

Sebastião Romero Franco

**MÉTODOS *MULTIGRID* ESPAÇO-TEMPO
PARA RESOLVER AS EQUAÇÕES DO CALOR
E DA POROELASTICIDADE**

Curitiba

2017

Sebastião Romero Franco

**MÉTODOS *MULTIGRID* ESPAÇO-TEMPO PARA
RESOLVER AS EQUAÇÕES DO CALOR E DA
POROELASTICIDADE**

Tese apresentada ao Programa de Pós-Graduação em Métodos Numéricos em Engenharia, Área de Concentração em Mecânica Computacional, Setor de Ciências Exatas e de Tecnologia, Universidade Federal do Paraná, como requisito parcial à obtenção do grau de Doutor em Ciências.

Orientador: Prof. Dr. Marcio Augusto Villela Pinto

Coorientador: Prof. Dr. Francisco José Gaspar Lorenz

Curitiba

2017

F825m

Franco, Sebastião Romero

Métodos multigrid espaço-tempo para resolver as equações do calor e da poroelasticidade / Sebastião Romero Franco. – Curitiba, 2017.

217 f. : il. color. ; 30 cm.

Tese - Universidade Federal do Paraná, Setor de Ciências Exatas,
Programa de Pós-Graduação em Métodos Numéricos em Engenharia, 2017.

Orientador: Marcio Augusto Villela Pinto – Co-orientador: Francisco José
Gaspar Lorenz,.

Bibliografia: p. 187-192.

1. Equação do Calor. 2. Equações da Poroelasticidade. 3. Equações
diferenciais parciais. I. Universidade Federal do Paraná. II. Pinto, Marcio
Augusto Villela. III. Gaspar Lorenz, Francisco José . IV. Título.

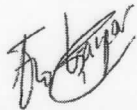
CDD: 519.2

Sebastião Romero Franco

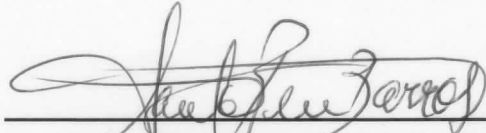
MÉTODOS *MULTIGRID* ESPAÇO-TEMPO PARA RESOLVER AS EQUAÇÕES DO CALOR E DA POROELASTICIDADE

Tese apresentada como requisito parcial à
obtenção do grau Doutor em Ciências, pelo
Programa de Pós-Graduação em Métodos
Numéricos em Engenharia, Setor de Ciências
Exatas e Tecnologia, Universidade Federal do
Paraná.

Trabalho aprovado. Curitiba, 12 de Dezembro de 2017:



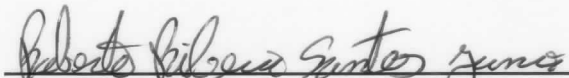
Francisco José Gaspar Lorenz
Universidad de Zaragoza – España
Coorientador



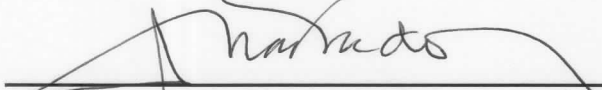
Saulo Rabelo Maciel de Barros
Universidade de São Paulo



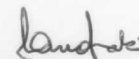
Cosmo Damião Santiago
Universidade Tecnológica Federal do Paraná



Roberto Ribeiro Santos Júnior
Universidade Federal do Paraná



Roberto Dalledone Machado
Universidade Federal do Paraná



Luciano Kiyoshi Araki
Universidade Federal do Paraná

Curitiba
Dezembro de 2017

À minha amada esposa Carine, por estar sempre ao meu lado em todos os momentos, pelo amor recebido, amizade, carinho, incentivo e compreensão pelo tempo dedicado aos estudos.

Às minhas filhas Isabela e Geovana Franco por me fazer feliz.

À meus pais Rosonir de Oliveira Franco (*in memoriam*) e Olinda Bochenek Franco que me mostraram, através de seus atos, que a simplicidade e a humildade são grandes virtudes do ser humano.

Agradecimentos

Agradeço a Deus, por tudo, pois sem Ele nada disso seria possível.

Agradeço muito ao meu orientador, Prof. Dr. Marcio Augusto Villela Pinto, por ter aceitado me orientar neste trabalho, pelo conhecimento recebido e também pela paciência, amizade e confiança.

Agradezco mucho a mi codirector, Prof. Dr. Francisco José Gaspar Lorenz y a la Prof^a. Dra. Carmen Rodrigo por la oportunidad de estudio, por las enseñanzas que recibí durante mi estancia en la Universidad de Zaragoza (y sigo recibiendo) y también por la atención y la amistad, no sólo conmigo, sino también con mi esposa Carine e hija Isabela.

