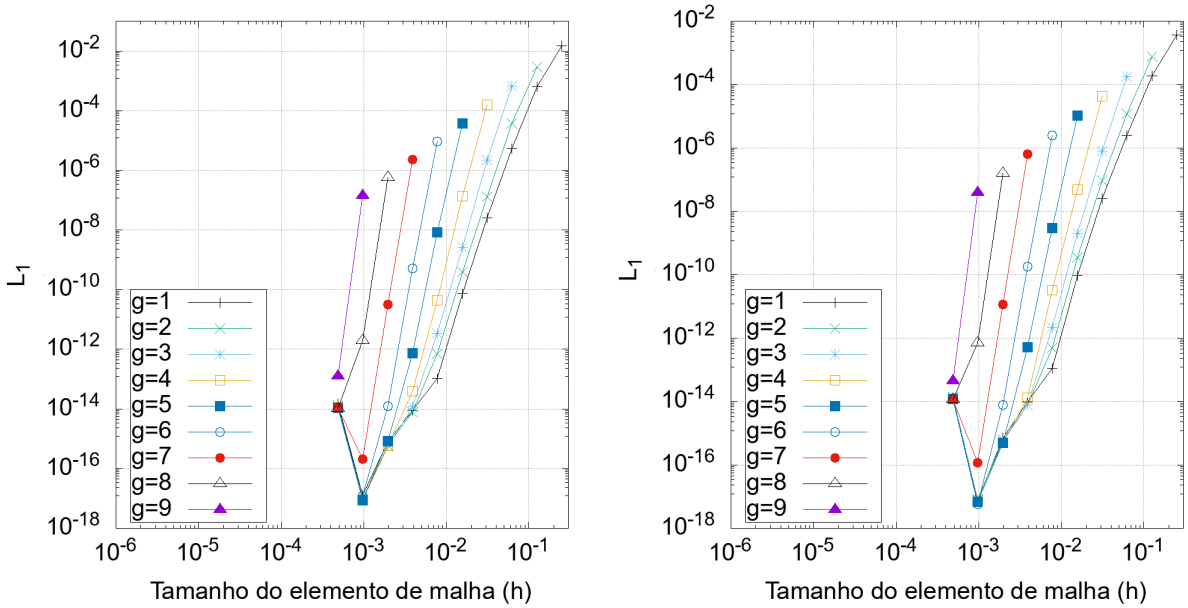
7.1 SMER

Para ambas as velocidades u e v, foram obtidos resultados semelhantes, houve significativa redução do erro com concomitante aumento das ordens de acurácia. Novamente, pode ser observado que SMER beneficia as malhas mais grossas.

Estes resultados são ilustrados de forma qualitativa pelas figuras 60, 61 e 62. Novamente SMER se mostrou estável, em um cenário com mais malhas disponíveis provavelmente haveria mais redução do erro e aumento das ordens de acurácia.

Figura 60 – Resultados: SMER, Variável ł$_1$, Malhas $g = 1,2,\cdots,9$, Burgers 2D, $Re = 1$

(a) Velocidade u (b) Velocidade v



FONTE: Autor(2020).

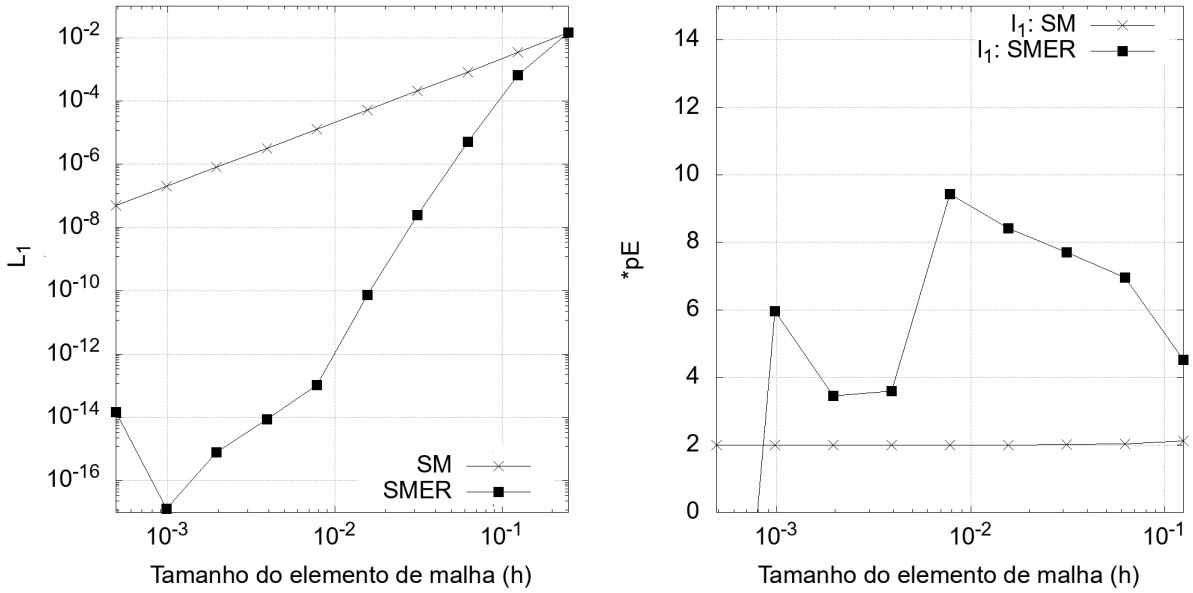
7.2 SMER INTERPOLADO

Para ambas as velocidades u e v foram obtidos resultados semelhantes, houve significativa redução do erro e aumento das ordens de acurácia. Estes resultados são ilustrados pelas figuras 63 e 64. Analisou-se o desempenho de SMER-I no conjunto β_2, o mesmo utilizado para Laplace 2D.

Novamente SMER-I se mostrou estável quando são utilizadas malhas uniformes, neste cenário ele se equipara com SMER.

Figura 61 – Resultados: SMER, Malha base 1, Variável l_1, Burgers 2D, $Re = 1$, velocidade u

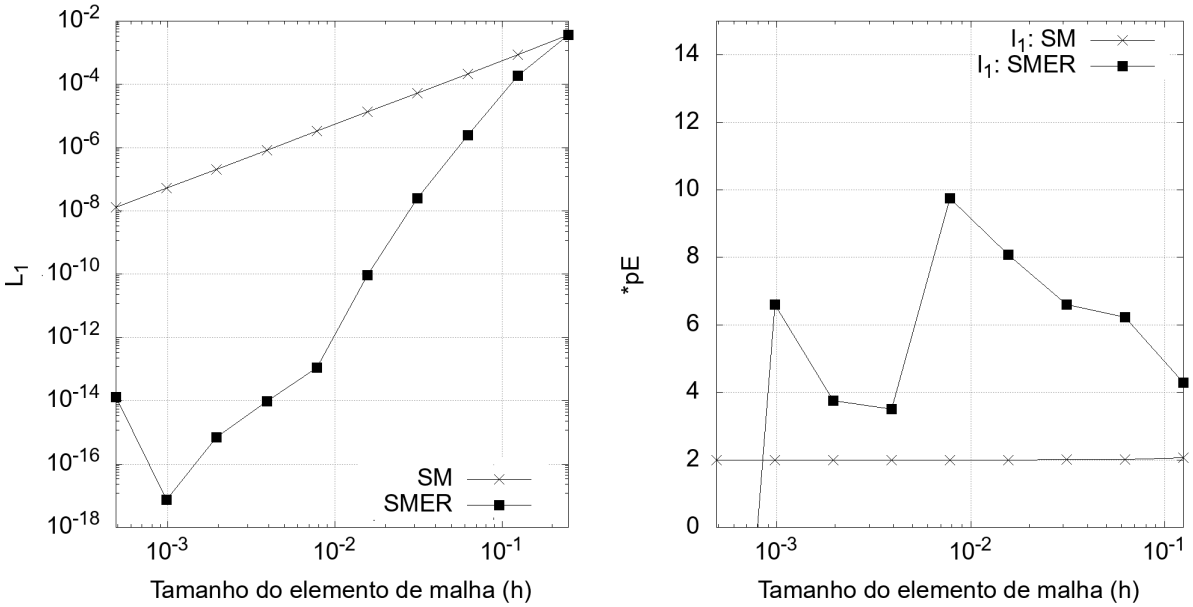
(a) l_1, Malha 1 (b) p_E^*, Malha 1



FONTE: Autor(2020).

Figura 62 – Resultados: SMER, Malha base 1, Variável l_1, Burgers 2D, $Re = 1$, velocidade v

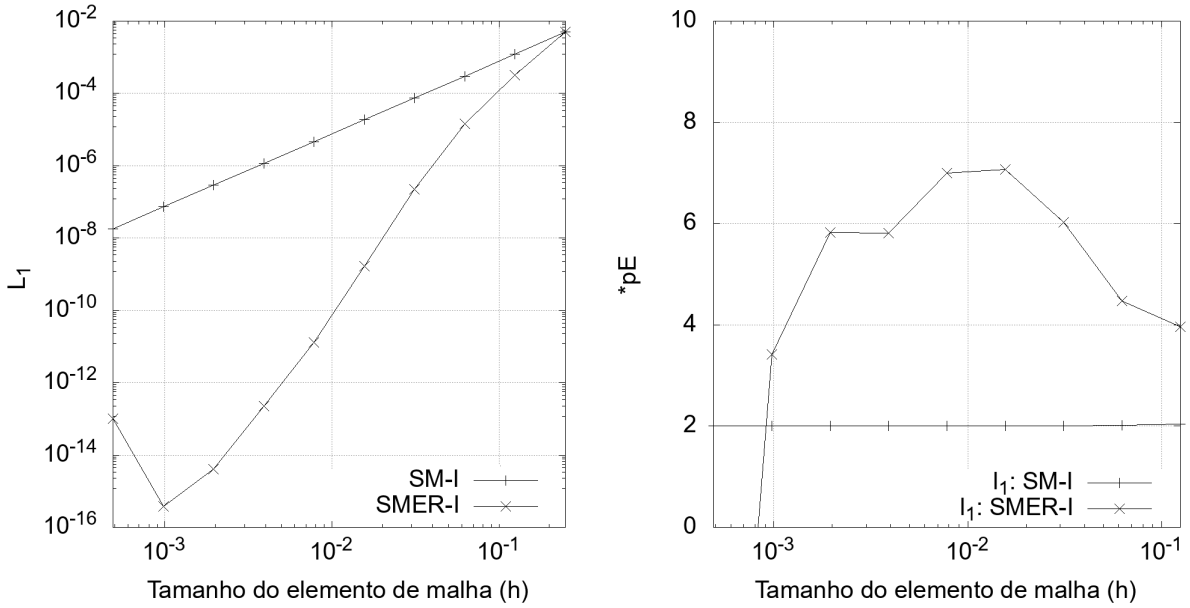
(a) l_1, Malha 1 (b) p_E^*, Malha 1



FONTE: Autor(2020).

Figura 63 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Burgers 2D, $Re = 1$, velocidade u

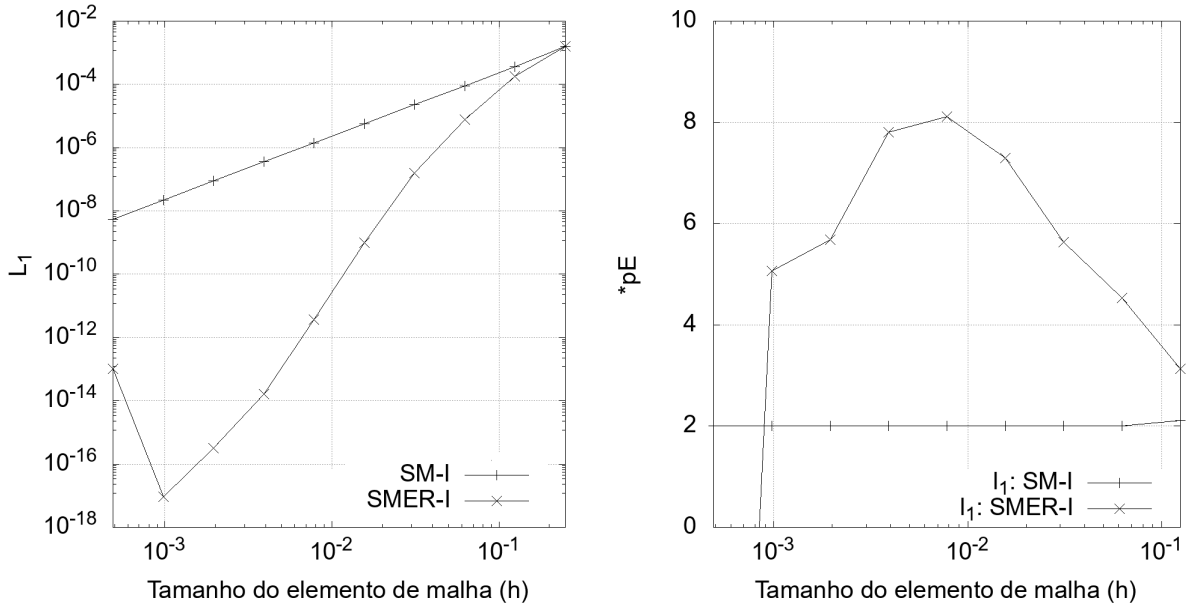
(a) l_1 (b) p^*_E



FONTE: Autor(2020).

Figura 64 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Burgers 2D, $Re = 1$, velocidade v

(a) l_1 (b) p^*_E



FONTE: Autor(2020).

7.3 CRE clássico e com interpolação

CRE clássico (CRE) novamente se mostrou estável, cumprindo com a sua proposta de progredir ao menos uma ordem de acurácia do método, com redução do erro para l_1, para ambas as velocidades.

CRE com interpolação (CRE-I) conseguiu reduzir o erro com mais efetividade e obteve maiores ordens de acurácia. Neste cenário se equiparou a SMER e SMER-I. Do mesmo que SMER-I, para malhas uniformes CRE-I novamente apresentou estabilidade. Estes resultados estão ilustrados pelas Figuras 65, 66, 67 e 68.

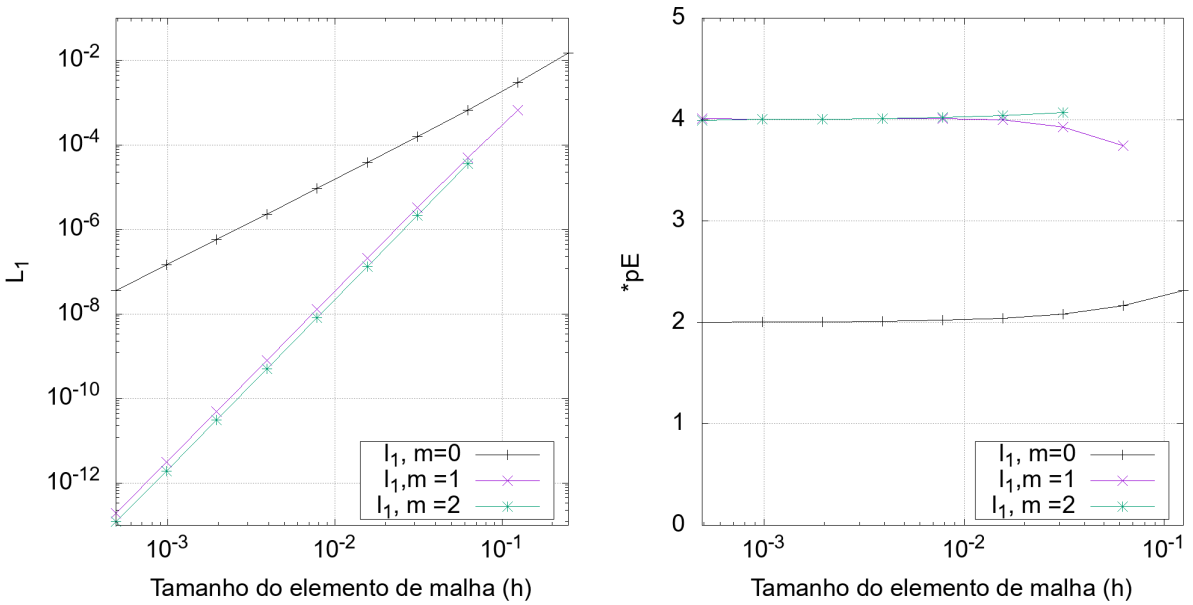
7.4 SER

SER apresentou desempenho similar, comparando com os resultados obtidos com CRE-I, SMER e SMER-I. Novamente como SMER, apresentou mais uma vez estabilidade independente do modelo ou tipo de malha utilizados.

Os resultados para variável l_1 são ilustrados pelas figuras 69 e 70.

Figura 65 – Resultados: CRE, Variável l_1, Burgers 2D, $Re = 1$, velocidade u

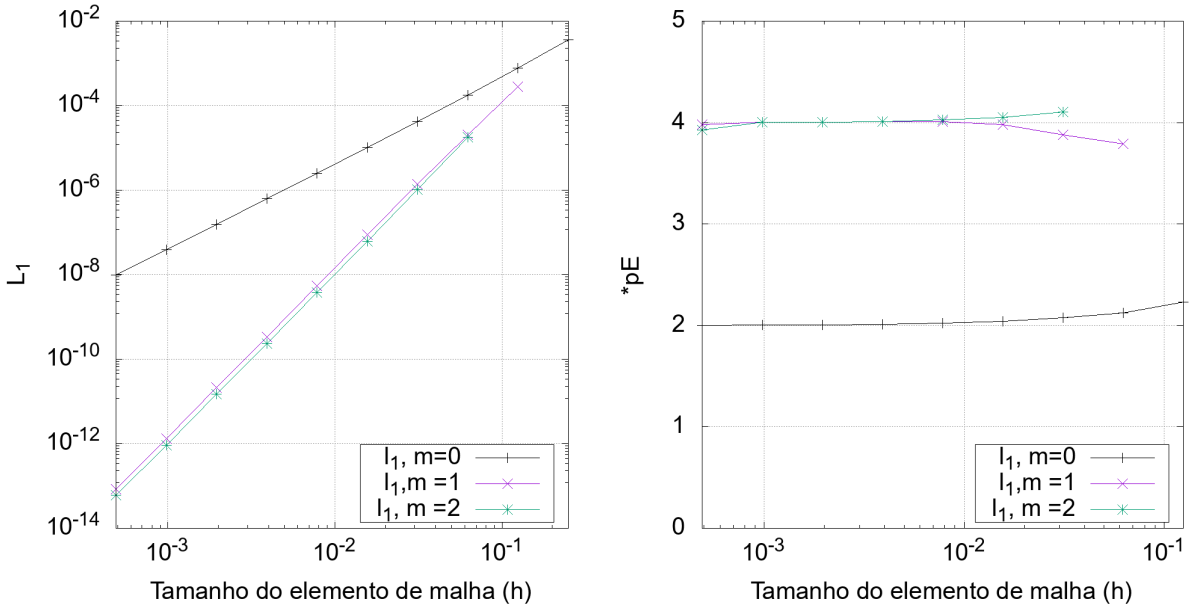
(a) l_1 (b) p^*_E



FONTE: Autor(2020).

Figura 66 – Resultados: CRE, Variável l_1, Burgers 2D, $Re = 1$, velocidade v

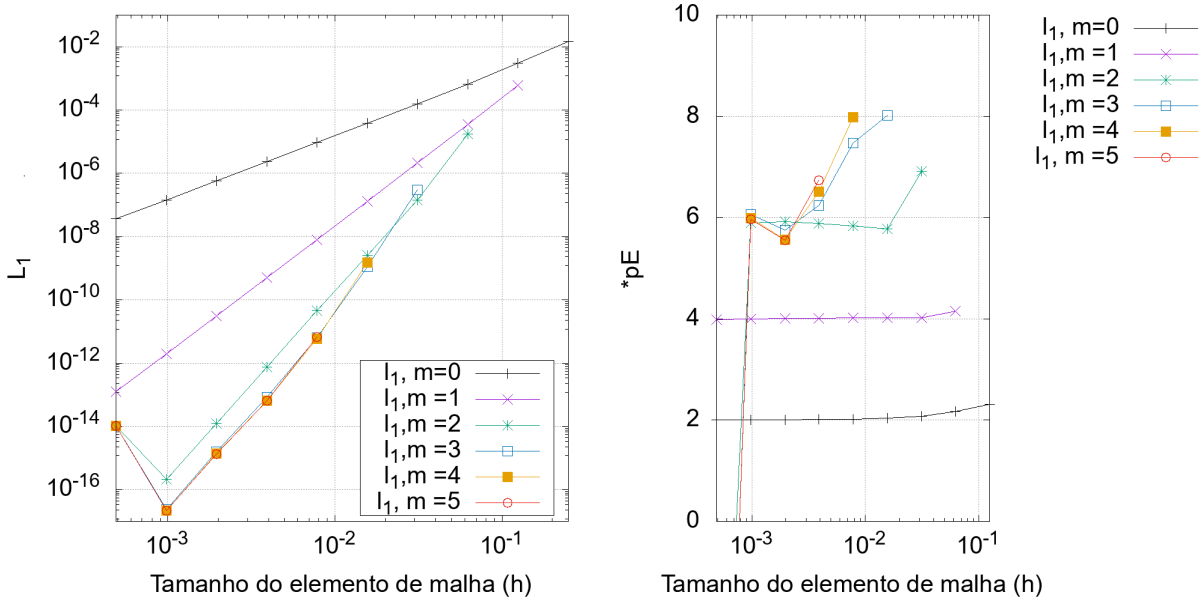
(a) l_1 (b) p^*_E



FONTE: Autor(2020).

Figura 67 – Resultados: CRE-I, Variável l_1, Burgers 2D, $Re = 1$, velocidade u

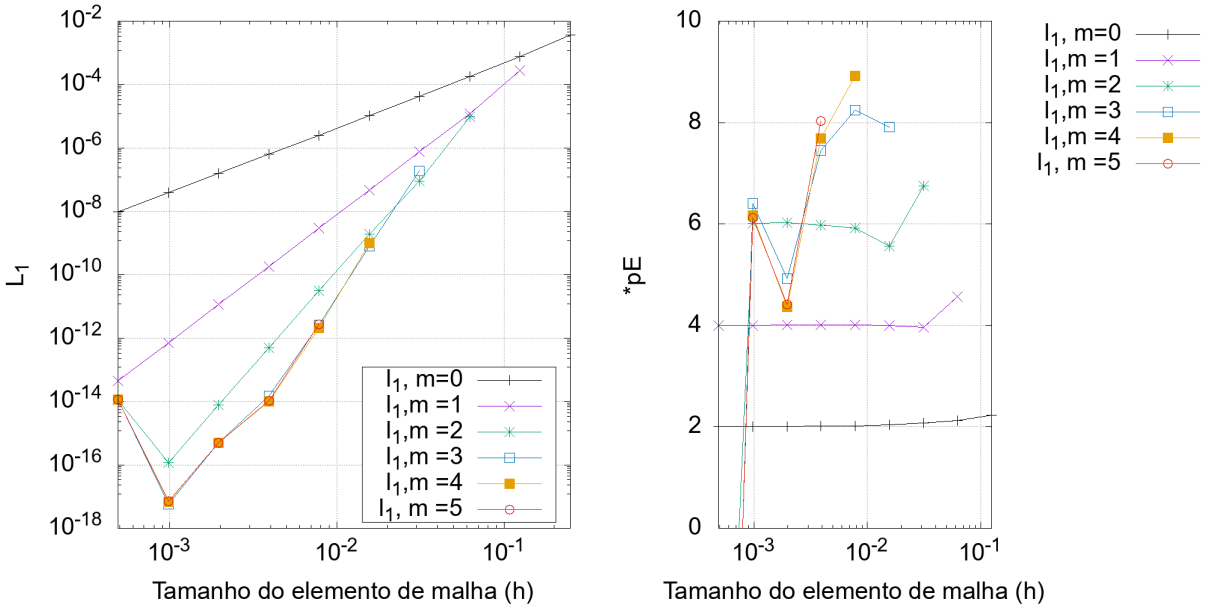
(a) l_1 (b) p^*_E



FONTE: Autor(2020).

Figura 68 – Resultados: CRE-I, Variável l_1, Burgers 2D, $Re = 1$, velocidade v

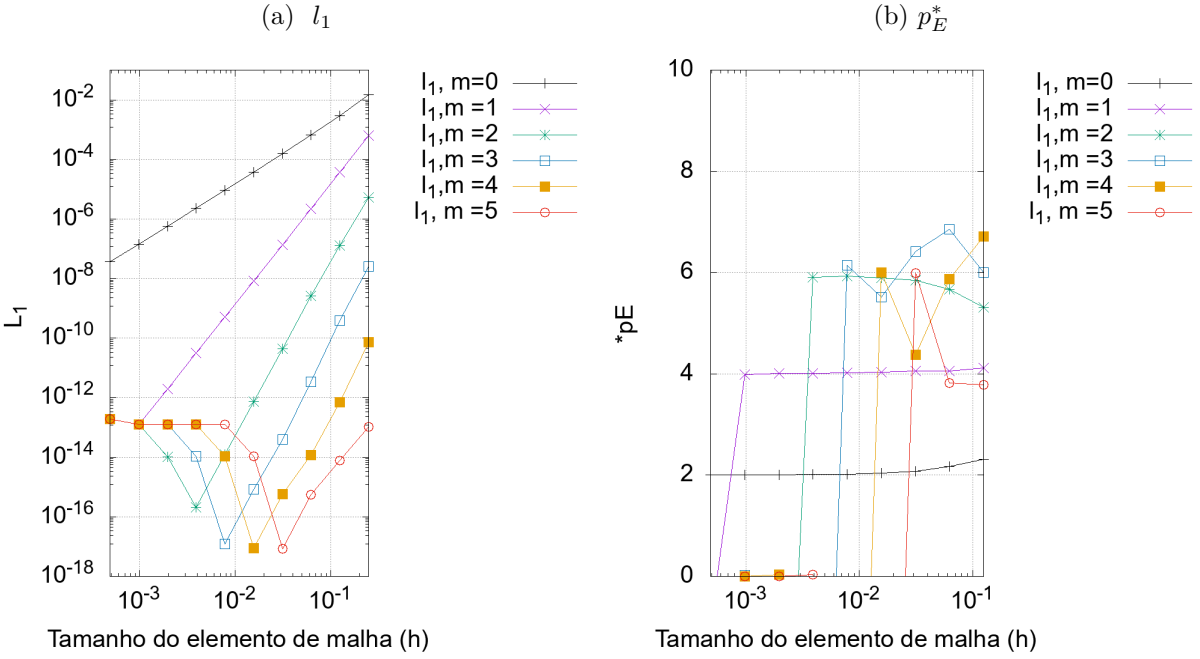
(a) l_1 (b) p^*_E



FONTE: Autor(2020).

Figura 69 – Resultados: SER, Variável l_1, Burgers 2D, $Re = 1$, velocidade u

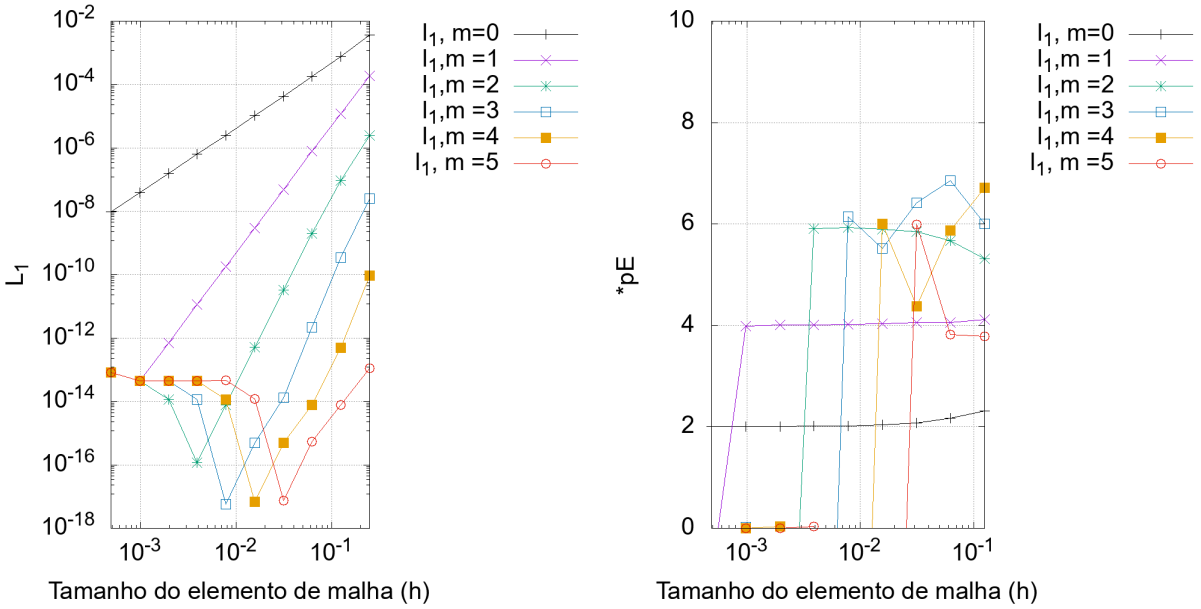
(a) l_1 (b) p^*_E



FONTE: Autor(2020).

Figura 70 – Resultados: SER, Variável l_1, Burgers 2D, $Re = 1$, velocidade v

(a) l_1 (b) p_E^*



FONTE: Autor(2020).

7.5 Resumo do capítulo e conclusões

Os resultados obtidos para este modelo se equiparam qualitativamente aos apresentados no Capítulo 4.

CRE-I, SMER, SER, SMER-I apresentaram desempenho similar, com leve vantagem para CRE-I porque consegue extrapolar as malhas mais finas. A última citada, em particular, é a mais custosa operacionalmente dentre todas para funcionar, mas é a mais completa, por conseguir extrapolar qualquer malha e conjunto de pontos dentro do domínio do cálculo. No entanto, ela não deve ser priorizada, caso alguma das outras técnicas seja aplicável no problema.

CRE apresentou novamente desempenho estável, dentro da sua proposta: conseguir progredir uma ordem verdadeira do método de $p_0 = 2$ para $p_1 = 4$ com redução do erro. As outras técnicas conseguiram ordens superiores a 4 e uma maior redução do erro. Mas CRE cumpriu com a sua proposta em todos os modelos utilizados até agora.

Alguns resultados para l_∞ podem ser vistos nos apêndices, em resumo foram obtidos resultados similares com os apresentados de l_1.

8 RESULTADOS - NAVIER-STOKES

Neste capítulo são apresentados os resultados para as equações de NAVIER-STOKES, discretizada com malhas uniformes. Na Figura 71 é apresentado o gráfico das velocidades u e v com suas soluções numéricas e analíticas.

Foram geradas nove malhas para este modelo, cuja malha mais grossa foi apresentada na Seção 3.6, as restantes foram obtidas através do seu refinamento. Em relação ao tempo de CPU, graças ao Multigrid foi necessário apenas dez minutos aproximadamente, para obter as nove malhas.