Agradeço aos membros da banca examinadora, Prof. Dr. Francisco José Gaspar, Prof. Dr. Saulo Rabello Maciel de Barros, Prof. Dr. Cosmo Damião Santiago, Prof. Dr. Roberto Ribeiro Santos Junior, Prof. Dr. Roberto Dalledone Machado e Prof. Dr. Luciano Kiyoshi Araki, pelo tempo dispensado à leitura deste trabalho e pelas importantes sugestões apontadas.

Agradeço ao Programa de Pós-Graduação em Métodos Numéricos em Engenharia (PPGMNE) da Universidade Federal do Paraná (UFPR) pela oportunidade de cursar o doutorado e ao Departamento de Matemática da Universidade Estadual do Centro-Oeste (UNICENTRO) pela licença concedida para a dedicação aos estudos.

Agradeço à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), ao CNPq (especialmente ao programa *Ciências sem Fronteiras*) e à UNICENTRO pelo suporte financeiro.

Agradeço aos colegas do PPGMNE e PGMec pela amizade e disposição em ajudar.

Agradeço aos amigos e familiares, em especial aos meus irmãos: Rosari, Henrique Rogério, Rosimara e Amando Ronaldo Franco pela energia positiva durante este percurso.

Resumo

Nesta tese apresenta-se um estudo dos métodos usados para resolver equações diferenciais parciais transientes com o uso do método *multigrid*. Os modelos matemáticos usados são dados pela equação do calor e as equações da poroelasticidade. O modelo numérico é obtido através do emprego do Método das Diferenças Finitas, usando aproximação central de segunda ordem para a discretização espacial e os métodos de Euler e Crank-Nicolson para as discretizações no tempo. Na solução do sistema de equações resultante da discretização, utilizou-se o método *multigrid* geométrico com esquema CS, ciclos V , F e W , operador de restrição por ponderação completa, prolongação por interpolação linear nos casos unidimensionais e bilinear nos casos bidimensionais, e razão de engrossamento padrão nas direções das coordenadas espaciais. Utilizou-se o suavizador Gauss-Seidel para a equação do calor e o suavizador Vanka de 3 e 5 pontos para as equações da poroelasticidade. Com o objetivo de desenvolver algoritmos paralelizáveis, além da ordenação lexicográfica, usou-se a ordenação colorida para a suavização das incógnitas. Para os primeiros estudos com a equação do calor utilizou-se os métodos *Time-Stepping - multigrid*, *Waveform Relaxation - multigrid* e *Space-Time - multigrid*. Visando algoritmos paralelizáveis, propôs-se um novo método, o *Space-Time* com engrossamento padrão. Esse método consiste em usar engrossamento padrão em todos os níveis de malha, um apropriado operador de restrição e prolongação, e uma estratégia de suavização baseada em um processo que depende do grau de anisotropia de cada malha; processo esse, que contou com a ajuda da análise de Fourier local (LFA). Propôs-se o uso do método da dupla discretização associado ao *Space-Time* com engrossamento padrão para garantir aproximações de 2ª ordem de acurácia. Essa técnica apresentou robustez, bons fatores de convergência e possibilitou o uso de algoritmos altamente paralelizáveis no espaço e tempo, podendo ser considerado como um excelente método para resolver esse tipo de problemas. Propôs-se também o método *Waveform Relaxation - multigrid* para o sistema de equações da poroelasticidade. Esse método permite o desenvolvimento de algoritmos que podem ter um maior grau de paralelização que os algoritmos usuais descritos com o método *Time-Stepping*. O uso do método *Waveform Relaxation - multigrid* associado ao suavizador Vanka com ordenação colorida, além de possibilitar o desenvolvimento de algoritmos paralelizáveis no espaço e tempo, apresenta robustez e bons fatores de convergência. Com isso, pode ser considerado como um excelente método para resolver os problemas propostos.

Palavras-chave: *Multigrid*. *Waveform Relaxation*. *Space-Time* com Engrossamento Padrão. Equação do Calor. Dupla Discretização. Equações da Poroelasticidade. Suavizador Vanka.

Abstract

This thesis presents a study of the methods used to solve transient partial differential equations with the application of the multigrid method. The mathematical models employed are given by heat equation and poroelasticity equations. The numerical model is obtained by means of the Finite Difference Method, with the application of second-order central approximation for spatial discretization and Euler and Crank-Nicolson methods for time discretization. For the solution of the equation system that resulted from the discretization, the multigrid geometric method was used with CS scheme, V , F and W -cycles, full-weighting restriction operator, linear interpolation prolongation for one-dimensional cases and bilinear for bi-dimensional cases, and standard coarsening ratio in the directions of the spatial coordinates. The Gauss-Seidel smoother was used for heat equation and the 3-point and 5-point Vanka smoothers for the poroelasticity equations. Aiming at developing parallelizable algorithms, besides the lexicographical ordering, color ordering was employed in the unknowns smoothing. In the first studies with the heat equation, the Time-Stepping-multigrid, Waveform Relaxation-multigrid and Space-Time-multigrid methods were employed. A new method, the Space-Time with standard coarsening, was proposed aimed at parallelizable algorithms. This method consists in using standard coarsening in every level of the mesh, an adequate prolongation and restriction operator and a smoothing strategy based on a process that depends on the anisotropy degree of each grid; process which was assisted by local Fourier analysis (LFA). It was proposed the use of the double discretization method in conjunction with the Space-Time method with standard coarsening in order to assure accurate second-order approximations. This technique presented robustness, good convergence factors and allowed the use of highly parallelizable algorithms in space and time, and can be considered as an excellent method to solve this type of problems. Moreover, the Waveform Relaxation-multigrid method was proposed for the poroelasticity equation system. This method enables the development of algorithms that might have a higher degree of parallelization than the algorithms usually described for the Time-Stepping method. The application of the Waveform Relaxation multigrid method together with the Vanka smoother with color ordering, besides allowing the development of parallelizable algorithms in space and time, presents robustness and good convergence factors. With this, it can be considered as an excellent method to solve the proposed problems.

Keywords: Multigrid. Waveform Relaxation. Space-Time with Standard Coarsening. Heat Equation. Double Discretization. Poroelasticity Equation. Vanka Smoothers.

Lista de ilustrações

Figura 1 – Exemplo de uma malha unidimensional não uniforme.	39
Figura 2 – Exemplo de uma malha bidimensional não uniforme.	39
Figura 3 – Malha unidimensional uniforme com tamanho dos elementos de malha h	39
Figura 4 – Reta tangente (Exata) à curva Φ no ponto P e as aproximações Atrasada, Adiantada e Central através da discretização pelo MDF.	40
Figura 5 – Suavizador Vanka (usado para o problema da poroelasticidade).	46
Figura 6 – Ordenação lexicográfica para atualização das incógnitas.	47
Figura 7 – Ordenação <i>red-black</i> para atualização das incógnitas: (\circ) <i>red</i> e (\square) <i>black</i>	48
Figura 8 – Ordenação para atualização das incógnitas com um método linha.	49
Figura 9 – Ordenação para atualização das incógnitas com um método zebra.	50
Figura 10 – Conexão espacial e temporal em um ponto i com formulações Explícita, Totalmente Implícita e Implícita (MALISKA, 2004, p. 39).	51
Figura 11 – Sequência de 3 malhas bidimensionais e uniformes com engrossamento padrão.	54
Figura 12 – Número de ondas $k = 4$ sobre uma malha fina com $N = 12$ pontos e sobre uma malha grossa com $N = 6$ pontos (BRIGGS et al., 2000, p. 32).	57
Figura 13 – Operadores de restrição por injeção 1D e 2D.	58
Figura 14 – Operador de restrição por meia ponderação.	59
Figura 15 – Operadores de restrição por ponderação completa para os casos 1D e 2D.	60
Figura 16 – Operador de interpolação linear para o caso 1D.	61
Figura 17 – Operador de Interpolação bilinear para o caso 2D.	62
Figura 18 – Estrutura dos ciclos V , F e W . O símbolo (\bullet) representa suavização e o símbolo (\circ) a solução exata.	64
Figura 19 – Regiões de componentes de altas e baixas frequências (TROTTEMBERG et al., 2001, p. 102).	68
Figura 20 – Modelo do problema da poroelasticidade unidimensional.	80
Figura 21 – Variáveis usadas pelo suavizador Vanka de 3 pontos para equação da poroelasticidade 1D. As variáveis que estão em um círculo (\circ) serão atualizadas na iteração atual.	91
Figura 22 – Suavizador Vanka de 3 pontos para equação da poroelasticidade 1D. Os valores abaixo da linha reta pertencem a um passo de suavização e os valores acima da linha pertencem ao passo subsequente. Apenas as variáveis que estão em um círculo (\circ) serão atualizadas na iteração atual.	91
Figura 23 – Ordenação de 4 cores para o suavizador Vanka: (\circ) <i>vermelho</i> ; (\square) <i>preto</i> ; (\triangle) <i>azul</i> e (∇) <i>verde</i>	91