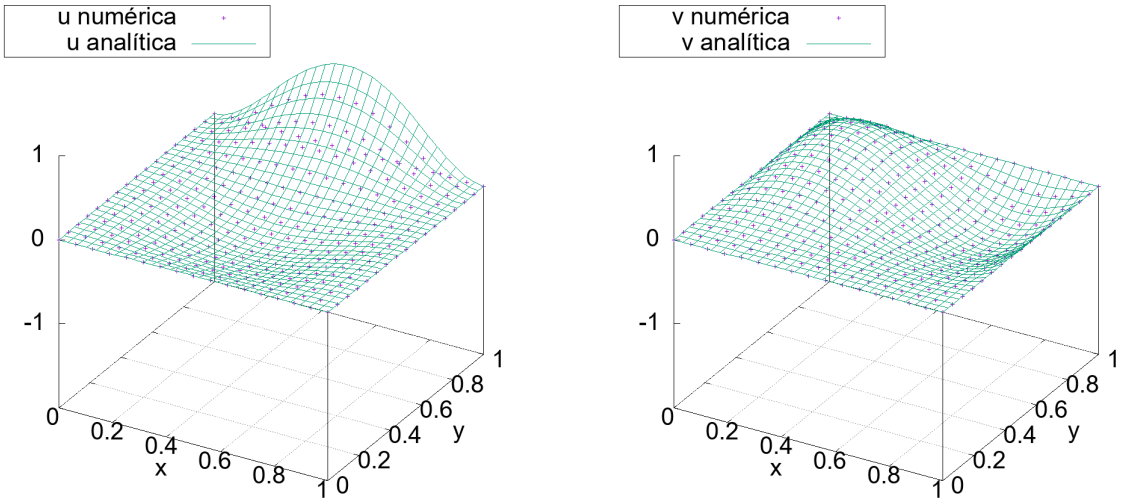
Novamente para aplicar as técnicas de pós-processamento, foi verificada a acurácia das soluções através de estimativas a posteriori. As malhas obtidas para os experimentos estão com suas ordens dentro do esperado.

Mais uma vez, são apresentados apenas alguns resultados de l_1, para ilustrar as soluções obtidas. No Apêndice C, são apresentados alguns resultados para l_∞. Todos os resultados podem ser vistos no link:

http://ftp.demec.ufpr.br/CFD/monografias/2020_Carlos-Carvalho_doutorado/

Figura 71 – $u(x,y)$, $v(x,y)$: Numérica vs Analítica, Navier-Stokes 2D, $Re = 1$

(a) Velocidade u. (b) Velocidade v.



FONTE: Autor(2020).

8.1 SMER

SMER apresentou ótimo desempenho para ambas as velocidades, alcançando nas malhas mais grossas erro em módulo inferiores a 10^{-10}, consideravelmente superior ao erro inicial. Resultados semelhantes aos já apresentados anteriormente.

Em todas as malhas em que SMER é aplicável, houve redução do erro e aumento nas ordens de acurácia. Com aproveitamento em menor escala para as malhas mais finas, e maior para as mais grossas. Conforme já observado anteriormente em outros experimentos. Estes resultados estão ilustrados de forma qualitativa nas figuras 72, 73 e 74.

8.2 SMER INTERPOLADO

SMER INTERPOLADO (SMER-I) novamente apresentou desempenho abaixo do esperado, não convergindo para velocidade u e obtendo baixo desempenho para velocidade v, em comparação com SMER. Estes resultados são expostos pelas figuras 75 e 76.

Os motivos para não convergência possivelmente são os mesmos já mencionados no capítulo 5. Infelizmente, este é um resultado novo **negativo**, porque SMER-I estava convergindo nas malhas uniformes.

8.3 CRE clássico e com interpolação

CRE com interpolação (CRE-I) apresentou desempenho abaixo do esperado, apresentando degeneração das ordens verdadeiras para velocidades u e v. Pela primeira vez nos experimentos CRE não conseguiu progredir uma ordem de acurácia do método, também apresentando degeneração das ordens. Ambas as técnicas alcançaram acurácia 3, uma unidade abaixo do esperado ($p_1 = 4$), mas houve redução do erro.

Estes resultados são expostos nas figuras 77, 78 79 e 80. Neste cenário ambas as técnicas apresentaram desempenho inferior em comparação com SMER.

Um possível motivo para este acontecimento seria as interpolações realizadas nos pontos não coincidentes, como já comentado no capítulo 5. Infelizmente, novamente este é um resultado novo **negativo**, porque CRE-I e CRE estavam convergindo nas malhas uniformes.

8.4 SER

SER apresentou ótimo desempenho, apresentando aumento para ordens de acurácia para as velocidades u e v com redução erro significativa. Resultados semelhantes aos já apresentados anteriormente.

Estes resultados são expostos pelas figuras 81 e 82. Neste cenário SER se equiparou a SMER, atingindo erro em módulo inferiores a 10^{-15} e ordens de acurácias entre 4 e 7.

8.5 Resumo do capítulo e conclusões

Neste capítulo ocorreram novidades **negativas** em relação aos experimentos anteriores que utilizaram somente malhas uniformes.

CRE e CRE-I apresentaram degeneração das ordens, não esperado e é uma questão para ser investigada. Pela primeira vez nos experimentos, CRE não conseguiu progredir para a ordem de acurácia seguinte do método, ou seja, $p_1 = 4$.

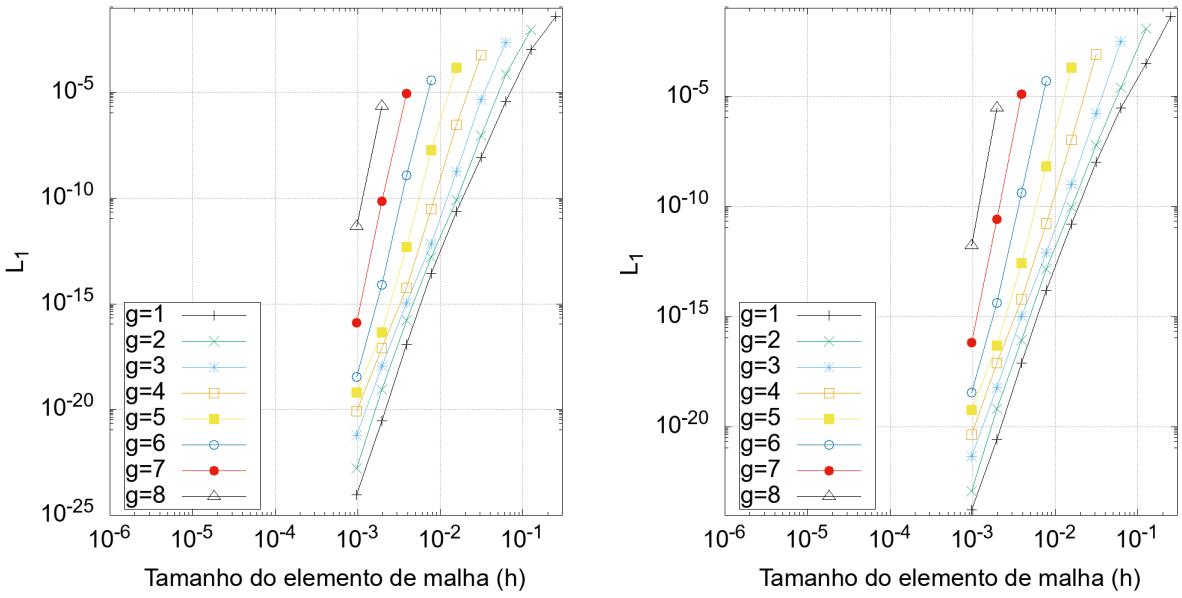
SMER-I apresentou os **piores resultados**, as suas soluções numéricas sem MER (**SM**) já não convergiram para as ordens estimadas a priori. Neste cenário já era esperado que MER não iria conseguir melhorar seus resultados, porque claramente os erros das soluções estão afetados não somente pelo erro de discretização e, portanto, já não é indicada a aplicação da técnica.

SER e SMER alcançaram os **melhores resultados**, apresentando grande superioridade em comparação com as técnicas listadas anteriormente. Novamente apresentando estabilidade.

Para as variáveis l_∞ foram obtidos resultados similares, mas um pouco inferiores. Talvez possa ajudar no entendimento do que aconteceu com CRE, CRE-I e SMER-I em trabalhos futuros. Seus resultados são expostos de forma qualitativa nos apêndices.

Figura 72 – Resultados: SMER, Variável ł$_1$, Malhas $g = 1,2,\cdots,8$, Navier-Stokes 2D

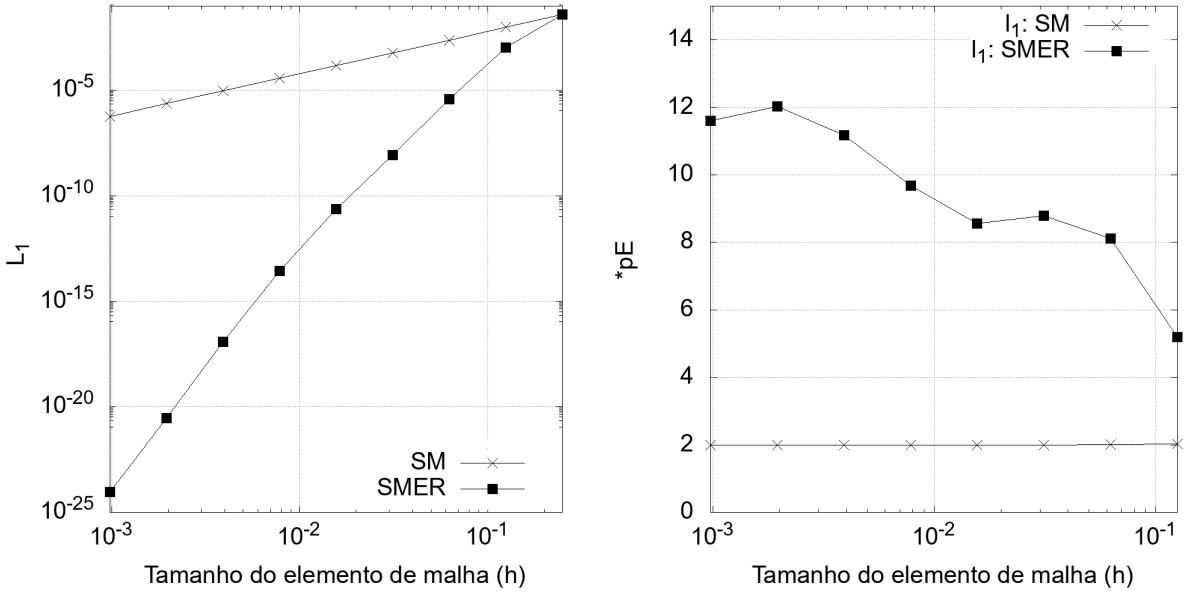
(a) Velocidade u (b) Velocidade v



FONTE: Autor(2020).

Figura 73 – Resultados: SMER, Malha base 1, Variável l_1, Navier-Stokes 2D, velocidade u

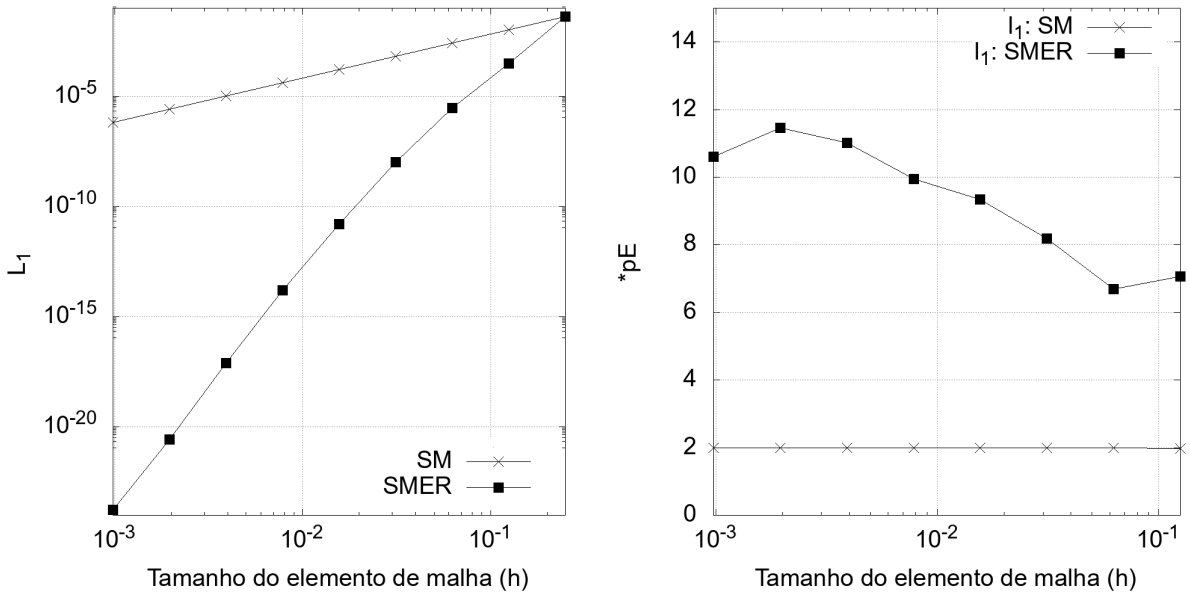
(a) l_1, Malha 1 (b) p^*_E, Malha 1



FONTE: Autor(2020).

Figura 74 – Resultados: SMER, Malha base 1, Variável l_1, Navier-Stokes 2D, velocidade v

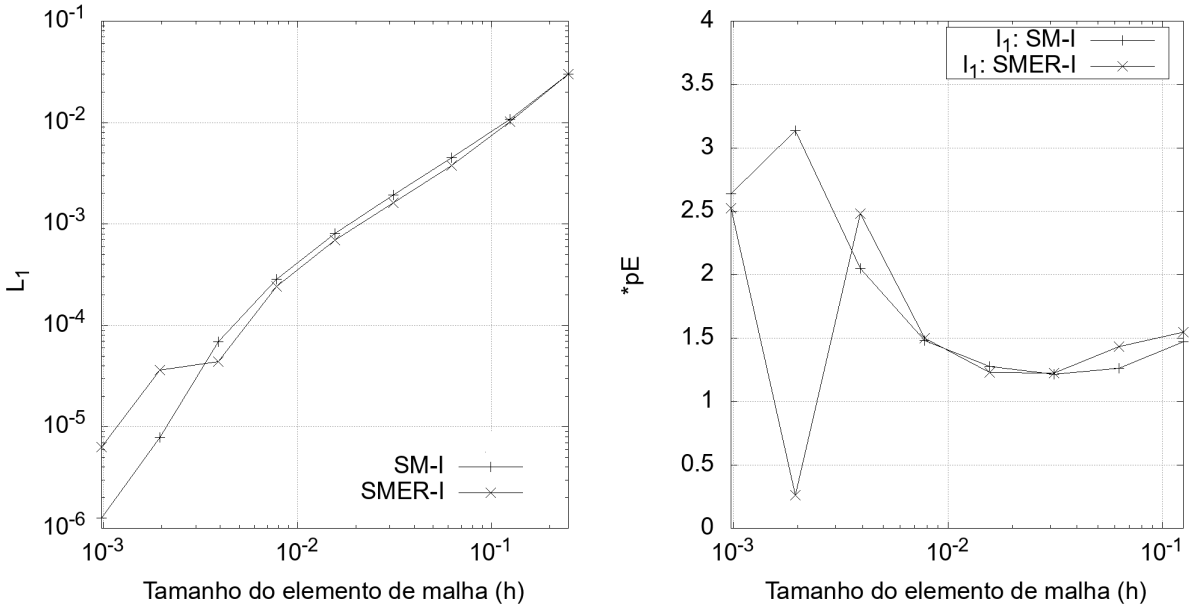
(a) l_1, Malha 1 (b) p^*_E, Malha 1



FONTE: Autor(2020).

Figura 75 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Navier-Stokes 2D, velocidade u

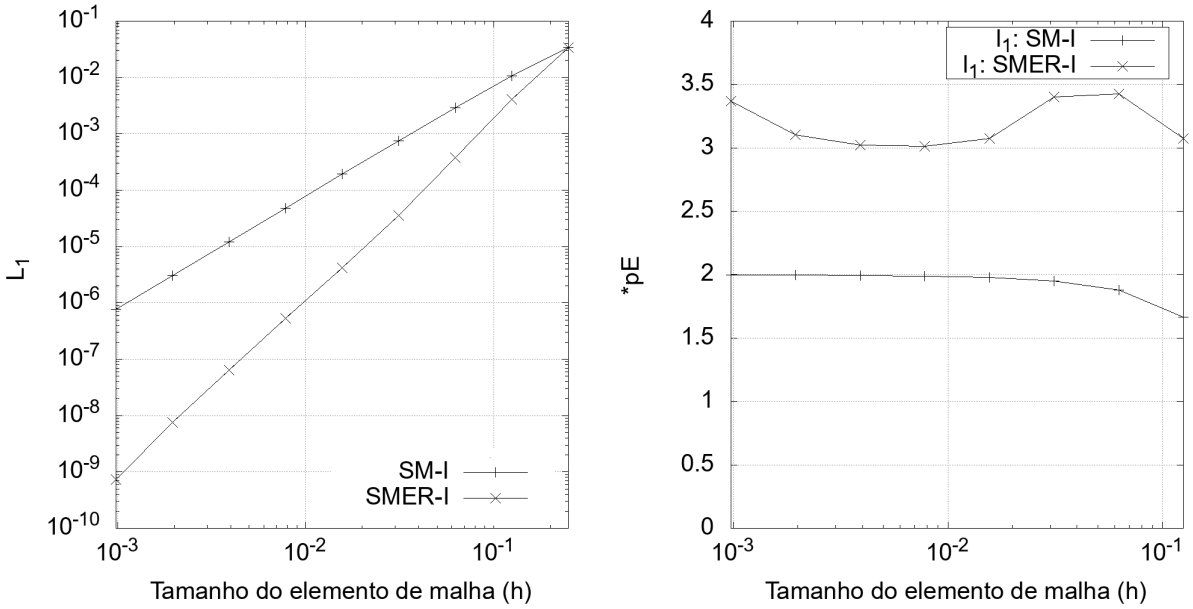
(a) l_1 (b) p^*_E



FONTE: Autor(2020).

Figura 76 – Resultados: SMER-I, Conjunto β_2, Variável l_1, Navier-Stokes 2D, velocidade v

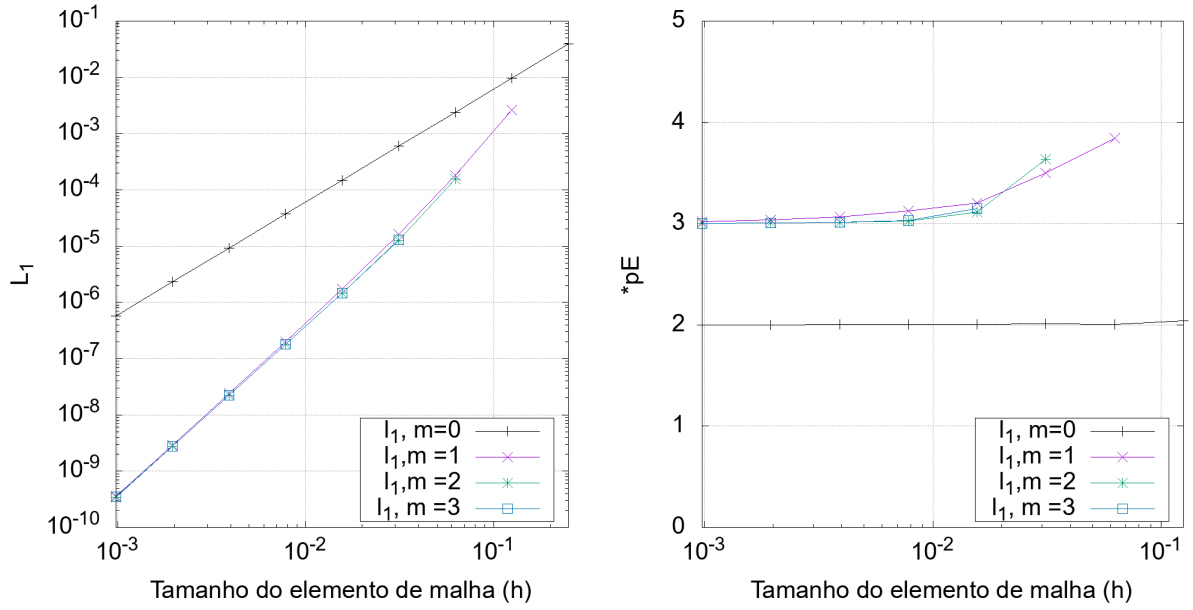
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 77 – Resultados: CRE-I, Variável l_1, Navier-Stokes 2D, velocidade u

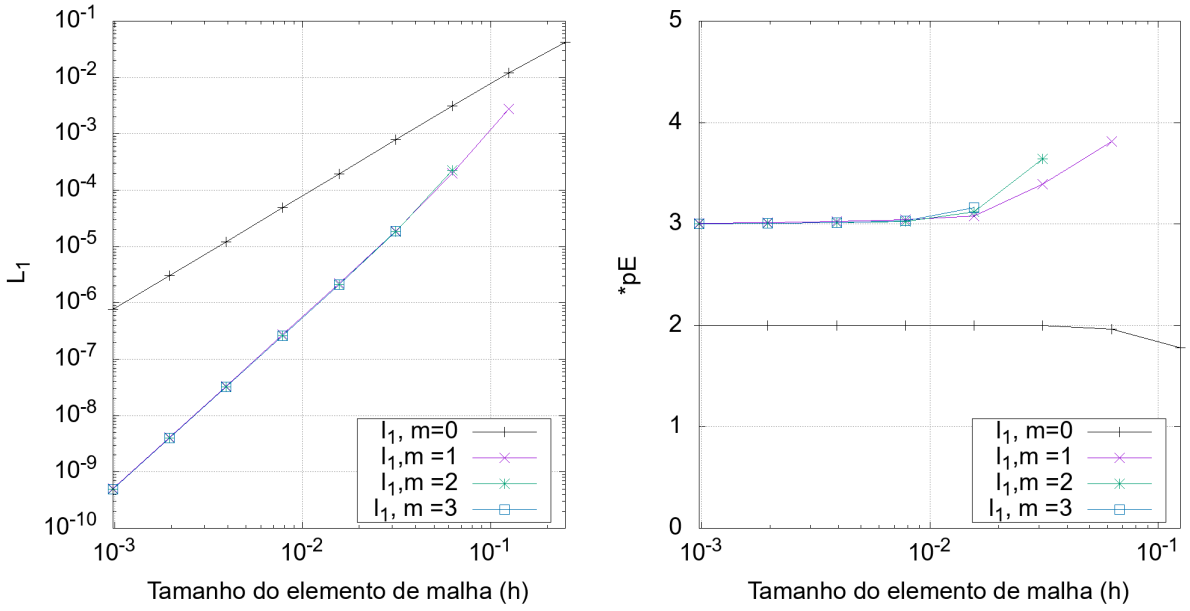
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 78 – Resultados: CRE-I, Variável l_1, Navier-Stokes 2D, velocidade v

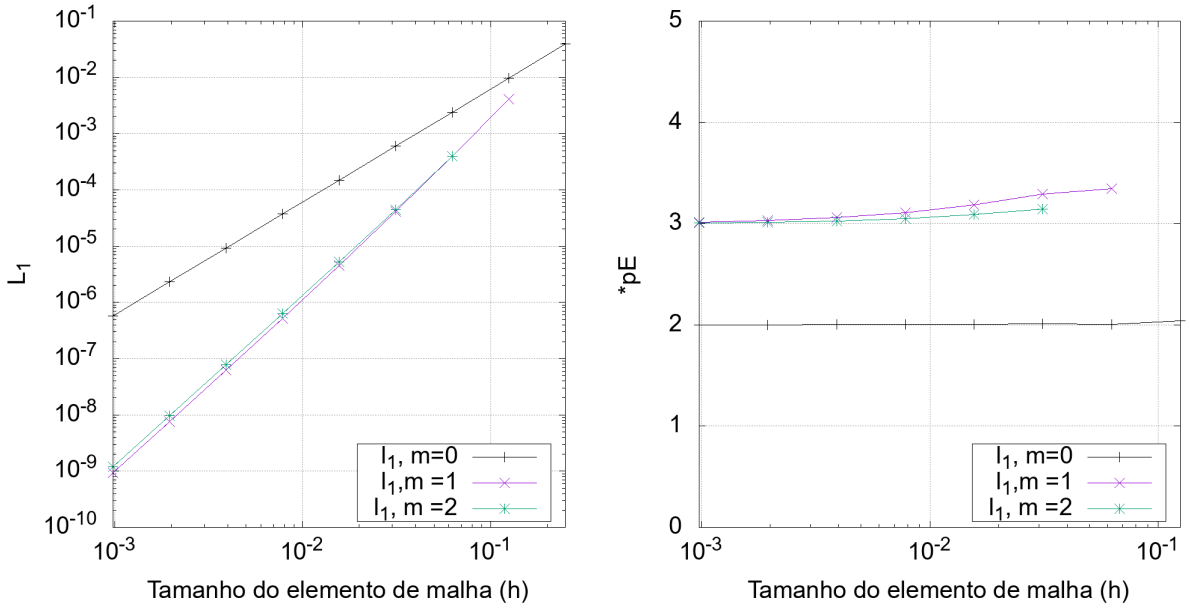
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 79 – Resultados: CRE, Variável l_1, Navier-Stokes 2D, velocidade u

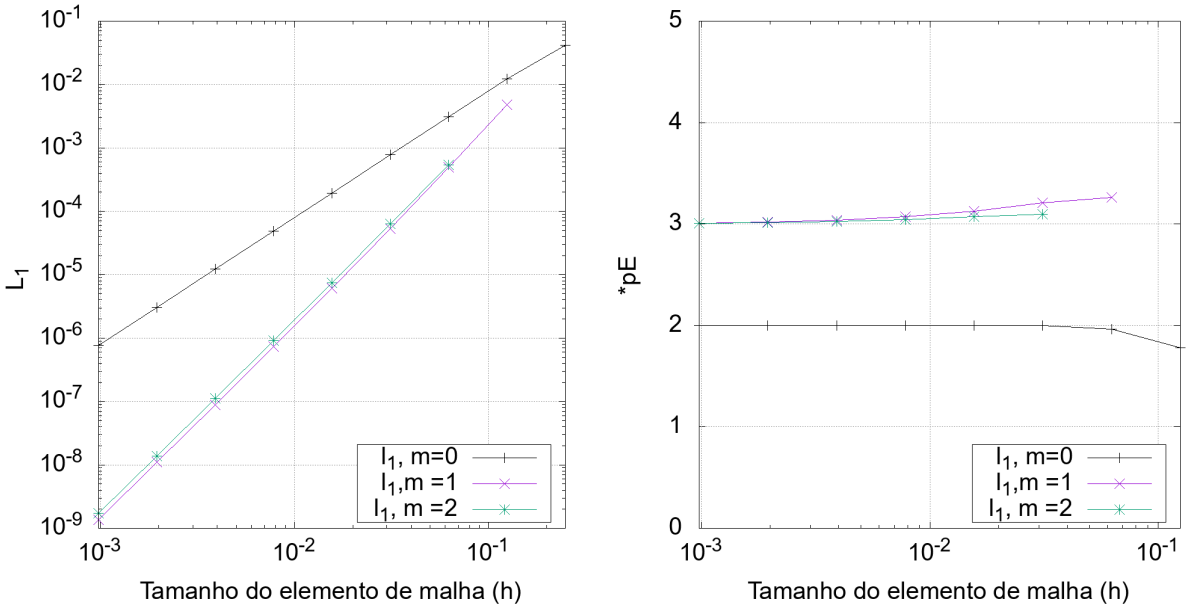
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 80 – Resultados: CRE, Variável l_1, Navier-Stokes 2D, velocidade v

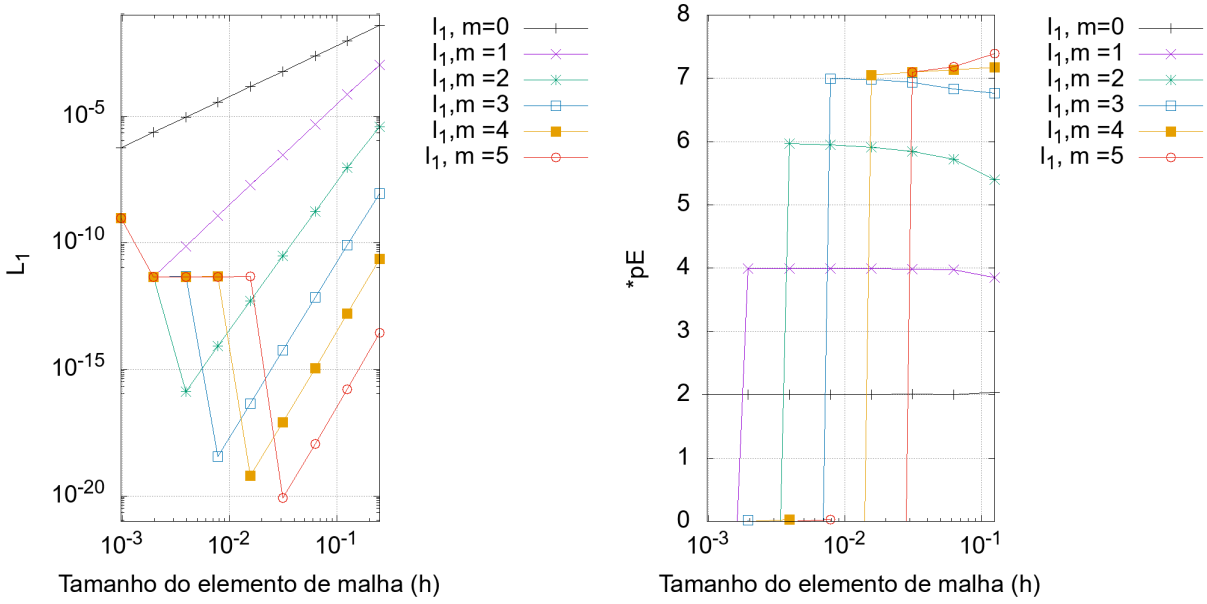
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 81 – Resultados: SER, Variável l_1, Navier-Stokes 2D, velocidade u

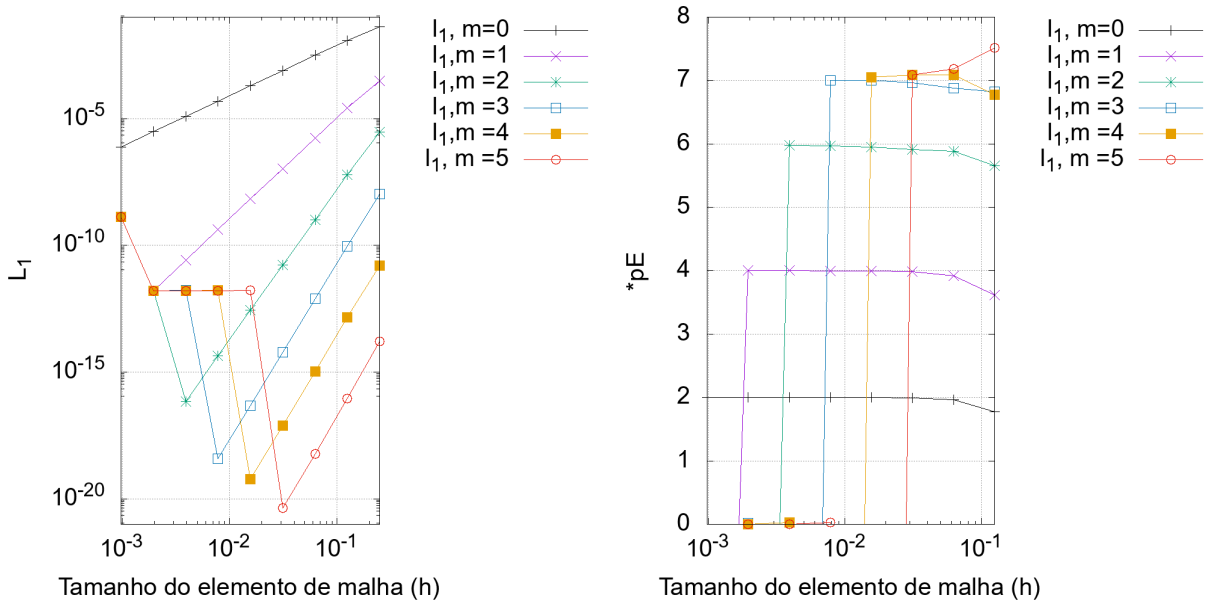
(a) l_1 (b) p_E^*



FONTE: Autor(2020).