Figura 24 – Variáveis usadas pelo suavizador Vanka de 5 pontos para equação da poroelasticidade 2D. As variáveis que estão em um círculo (○) são atualizadas na iterada atual.	92
Figura 25 – Ordenação de 12 cores usada para o problema da poroelasticidade 2D com o suavizador Vanka de 5 pontos.	93
Figura 26 – Procedimento de atualização das incógnitas no método <i>Time-Stepping</i> (VANDEWALLE, 1993, p. 18) para o caso 2D.	95
Figura 27 – Procedimento de atualização das incógnitas no método <i>Waveform Relaxation</i> (LENT, 2006, p. 94) para o caso 1D.	97
Figura 28 – Método <i>Waveform Relaxation</i> Vanka de três pontos, usada para o problema da poroelasticidade 1D.	99
Figura 29 – Método <i>Waveform Relaxation</i> Vanka de cinco pontos, usada para o problema da poroelasticidade 2D.	100
Figura 30 – Tempo computacional usado para resolver a equação do calor 1D com os métodos <i>singlegrid</i> e <i>multigrid</i> e com aproximações temporal usando Euler e CN.	106
Figura 31 – <i>Speed-up</i> dos métodos <i>singlegrid</i> e <i>multigrid</i> para as aproximações temporal usando Euler e CN.	106
Figura 32 – Ordens efetiva, aparente e assintótica considerando a equação do calor 1D e o método de Euler.	107
Figura 33 – Ordens efetiva, aparente e assintótica considerando a equação do calor 1D e o método de CN.	108
Figura 34 – Tempo computacional usado para resolver a equação do calor 2D com os métodos <i>singlegrid</i> e <i>multigrid</i> e com aproximações temporal usando Euler e CN.	109
Figura 35 – <i>Speed-up</i> entre os métodos <i>singlegrid</i> e <i>multigrid</i> para as aproximações temporal usando Euler e CN aplicados na equação do calor 2D.	110
Figura 36 – Ordens efetiva, aparente e assintótica considerando a equação do calor 2D e o método de Euler.	111
Figura 37 – Ordens efetiva, aparente e assintótica considerando a equação do calor 2D e o método de CN.	112
Figura 38 – Tempo computacional usado para resolver a equação da poroelasticidade 1D com os métodos <i>singlegrid</i> e <i>multigrid</i> e com aproximações temporal usando Euler e CN. $tol = 10^{-6}$	114
Figura 39 – <i>Speed-up</i> entre os métodos <i>singlegrid</i> e <i>multigrid</i> para as aproximações temporal usando Euler e CN.	114
Figura 40 – Ordens efetiva, aparente e assintótica considerando a variável u da equação da poroelasticidade 1D e o método de Euler.	116

Figura 41 – Ordens efetiva, aparente e assintótica considerando a variável p da equação da poroelasticidade 1D e o método de Euler.	117
Figura 42 – Ordens efetiva, aparente e assintótica considerando a variável u da equação da poroelasticidade 1D e o método de CN.	118
Figura 43 – Ordens efetiva, aparente e assintótica considerando a variável p da equação da poroelasticidade 1D e o método de CN.	119
Figura 44 – Tempo computacional usado para resolver a equação da poroelasticidade 2D com os métodos <i>singlegrid</i> e <i>multigrid</i> e com aproximações temporal usando Euler e CN. $tol = 10^{-6}$	120
Figura 45 – <i>Speed-up</i> entre os métodos <i>singlegrid</i> e <i>multigrid</i> para as aproximações temporal usando Euler e CN na equação da poroelasticidade 2D.	121
Figura 46 – Ordens efetiva, aparente e assintótica considerando a variável u da equação da poroelasticidade 2D e o método de Euler.	122
Figura 47 – Ordens efetiva, aparente e assintótica considerando a variável v da equação da poroelasticidade 2D e o método de Euler.	123
Figura 48 – Ordens efetiva, aparente e assintótica considerando a variável p da equação da poroelasticidade 2D e o método de Euler.	124
Figura 49 – Ordens efetiva, aparente e assintótica considerando a variável u da equação da poroelasticidade 2D e o método de CN.	125
Figura 50 – Ordens efetiva, aparente e assintótica considerando a variável v da equação da poroelasticidade 2D e o método de CN.	126
Figura 51 – Ordens efetiva, aparente e assintótica considerando a variável p da equação da poroelasticidade 2D e o método de CN.	127
Figura 52 – Efetividade do estimador bicoeficiente considerando a variável p da equação da poroelasticidade 2D e o método de Euler e condutividade hidráulica $K = 10^{-9}$	128
Figura 53 – Efetividade do estimador bicoeficiente considerando a variável p da equação da poroelasticidade 2D, o método de Euler e a condutividade hidráulica $K = 10^{-3}$	128
Figura 54 – Fator de convergência médio (ρ_m) do método <i>singlegrid</i> após 20 iterações com os suavizadores zebra no tempo e zebra no espaço para a equação do calor 1D.	130
Figura 55 – Fator de convergência calculado através do método de 2 malhas (LFA) para a equação do calor 1D e aproximação temporal feita com o método de Euler, considerando $N_x = N_t = 128$, $\nu_1 = 2$ (pré) e $\nu_2 = 1$ (pós suavização). No método <i>Space-Time - multigrid</i> a linha sólida indica engrossamento apenas na direção espacial e a linha tracejada indica engrossamento apenas na direção temporal. Figura retirada de Vandewalle e Horton (1993, p. 103).	131

Figura 56 – Robustez do método <i>Time-Stepping - multigrid</i> e ciclo $V(0, 1)$ para resolver a equação do calor 2D.	134
Figura 57 – Robustez do método <i>Waveform Relaxation - multigrid</i> e ciclo $V(0, 1)$ para resolver a equação do calor 1D.	135
Figura 58 – Robustez do método <i>Waveform Relaxation - multigrid</i> e ciclo $V(0, 1)$ para resolver a equação do calor 2D.	136
Figura 59 – Robustez do método <i>Waveform Relaxation - multigrid</i> e ciclo $V(0, 1)$ para resolver a equação do calor 2D. Neste caso considerou-se a malha espacial mais refinada fixa com $N_x = 2^5 + 1$	137
Figura 60 – Robustez do método <i>Waveform Relaxation - multigrid</i> e ciclo $V(0, 1)$ para resolver a equação do calor 2D. Neste caso considerou-se a aproximação temporal mais refinada fixa $N_t = 2^5 + 1$	138
Figura 61 – Otimização dos parâmetros relacionados ao método <i>Time-Stepping - multigrid</i> para a equação do calor 2D com o método de Euler e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.	140
Figura 62 – Otimização dos parâmetros relacionados ao método <i>Time-Stepping - multigrid</i> para a equação do calor 2D com o método de CN e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.	141
Figura 63 – Otimização dos parâmetros relacionados ao método <i>Time-Stepping - multigrid</i> para a equação do calor 2D com os métodos de Euler e de CN, considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção e 1 suavização com os ciclos V , W ou F	142
Figura 64 – Otimização dos parâmetros relacionados ao método <i>Waveform Relaxation - multigrid</i> para a equação do calor 1D com o método de Euler e considerando uma discretização com $N = 2^{13} + 1$ pontos em cada direção.	143
Figura 65 – Otimização dos parâmetros relacionados ao método <i>Waveform Relaxation - multigrid</i> para a equação do calor 1D com o método de CN e considerando uma discretização com $N = 2^{13} + 1$ pontos em cada direção.	144
Figura 66 – Otimização dos parâmetros relacionados ao método <i>Waveform Relaxation - multigrid</i> para a equação do calor 2D com o método de Euler e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.	145
Figura 67 – Otimização dos parâmetros relacionados ao método <i>Waveform Relaxation - multigrid</i> para a equação do calor 2D com o método de CN e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.	146

Figura 68 – Fator de convergência ρ obtido através da análise de duas malhas com LFA para o método de Euler (BDF1) e o método de Crank-Nicolson (CN), com 2 passos de suavização (esquerda) e 3 passos de suavização (direita). A linha sólida indica engrossamento apenas na direção espacial, a linha tracejada indica engrossamento apenas na direção temporal e a linha pontilhada indica engrossamento padrão. Figura retirada de Horton e Vandewalle (1995, p. 858).	147
Figura 69 – Otimização dos parâmetros relacionados ao método <i>Space-Time - multigrid</i> para a equação do calor 1D com o método de Euler e considerando uma discretização com $N = 2^{13} + 1$ pontos em cada direção.	148
Figura 70 – Otimização dos parâmetros relacionados ao método <i>Space-Time - multigrid</i> para a equação do calor 2D com o método de Euler e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.	149
Figura 71 – Tempo computacional para resolver a equação do calor 2D usando paralelização de computadores, com o método de Euler e considerando uma discretização espacial com $N_x = N_y = 2^7 + 1$ pontos e temporal com $N_t = 16.384$ pontos. Figura retirada de Falgout et al. (2017, p. 140).	151
Figura 72 – Fatores de convergência preditos por LFA (ρ_{loc}) correspondentes ao método <i>multigrid</i> com suavizadores zebra com linha no tempo e zebra com linha no espaço, e fatores de convergência médio (ρ_m) obtido experimentalmente, para diferentes valores do parâmetro λ aplicados ao problema modelo unidimensional da equação do calor.	158
Figura 73 – Robustez do método <i>Space-Time</i> com engrossamento padrão para resolver a equação do calor 1D com o método de Euler, ciclo $V(0, 1)$ e $N_x = 2^7 + 1$.	159
Figura 74 – Robustez do método <i>Space-Time</i> com engrossamento padrão para resolver a equação do calor 1D com o método de Euler, ciclo $V(0, 1)$ e $N_t = 2^7 + 1$.	159
Figura 75 – Robustez do método <i>Space-Time</i> com engrossamento padrão para resolver a equação do calor 2D com o método de Euler, ciclo $V(0, 1)$ e $N_x = N_y = 2^5 + 1$.	160
Figura 76 – Robustez do método <i>Space-Time</i> com engrossamento padrão para resolver a equação do calor 2D com o método de Euler, ciclo $V(0, 1)$ e $N_t = 2^5 + 1$.	161
Figura 77 – Otimização dos parâmetros <i>multigrid</i> com o método <i>Space-Time - multigrid</i> com engrossamento padrão usado na solução da equação do calor 1D.	162