Figura 82 – Resultados: SER, Variável l_1, Navier-Stokes 2D, velocidade v

(a) l_1 (b) p_E^*



FONTE: Autor(2020).

9 CONCLUSÃO

Depois de **muitos** testes, em geral as técnicas utilizadas obtiveram ótimos resultados, reduzindo o erro nos campos de forma significativa e aumentando suas ordens de acurácia, sem a necessidade da realização de novas simulações.

Agora em partes, começando por FRE. Embora só tenha sido testada apenas em problemas unidimensionais se mostrou eficaz. Foi possível reproduzir os resultados encontrados na literatura, para as equações de Burgers 1D e Poisson 1D com $T = \text{sen}(\pi x)$, e com a ressalva que foram discretizadas para malhas uniformes.

Trocando para Poisson 1D com $T = \frac{-1}{5}e^{3x} + 3x + 2$, seja com malha uniforme ou não e para os outros modelos mencionados anteriormente, mas agora com malhas não uniformes. Neste caso, não foi possível mensurar a ordem de acurácia de FRE, devido as oscilações apresentadas nas suas ordens de acurácia, acontecimento que não deveria acorrer. Porém, houve redução do erro, mas abaixo do que foi apresentado nos modelos anteriores. Além disso, CRE para estes modelos se equipara a FRE.

CRE por sua vez obteve desempenho esperado em todos os modelos apresentados com exceção de Navier-Stokes, onde foi apresentado uma degeneração de ordem. Era esperado ao menos o aumento de uma ordem verdadeira do método com redução do erro. Fato que aconteceu em todos os experimentos com exceção de Navier-Stokes.

Para Navier-Stokes, não só CRE, mas também CRE-I e SMER-I apresentaram resultados abaixo do esperado. Provavelmente o problema para os resultados negativos estão nos termos de correção dos pontos não coincidentes. Questão pertinente para ser investigada para trabalhos futuros.

CRE pode ser considerada uma versão menos robusta de CRE-I, porque é utilizada uma interpolação linear nos termos de correção. Informação que pode explicar o motivo do resultado negativo em Navier-Stokes, as interpolações. Foram elas provavelmente as responsáveis pelos resultados inferiores de CRE-I e SMER-I nas malhas não uniformes e Navier-Stokes.

A ideia de **FRE** ou até mesmo **CRE com interpolação** (CRE-I) é fazer com CRE consiga progredir, para mais extrapolações, tanto na redução do erro como no aumento da ordem de acurácia. CRE inicialmente consegue progredir ao menos uma ordem de acurácia do método com redução do erro, mesmo teoricamente utilizando somente uma interpolação linear no seu termo de correção. Já foi verificado em Giacomini (2013) que CRE para Poisson 1D com uma aproximação do tipo CDS de ordem $p_0 = 4$ consegue progredir para ordem seguinte $p_1 = 6$.

Tanto FRE quanto CRE-I modificam o termo de correção C para os nós não

coincidentes, e enquanto se mantiverem estáveis, conseguem avanços em seus resultados para $m > 1$, porque para os pontos coincidentes, variáveis do Tipo I (Seção 2.5.1) obtidas com MER ou ER (para $m = 1$) é eficaz em geral, e dificilmente não vai funcionar com condições ideais, isto é, quando o erro numérico esta afetado somente por Eh.

O termo estável neste contexto, refere-se ao fato de não influenciar negativamente no desempenho, fato que pode ser observado já no caso unidimensional, tanto FRE e CRE-I, quando funcionaram. Por exemplo, Poisson com $T = \text{sen}(\pi x)$ e malha uniforme, obtiveram resultados similares ou até melhores, em comparação com **SER** e **SMER**, porque conseguirem manter a qualidade obtida com MER (sobre os pontos coincidentes) em pontos não coincidentes. Neste contexto, FRE e CRE-I tem vantagem sobre SER e SMER por conseguirem extrapolar a malha mais fina, com resultados significativos.

Porém, trocando a discretização, SMER e SER se **destacam**, porque a estabilidade mencionada anteriormente é afetada nestes modelos. No geral SMER e SER que utilizam o conceito de submalha obtiveram os **melhores resultados**, porque simplesmente conseguiram soluções similares e algumas até melhores em todos os modelos empregados, em comparação as outras técnicas.

SER e SMER são aplicações de variáveis de MER do **Tipo I**, fator que explica sua convergência em todos os modelos, porque já é conhecido na literatura sua eficácia, a questão da novidade nestes experimentos é que o **conjunto** de variáveis do Tipo I também converge, até mesmo nas malhas não uniformes.

Os trabalhos de Celik e Karatekin (1997) e Carvalho Junior, Marchi e Martins (2015) já davam fortes indícios que SMER e SER poderiam funcionar neste cenário de não uniformidade, desde que o refino seja uniforme. Provavelmente para Laplace com malha não uniforme, ambas as técnicas poderiam ter obtido melhores resultados, através de um maior número de malhas ou as mesmas, no entanto mais acuradas.

Versão estendida de SMER a técnica **SMER interpolado** (SMER-I) se mostrou uma agradável surpresa, porque mostra que é possível misturar variáveis do Tipo I, II e III e ainda conseguir resultados excelentes, podendo ser considerada um importante complemento para CRE-I e SMER, porque consegue extrapolar qualquer conjunto de pontos.

CRE-I, técnica onde são realizadas interpolações unidimensionais, conforme já comentado anteriormente na revisão bibliográfica não é novidade, já foi desenvolvida em outros trabalhos, por exemplo Moro (2018). Neste trabalho o autor até obteve resultados melhores para equação de Burgers 2D, já para Laplace 2D os resultados foram equivalentes, em todos os casos a malha das discretizações é uniforme. Outro complemento desta técnica para a literatura é a utilização de interpolação clássica bidimensional de alta ordem. Porém, deve ser dado preferência para a interpolação alternativa, caso ela seja aplicável, devido

ao custo operacional inferior em relação a clássica.

Uma curiosidade importante: as técnicas **SMER** e **SMER INTERPOLADO** levam esse nome pelo comportamento de suas soluções, idênticas com MER, isto é, aumento concomitante da ordem de acurácia e redução progressiva do erro numérico. Fato que pode ser facilmente observado nos resultados.

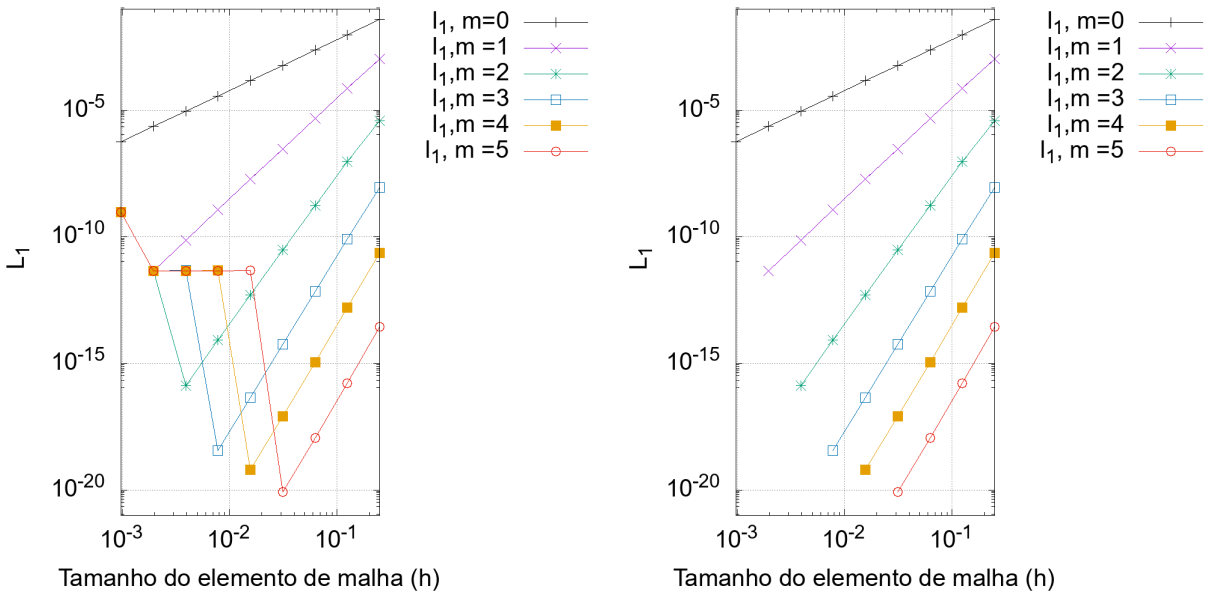
A ideia do nome **SER** segue o mesmo caminho, mas em comparação com ER. No entanto, deve ser feito uma alteração em seu algoritmo, bastando apenas retirar a malha fina em cada extrapolação, tornando esta técnica uma variação de CRE, mas de forma **inversa**, em que são beneficiadas as malhas iniciais ao contrário da técnica original, neste contexto SMER também pode ser considerado uma forma **inversa** de CRE. Um exemplo de SER com esta alteração pode ser observado na Figura 83.

Em resumo, as melhores técnicas utilizadas nos experimentos são **SER** e **SMER**, devido aos seus resultados, considerando todos os modelos utilizados. Analisando o tempo de CPU, todas as técnicas operam em questão de **segundos** exceto SMER-I, o que é ótimo, porque é possível obter soluções mais acuradas sem a necessidade de novas simulações e de forma rápida. A exceção de SMER-I, caso não seja utilizada alguma estratégia para otimizar a busca da região onde é realizada suas interpolações, é disparada a mais demorada e custosa dentre todas as técnicas utilizadas.

Figura 83 – SER, excluindo a malha mais fina em cada extrapolação, Navier-Stokes, velocidade u, variável l_1

(a) l_1: original

(b) l_1: excluindo a malha mais fina a cada extrapolação.



FONTE: Autor(2020).

9.1 CONTRIBUIÇÕES PARA LITERATURA

Através deste trabalho foi possível iniciar um estudo em relação as multiextrapolações para os campos, utilizando malhas diferentes das usuais, começando pelas não uniformes.

Neste trabalho foram testados métodos já existentes na literatura, onde foi mostrado sua queda no rendimento ao trocar o modelo de discretização, simultaneamente foi apresentado o conceito de submalha que a priori já consegue suprir este evento. Por meio deste trabalho portanto foi possível reduzir o erro de discretização para as variáveis de interesse estudadas, abordando malhas uni e bidimensionais uniformes e não uniformes.

Através deste trabalho foi possível analisar os efeitos ocasionados em MER pelo tipo de malha empregado no campo, associando seu desempenho com as variáveis do Tipo I, II e III de MER. Sendo que através de SMER-I foi possível verificar que é factível obter soluções acuradas com MER, misturando as estratégias utilizadas para obtenção destes três tipos de variáveis.

Através deste trabalho além dos conceitos de submalhas, foi abordado uma nova forma de realizar um procedimento já existente na literatura, é o caso da interpolação bidimensional, utilizada para complementar a técnica CRE-I. Embora não tenha conseguido melhorar os resultados nas malhas não uniformes, eles foram similares e possivelmente seja aplicável em outros modelos onde as interpolações unidimensionais não sejam aplicáveis, é o caso por exemplo em problemas envolvendo malhas não ortogonais.

Através deste trabalho foi possível estabelecer novos conceitos sobre o tema, porque claramente já é necessário pensar em novas estratégias para empregar as multiextrapolações no campo, graças aos inúmeros experimentos realizados.

9.2 TRABALHOS FUTUROS

Sugestões de trabalhos futuros:

- Trocar as condições de contorno dos modelos utilizados: É interessante verificar o que acontece, por exemplo para a condição de contorno de Neumann. Em particular já para o caso unidimensional devido a sua facilidade até os casos mais complexos como Navier-Stokes e Burgers.
- Variar o número de Reynolds: Só foi testado o caso $Re = 1$, é interessante verificar o que acontece para outros valores.
- Escolha dos vetores h: Escolher as dimensões e entradas de h para casos mais específicos.

- Utilizar outras técnicas de discretização: Por exemplo, para Volumes Finitos não teria o caso dos pontos coincidentes em geral, sendo necessário utilizar outras estratégias. Poderia começar com abordagem utilizada em Ferreira (2017).

- Abordar outros tipos de discretização: No caso das malhas não ortogonais, por exemplo, é interessante verificar a questão se é somente a não uniformidade que afeta os resultados. O mesmo procedimento poderia ser realizado em malhas triangulares.

- Se possível, encontrar uma forma de melhorar as malhas mais finas, através das técnicas SMER e SER: Embora seja improvável, devido as justificativas apresentadas para sua ineficácia, é uma questão em aberto.

- Aprimorar a técnica SMER-I: É preciso pensar em outras formas de determinar a região de interpolação, além das já abordadas neste trabalho. Caso isso seja possível, o tempo de CPU da técnica é reduzido consideravelmente, caso a região seja pré-determinada.

- Investigar problemas de convergência e degeneração de ordens relados nos experimentos: Uma sugestão seria verificar o erro de discretização (Eh) em todos os pontos de uma malha específica, para os dois casos estudados.

- Abordar mais variáveis de interesse de MER: Ficou faltando as do tipo IV e V. Variáveis do Tipo V são na teoria as mais complexas.

- Aplicabilidade da extrapolações para campos em escoamentos reais: Verificar a aplicabilidade em problemas de turbulência, por exemplo. Porém, o primeiro passo é verificar se MER funciona para pontos específicos do domínio, para depois ajustar no conjunto todo (Campo).

REFERÊNCIAS

AIAA. **Guide: Guide for the Verification and Validation of Computational Fluid Dynamics Simulations (AIAA G-077-1998(2002))**. [s.n.], 2002. Disponível em: <https://arc.aiaa.org/doi/abs/10.2514/4.472855.001>. Citado 2 vezes nas páginas 36 e 43.

ALVES, C. A. **Verificação de soluções numéricas da equação de Laplace 2D com malhas triangulares e múltiplas extrapolações de Richardson**. Tese de Doutorado em Engenharia Mecânica — Universidade Federal do Paraná, Curitiba, PR, 2010. Citado na página 28.

ANDREEV, V. F.; POPOV, A. M. Using Richardson's method to construct high-order accurate adaptive grids. **Computational Mathematics and Modeling**, v. 10, n. 3, p. 227–238, Jul 1999. ISSN 1573-837X. Disponível em: <https://doi.org/10.1007/BF02358942>. Citado na página 27.

ANUNCIAÇÃO, M. A. **Redução do erro de iteração e aceleração do método Multigrid com o uso de extrapoladores**. Dissertação de Mestrado em Métodos Numéricos em Engenharia — Universidade Federal do Paraná, Curitiba, PR, 2013. Citado na página 23.

BRIGGS, W. L.; HENSON, V. E.; MCCORMICK, S. F. **A multigrid tutorial**. [S.l.]: Siam, 2000. Citado na página 73.

CARVALHO JUNIOR, C. A. R. **Interpolação polinomial com Multiextrapolação de Richardson para reduzir o erro de discretização em malhas não uniformes 1D**. Dissertação de Mestrado em Métodos Numéricos em Engenharia — Universidade Federal do Paraná, Curitiba, PR, 2015. Citado 4 vezes nas páginas 25, 27, 69 e 70.

CARVALHO JUNIOR, C. A. R.; MARCHI, C. H.; MARTINS, M. A. Multiextrapolação de Richardson com interpolação para reduzir o erro de discretização em malhas não uniformes 1D. **Proceedings of the Brazilian Society of Applied and Computational Mathematics**, Vitória, Brazil, v. 3, n. 2, 2015. Disponível em: <http://dx.doi.org/10.5540/03.2015.003.02.0068>. Citado 2 vezes nas páginas 29 e 133.

CELIK, I.; KARATEKIN, O. Numerical Experiments on Application of Richardson Extrapolation With Nonuniform Grids. **Journal of Fluids Engineering**, v. 119, n. 3, p. 584–590, 09 1997. ISSN 0098-2202. Disponível em: <https://doi.org/10.1115/1.2819284>. Citado 3 vezes nas páginas 27, 29 e 133.

CELIK, I.; LI, J.; HU, G.; SHAFFER, C. Limitations of Richardson extrapolation and some possible remedies. **Journal of Fluids Engineering-transactions of The Asme - J FLUID ENG**, v. 127, 07 2005. Citado na página 28.

DAHLQUIST, G.; BJORCK, A. **Numerical Methods in Scientific Computing: Volume 1**. Society for Industrial and Applied Mathematics, 2008. (Other Titles in Applied Mathematics). ISBN 9780898717785. Disponível em: <https://books.google.com.br/books?id=qy83gXoRps8C>. Citado na página 41.

DAI, R.; LIN, P. Analysis on sixth-order compact approximations with Richardson extrapolation for 2d poisson equation. **Journal of Applied Mathematics and Physics**, v. 6, n. 6, p. 1139–1159, 2018. Disponível em: <https://www.scirp.org/pdf/JAMP_2018060514032349.pdf>. Citado na página 29.

Dai, R.; Zhang, J.; Wang, Y. Sixth order compact approximation with completed Richardson extrapolation. In: **2013 International Conference on Computational and Information Sciences**. [S.l.: s.n.], 2013. p. 966–971. ISSN null. Citado na página 28.

DAI, R.; ZHANG, J.; WANG, Y. Higher order adi method with completed Richardson extrapolation for solving unsteady convection-diffusion equations. **Comput. Math. Appl.**, Pergamon Press, Inc., USA, v. 71, n. 1, p. 431–442, jan. 2016. ISSN 0898-1221. Disponível em: <https://doi.org/10.1016/j.camwa.2015.12.007>. Citado na página 29.

FERREIRA, A. E. G. **Multiextrapolação de Richardson completa para o método de volumes finitos**. Tese de Doutorado em Engenharia Mecânica — Universidade Federal do Paraná, Curitiba, PR, 2017. Citado 2 vezes nas páginas 29 e 136.

FERZIGER, J.; PERIĆ, M.; STREET, R. **Computational Methods for Fluid Dynamics**. [S.l.: s.n.], 2020. ISBN 978-3-319-99691-2. Citado 3 vezes nas páginas 24, 32 e 36.

FORTUNA, A. **Técnicas computacionais para dinâmica de fluidos**. São Paulo: EDUSP, 2000. Citado 2 vezes nas páginas 22 e 39.

FRANCO, N. B. **Cálculo Numérico: Aspectos Teóricos e Computacionais**. São Paulo: Pearson Prentice Hall, 2006. Citado 2 vezes nas páginas 39 e 40.

FRANKE, J.; FRANK, W. Application of generalized Richardson extrapolation to the computation of the flow across an asymmetric street intersection. **Journal of Wind Engineering and Industrial Aerodynamics**, v. 96, n. 10, p. 1616 – 1628, 2008. ISSN 0167-6105. 4th International Symposium on Computational Wind Engineering (CWE2006). Disponível em: <http://www.sciencedirect.com/science/article/pii/S0167610508000342>. Citado na página 27.

FROM, J. E. The time dependent flow of an incompressible viscous fluid. v. 3, p. 345 – 382, 05 1964. Citado na página 76.

GERMER, E. M. **Verificação de funções de interpolação em advecção - difusão 1D com volumes finitos**. Dissertação de Mestrado em Engenharia Mecânica — Universidade Federal do Paraná, Curitiba, PR, 2009. Citado na página 42.

GERMER, E. M.; MARCHI, C. H. Verificação de esquemas advectivo-difusivos 1d com e sem multiextrapolação de Richardson. **Proceedings of the XXX Iberian Latin American Congress on the Computational Methods in Engineering (CILAMCE)**, Armação dos Búzios , Brazil, 2009. Citado na página 28.

GIACOMINI, F. F. **Multiextrapolação de Richardson completa para reduzir o erro de discretização**. Tese de Doutorado em Engenharia Mecânica — Universidade Federal do Paraná, Curitiba, PR, 2013. Citado 6 vezes nas páginas 28, 29, 55, 84, 85 e 132.

MALISKA, C. R. **Transferência de calor e mecânica dos fluidos computacional**. 2. ed. Rio de Janeiro: LTC, 2004. Citado 4 vezes nas páginas 22, 24, 38 e 39.

MARCHI, C. H. **Verificação de soluções numéricas unidimensionais em dinâmica de fluidos**. Tese de Doutorado em Engenharia Mecânica — Universidade Federal de Santa Catarina, Florianópolis, SC, 2001. Citado 6 vezes nas páginas 22, 23, 31, 32, 34 e 35.

MARCHI, C. H.; GIACOMINI, F. F. Efeito da razão de refino de malha sobre o uso de multiextrapolação de Richardson em CFD. **XXXV Congresso nacional de matemática aplicada e computacional**, Natal, Brazil, 2014. Citado na página 28.

MARCHI, C. H.; NOVAK, L. A.; COSMO, D. S.; VARGAS, A. P. S. Highly accurate numerical solutions with repeated Richardson extrapolation for 2D Laplace equation. **Applied Mathematical Modelling**, 2013. Citado na página 33.

MARCHI, C. H.; SUERO, R.; ARAKI, L. K. The lid-driven square cavity flow: numerical solution with a 1024 x 1024 grid. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, scielo, v. 31, p. 186 – 198, 09 2009. ISSN 1678-5878. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1678-58782009000300004&nrm=iso>. Citado na página 44.

MARTINS, M. A. **Multiextrapolação de Richardson com interpolação para reduzir e estimar o erro em CFD**. Tese de Doutorado em Engenharia Mecânica — Universidade Federal do Paraná, Curitiba, PR, 2013. Citado 8 vezes nas páginas 23, 28, 31, 39, 41, 45, 46 e 47.

MARTINS, M. A.; MARCHI, C. H. Estimate of iteration errors in computational fluid dynamics. **Numerical Heat Transfer, Part B: Fundamentals**, v. 53, n. 3, p. 234–245, 2008. Disponível em: <http://dx.doi.org/10.1080/10580530701790142>. Citado na página 32.

MARTINS, M. A.; MARCHI, C. H.; NOVAK, L. A.; ARAKI, L. K.; PINTO, M. A. V.; GONÇALVES, S. F. T. Multiextrapolação de Richardson com interpolação para reduzir o erro de discretização em CFD. **CMAC Sudeste**, 2013. Citado 3 vezes nas páginas 46, 47 e 48.

MORO, D. F. **Desenvolvimento de técnicas para reduzir os erros de iteração e discretização em CFD**. Tese de Doutorado em Engenharia Mecânica — Universidade Federal do Paraná, Curitiba, PR, 2018. Citado 4 vezes nas páginas 8, 29, 66 e 133.

MUNYAKAZI, J. B.; PATIDAR, K. C. On Richardson extrapolation for fitted operator finite difference methods. **Applied Mathematics and Computation**, n. 201, p. 465–480, 2008. Citado na página 27.

OLIVEIRA, A. G. de. **Resolução numérica de equações de advecção-difusão empregando malhas adaptativas**. Dissertação de mestrado em matemática aplicada — Universidade de São Paulo, São Paulo, SP, 2015. Citado na página 26.

PRITCHARD, P. **Fox and McDonald's Introduction to Fluid Mechanics, 8th Edition**. John Wiley & Sons, 2010. ISBN 9781118139455. Disponível em: <https://books.google.com.br/books?id=RdIbAAAAQBAJ>. Citado na página 75.

RANE, S.; KOVACEVIC, A. Application of numerical grid generation for improved cfd analysis of multiphase screw machines. **IOP Conference Series: Materials Science and Engineering**, v. 232, p. 012017, 08 2017. Citado na página 24.

REYNOLDS, O. **An Experimental Investigation of the Circumstances which Determine Whether the Motion of Water Shall be Direct Or Sinuous: And of the Law of Resistance in Parallel Channels**. Royal Society of London, 1883. (Philosophical transactions). Disponível em: <https://books.google.com.br/books?id=zz_NHAAACAAJ>. Citado na página 76.

RICHARDS, A. S. Completed Richardson extrapolation in space and time. **Commun. Numer. Meth. Eng**, v. 13, p. 573, 1997. Citado na página 28.

RICHARDSON, L. F. The approximate arithmetical solution by finite differences of physical problems involving differential equation, with an application to the stresses in a masonry dam. **Phylosophical Proceedings of the Royal Society of London**, v. 210, Serial A, p. 307–357, 1910. Citado 2 vezes nas páginas 27 e 43.

RICHARDSON, L. F.; GAUNT, J. A. The differed approach to the limit. **Phylosophical Proceedings of the Royal Society of London**, v. 226, Serial A, p. 229–361, 1927. Citado na página 27.

ROACHE, P. **Fundamentals of verification and validation**. USA: Hermosa, 2009. Citado 2 vezes nas páginas 32 e 36.

ROACHE, P. J.; KNUPP, P. M. Completed Richardson extrapolation. v. 9, p. 365 – 374, 05 1993. Citado 10 vezes nas páginas 8, 25, 28, 48, 49, 51, 52, 54, 85 e 153.

ROY, J. Review of code and solution verification procedures for computational simulation. **Journal of Computational Physics**, n. 205, p. 131–156, 2005. Citado na página 42.

SCHNEIDER, F. A. **Verificação de Soluções Numéricas em Problemas Difusivos e Advectivos com Malhas não Uniformes**. Tese de Doutorado em Métodos Numéricos em Engenharia — Universidade Federal do Paraná, Curitiba, PR, 2007. Citado 4 vezes nas páginas 23, 31, 35 e 38.

SHIH, T. M.; TAN, C. H.; HWANG, B. C. Effects of grid staggering on numerical schemes. **International Journal for Numerical Methods in Fluids**, v. 9, n. 2, p. 193–212, 1989. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.1650090206>. Citado 2 vezes nas páginas 74 e 75.

SILVA, N. D. P. da. **Extrapolação de Richardson completa e repetida para escoamentos com fluido compressível**. Tese de Doutorado em Engenharia Mecânica — Universidade Federal do Paraná, Curitiba, PR, 2019. Citado na página 29.

SONI, B. K.; SHIH, A. M.; ITO, Y. Grid generation techniques. In: _____. **Encyclopedia of Aerospace Engineering**. American Cancer Society, 2010. ISBN 9780470686652. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470686652.eae051>. Citado na página 24.

STEFFENSEN, J. F. **Interpolation**. 2. ed. New York: Chelsea Publishing Company, 1950. Citado 2 vezes nas páginas 42 e 67.

VARGAS, A. P. S. **Multiextrapolação de Richardson e esquemas de 1^a e 2^a ordens, mistos e crank−nicolson sobre as equações 2D de Advecção−Difusão e Fourier**. Tese de Doutorado em Engenharia Mecânica — Universidade Federal do Paraná, Curitiba, PR, 2013. Citado 5 vezes nas páginas 27, 28, 31, 44 e 45.

WANG, Y.; ZHANG, J. Sixth order compact scheme combined with multigrid method and extrapolation technique for 2d poisson equation. **Journal of Computational Physics**, v. 228, n. 1, p. 137 – 146, 2009. ISSN 0021-9991. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0021999108004580>. Citado na página 28.

YANG, X.; HAYES, M. Applications of grid techniques in the cfd field. In: . [S.l.: s.n.], 2008. Citado na página 24.

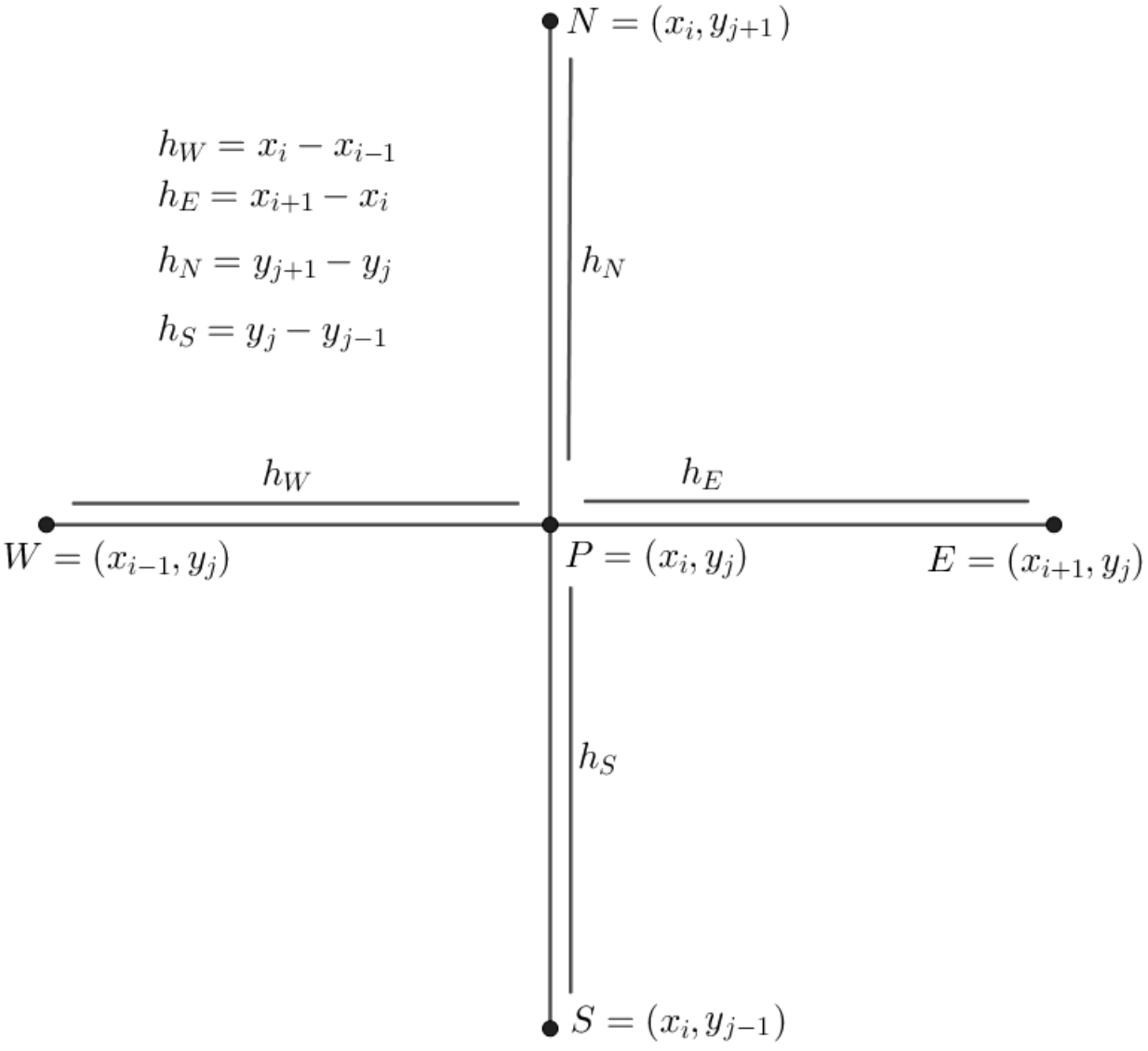
ZHANG, C.; ZHANG, J. Interpolation strategies in repeated Richardson extrapolation. **Journal of Intelligent and Fuzzy Systems**, v. 31, n. 6, 2016. Citado na página 29.

APÊNDICE A – DISCRETIZAÇÕES

Neste apêndice são apresentadas as discretizações empregadas e algoritmos para Diferenças Finitas, utilizados na resolução dos modelos matemáticos, cuja soluções numéricas não foram obtidas através de códigos cedidos pelo grupo de CFD da UFPR.

A estrutura das malhas utilizadas é apresentada pela Figura 84, cuja base parte da discretização feita em torno nó P, e seus vizinhos W, E, N e S, as variáveis h_W, h_E, h_N e h_S representam a distância entre estes pontos em relação ao ponto P, respectivamente, conforme é ilustrado na Figura 84. Note que o caso unidimensional é análogo, basta desconsiderar os pontos N e S. Naturalmente os nós P, E e W são unidimensionais, assim como suas derivadas, nesta situação.