Figura 78 – Comparação entre os parâmetros <i>multigrid</i> ótimos para o método <i>Space-Time - multigrid</i> com engrossamento padrão usado na solução da equação do calor 1D.	163
Figura 79 – Otimização dos parâmetros <i>multigrid</i> através do método <i>Space-Time</i> com engrossamento padrão, método de Euler e ciclo <i>V</i> para resolver a equação do calor 2D.	164
Figura 80 – Otimização dos parâmetros <i>multigrid</i> através do método <i>Space-Time</i> com engrossamento padrão, método de Euler e ciclo <i>F</i> para resolver a equação do calor 2D.	165
Figura 81 – Otimização dos parâmetros <i>multigrid</i> através do método <i>Space-Time</i> com engrossamento padrão, método de Euler e ciclo <i>W</i> para resolver a equação do calor 2D.	166
Figura 82 – Comparação dos resultados descritos nas Fig. 79, Fig. 80 e Fig. 81. No eixo vertical são descritas as somas de todos os t_{CPU} relacionados aos casos usados para a suavização por planos para cada λ	167
Figura 83 – Comparação dos resultados descritos nas Figs. 79 a 81. No eixo vertical são mostrados os valores máximo dos t_{CPU} considerando-se os ciclos $V(0, 1)$, $F(0, 1)$ e $W(0, 1)$	168
Figura 84 – Fator de convergência médio do método <i>Space-Time</i> com engrossamento padrão e ciclo $V(0, 1)$	169
Figura 85 – Experimentos com o método <i>multigrid</i> , aproximação temporal com o método de CN, engrossamento padrão, ciclo $V(0, 1)$ e suavizadores <i>zebra com linha no tempo</i> e <i>zebra com linha no espaço</i> para resolver a equação do calor 1D.	170
Figura 86 – Número de ciclos $V(0, 1)$ necessários para alcançar o erro de máquina usando o método de dupla discretização.	172
Figura 87 – Fator de convergência médio do método <i>Space-Time</i> com engrossamento padrão associado a técnica da dupla discretização e ciclo $V(0, 1)$	174
Figura 88 – Fator de convergência após 20 ciclos <i>multigrid</i> , considerando $E = 10^4 N/m^2$, $K = 10^{-6}$, solução analítica nula, método de Euler e o suavizador Vanka de 3 pontos e 4 cores para o problema da poroelasticidade 1D.	176
Figura 89 – Fator de convergência após 20 ciclos <i>multigrid</i> , considerando $E = 10^4$, $\nu = 0, 2$, $K = 10^{-6}$, solução analítica nula, método de Euler e o suavizador Vanka de 5 pontos e 12 cores para o problema da poroelasticidade 2D.	177
Figura 90 – Fator de convergência para mostrar a robustez do método <i>Waveform Relaxation - multigrid</i> para o problema da poroelasticidade 1D, usando o ciclo $W(0, 1)$	178

Figura 91 – Fator de convergência para mostrar a robustez do método <i>Waveform Relaxation - multigrid</i> para o problema da poroelasticidade 2D, usando o ciclo $W(0,1)$	179
Figura 92 – Fator de convergência médio ρ_m e t_{CPU} considerando o método <i>Waveform Relaxation - multigrid</i> , Euler e o suavizador Vanka de 3 pontos com ordenação lexicográfica para o problema da poroelasticidade 1D. .	180
Figura 93 – Fator de convergência médio ρ_m e t_{CPU} considerando o método <i>Waveform Relaxation - multigrid</i> , Euler e o suavizador Vanka de 3 pontos com ordenação de 4 cores para o problema da poroelasticidade 1D.	181
Figura 94 – Fator de convergência médio ρ_m e t_{CPU} considerando o método <i>Waveform Relaxation - multigrid</i> , Euler e o suavizador Vanka de 5 pontos com ordenação lexicográfica para o problema da poroelasticidade 2D. .	182
Figura 95 – Fator de convergência médio ρ_m e t_{CPU} considerando o método <i>Waveform Relaxation - multigrid</i> , Euler e o suavizador Vanka de 5 pontos com ordenação de 12 cores para o problema da poroelasticidade 2D.	183
Figura 96 – Posições dos pontos em um sistema de coordenadas cartesianas e em relação a um ponto central.	204