Figura 84 – Malha não uniforme, discretizada para diferenças finitas.



FONTE: Autor(2020).

Avançando, as aproximações utilizadas pela série de Taylor em torno do nó P, são as seguintes:

$$T_W = T_P - \left(\frac{\partial T}{\partial x}\bigg|_P\right) h_W + \left(\frac{\partial^2 T}{\partial x^2}\bigg|_P\right) \frac{h_W^2}{2} + \cdots \sum_{i=3}^{\infty} \left(\frac{\partial^i T}{\partial x^i}\bigg|_P\right) \frac{(-h_W)^i}{i!}, \tag{A.1}$$

$$T_E = T_P + \left(\frac{\partial T}{\partial x}\bigg|_P\right) h_E + \left(\frac{\partial^2 T}{\partial x^2}\bigg|_P\right) \frac{h_E^2}{2} + \cdots \sum_{i=3}^{\infty} \left(\frac{\partial^i T}{\partial x^i}\bigg|_P\right) \frac{(h_E)^i}{i!}, \tag{A.2}$$

$$T_S = T_P - \left(\frac{\partial T}{\partial y}\bigg|_P\right) h_S + \left(\frac{\partial^2 T}{\partial y^2}\bigg|_P\right) \frac{h_S^2}{2} + \cdots \sum_{i=3}^{\infty} \left(\frac{\partial^i T}{\partial y^i}\bigg|_P\right) \frac{(-h_S)^i}{i!}, \tag{A.3}$$

$$T_N = T_P + \left(\frac{\partial T}{\partial y}\bigg|_P\right) h_N + \left(\frac{\partial^2 T}{\partial y^2}\bigg|_P\right) \frac{h_N^2}{2} + \cdots \sum_{i=3}^{\infty} \left(\frac{\partial^i T}{\partial y^i}\bigg|_P\right) \frac{(h_N)^i}{i!}. \tag{A.4}$$

Deste modo, obtêm-se aproximações do do tipo CDS, vistas no capítulo 2:

- Caso não uniforme

$$\left(\frac{\partial^2 T}{\partial x^2}\right)_P \approx \frac{2(h_E T_W + h_W T_E - (h_W + h_E) T_P)}{(h_W + h_E) h_W h_E}, \tag{A.5}$$

$$\left(\frac{\partial^2 T}{\partial y^2}\right)_P \approx \frac{2(h_N T_S + h_S T_N - (h_S + h_N) T_P)}{(h_S + h_N) h_S h_N} \tag{A.6}$$

e

$$\left(\frac{\partial T}{\partial x}\right)_P \approx \frac{T_E - T_W}{h_W + h_E}. \tag{A.7}$$

- Caso uniforme

  Considerando

$$h_N = h_S = h_y \text{ e } h_E = h_W = h_x, \tag{A.8}$$

  obtêm-se

$$\left(\frac{\partial^2 T}{\partial x^2}\right)_P \approx \frac{T_W + T_E - 2T_P}{h_x^2}, \tag{A.9}$$

$$\left(\frac{\partial^2 T}{\partial y^2}\right)_P \approx \frac{T_S + T_N - 2T_P}{h_y^2} \tag{A.10}$$

e

$$\left(\frac{\partial T}{\partial x}\right)_P \approx \frac{T_E - T_W}{2h_x}. \tag{A.11}$$

Nas próximas seções são apresentadas as discretizações e algoritmos para cada modelo matemático, em que são utilizadas estas aproximações.

A.1 Laplace 2D

Dado o modelo matemático apresentado na Seção 3.6.2

$$\begin{cases} \dfrac{\partial^2 T}{\partial x^2} + \dfrac{\partial^2 T}{\partial y^2} = 0, x,y \in [0,1] \\ T(x,1) = \text{sen}(\pi x),\ T(0,y) = T(1,y) = T(x,0) = 0 \end{cases} \tag{A.12}$$

Para aproximar as derivadas $\frac{\partial^2 T}{\partial x^2}$ e $\frac{\partial^2 T}{\partial y^2}$, utiliza-se as aproximações dadas pelas equações (A.5) e (A.6). Portanto, a Equação (A.12) pode ser reescrita aproximadamente da seguinte forma, para o ponto P:

$$\frac{2(h_E T_W + h_W T_E - (h_W + h_E)T_P)}{(h_W + h_E)h_W h_E} + \frac{2(h_N T_S + h_S T_N - (h_S + h_N)T_P)}{(h_S + h_N)h_S h_N} = 0 \tag{A.13}$$

ou

$$T_P\left(\frac{2}{(h_W + h_E)h_E} + \frac{2}{(h_W + h_E)h_W} + \frac{2}{(h_S + h_N)h_S} + \frac{2}{(h_S + h_N)h_N}\right) = \tag{A.14}$$
$$T_E\frac{2}{(h_W + h_E)h_E} + T_W\frac{2}{(h_W + h_E)h_W} + T_S\frac{2}{(h_S + h_N)h_S} + T_N\frac{2}{(h_S + h_N)h_N}$$

reagrupando no formato

$$a_P T_P = a_E T_E + a_W T_W + a_N T_N + a_S T_S \tag{A.15}$$

obtêm-se os coeficientes internos do modelo, para os casos:

- Caso não uniforme

$$a_E = \frac{2}{(h_W + h_E)h_E} \tag{A.16}$$
$$a_W = \frac{2}{(h_W + h_E)h_W} \tag{A.17}$$
$$a_S = \frac{2}{(h_S + h_N)h_S} \tag{A.18}$$
$$a_N = \frac{2}{(h_S + h_N)h_N} \tag{A.19}$$
$$a_P = a_E + a_W + a_S + a_N \tag{A.20}$$

- Caso uniforme

$$h_N = h_S = h_y \tag{A.21}$$
$$h_E = h_W = h_x \tag{A.22}$$
$$a_E = \frac{1}{h_x^2} \tag{A.23}$$
$$a_W = \frac{1}{h_x^2} \tag{A.24}$$
$$a_S = \frac{1}{h_y^2} \tag{A.25}$$
$$a_N = \frac{1}{h_y^2} \tag{A.26}$$
$$a_P = a_E + a_W + a_S + a_N \tag{A.27}$$

Os contornos são os seguintes:

$$T_P = \text{sen}(\pi x), \texttt{ para } P = (x,1) \tag{A.28}$$
$$T_P = 0, \texttt{ para } P = (0,y),\ P = (1,y) \texttt{ e } P = (x,0) \tag{A.29}$$
$$x,y \in [0,1]. \tag{A.30}$$

Para resolver a equação discretizada para os contornos internos utiliza-se o solver Gauss-Seidel:

$$T_P = \frac{a_E T_E + a_W T_W + a_N T_N + a_S T_S}{a_P}. \tag{A.31}$$

O Algoritmo 8, apresenta os passos da resolução do modelo.

Algoritmo 8: Resolução numérica da Equação de Laplace.

1. Carregue os contornos, dados pelas equações (A.28) e (A.29);
2. Faça $T_P = 0$, para os pontos fora do contorno do domínio;
3. Obtenha os coeficientes a_P,a_E,a_W,a_N e a_S internos, dados pelas equações (A.16-A.20) (caso não uniforme) ou (A.21-A.27) (caso uniforme);
4. Aplique o Gauss-Seidel somente para os pontos internos (Equação (A.29));
5. Repita o passo anterior até satisfazer um critério de parada preestabelecido.
6. Escreva os campos e variáveis de interesse.

A.2 Burgers 1D

Dada a variação da equação de Burgers 1D, apresentada na seção 3.6.3

$$\begin{cases} T\dfrac{dT}{dx}Re = \dfrac{d^2T}{dx^2} + \dfrac{Re^2e^{xRe}(e^{xRe}-e^{Re})}{(e^{Re}-1)^2} & x \in [0,1] \\ T(0)=0; T(1)=1 \end{cases} . \tag{A.32}$$

Para aproximar o termo $\frac{d^2T}{dx^2}$ avaliado no ponto P, utiliza-se uma aproximação do tipo CDS, dada pela Equação (A.5), a variável $T\frac{\partial T}{\partial x}$ avaliada no ponto P, para o termo $\frac{\partial T}{\partial x}$ utiliza-se a aproximação CDS, dada pela equação (A.6), o termo T é linearizado (T_P^*), isto é

$$\left(T\frac{dT}{dx}\right)_P = T_P\left(\frac{dT}{dx}\right)_P \approx T_P^*\left(\frac{T_E - T_W}{h_W + h_E}\right), \tag{A.33}$$

em que T_P^* representa a temperatura avaliada em P, na iteração i anterior, isto é

$$T_P^* = T_P^{(i-1)}, \quad i = 1,2,3,\cdots \tag{A.34}$$

e

$$T_P^{(0)} = 0. \tag{A.35}$$

Deste modo, a Equação (A.32) pode ser reescrita aproximadamente da seguinte forma, para o ponto P

$$\frac{Re(T_P^*T_E - T_P^*T_W)}{h_W + h_E} = \frac{2(h_ET_W + h_WT_E - (h_W+h_E)T_P)}{(h_W+h_E)h_Wh_E} + \frac{Re^2e^{xRe}(e^{xRe}-e^{Re})}{(e^{Re}-1)^2} \tag{A.36}$$

ou

$$4T_P = T_W\frac{2(h_Wh_E)ReT_P^* + 4h_E}{h_W+h_E} + T_E\frac{-2(h_Wh_E)ReT_P^* + 4h_W}{h_W+h_E} + 2S_Ph_Wh_E \tag{A.37}$$

em que

$$S_P = \frac{Re^2e^{PRe}(e^{PRe}-e^{Re})}{(e^{Re}-1)^2}. \tag{A.38}$$

Portanto, obtém-se os termos dos contornos internos da discretização

$$a_P = 4 \tag{A.39}$$

$$a_W = \frac{2(h_Wh_E)ReT_P^* + 4h_E}{h_W+h_E} \tag{A.40}$$

$$a_E = \frac{-2(h_Wh_E)ReT_P^* + 4h_W}{h_W+h_E} \tag{A.41}$$

$$b_P = 2S_Ph_Wh_E \tag{A.42}$$

o contorno esquerdo e direito são os seguintes:

- Contorno **esquerdo**

$$a_W = 0; \tag{A.43}$$
$$a_E = 0; \tag{A.44}$$
$$a_P = 1; \tag{A.45}$$
$$b_P = 0. \tag{A.46}$$

- Contorno **direito**

$$a_W = 0; \tag{A.47}$$
$$a_E = 0; \tag{A.48}$$
$$a_P = 1; \tag{A.49}$$
$$b_P = 1. \tag{A.50}$$

Para resolver a equação discretizada utiliza-se o TDMA de forma iterativa, porque é necessário atualizar os termos T_P^*. Em resumo o Algoritmo 9 mostra os passos da resolução do modelo.

Algoritmo 9: Resolução numérica da Equação de Burgers 1D.

1. Obtenha os coeficientes a_P,a_E,a_W e b_P internos e contornos (equações (A.39-A.50)).
2. Aplique o TDMA, utilizando os coeficientes a_P,a_E,a_W e b_P, obtidos no item anterior.
3. Atualize os coeficientes internos a_W e a_E (equações (A.40) e (A.41)).
4. Aplique o TDMA novamente, utilizando os novos coeficientes internos.
5. Repita os passos 3 e 4, até alcançar algum critério de parada preestabelecido.
6. Escreva os campos e variáveis de interesse.

A.3 Poisson 1D

Dados os modelos matemáticos apresentados na Seção 3.6.1

$$\begin{cases} \dfrac{d^2T_1}{dx^2} = \dfrac{-9}{5}e^{3x} \quad x \in [0,1] \\ T_1(0) = \dfrac{9}{5} \ ; \ T_1(1) = \dfrac{-e^3}{5} + 5 \end{cases} \quad (\texttt{caso 1}), \tag{A.51}$$

$$\begin{cases} \dfrac{d^2T_2}{dx^2} = -\pi^2 \operatorname{sen}(\pi x) \quad x \in [0,1] \\ T_2(0) = T_2(1) = 0 \end{cases} \quad (\texttt{caso 2}). \tag{A.52}$$

Para aproximar a derivada de segunda ordem, utiliza-se uma aproximação do tipo CDS dado pela Equação (A.5), deste modo pode-se reescrever os modelos matemáticos das

seguintes formas aproximadas:

$$2\frac{[h_E T_W + h_W T_E - (h_W + h_E)T_P]}{(h_W + h_E)h_W h_E} = S_P \tag{A.53}$$

ou

$$T_P = \frac{h_E}{(h_W + h_E)}T_W + \frac{h_W}{(h_W + h_E)}T_E - h_W h_E \frac{S_P}{2}. \tag{A.54}$$

Portanto da equação (A.54), obtêm-se os elementos do contorno **interno** da discretização:

$$a_W = \frac{h_E}{h_W + h_E}, \tag{A.55}$$

$$a_e = \frac{h_W}{(h_W + h_E)}, \tag{A.56}$$

$$a_p = 1, \tag{A.57}$$

$$b_p = -h_W h_E \frac{S_P}{2} \tag{A.58}$$

em que

$$S_P = \begin{cases} \dfrac{-9}{5}e^{3P}, \ (\texttt{caso 1}) \\ -\pi^2 \operatorname{sen}(\pi P), \ (\texttt{caso 2}) \end{cases} \tag{A.59}$$

O contorno esquerdo e direito, são os seguintes:

- Contorno **esquerdo**

$$a_W = 0; \tag{A.60}$$

$$a_E = 0; \tag{A.61}$$

$$a_P = 1; \tag{A.62}$$

$$b_P = F(0). \tag{A.63}$$

- Contorno **direito**

$$a_W = 0; \tag{A.64}$$

$$a_E = 0; \tag{A.65}$$

$$a_P = 1; \tag{A.66}$$

$$b_P = F(1) \tag{A.67}$$

em que

$$F(x) = \begin{cases} \frac{-1}{5}e^{3x} + 3x + 2, \ (\texttt{caso 1}) \\ \operatorname{sen}(\pi x), \ (\texttt{caso 2}) \end{cases} \qquad x \in [0,1]. \tag{A.68}$$

Com a discretização do modelo matemático, obtendo os elementos do contorno interno, direito e esquerdo, aplicou-se o método TDMA, para resolver o sistema linear

tridiagonal, gerado desta discretização, o algoritmo 10 apresenta a resolução do modelo de forma resumida.

Algoritmo 10: Resolução numérica da Equação de Poisson 1D.

1. Obtenha os coeficientes a_P,a_E,a_W e b_P internos e contornos (equações (A.55-A.67)).
2. Aplique o TDMA, utilizando os coeficientes a_P,a_E,a_W e b_P, obtidos no item anterior.
3. Escreva os campos e variáveis de interesse, para cada caso.

APÊNDICE B – ALGORITMOS GERAIS

Neste apêndice são apresentados algoritmos gerais detalhados em formato de pseudocódigo da maioria das técnicas utilizadas no trabalho, lembrando da notação utilizada para variáveis do tipo $\phi^m_{g,p}$, em que

- m: representa o nível da extrapolação,
- ϕ: variável de interesse,
- g: número da malha,
- p: posição do ponto.

Naturalmente, existem outras formas de realizar as operações, talvez mais econômicas e rápidas. Os algoritmos apresentados foram citados no trabalho e são voltados para Diferenças Finitas, considerando a notação lexicográfica e pontos com índices 0, 1, 2, $\cdots$.

Algoritmo 11: Algoritmo SUBER 2D, para diferenças finitas.

Entrada: Obter G soluções numéricas distintas: $\phi^0_{1,p}$ com $Nx+1$ e $Ny+1$ ($Nx>0$ e $Ny>0$ pares) nós na direção x e y respectivamente; $\phi^0_{2,p}$ com $2Nx+1$ e $2Ny+1$ nós na direção x e y respectivamente; $\cdots$,$\phi^0_{G,p}$ com $2^{G-1}Nx+1$ e $2^{G-1}Ny+1$ nós na direção x e y respectivamente. Fornecer o valor de p_L (ordem de acurácia das soluções numéricas).

Saída: $\phi_{g,1}$, para g = 1,2,3,$\cdots$,G.

para $g = 1,2,\cdots,G-1$ **faça**
  $NxI = 2^g Nx + 1$
  $NyI = 2^g Ny + 1$
  $k = 1$
  para j = 1,3,5,$\cdots$,NyI **faça**
    para i = 1,3,5,$\cdots$,NxI **faça**
      $p = i + NxI(j-1)$
      $\phi^1_{g,k} = \phi^0_{g+1,p} + \dfrac{\phi^0_{g+1,p} - \phi^0_{g,k}}{2^{p_L} - 1}$
      $k = k + 1$

para $g = G$ **faça**
  Obtenha $\phi^1_{g,p}$ através do Algoritmo 3 (CRE 2D clássico), para g = G.

Algoritmo 12: Algoritmo SMER 1D, para diferenças finitas.

Entrada: Fornecer o valor de $N_{inicial}$, p_0, r e dv (número de nós da malha mais grossa, ordem de acurácia das soluções numéricas, razão de refino das malhas e variação das ordens verdadeiras p_0,p_1,p_2,.., respectivamente). Obter G soluções numéricas distintas: $\phi^0_{1,p}$ com nós $p = 0,1,\cdots,N_{inicial}$; $\phi^0_{2,p}$ com nós $p = 0,1,2,\cdots,rN_{inicial}$; $\cdots$; $\phi^0_{G,p}$ com nós $p = 0,1,2,\cdots,r^{G-1}N_{inicial}$.

para $g = 1,2,3,\cdots,G-1$ **faça**

  $N_c = r^{g-1}N_{inicial}$

  $\overline{G} = G - g + 1$

  $\overline{it} = 1$

  para $it = g,g+1,g+2,\cdots,G$ **faça**

    $N = r^{it-1}N_{inicial}$

    $\overline{\phi}^0_{\overline{it},0} = \phi^0_{it,0}$

    $\widetilde{N} = r^{\overline{it}-1}$

    para $i = 1,2,\cdots,N_c - 1$ **faça**

      $\overline{\phi}^0_{\overline{it},i} = \phi^0_{it,i\widetilde{N}}$

    $\overline{\phi}^0_{\overline{it},N_c} = \phi^0_{it,N}$

    $\overline{it} = \overline{it} + 1$

  para $i = 0,1,\cdots,N_c$ **faça**

    para $it = 1,2,\cdots,\overline{G}$ **faça**

      $\sigma_{it,0} = \overline{\phi}^0_{it,i}$

    para $m = 1,2,\cdots,\overline{G}-1$ **faça**

      $p_{m-1} = p_0 + (m-1)dv$

      para $\overline{g} = m+1, m+2,\cdots,\overline{G}$ **faça**

        $$\sigma_{\overline{g},m} = \sigma_{\overline{g},m-1} + \frac{\sigma_{\overline{g},m-1} - \sigma_{\overline{g}-1,m-1}}{r^{p_{m-1}} - 1}$$

    para $it = 1,2,\cdots,\overline{G}$ **faça**

      $\overline{\phi}^1_{it,i} = \sigma_{it,it-1}$

  Calcule e/ou escreva as variáveis de interesse, campos, normas etc., para $\overline{\phi}^1_{it,i}$ e $\phi^0_{it,i}$, com $it = 1,2,\cdots,\overline{G}$ e $i = 0,1,2,\cdots,N_c$, referentes a malha g.

Algoritmo 13: Algoritmo SMER 2D, para diferenças finitas.

Entrada: Fornecer o valor de p_0, r e dv (ordem de acurácia das soluções numéricas, razão de refino entre as malhas e variação das ordens verdadeiras p_0,p_1,p_2,.., respectivamente).
Obter G soluções numéricas distintas: $\phi^0_{1,p}$ com $Nx+1$ e $Ny+1$ ($Nx>0$ e $Ny>0$ pares) nós na direção x e y respectivamente; $\phi^0_{2,p}$ com $rNx+1$ e $rNy+1$ nós na direção x e y respectivamente; $\cdots$,$\phi^0_{G,p}$ com $r^{G-1}Nx+1$ e $r^{G-1}Ny+1$ nós na direção x e y respectivamente.

para $g = 1,2,3,\cdots,G-1$ **faça**
- $\overline{Nx} = r^{g-1}Nx+1$
- $\overline{Ny} = r^{g-1}Ny+1$
- $N_c = \overline{Nx}\ \overline{Ny}$
- $\overline{G} = G-g+1$
- $\overline{it} = 1$
- **para** $it = g,g+1,g+2,\cdots,G$ **faça**
  - $\overline{Nx} = r^{g-1}Nx+1$
  - $\overline{Ny} = r^{g-1}Ny+1$
  - $k = 1$
  - $\widetilde{N} = r^{\overline{it}-1}$
  - **para** $j = 1,\widetilde{N}+1,2\widetilde{N}+1,3\widetilde{N}+1,\cdots,\overline{Ny}$ **faça**
    - **para** $i = 1,\widetilde{N}+1,2\widetilde{N}+1,3\widetilde{N}+1,\cdots,\overline{Nx}$ **faça**
      - $p = i+\overline{Nx}(j-1)$
      - $\overline{\phi}^0_{\overline{it},k} = \phi^0_{it,p}$
      - $k = k+1$
  - $\overline{it} = \overline{it}+1$
- **para** $i = 1,2,3\cdots,N_c$ **faça**
  - **para** $it = 1,2,\cdots,\overline{G}$ **faça**
    - $\sigma_{it,0} = \overline{\phi}^0_{it,i}$
  - **para** $m = 1,2,\cdots,\overline{G}-1$ **faça**
    - $p_{m-1} = p_0+(m-1)dv$
    - **para** $\overline{g} = m+1,m+2,\cdots,\overline{G}$ **faça**
      - $\sigma_{\overline{g},m} = \sigma_{\overline{g},m-1}+\dfrac{\sigma_{\overline{g},m-1}-\sigma_{\overline{g}-1,m-1}}{r^{p_{m-1}}-1}$
  - **para** $it = 1,2,\cdots,\overline{G}$ **faça**
    - $\overline{\phi}^1_{it,i} = \sigma_{it,it-1}$
- Calcule e/ou escreva as variáveis de interesse, campos, normas etc., para $\overline{\phi}^1_{it,i}$ e $\phi^0_{it,i}$, com $it = 1,2,\cdots,\overline{G}$ e $i = 1,2,\cdots,N_c$, referentes a malha g.

Algoritmo 14: CRE clássico com Interpolação e FRE 1D (adaptado de Roache e Knupp (1993)).

Entrada: Obter G soluções numéricas distintas: $\phi^0_{1,p}$ com nós $p = 0,1,\cdots,N_{inicial}$; $\phi^0_{2,p}$ com nós $p = 0,1,2,\cdots,2N_{inicial}$; $\cdots$; $\phi^0_{G,p}$ com nós $p = 0,1,2,\cdots,2^{G-1}N_{inicial}$.

Fornecer o valor de p_0 (ordem de acurácia das soluções numéricas).

Saída: $\phi^1_{g,p}$, para $g = 2,3,\cdots,G$ e $p = 0,1,2,\cdots,2^{g-1}N_{inicial}$.

para $g = 2,3,\cdots,G$ **faça**
- $N = 2^{g-1}N_{inicial}$
- **para** $i = 0,2,4,6,..,N$ **faça**
  - $C^1_{g,i} = \dfrac{\phi^0_{g,i} - \phi^0_{g-1,i/2}}{2^{p_0} - 1}$
  - $\phi^1_{g,i} = \phi^0_{g,i} + C^1_{g,i}$
- **para** $i = 0,2,4,6,..,N-2$ **faça**
  - **selecione** *Correção* **faça**
    - **caso** *Clássica* **faça**
      - $C^1_{g,i+1} = \dfrac{C^1_{g,i} + C^1_{g,i+2}}{2}$
    - **caso** *FRE* **faça**
      - $k = \dfrac{\phi^0_{g,i+1} - \phi^0_{g,i}}{\phi^0_{g,i+2} - \phi^0_{g,i}}$
      - $C^1_{g,i+1} = C^1_{g,i} + k(C^1_{g,i+2} - C^1_{g,i})$
    - **caso** *Interpolado* **faça**
      - Aplique o Algoritmo 15 para o nó i (Mais detalhes, veja Seção 3.5)
  - $\phi^1_{g,i+1} = \phi^1_{g,i+1} + C^1_{g,i+1}$

Algoritmo 15: Algoritmo cálculo da correção (C) por interpolação, para os nós não coincidentes, caso unidimensional.

Entrada: Fornecer a correção (C) com seus nós disponíveis da malha g:
$0,2,4,\cdots,N$ (N par).
Fornecer os nós da malha (x).
Fornecer o grau máximo (p) da interpolação desejada.
Fornecer a posição i do nó, antecessora do nó a ser interpolado ($i+1$).

Saída: $C^1_{g,i+1}$.

$j = 0$; $c_1 = 1$; $c_2 = 1$; $k = 2$; $k1 = i + k$; $k2 = 0$; $k3 = i + k2$;
enquanto $j \leq p$ *e* $c_1 + c_2 > 0$ **faça**
  se $k1 \leq N$ **então**
    se $j \geq 0$ *e* $j \leq p$ **então**
      $ponto(j) = k1$
      $k = k + 2$
      $k1 = i + k$
      $j = j + 1$
  senão
    $c_1 = 0$
  se $k3 \geq 0$ **então**
    se $j \geq 0$ *e* $j \leq p$ **então**
      $ponto(j) = k3$
      $k2 = k2 - 2$
      $k3 = i + k2$
      $j = j + 1$
  senão
    $c_2 = 0$
Calcule a correção $C^1_{g,i+1}$, através da interpolação de grau máximo $j-1$, avaliada em $x(i+1)$, passando pelos pontos: $(x(ponto(k)), C^1_{g,ponto(k)}$, $k = 0,1,\cdots,j-1)$.

APÊNDICE C – RESULTADOS ADICIONAIS

Neste apêndice são apresentados resultados adicionais para a variável l_∞ referente aos capítulos 4, 5, 6, 7 e 8.

Estão publicados todos os resultados e simulações de forma integral neste link:

http://ftp.demec.ufpr.br/CFD/monografias/2020_Carlos-Carvalho_doutorado/

C.1 MALHA NÃO UNIFORME 1D

C.1.1 Poisson

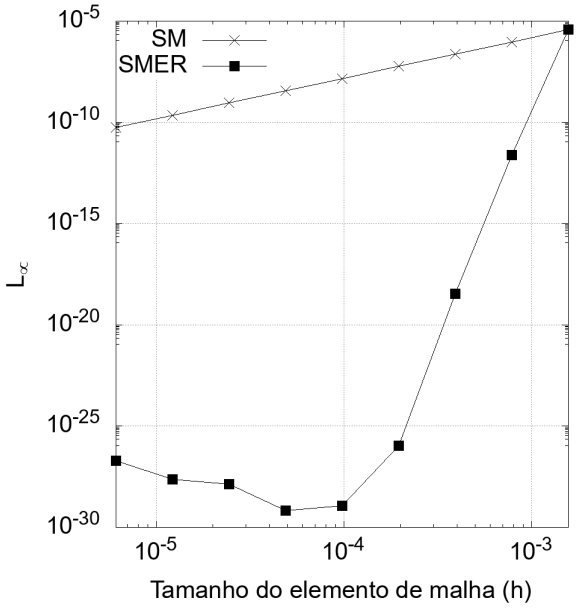
Tabela 16 – Resultados: SMER INTERPOLADO, conjunto β_2, Variável l_∞, Malha não uniforme, Poisson1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$

Malha	h	l_∞: SM-I	l_∞: SMER-I	p_E^*: l_∞, SM-I	p_E^*: l_∞, SMER-I
1	1.00E-01	8.48836E-02	8.48836E-02		
2	5.00E-02	6.03663E-03	3.15517E-02	3.81367E+00	1.42777E+00
3	2.50E-02	1.49333E-03	3.80227E-03	2.01521E+00	3.05278E+00
4	1.25E-02	3.69191E-04	1.61679E-04	2.01610E+00	4.55566E+00
5	6.25E-03	9.17485E-05	3.94693E-06	2.00861E+00	5.35626E+00
6	3.13E-03	2.28669E-05	4.96512E-07	2.00442E+00	2.99083E+00
7	1.56E-03	5.70786E-06	6.20594E-08	2.00224E+00	3.00010E+00
8	7.81E-04	1.42585E-06	7.76430E-09	2.00113E+00	2.99872E+00
9	3.91E-04	3.56323E-07	9.70877E-10	2.00057E+00	2.99950E+00
10	1.95E-04	8.90633E-08	1.21379E-10	2.00028E+00	2.99977E+00
11	9.77E-05	2.22636E-08	1.51628E-11	2.00014E+00	3.00091E+00
12	4.88E-05	5.56564E-09	1.89427E-12	2.00007E+00	3.00082E+00
13	2.44E-05	1.39137E-09	2.35156E-13	2.00004E+00	3.00995E+00
14	1.22E-05	3.47837E-10	2.93947E-14	2.00002E+00	2.99999E+00
15	6.10E-06	8.69588E-11	3.67436E-15	2.00001E+00	2.99999E+00

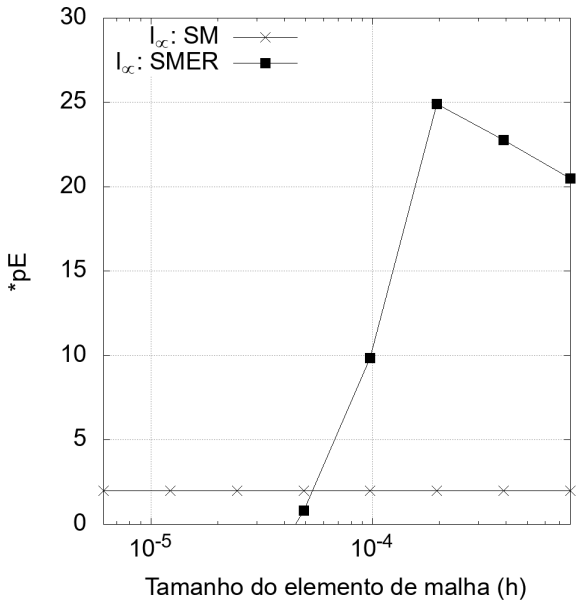
FONTE: Autor(2020).

Figura 85 – Resultados: SMER, Malhas 7-8, Variável l_∞, Poisson 1D com solução $T(x) = \text{sen}(\pi x)$, Malha não uniforme

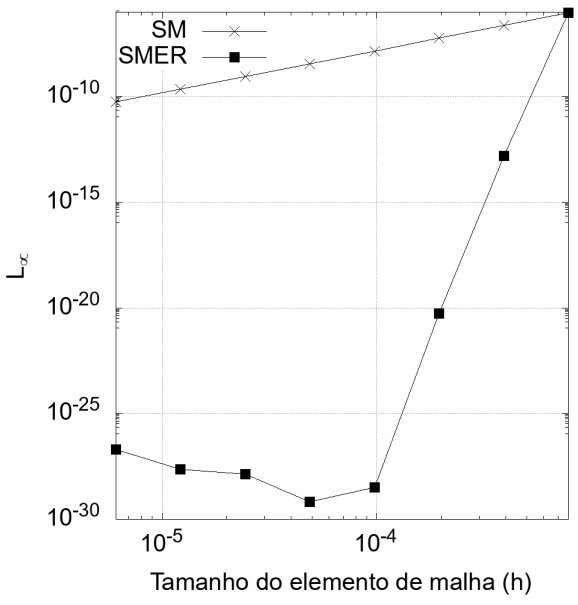
(a) l_∞, Malha 7



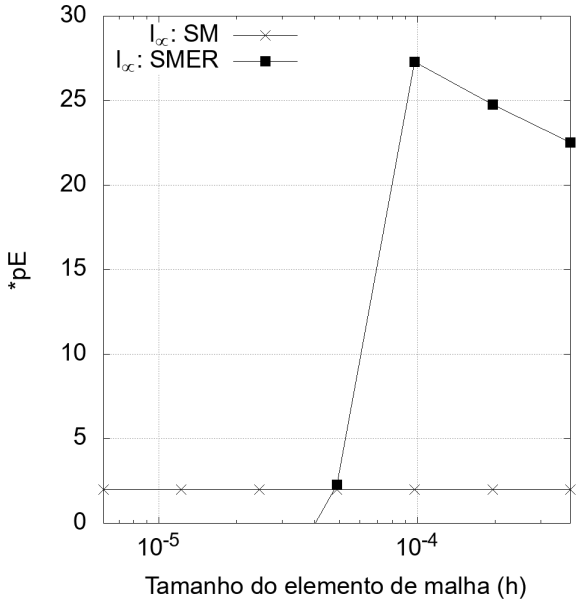
(b) p_E^*, Malha 7



(c) l_∞, Malha 8

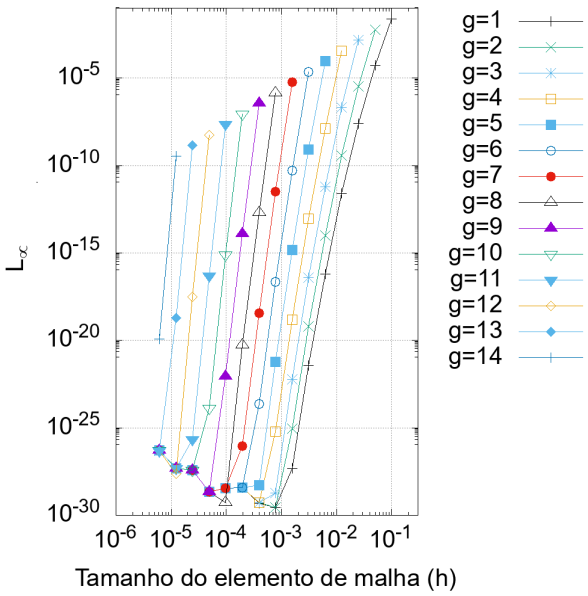


(d) p_E^*, Malha 88



FONTE: Autor(2020).

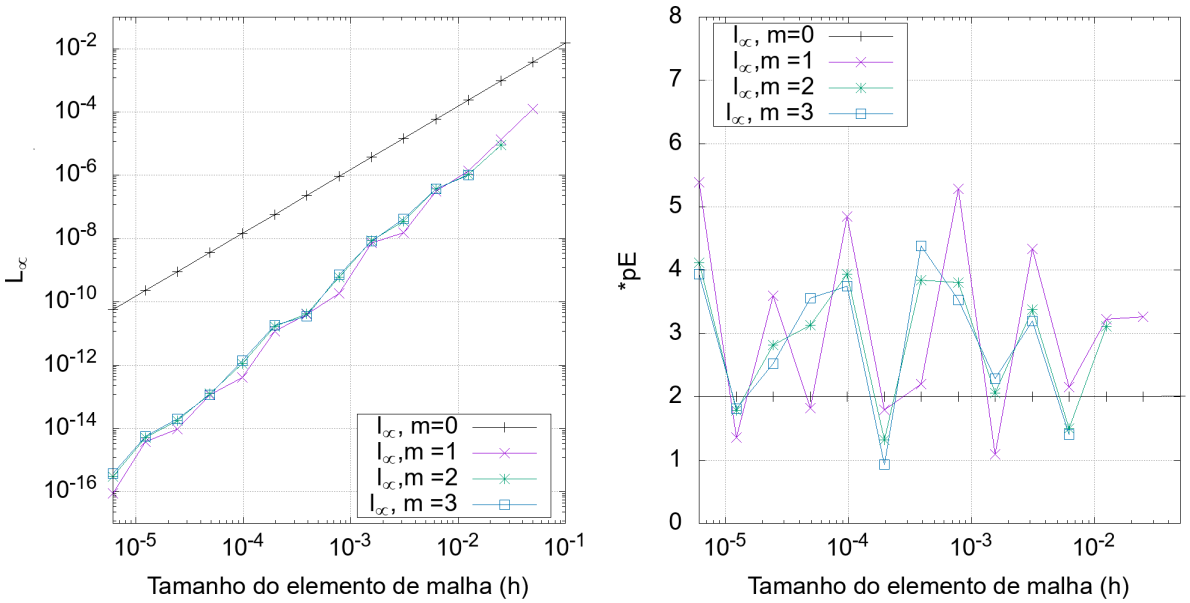
Figura 86 – Resultados: SMER, Malhas $g = 1{,}2\cdots{,}14$, Variável l_∞, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha não uniforme



FONTE: Autor(2020).

Figura 87 – Resultados: FRE, Variável l_∞, Poisson 1D com solução $T(x) = \text{sen}(\pi x)$, Malha não uniforme

(a) l_∞ (b) p_E^*

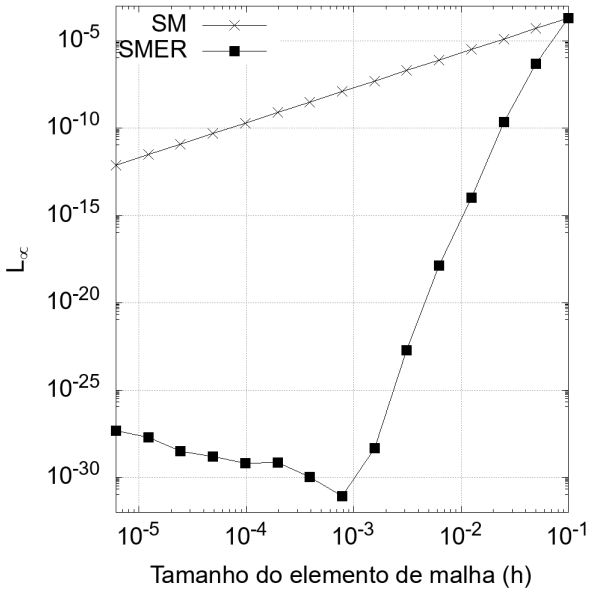


FONTE: Autor(2020).

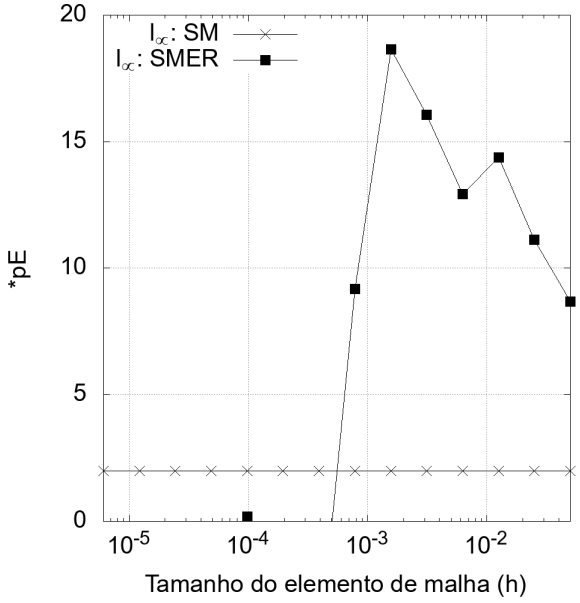
C.1.2 Burgers

Figura 88 – Resultados: SMER, Malhas 1-2, Variável l_∞, Burgers 1D com solução $T(x) = \frac{e^{xRe}-1}{e^{Re}-1}$, $Re = 1$, Malha não uniforme

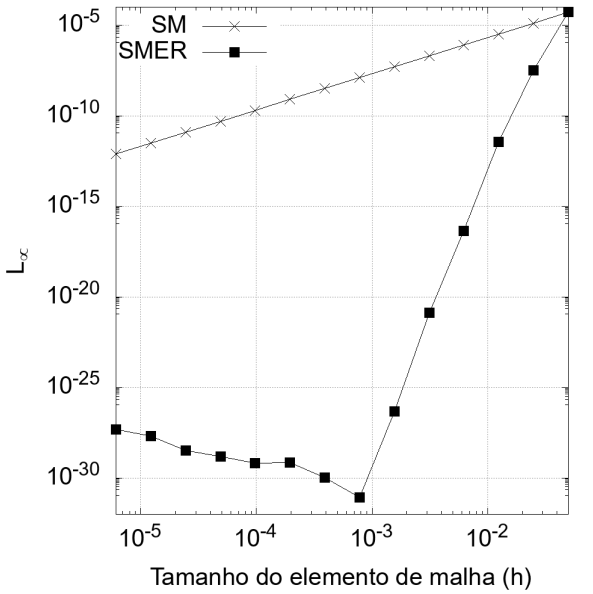
(a) l_∞, Malha 1



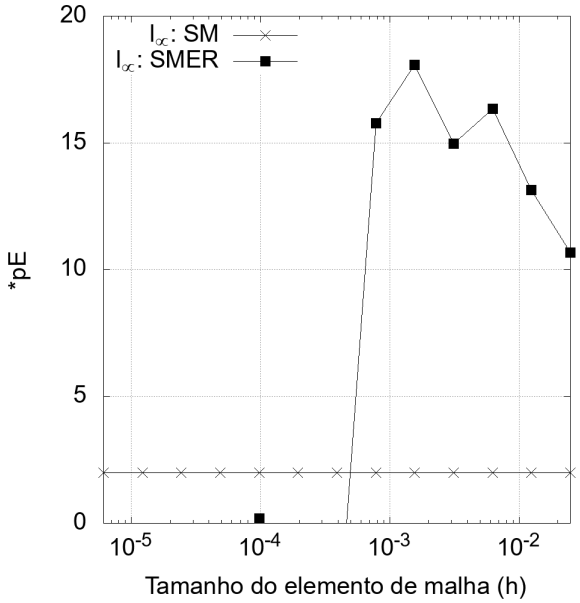
(b) p_E^*, Malha 1



(c) l_∞, Malha 2



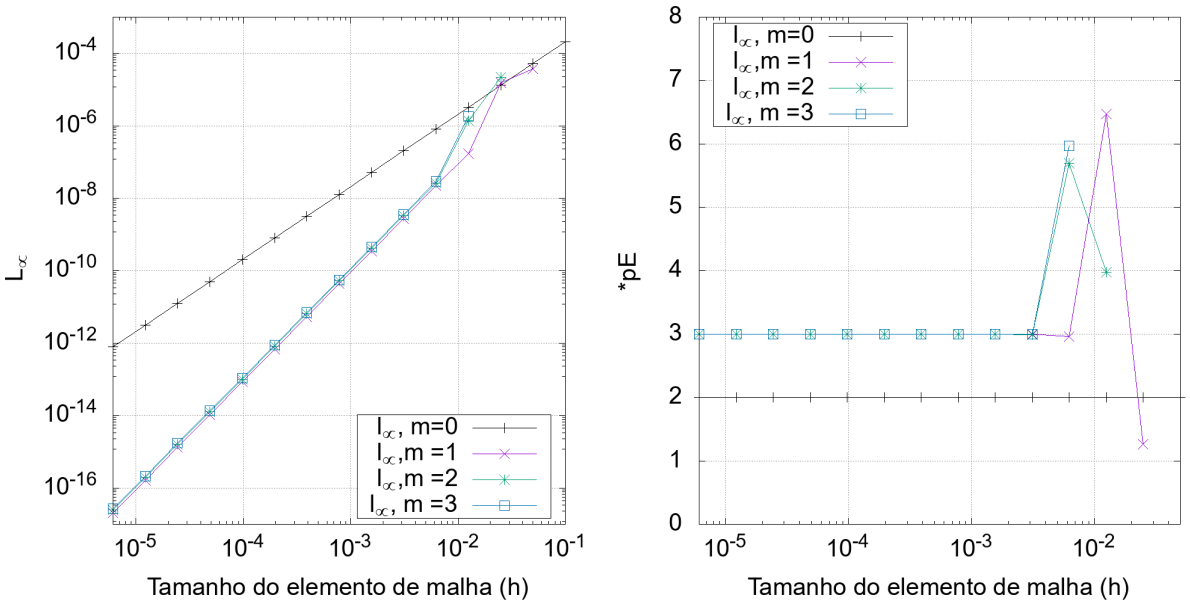
(d) p_E^*, Malha 2



FONTE: Autor(2020).

Figura 89 – Resultados: CRE-I, Variável l_∞, Burgers 1D com solução $T(x) = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme

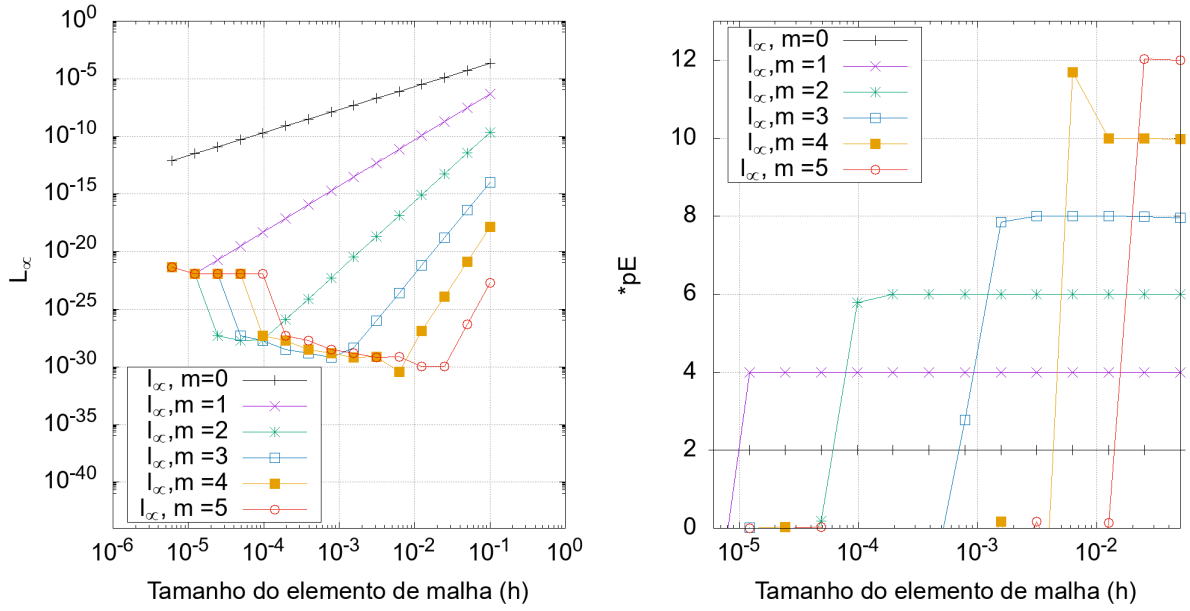
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 90 – Resultados: SER, Variável l_∞, Burgers 1D com solução $T(x) = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha não uniforme

(a) l_∞ (b) p_E^*



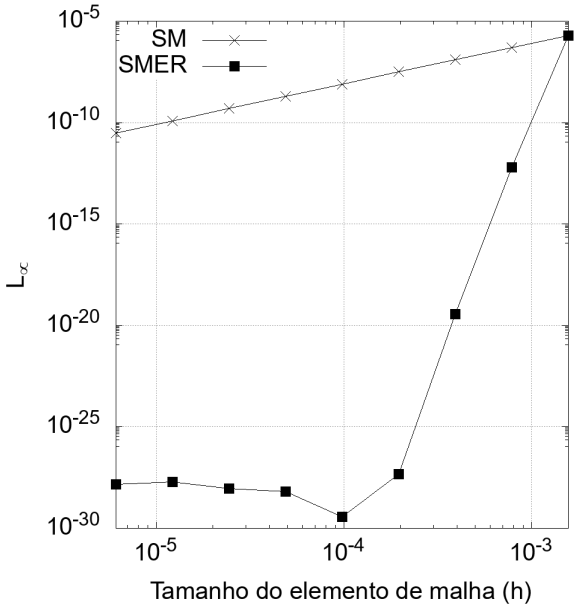
FONTE: Autor(2020).

C.2 MALHA UNIFORME 1D

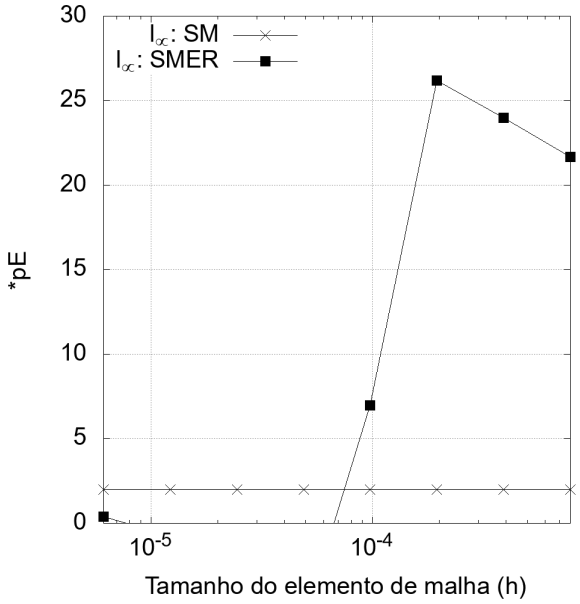
C.2.1 Poisson

Figura 91 – Resultados: SMER, Malhas 7-8, Variável l_∞, Poisson 1D com solução $T = \text{sen}(\pi x)$, Malha uniforme

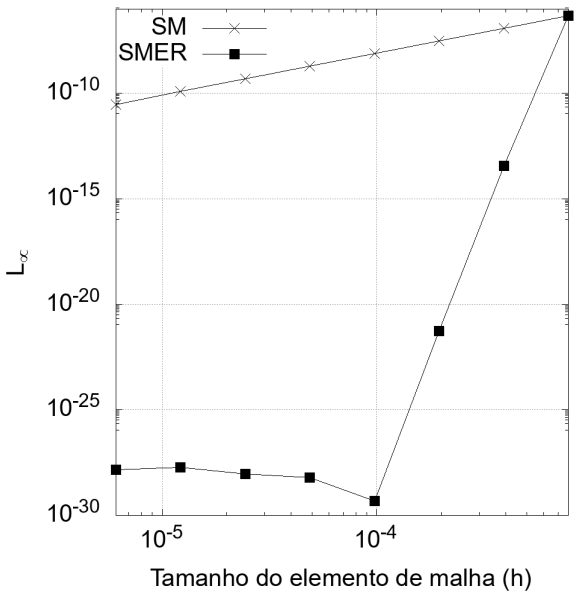
(a) l_∞, Malha 7



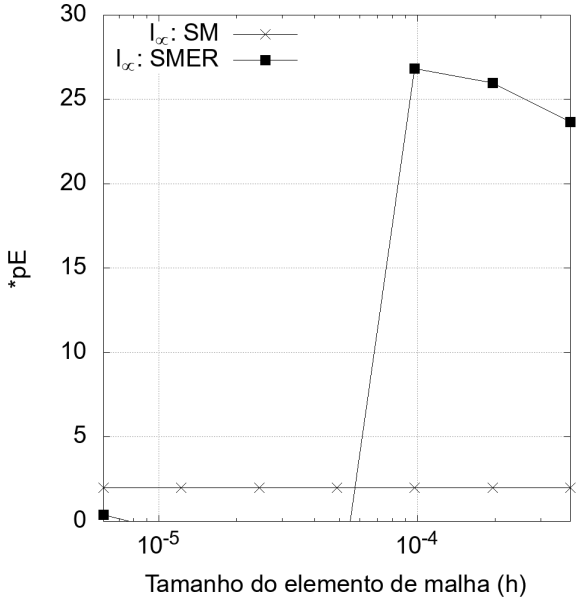
(b) p_E^*, Malha 7



(c) l_1, Malha 8



(d) p_E^*, Malha 8



FONTE: Autor(2020).

Tabela 17 – Resultados: SMER, Malha base 13, Variável l_∞, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme

Malha	h	l_∞: SM	l_∞: SMER	p_E^*: l_∞, SM	p_E^*: l_∞, SMER
13	2.44E-05	5.73062E-10	5.73062E-10		
14	1.22E-05	1.43266E-10	3.84268E-20	2.00000E+00	3.37959E+01
15	6.10E-06	3.58164E-11	3.66033E-28	2.00000E+00	2.66456E+01

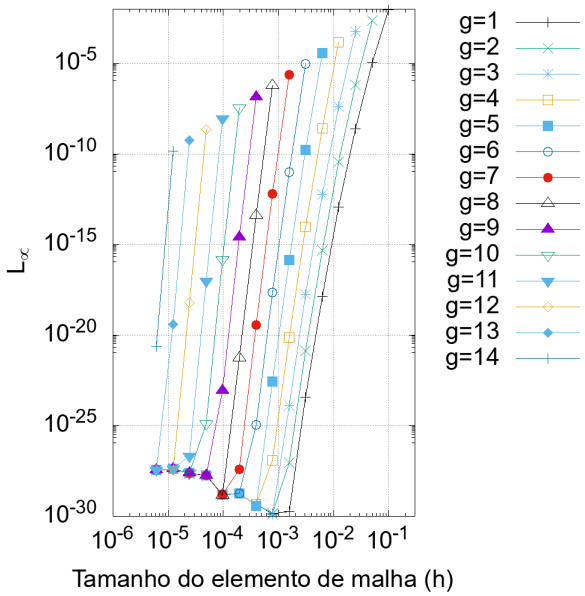
FONTE: Autor(2020).

Tabela 18 – Resultados: SMER, Malha base 14, Variável l_∞, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme

Malha	h	l_∞: SM	l_∞: SMER	p_E^*: l_∞, SM	p_E^*: l_∞, SMER
14	1.22E-05	1.43266E-10	1.43266E-10		
15	6.10E-06	3.58164E-11	2.40168E-21	2.00000E+00	3.57959E+01

FONTE: Autor(2020).

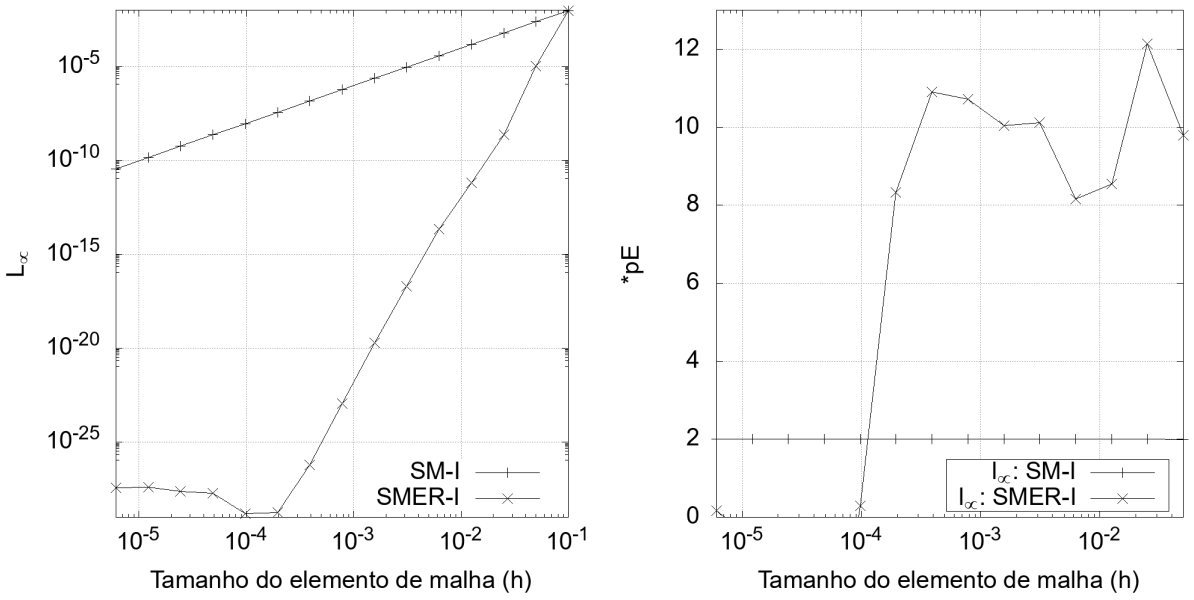
Figura 92 – Resultados: SMER, Malhas $g = 1{,}2\cdots{,}14$, Variável l_∞, Poisson 1D com solução $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme.



FONTE: Autor(2020).

Figura 93 – Resultados: SMER INTERPOLADO (SMER-I), Poisson 1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$, Variável l_∞, Malha uniforme, conjunto β_1

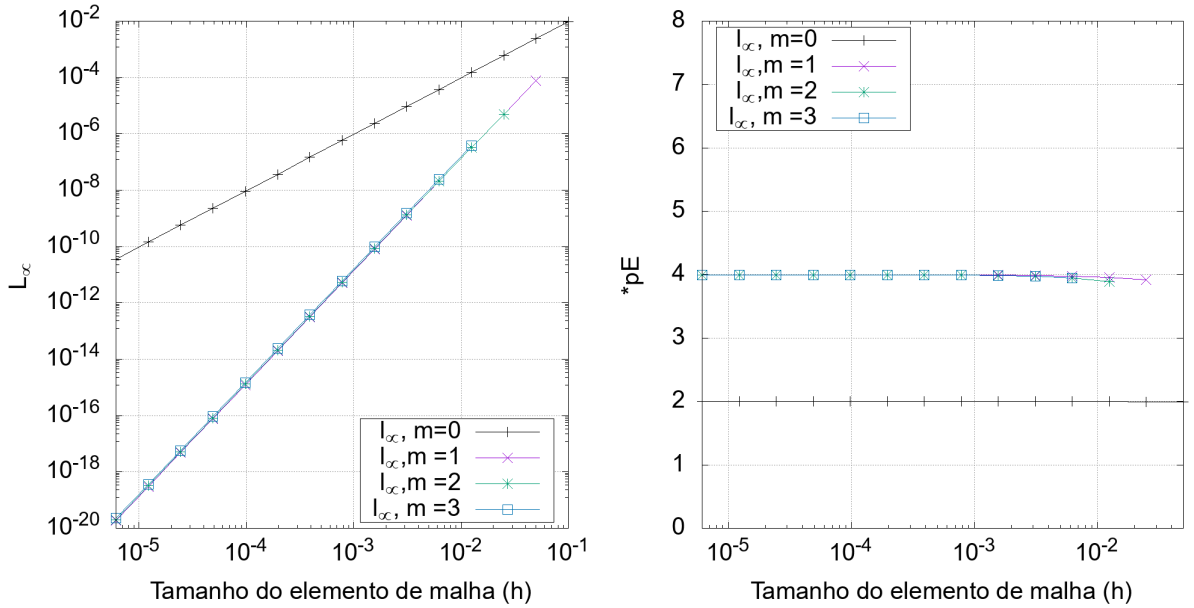
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 94 – Resultados: CRE clássico (CRE), Variável l_∞, Poisson 1D, $T = \frac{-1}{5}e^{3x} + 3x + 2$, Malha uniforme.

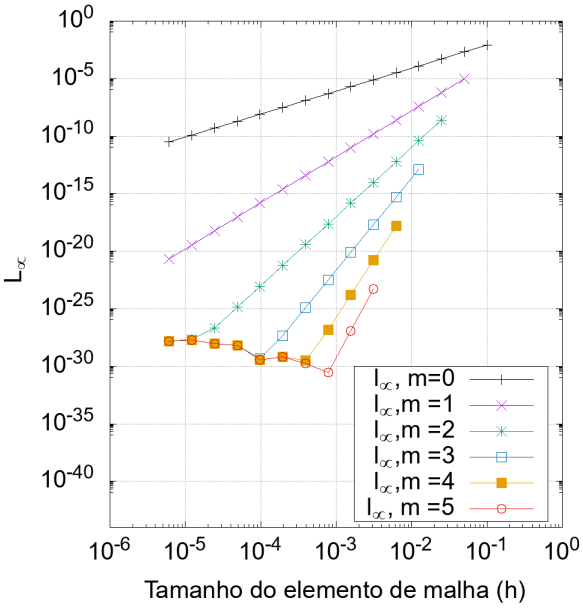
(a) l_∞ (b) p_E^*



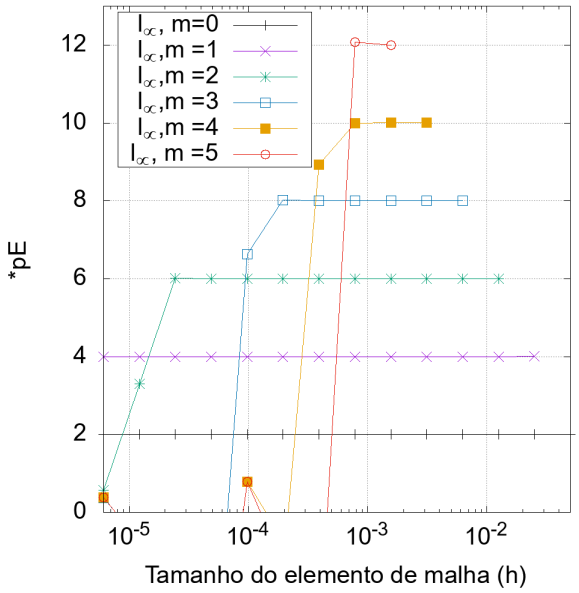
FONTE: Autor(2020).

Figura 95 – Resultados: Normas l_∞ e p^*_E, CRE com FRE (FRE), Poisson 1D, Malha uniforme

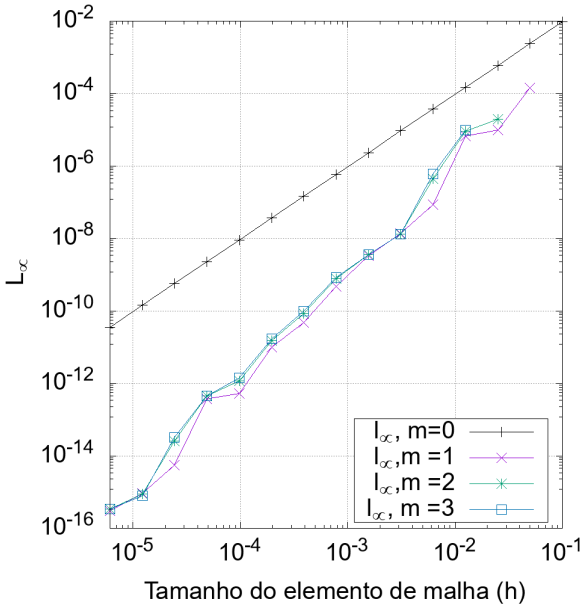
(a) l_∞, $T = \text{sen}(\pi x)$



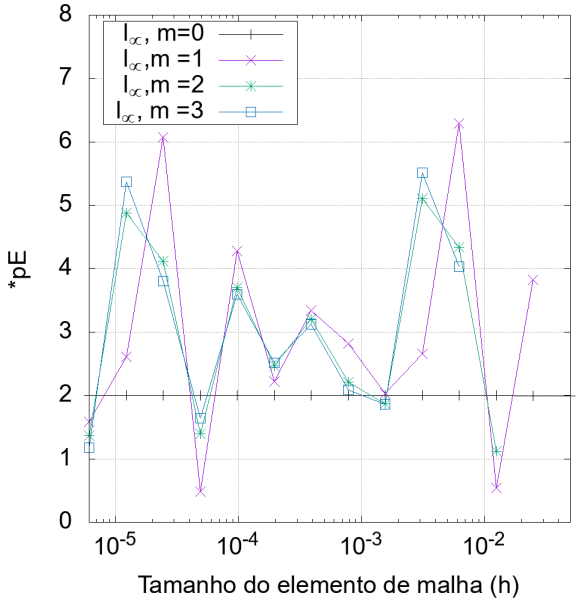
(b) p^*_E, $T = \text{sen}(\pi x)$



(c) l_∞, $T = \frac{-1}{5}e^{3x} + 3x + 2$



(d) p^*_E, $T = \frac{-1}{5}e^{3x} + 3x + 2$

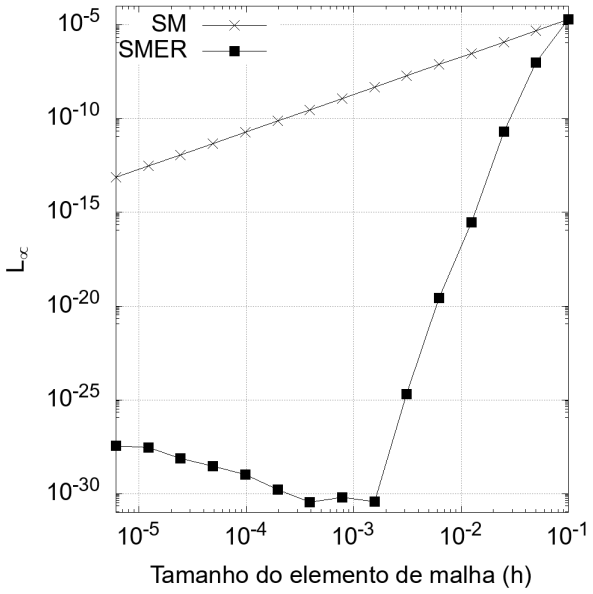


FONTE: Autor(2020).