Lista de tabelas

Tabela 1	– Autovalores dos operadores de restrição usados na LFA.	72
Tabela 2	– Valor da $\ \cdot\ _\infty$ dos erros numéricos para a equação do calor 1D, usando o método <i>Waveform Relaxation</i> para os casos <i>singlegrid</i> e <i>multigrid</i> . . .	105
Tabela 3	– Coeficientes c e p de $t_{cpu}(\mathcal{N}) = c \cdot \mathcal{N}^p$ considerando a equação do calor 1D, os métodos de Euler e de CN com <i>singlegrid</i> e <i>multigrid</i>	105
Tabela 4	– Valor da $\ \cdot\ _\infty$ dos erros numéricos para a equação do calor 2D, usando o método <i>Waveform Relaxation</i> para os caso <i>singlegrid</i> e <i>multigrid</i> . . .	109
Tabela 5	– Coeficientes c e p de $t_{cpu}(\mathcal{N}) = c \cdot \mathcal{N}^p$ considerando a equação do calor 2D, os métodos de Euler e de CN com <i>singlegrid</i> e <i>multigrid</i>	110
Tabela 6	– Valor da $\ \cdot\ _\infty$ dos erros numéricos para a equação da poroelasticidade 1D, usando o método <i>Waveform Relaxation</i> e o suavizador Vanka de 3 pontos com ordenação de 4 cores e linha no tempo para os métodos <i>singlegrid</i> e <i>multigrid</i>	113
Tabela 7	– Coeficientes c e p de $t_{cpu}(\mathcal{N}) = c \cdot \mathcal{N}^p$ considerando a equação da poroelasticidade 1D, os métodos de Euler e de CN com <i>singlegrid</i> e <i>multigrid</i>	115
Tabela 8	– Valor da $\ \cdot\ _\infty$ dos erros numéricos para a equação da poroelasticidade 2D, usando o método <i>Waveform Relaxation</i> e o suavizador Vanka de 5 pontos com ordenação de 12 cores e linha no tempo para os métodos <i>singlegrid</i> e <i>multigrid</i>	120
Tabela 9	– Coeficientes c e p de $t_{cpu}(\mathcal{N}) = c \cdot \mathcal{N}^p$ considerando a equação da poroelasticidade 2D, os métodos de Euler e de CN com <i>singlegrid</i> e <i>multigrid</i>	121
Tabela 10	– Parâmetros que otimizam o t_{CPU} para resolver as equações do calor 1D e 2D usando os métodos <i>Time-Stepping - multigrid</i> , <i>Waveform Relaxation - multigrid</i> e <i>Space-Time - multigrid</i>	150
Tabela 11	– Norma máxima do erro numérico, $\ e\ _\infty$, para o problema 1D e diferentes tamanhos da malha com $h = \tau = \frac{1}{N}$	173
Tabela 12	– Norma máxima do erro numérico, $\ e\ _\infty$, para o problema 2D e diferentes tamanhos da malha com $h = \tau = \frac{1}{N}$	173

Lista de Algoritmos

1	MG- μ -ciclo (l)	65
2	MG-Ciclos-F (l)	65
3	LFA para análise do fator de suavização	70
4	LFA para o fator de convergência em duas malhas	73
5	Método <i>Time-Stepping</i> em uma malha	95
6	Método <i>Time-Stepping - multigrid</i>	96
7	Método <i>Waveform Relaxation</i> em uma malha.	98
8	Método <i>Waveform Relaxation - multigrid</i>	98
9	Método <i>Space-Time</i> em uma malha	99
10	Estratégia de engrossamento no método <i>Space-Time - multigrid</i>	100
11	Método <i>Space-Time - multigrid</i>	102
12	Estratégia de suavização	156
13	Ciclo $V(\nu_1, \nu_2)$ com o método de dupla discretização, nível de malha (l) . . .	171

Lista de abreviaturas e siglas

CDS	<i>Central Differencing Scheme</i>
CFD	<i>Computational Fluid Dynamics</i>
CN	Método de Crank-Nicolson
CPU	<i>Central Processing Unit</i>
CS	<i>Correction Scheme</i>
EDOs	Equações Diferenciais Ordinárias
EDPs	Equações Diferenciais Parciais
FAS	<i>Full Approximation Scheme</i>
FMG	<i>Full multigrid</i>
FW	Ponderação completa
HW	Meia ponderação
ILU	Fatoração Incompleta LU
Inj	Injeção
Lex	Ordem lexicográfica
LFA	<i>Local Fourier Analysis</i>
MDF	Método das Diferenças Finitas
MGRIT	<i>Multigrid</i> com redução no tempo
T^{alta}	Região de altas frequências
T^{baixa}	Região de baixas frequências
t_{CPU}	Tempo de CPU
tol	Tolerância
WRMG-CR	<i>Multigrid Waveform Relaxation</i> com redução cíclica