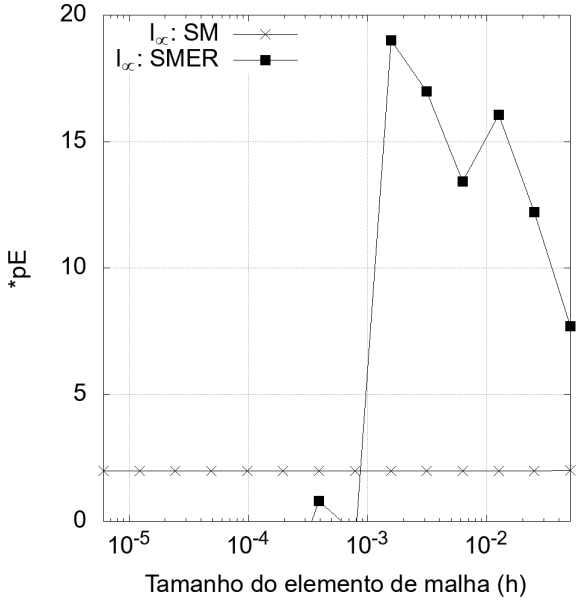
C.2.2 Burgers

Figura 96 – Resultados: SMER, Malhas 1-2, Variável l_∞, Burgers 1D com solução $T(x) = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, $Re = 1$, Malha uniforme

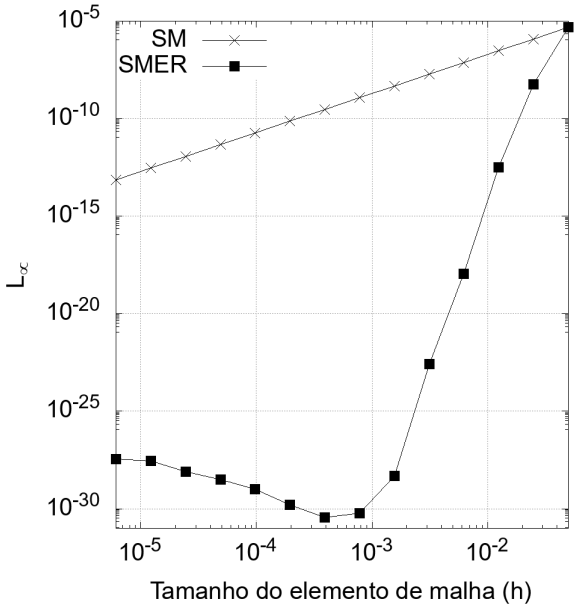
(a) l_∞, Malha 1



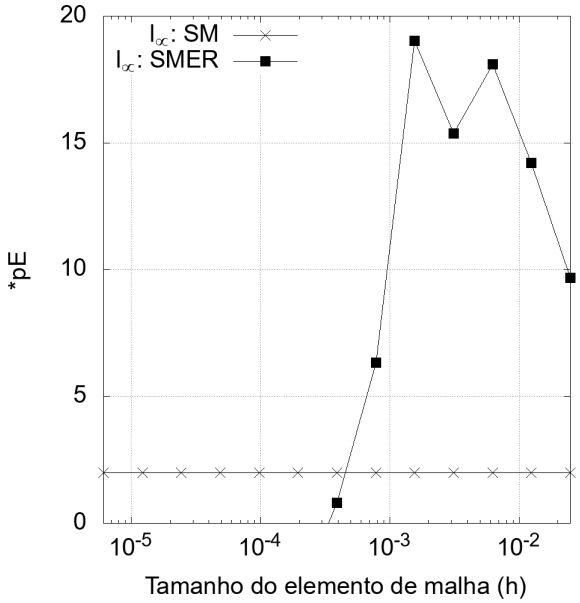
(b) p_E^*, Malha 1



(c) l_∞, Malha 2



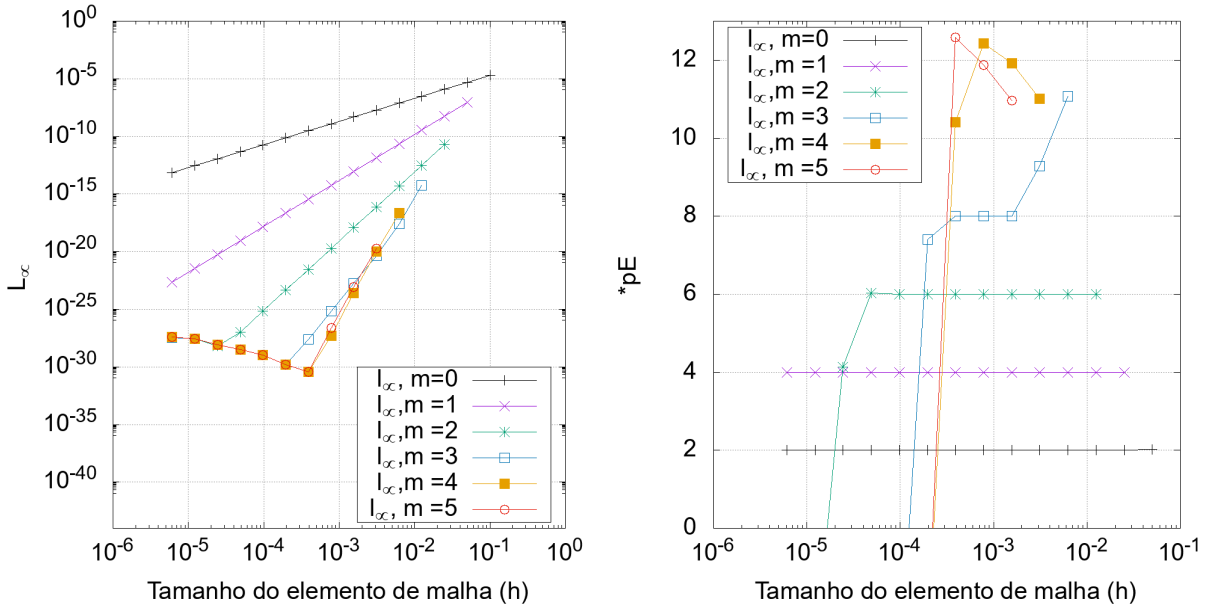
(d) p_E^*, Malha 2



FONTE: Autor(2020).

Figura 97 – Resultados: CRE com interpolação (CRE-I), Variável l_∞, Burgers 1D,$Re = 1$, $T = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, Malha uniforme

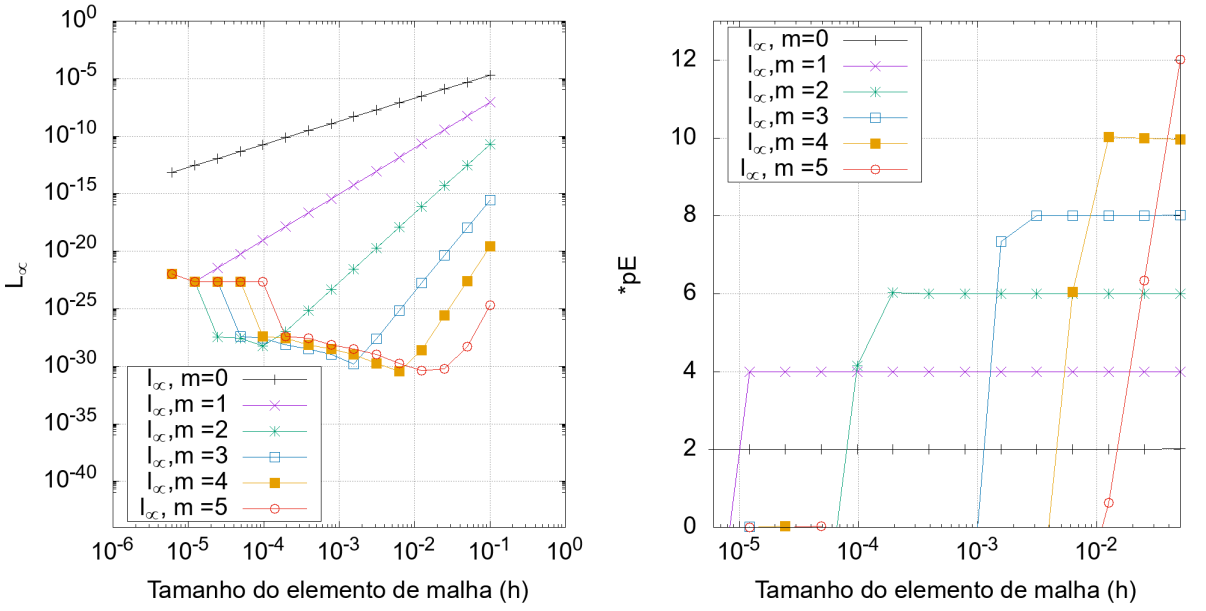
(a) l_∞. (b) p^*_E.



FONTE: Autor(2020).

Figura 98 – Resultados: SER, Variável l_∞, Burgers 1D,$Re = 1$, $T = \dfrac{e^{xRe} - 1}{e^{Re} - 1}$, Malha uniforme

(a) l_∞ (b) p^*_E



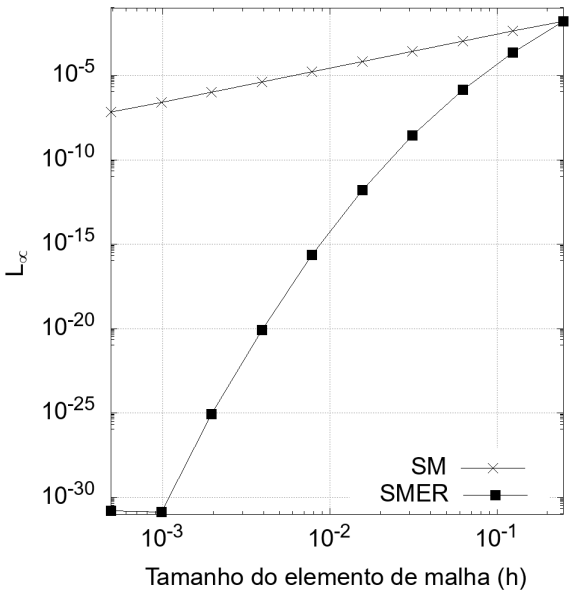
FONTE: Autor(2020).

C.3 LAPLACE 2D

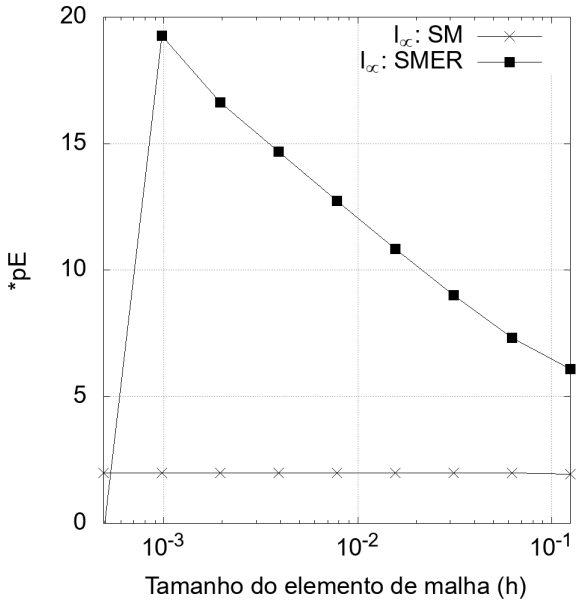
C.3.1 Malha uniforme

Figura 99 – Resultados: SMER, Malhas 1 e 6, Variável l_∞, Laplace 2D, Malha uniforme

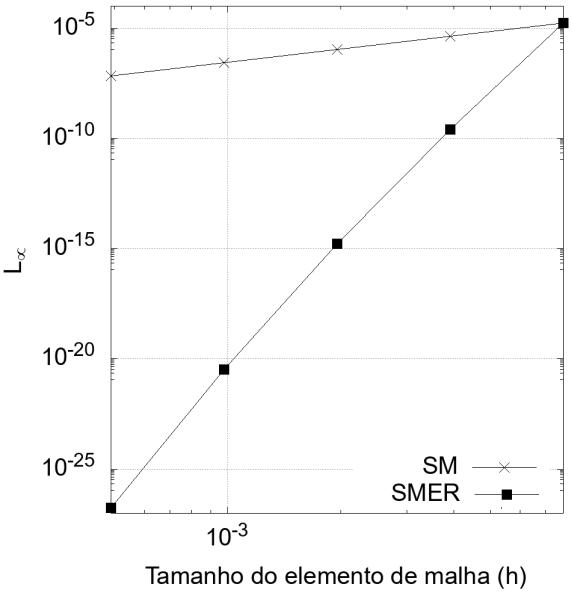
(a) l_∞, Malha 1



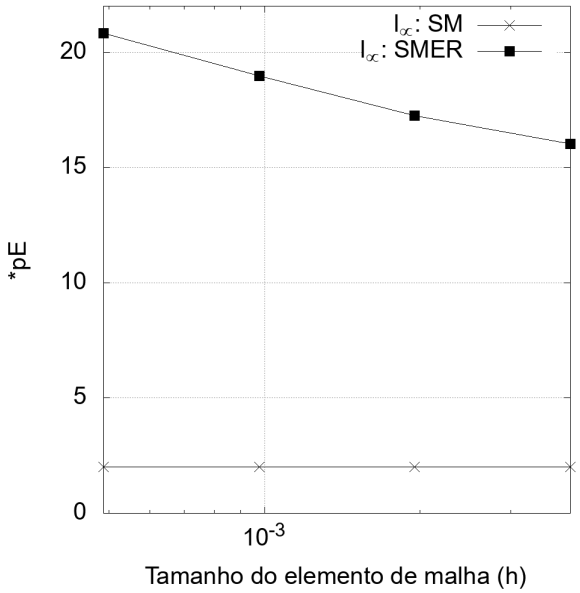
(b) p_E^*, Malha 1



(c) l_∞, Malha 6

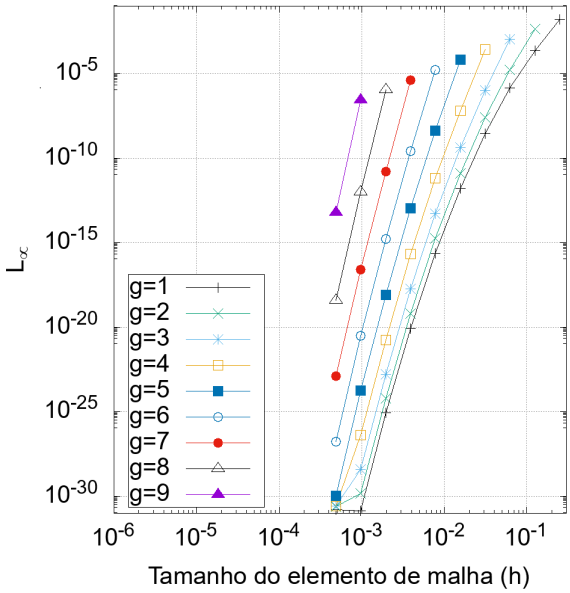


(d) p_E^*, Malha 6



FONTE: Autor(2020).

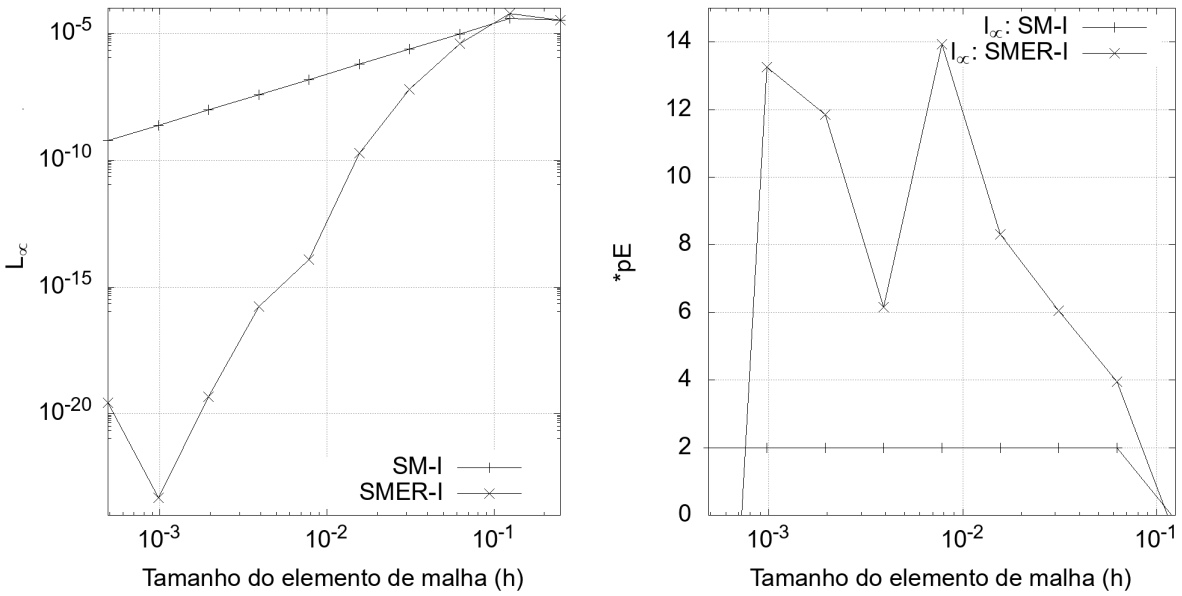
Figura 100 – Resultados: SMER, Malhas $g = 1,2,\cdots,9$, Variável l_∞, Laplace 2D, Malha uniforme



FONTE: Autor(2020).

Figura 101 – Resultados: SMER-I, Conjunto β_3, Variável l_∞, Laplace 2D, Malha uniforme

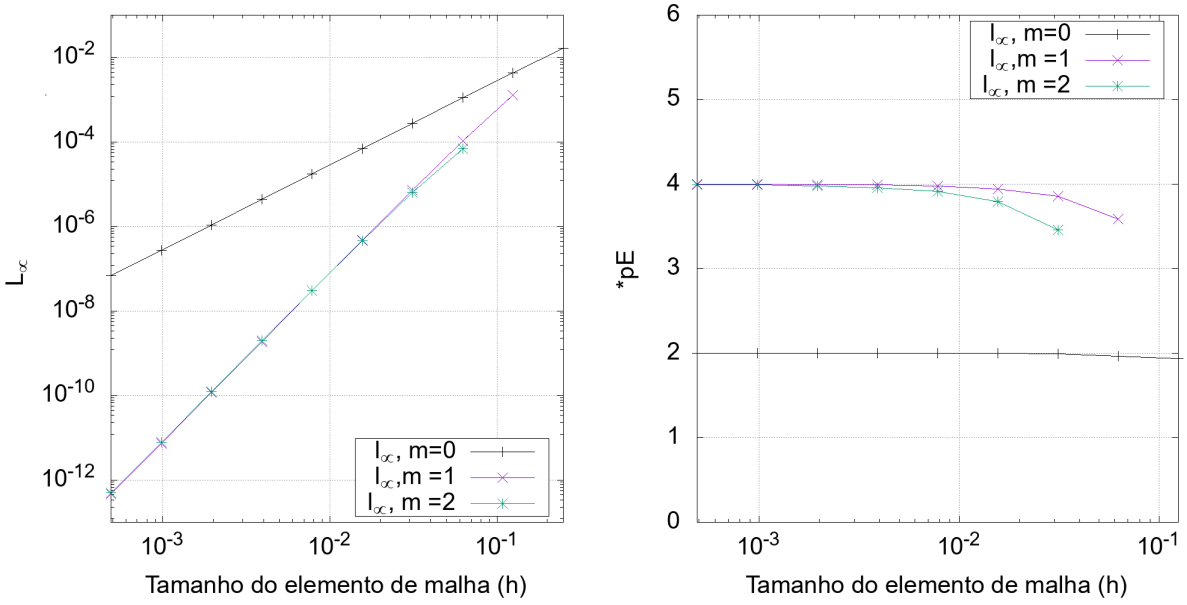
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 102 – Resultados: CRE, Variável l_∞, Laplace 2D, Malha uniforme

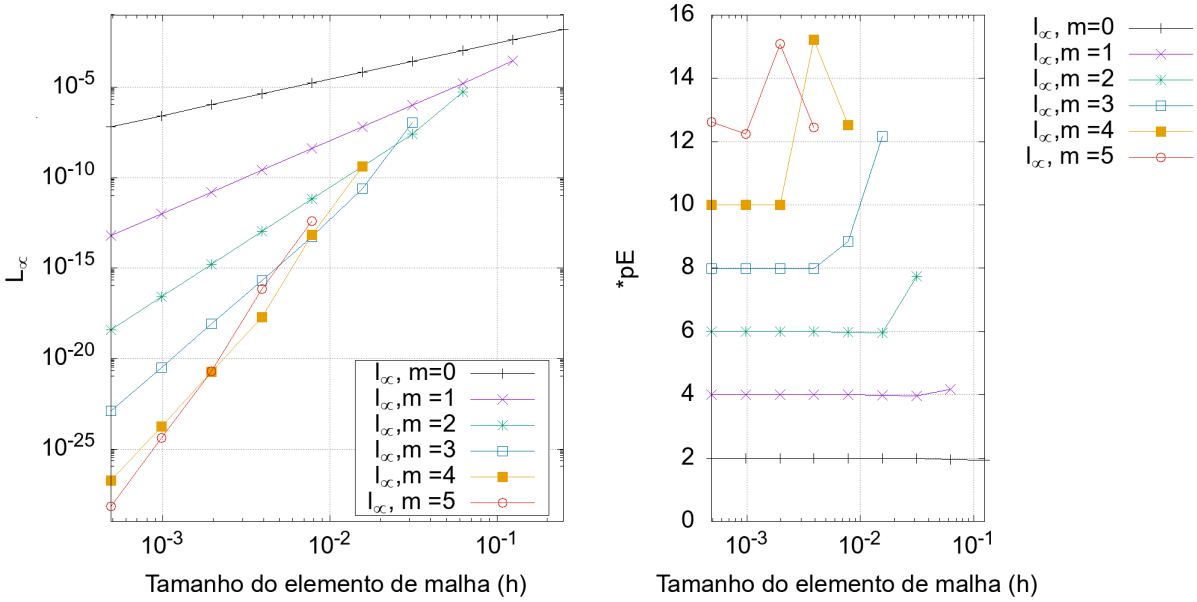
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 103 – Resultados: CRE-I, Variável l_∞, Laplace 2D, Malha uniforme

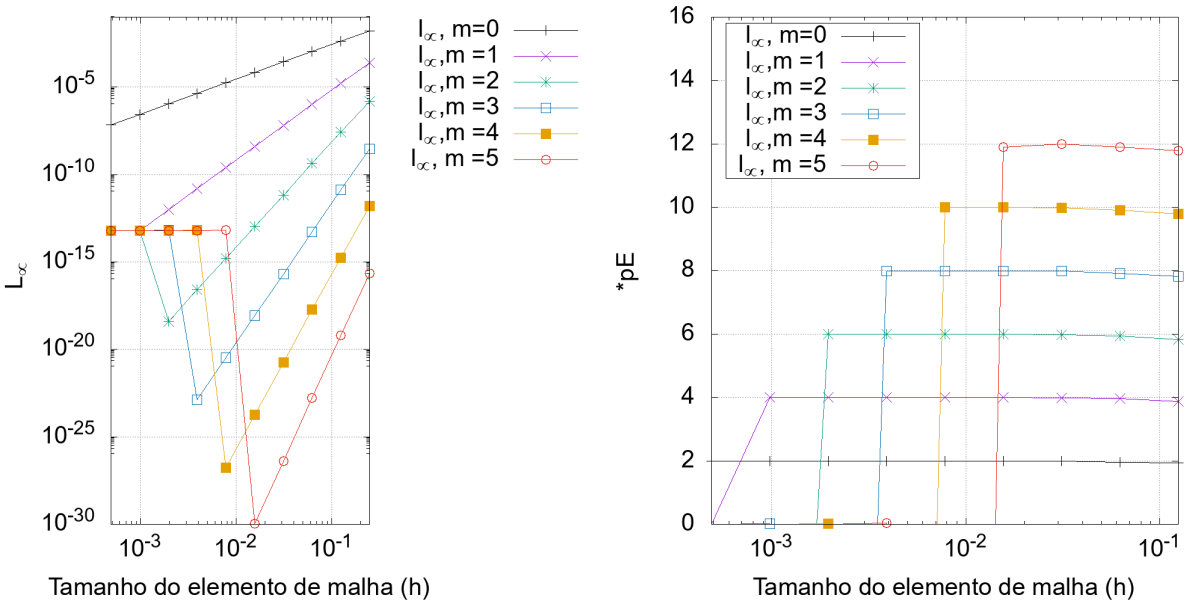
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 104 – Resultados: SER, Variável l_∞, Laplace 2D, Malha uniforme

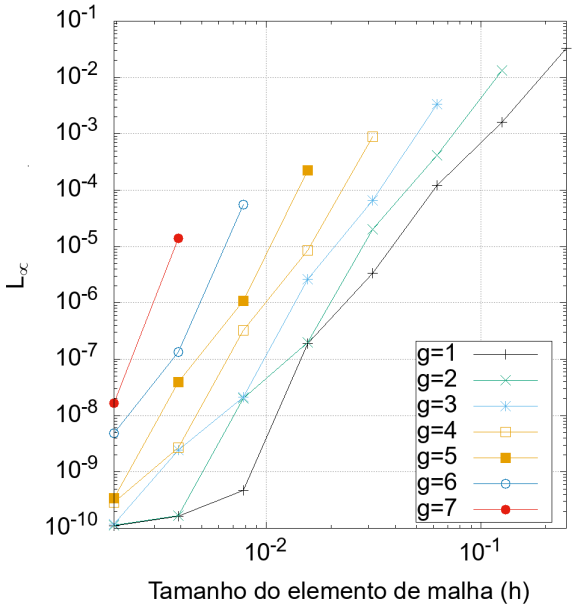
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

C.3.2 Malha não uniforme

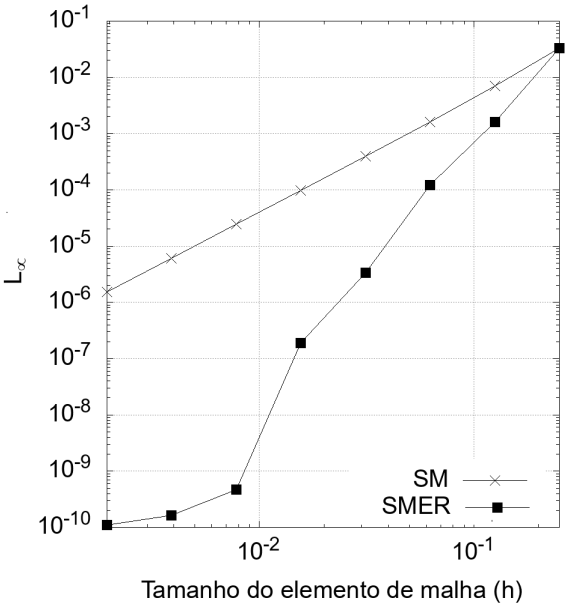
Figura 105 – Resultados: SMER, Malhas $g = 1,2,\cdots,9$, Variável l_∞, Laplace 2D, Malha não uniforme



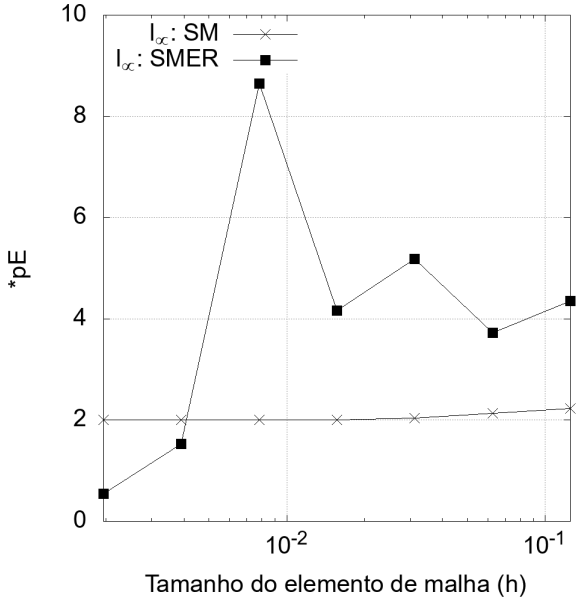
FONTE: Autor(2020).

Figura 106 – Resultados: SMER, Malhas 1 e 4, Variável l_∞, Laplace 2D, Malha não uniforme

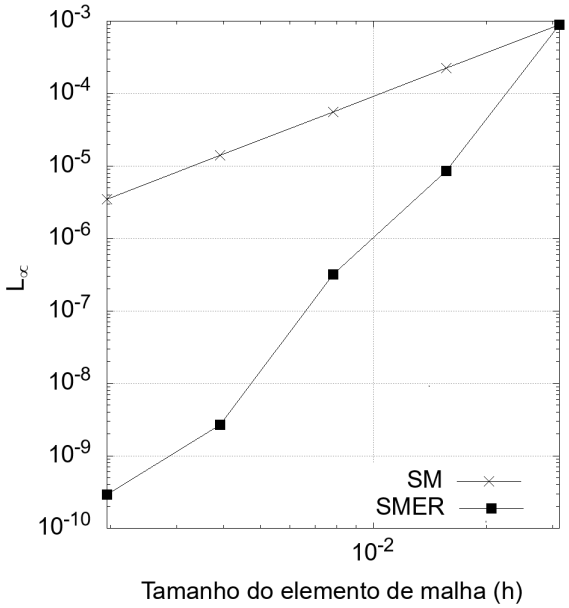
(a) l_∞, Malha 1



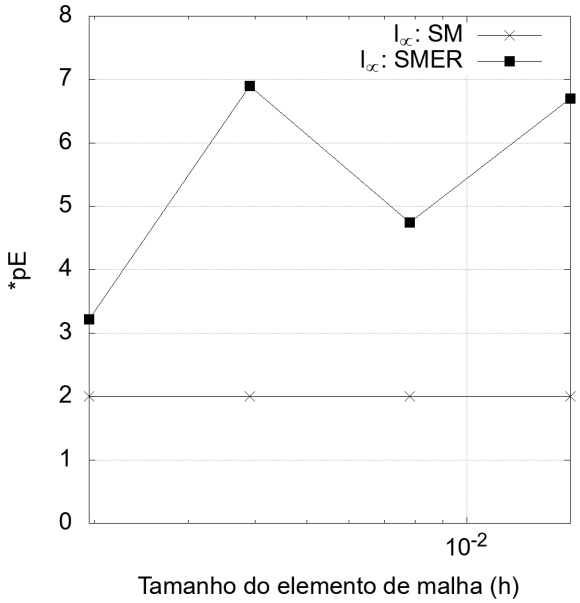
(b) p_E^*, Malha 1



(c) l_∞, Malha 4



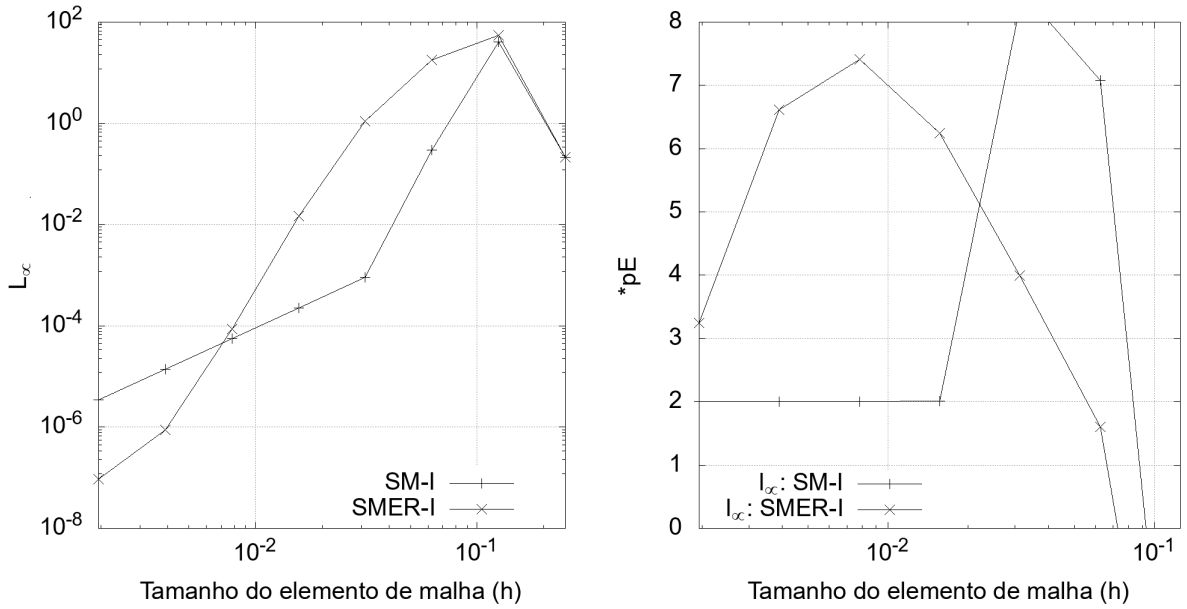
(d) p_E^*, Malha 4



FONTE: Autor(2020).

Figura 107 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Laplace 2D, Malha não uniforme

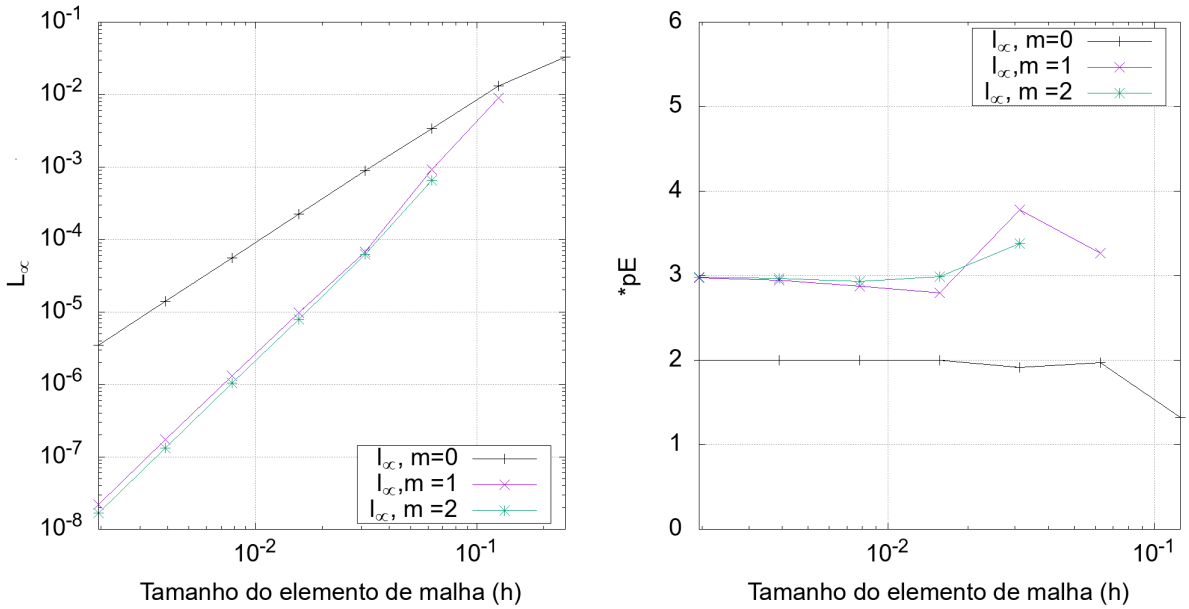
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 108 – Resultados: CRE, Variável l_∞, Laplace 2D, Malha não uniforme

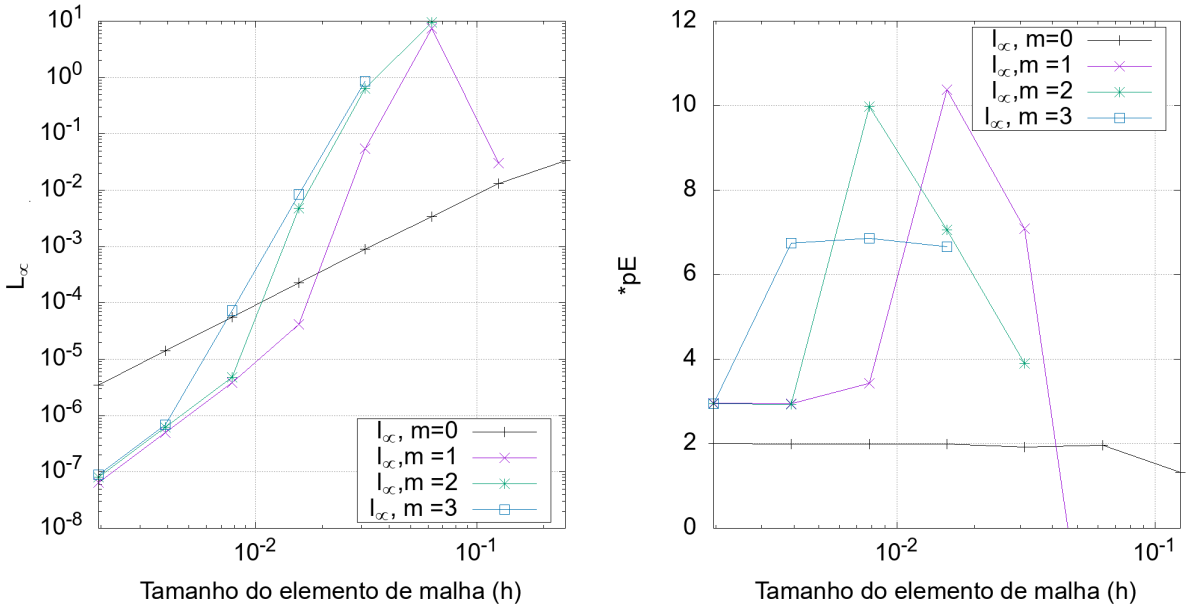
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 109 – Resultados: CRE-I, Variável l_∞, Laplace 2D, Malha não uniforme

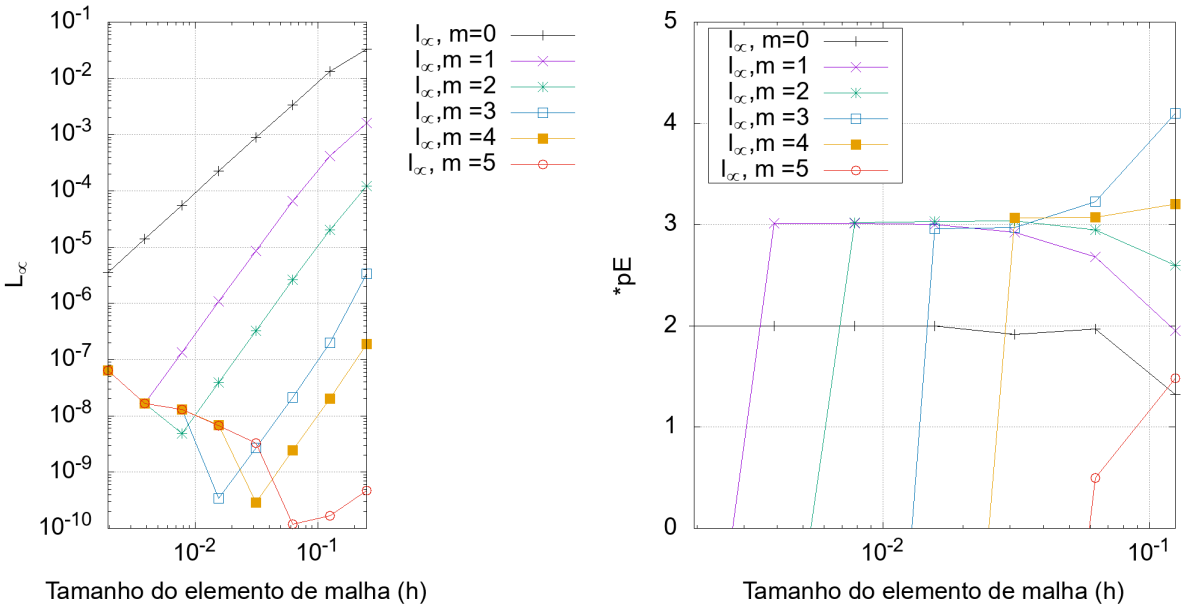
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 110 – Resultados: SER, Variável l_∞, Laplace 2D, Malha não uniforme

(a) l_∞ (b) p_E^*

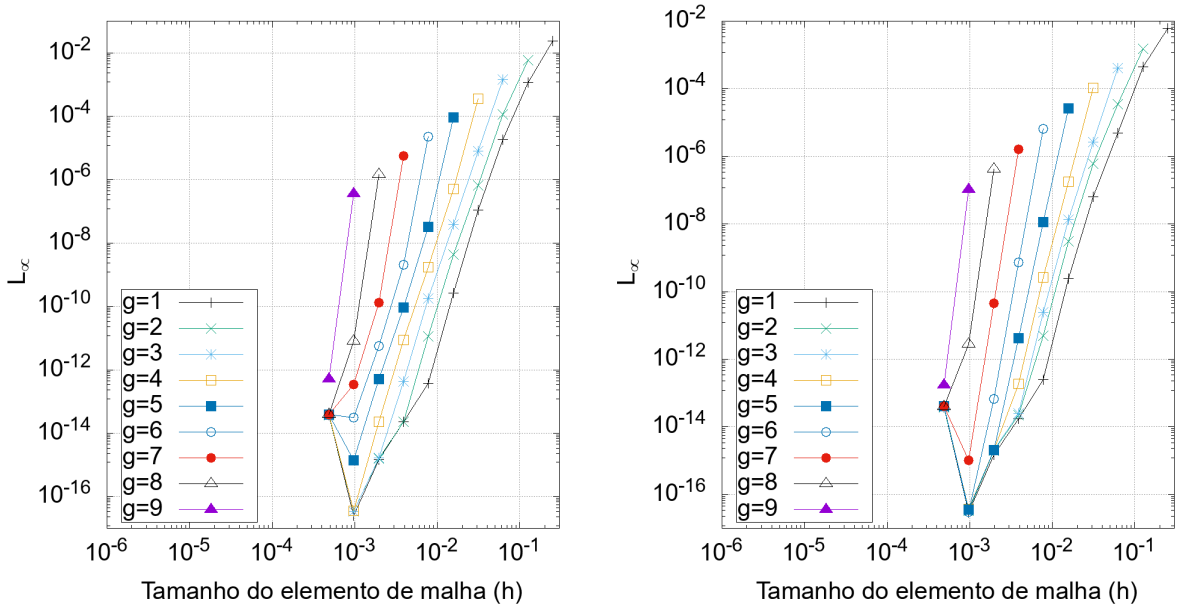


FONTE: Autor(2020).

C.4 BURGERS 2D

Figura 111 – Resultados: SMER, Malhas $g = 1{,}2{,}\cdots{,}9$, Burgers 2D, Variável ł$_\infty$, $Re = 1$

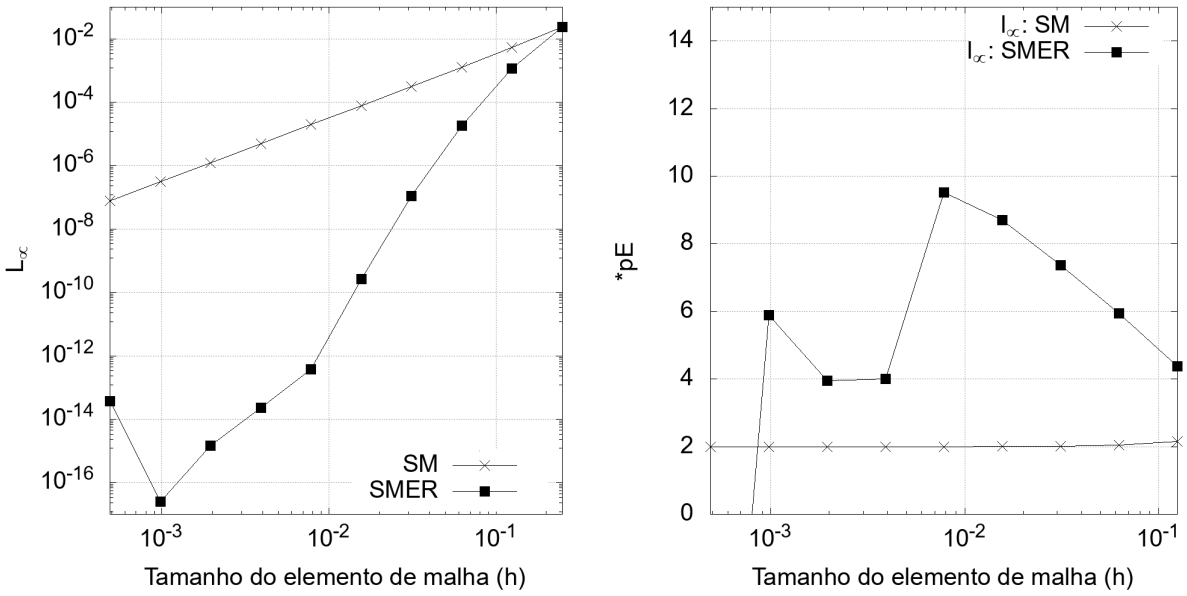
(a) Velocidade u (b) Velocidade v



FONTE: Autor(2020).

Figura 112 – Resultados: SMER, Malha base 1, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u

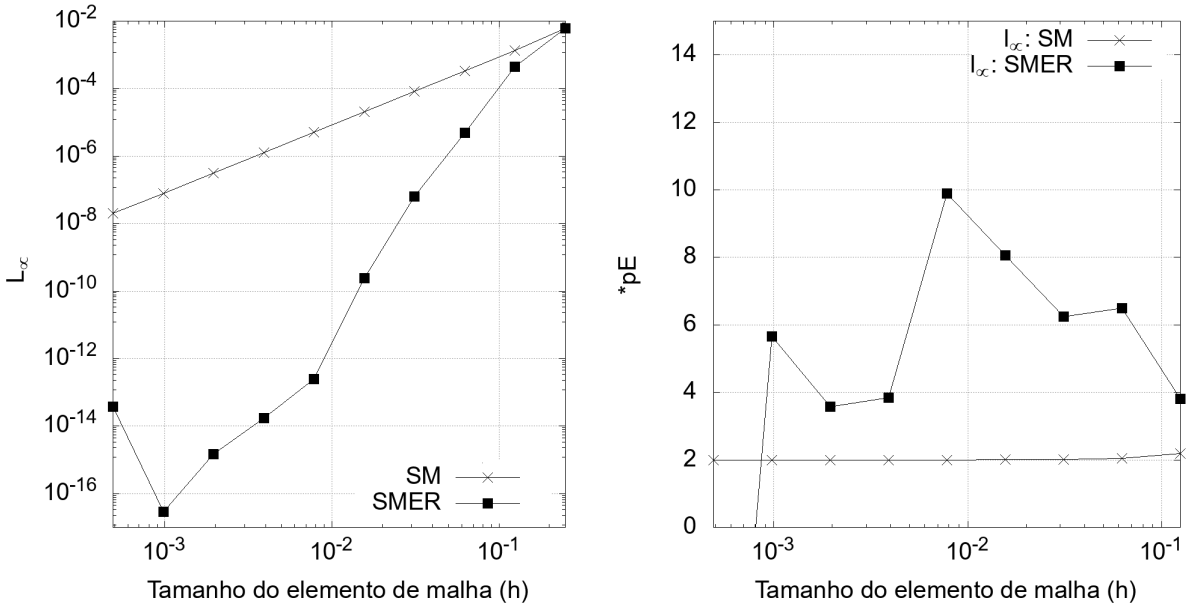
(a) l_∞, Malha 1 (b) p^*_E, Malha 1



FONTE: Autor(2020).

Figura 113 – Resultados: SMER, Malha base 1, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v

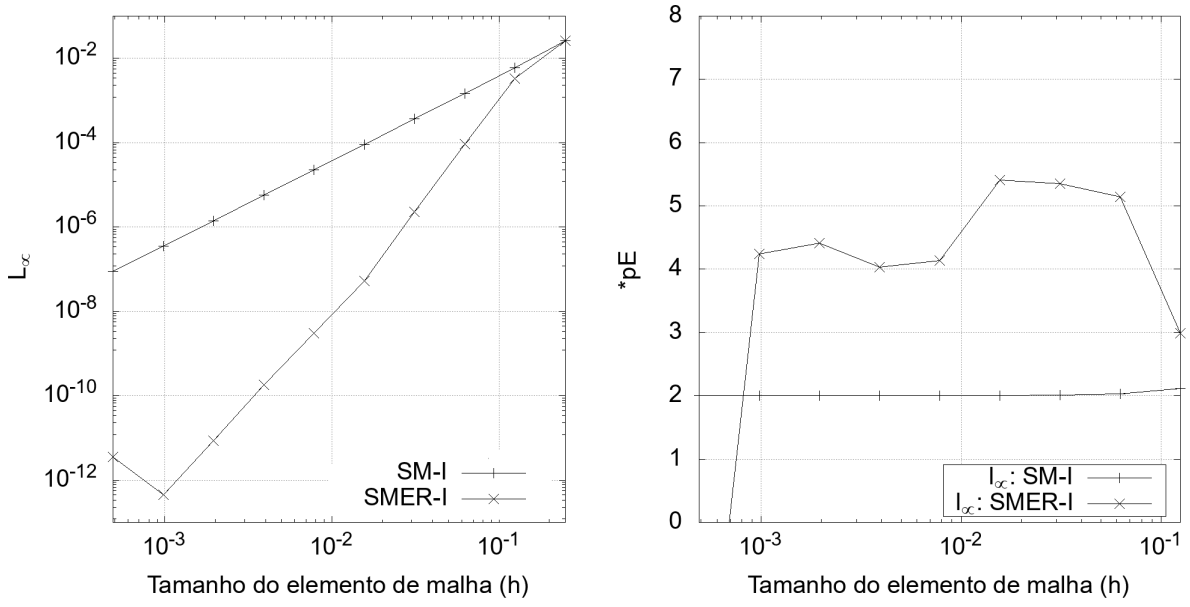
(a) l_∞, Malha 1 (b) p^*_E, Malha 1



FONTE: Autor(2020).

Figura 114 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u

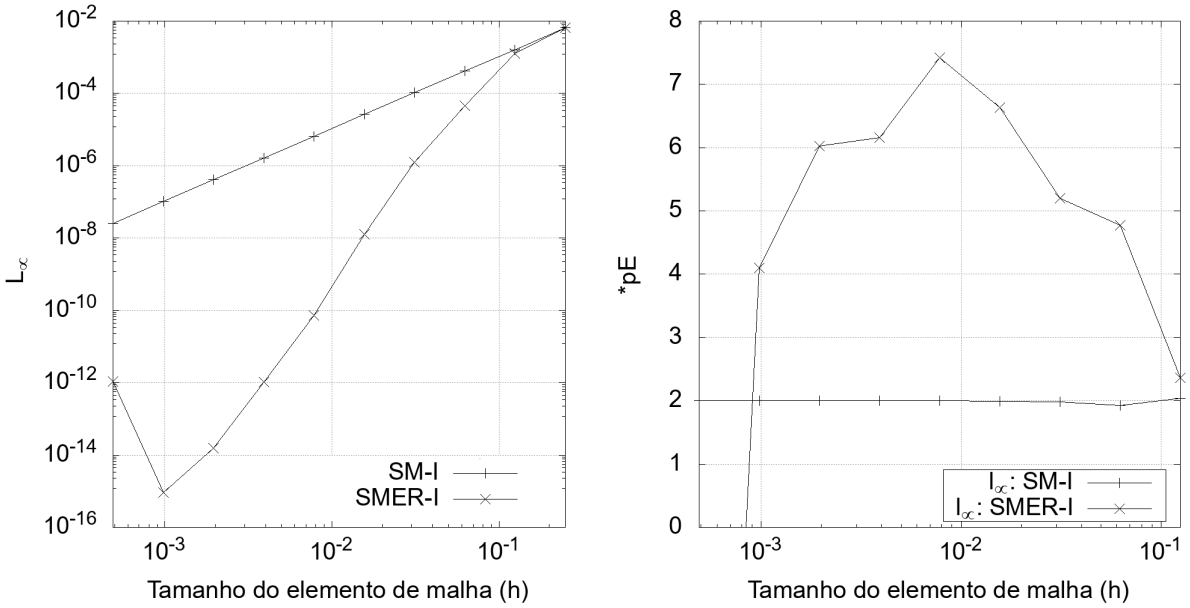
(a) l_∞ (b) p^*_E



FONTE: Autor(2020).

Figura 115 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v

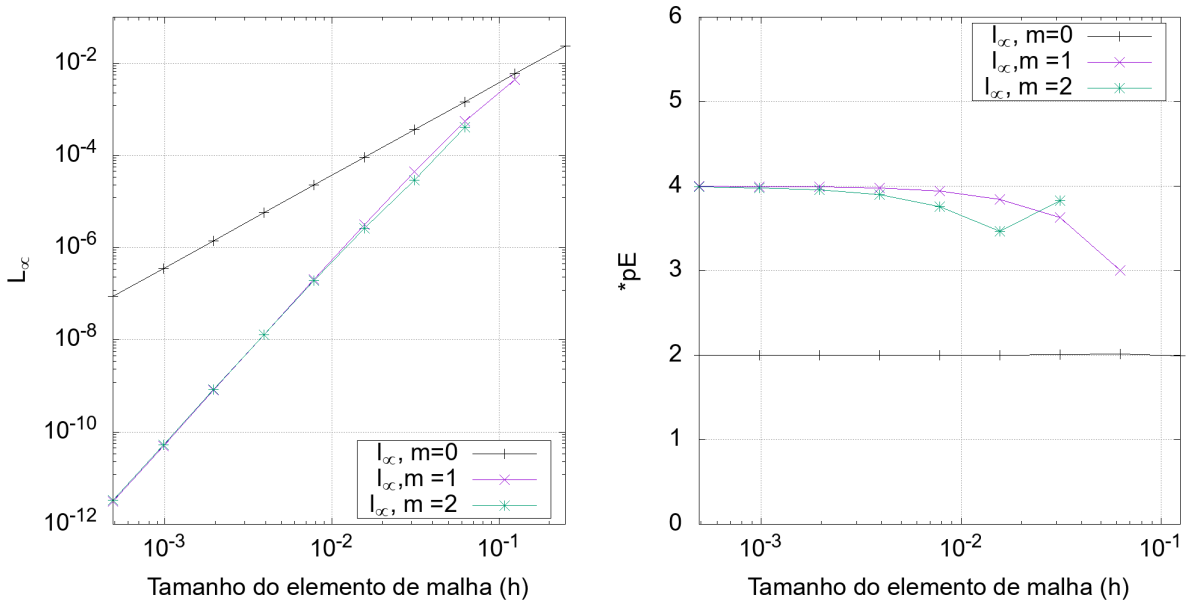
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 116 – Resultados: CRE, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u.

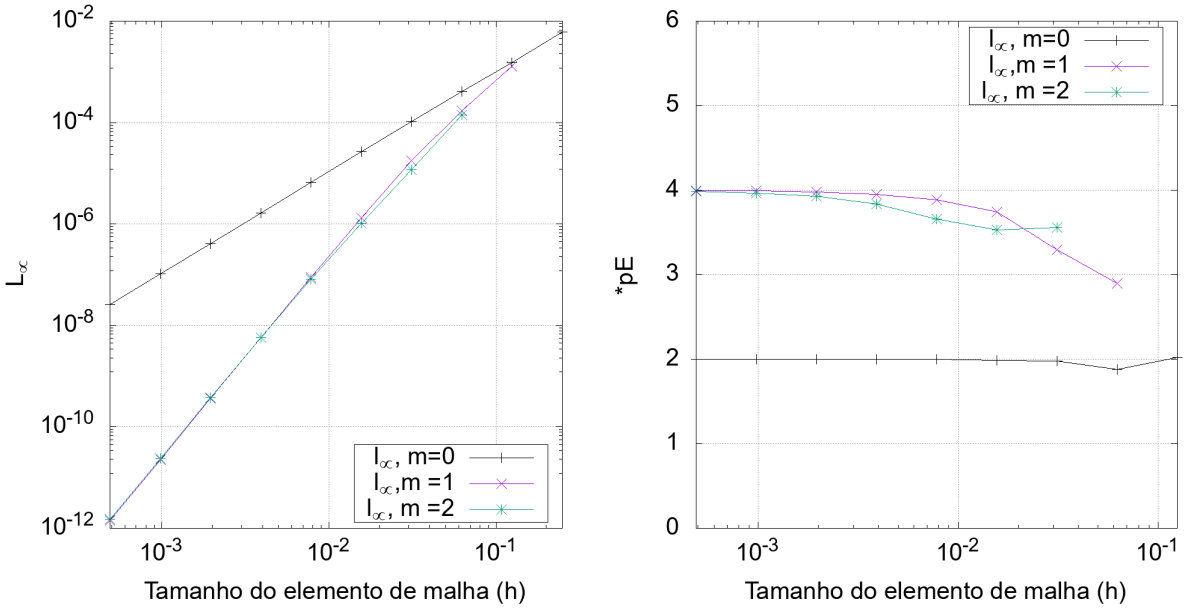
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 117 – Resultados: CRE, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v

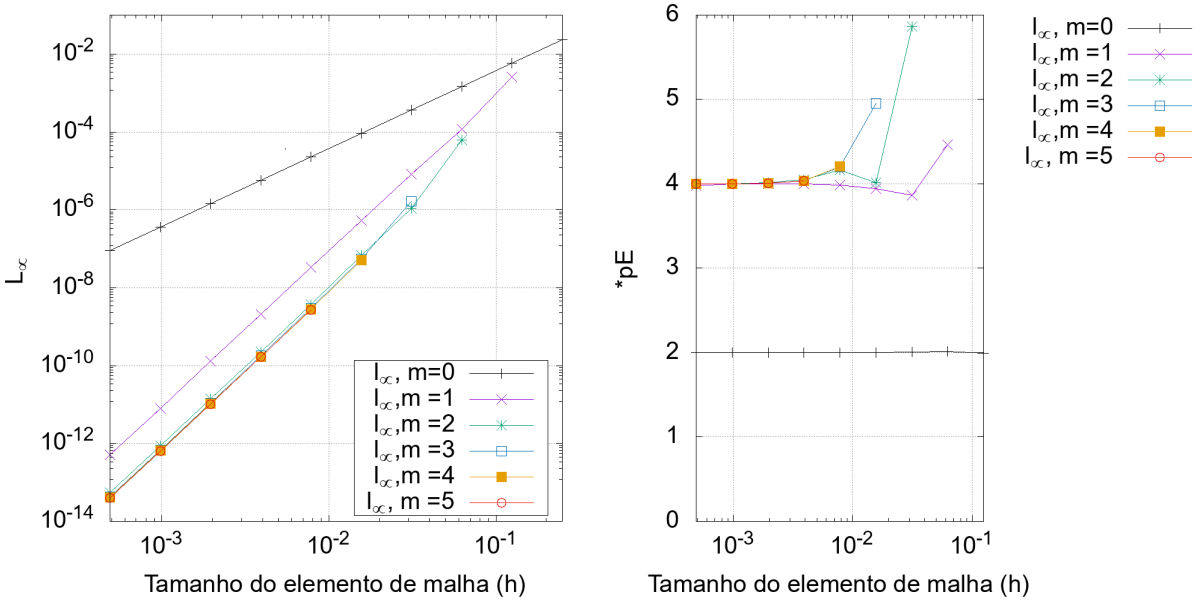
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 118 – Resultados: CRE-I, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u

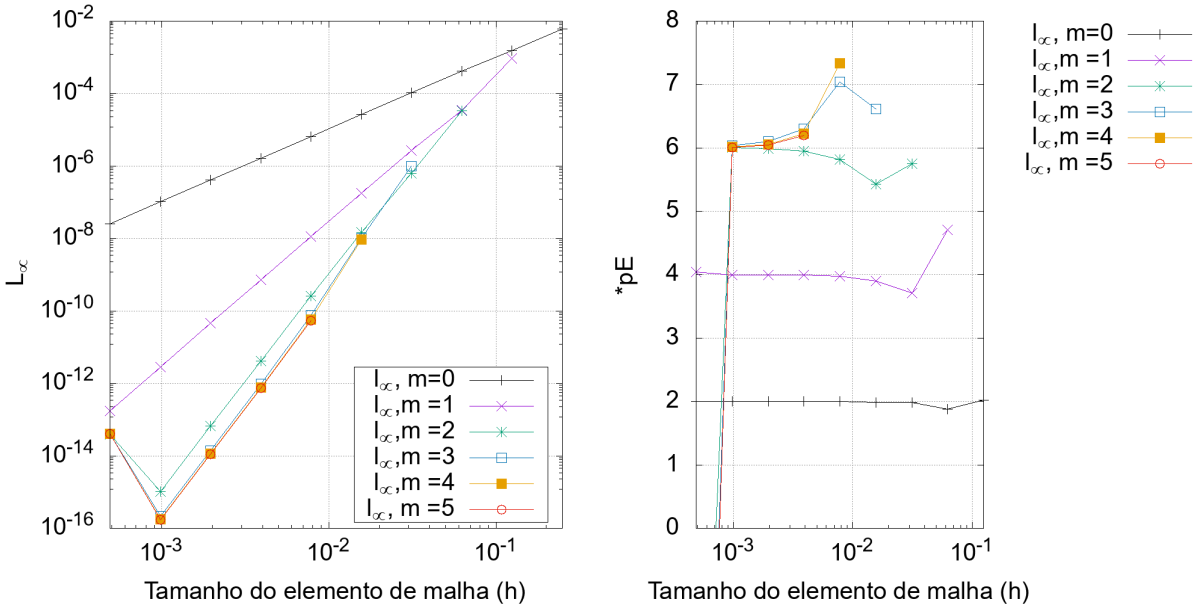
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 119 – Resultados: CRE-I, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v

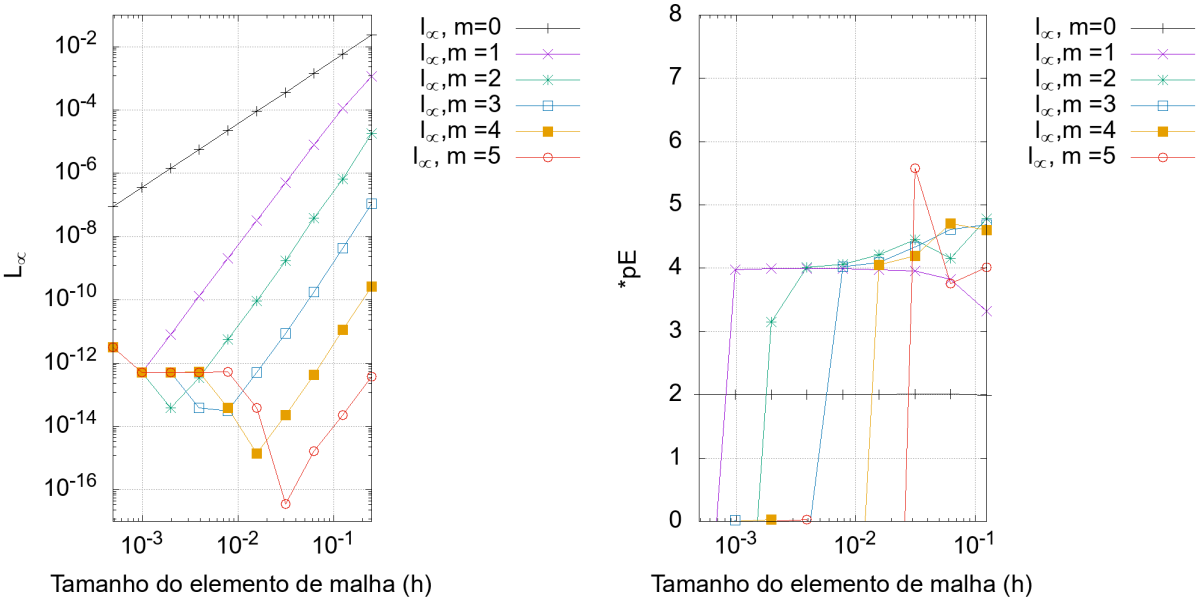
(a) l_∞ (b) p^*_E



FONTE: Autor(2020).

Figura 120 – Resultados: SER, Variável l_∞, Burgers 2D, $Re = 1$, velocidade u

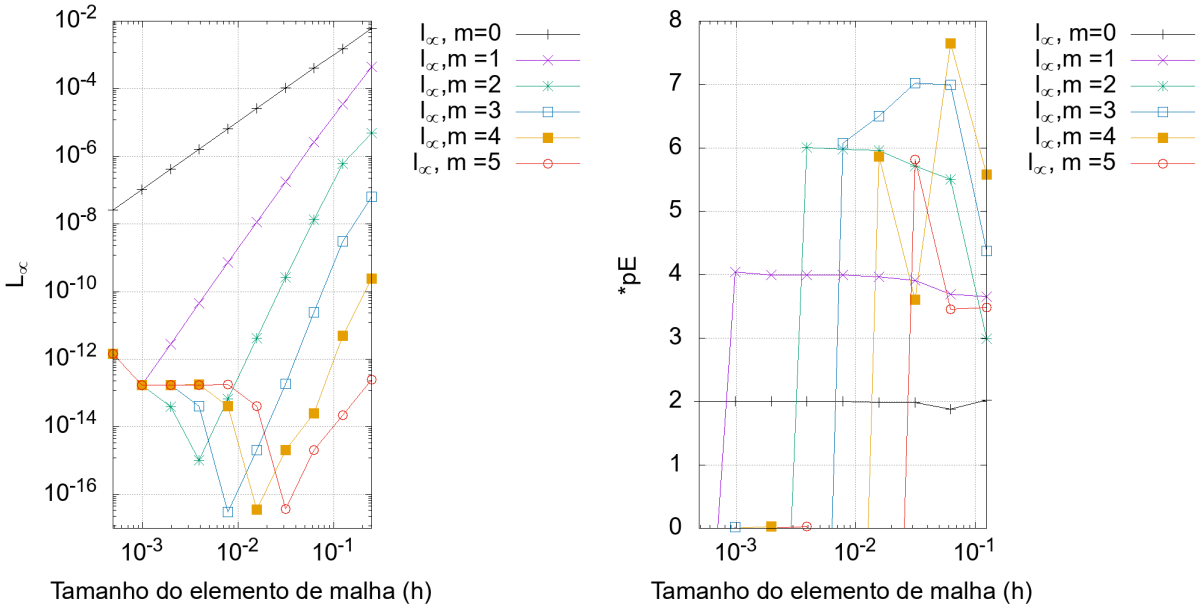
(a) l_∞ (b) p^*_E



FONTE: Autor(2020).

Figura 121 – Resultados: SER, Variável l_∞, Burgers 2D, $Re = 1$, velocidade v

(a) l_∞ (b) p^*_E

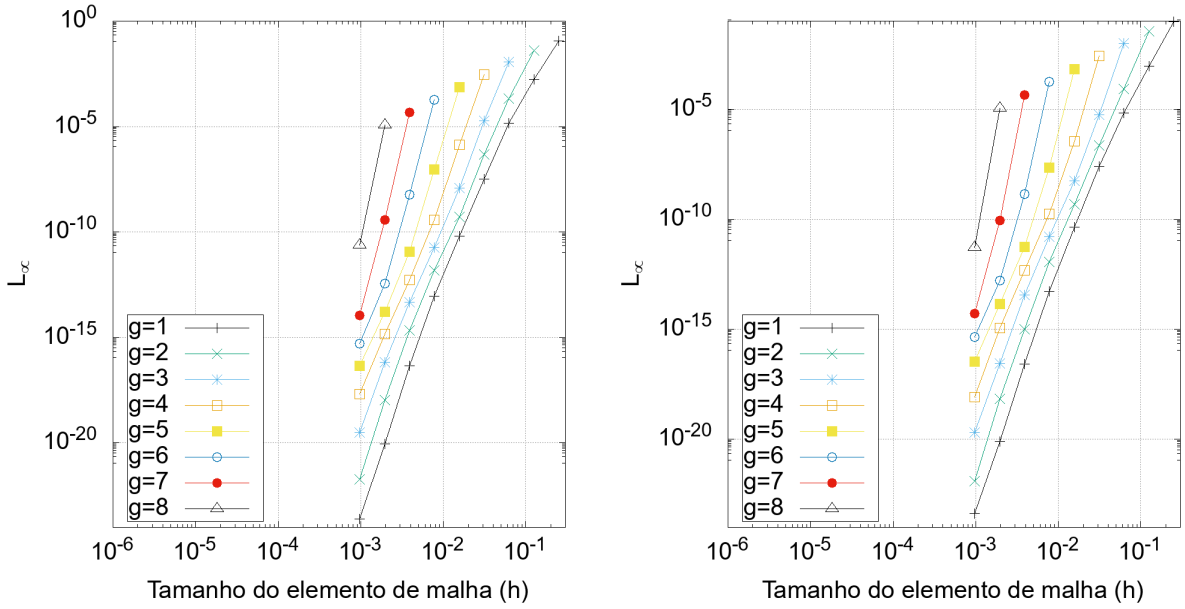


FONTE: Autor(2020).

C.5 NAVIER-STOKES

Figura 122 – Resultados: SMER, Variável ł$_\infty$, Malhas $g = 1,2,\cdots,8$, Navier-Stokes 2D

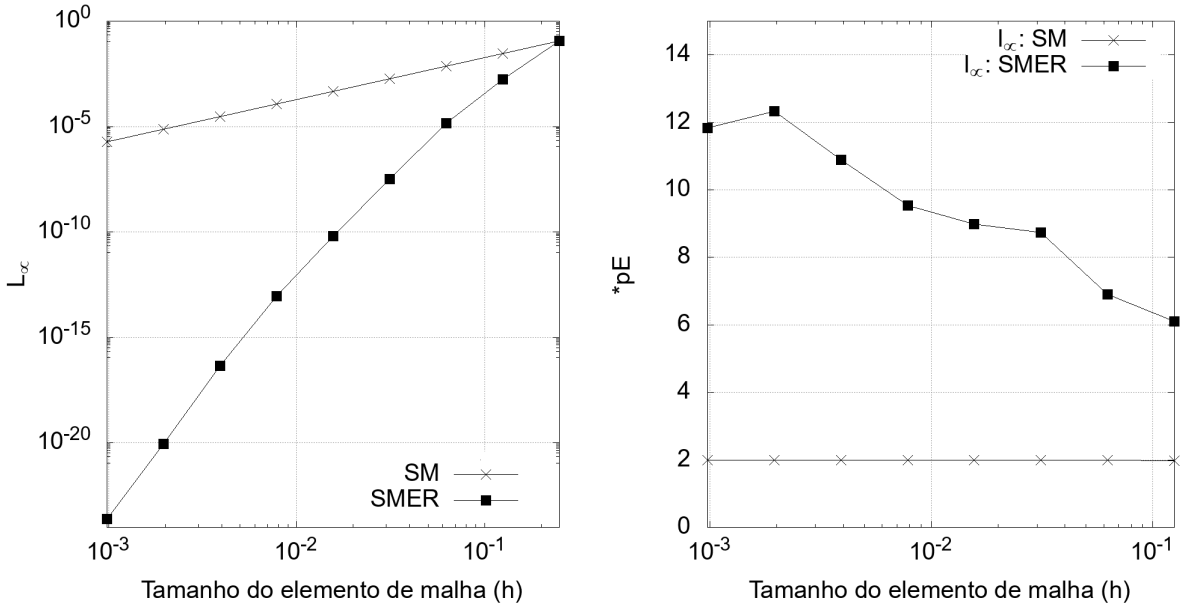
(a) Velocidade u (b) Velocidade v



FONTE: Autor(2020).

Figura 123 – Resultados: SMER, Malha base 1, Variável l_∞, Navier-Stokes 2D, velocidade u

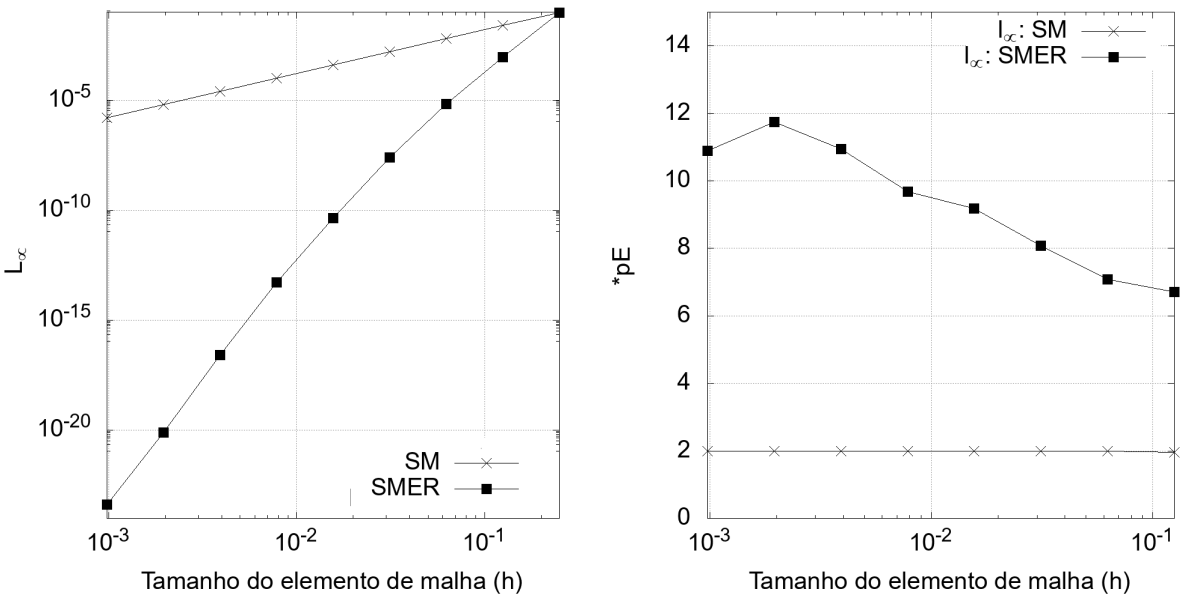
(a) l_∞, Malha 1 (b) p_E^*, Malha 1



FONTE: Autor(2020).

Figura 124 – Resultados: SMER, Malha base 1, Variável l_∞, Navier-Stokes 2D, velocidade v

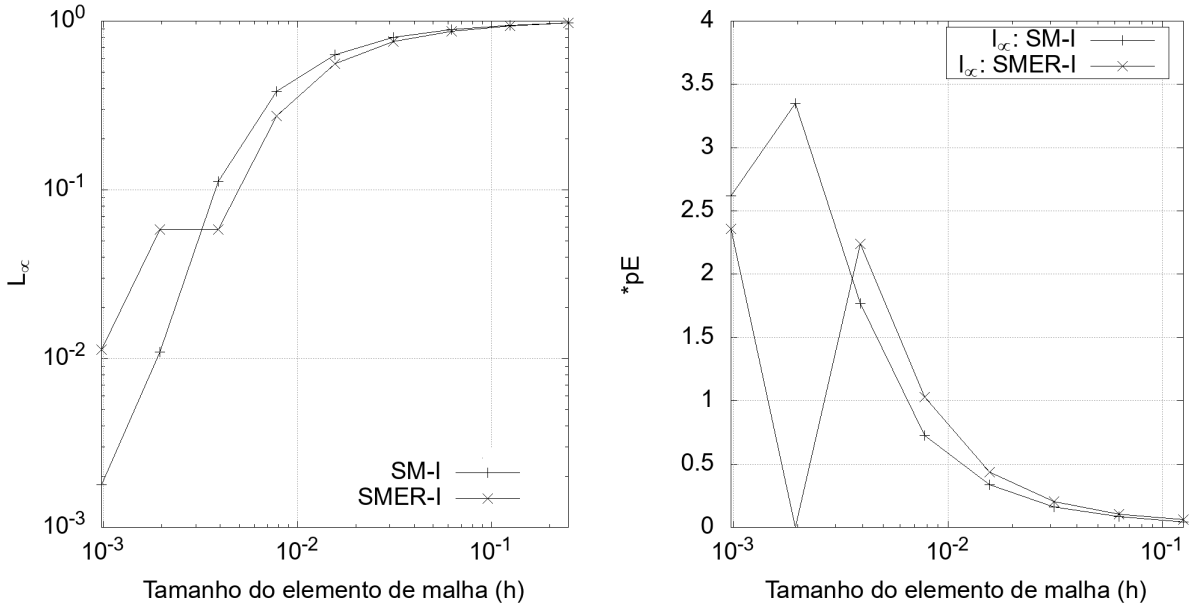
(a) l_∞, Malha 1 (b) p_E^*, Malha 1



FONTE: Autor(2020).

Figura 125 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Navier-Stokes 2D, velocidade u

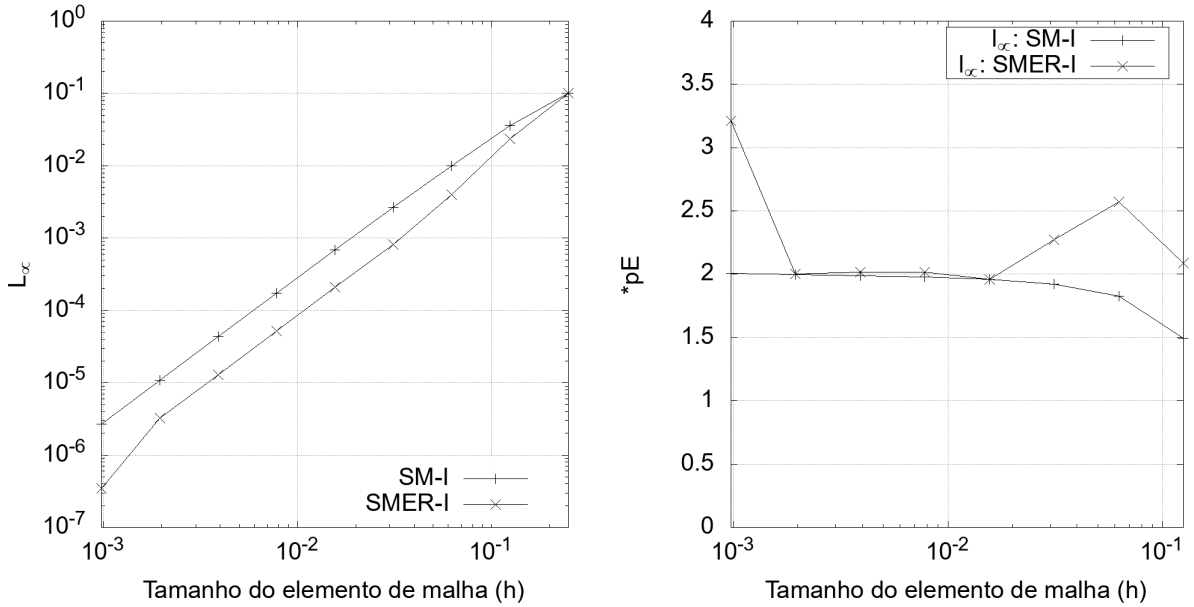
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 126 – Resultados: SMER-I, Conjunto β_2, Variável l_∞, Navier-Stokes 2D, velocidade v

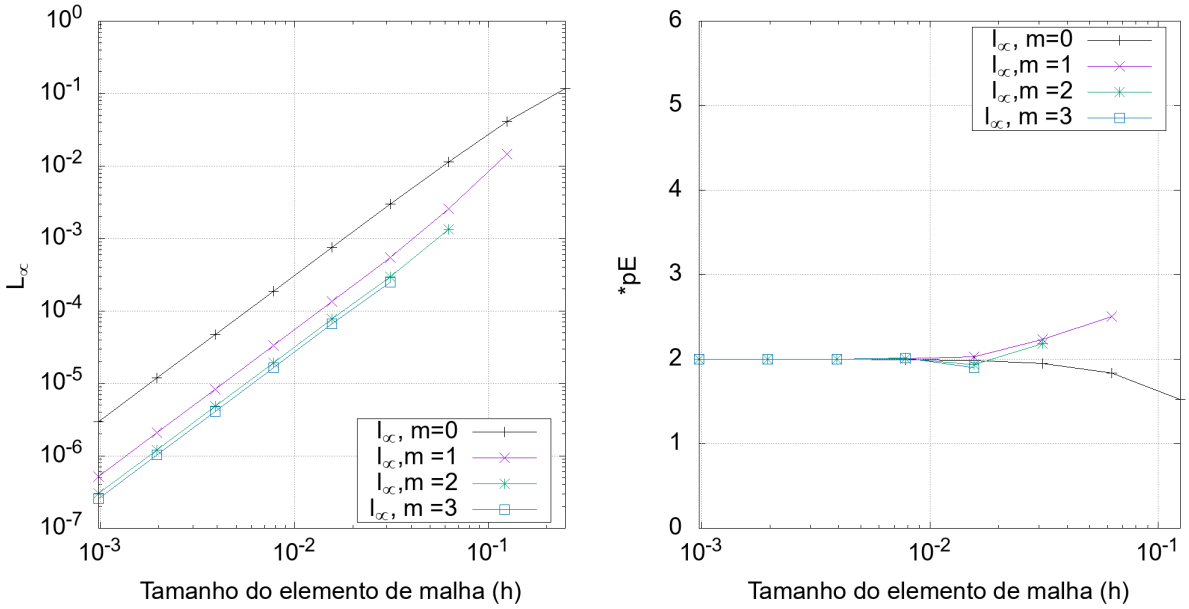
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 127 – Resultados: CRE-I, Variável l_∞, Navier-Stokes 2D, velocidade u

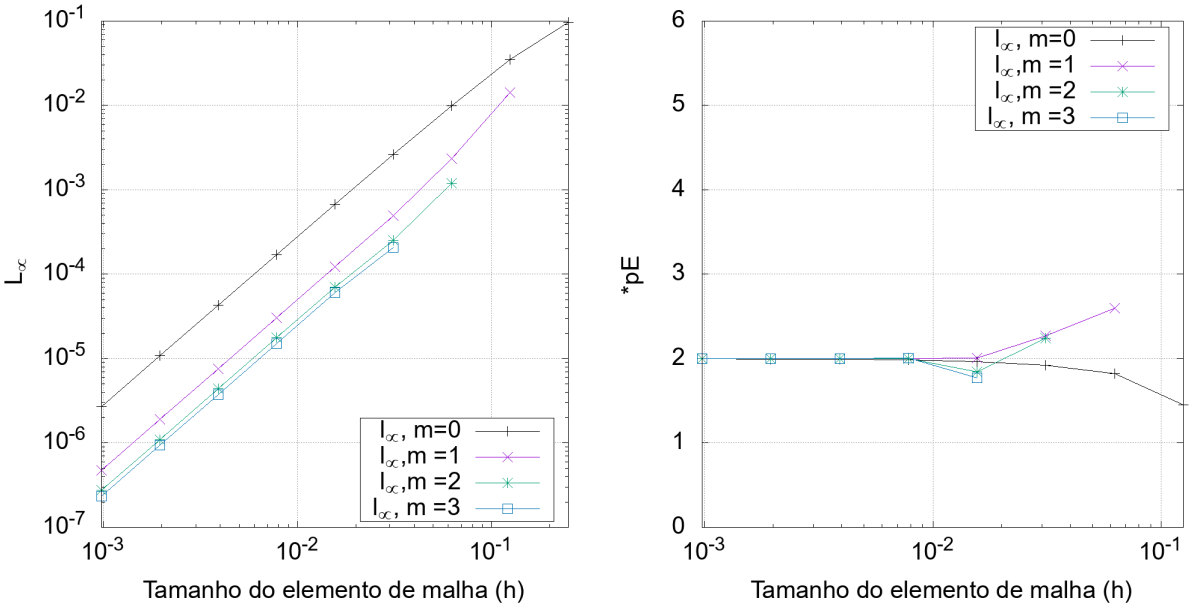
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 128 – Resultados: CRE-I, Variável l_∞, Navier-Stokes 2D, velocidade v

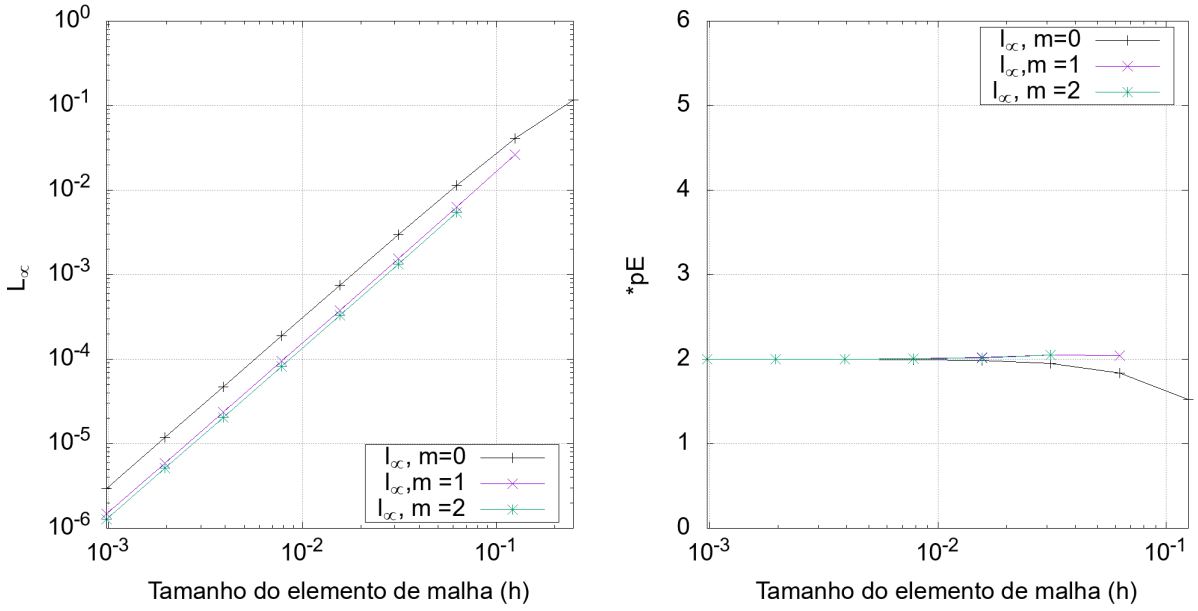
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 129 – Resultados: CRE, Variável l_∞, Navier-Stokes 2D, velocidade u

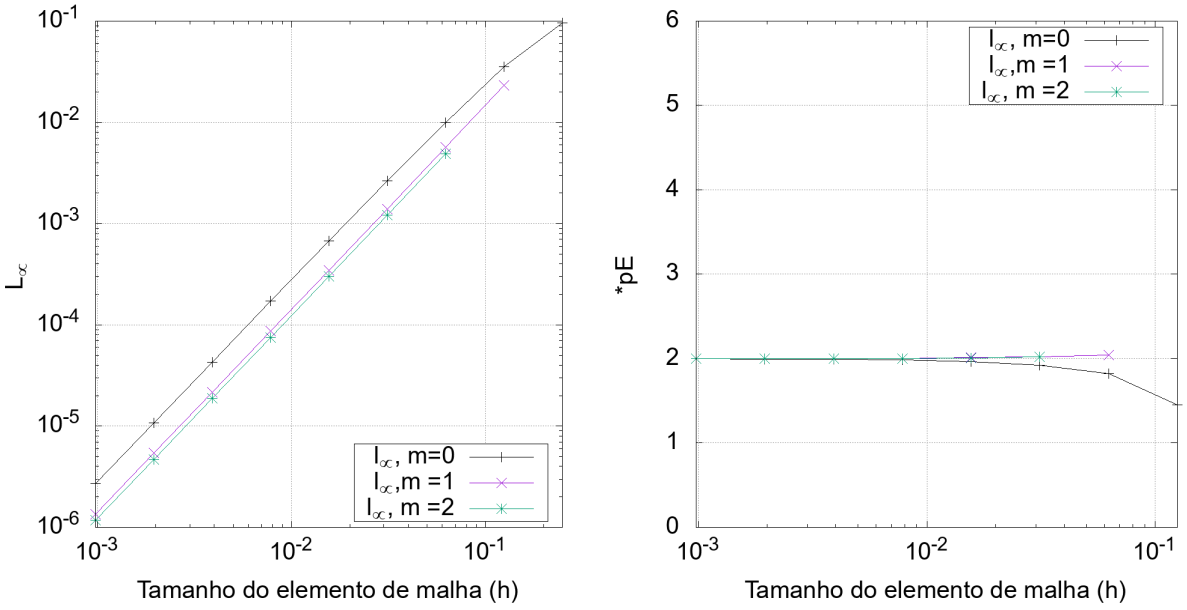
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 130 – Resultados: CRE, Variável l_∞, Navier-Stokes 2D, velocidade v

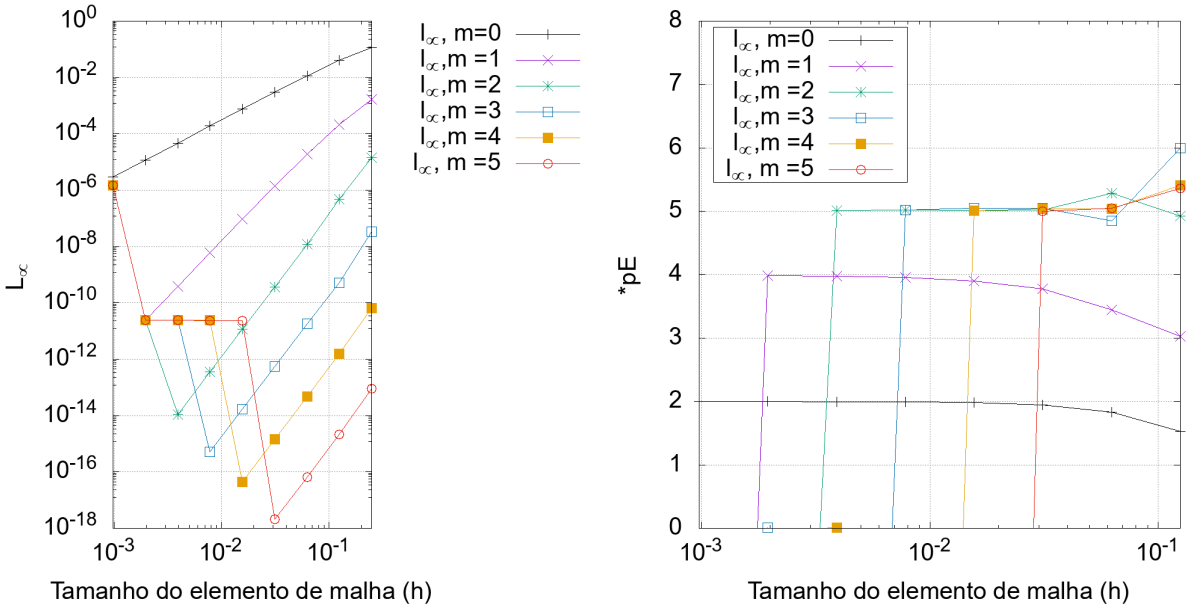
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 131 – Resultados: SER, Variável l_∞, Navier-Stokes 2D, velocidade u.

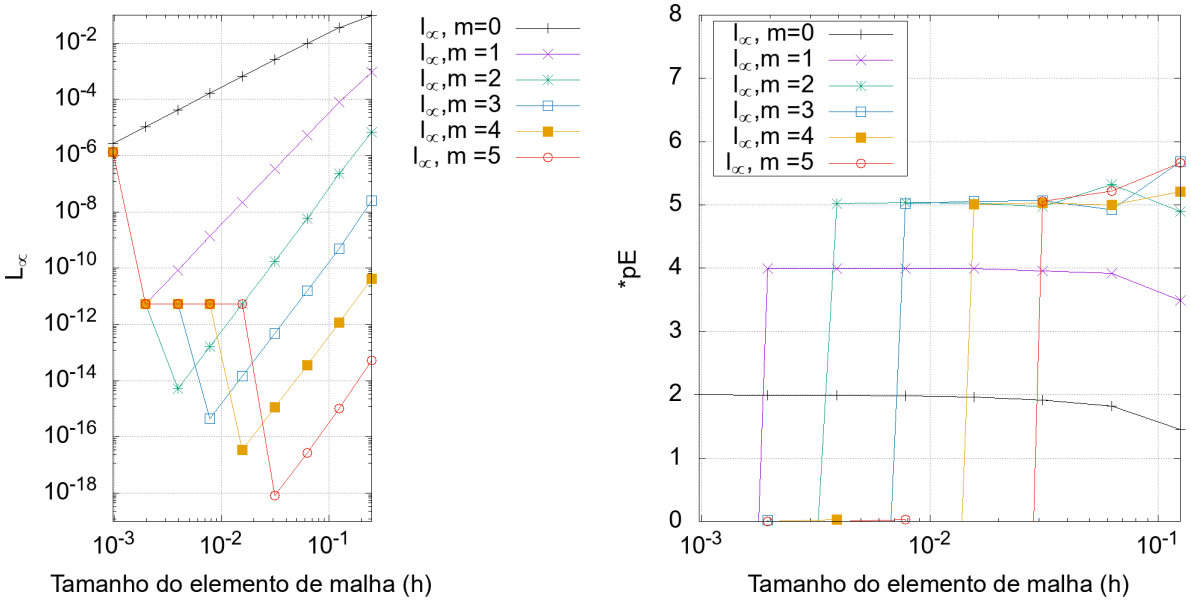
(a) l_∞ (b) p_E^*



FONTE: Autor(2020).

Figura 132 – Resultados: SER, Variável l_∞, Navier-Stokes 2D, velocidade v

(a) l_∞ (b) p_E^*



FONTE: Autor(2020).