Lista de símbolos

A	Matriz de coeficientes
c	Constante
D	Matriz diagonal, conjunto convexo
d_{ij}	Elementos da matriz D
e	Vetor dos erros numéricos
E	Módulo de Young
f	Termo fonte da equação do calor
\mathcal{U}	Termo fonte relacionado ao deslocamento na direção u
\mathcal{V}	Termo fonte relacionado ao deslocamento na direção v
\mathcal{P}	Termo fonte relacionado a pressão p
h_i	Distância entre os pontos da discretização espacial na malha Ω^i
h	Distância entre os pontos da discretização espacial
i, j	Contadores na direção espacial
I_h^{2h}	Operador de restrição
$(I_h^{2h})_{Inj}$	Operador de restrição por injeção
$(I_h^{2h})_{HW}$	Operador de restrição por meia ponderação
$(I_h^{2h})_{FW}$	Operador de restrição por ponderação completa
I_{2h}^h	Operador de prolongação
K	Condutividade hidráulica
k	Número de ondas ou modos de Fourier
K_h^{2h}	Operador de correção da malha grossa
L	Matriz triangular inferior, operador diferencial
L_h	Estêncil do operador diferencial Laplaciano em uma malha h

\tilde{L}_h	Autovalor de L_h
L^+	Termos já lido na iteração atual
L^-	Termos usados da iteração anterior
l_{ij}	Elementos da matriz L
l	Número de malhas usadas no <i>multigrid</i>
L_{\max}	Número máximo de níveis de malhas
m	Contador na direção temporal
M_h^{2h}	Operador de duas malhas
N	Número de pontos quando $N = N_x = N_t$ ou $N = N_x = N_y = N_t$
N_x	Número de pontos na discretização espacial x
N_y	Número de pontos na discretização espacial y
N_t	Número de pontos na discretização temporal t
\mathcal{N}	Número total de incógnitas
p	ordem do algoritmo (inclinação da curva em escala bilogarítmica na análise do t_{CPU})
p	Pressão (nas equações da poroelasticidade)
p_m	Valor médio da incógnita p
p_E	Ordem efetiva
p_L	Ordem assintótica
p_U	Ordem aparente
r	Vetor dos resíduos
$\ \cdot\ _\infty$	Norma do máximo (ou norma-infinito)
$\ r(0)\ _\infty$	Norma infinito do resíduo com a estimativa inicial
$\ r(it)\ _\infty$	Norma infinito do resíduo na iterada atual
q	Razão de engrossamento
S_h	Estêncil do suavizador em uma malha h

\tilde{S}_h	Autovalor de S_h
t_0	Tempo inicial
t_f	Tempo final
U	Matriz triangular superior
U_{bi}	Estimador bicoeficiente
u	Variável de interesse, temperatura no caso da equação do calor e deslocamento na direção coordenada x nas equações da poroelasticidade
u_m	Valor médio da incógnita u
v	Aproximação da variável de interesse u no caso da equação do calor e deslocamento na direção coordenada y nas equações da poroelasticidade
v_m	Valor médio da incógnita v
v_j	Modos de Fourier
w_h	Nova aproximação da variável u_h (usada em LFA)
\bar{w}_h	Aproximação anterior de u_h (usada em LFA)

Letras Gregas

Δ	Operador Laplaciano
Δ_h	Operador Laplaciano na malha h
κ	Posição no estêncil
λ	Fator de anisotropia $\lambda = \frac{\tau}{h^2}$ (usado na equação do calor)
λ	Constante de Lamé (usado nas equações da poroelasticidade)
λ_{crit}	Fator de anisotropia crítico
λ_l	Fator de anisotropia na malha $\Omega^{2^{l-1}h}$
μ	Constante de Lamé
μ -ciclo	Tipo de ciclo usado no processo <i>multigrid</i>
μ_{loc}	Fator de suavização obtido com LFA
ν	Número total de suavizações em LFA ou razão de Poisson no problema da poroelasticidade

ν_1	Número de pré-suavizações
ν_2	Número de pós-suavizações
Ω^h	Malha discretizada atual
ρ	Fator de convergência assintótica
ρ_m	Fator de convergência médio
ρ_{loc}	Fator de convergência obtido com LFA
ϕ	Solução numérica
ϕ_∞	Valor estimado para a solução analítica exata
τ	Distância entre os pontos da aproximação temporal
θ	Frequência da função de malha φ
φ	Função de malha

índices

h	Malha fina ($l = 1$)
l	Indicador da malha $\Omega^{2^{l-1}h}$
lh	Malha Grossa ($l = 2, 4, 8, \dots$)
i	posição do nó na direção x
j	posição do nó na direção y
m	posição do nó na direção t
ν	iterada
x	derivada com relação a x

Sumário

	Agradecimentos	5
	Sumário	22
1	INTRODUÇÃO	25
1.1	Generalidades em CFD	25
1.2	Método <i>multigrid</i>	26
1.3	Motivação	27
1.4	Objetivos	28
1.5	Delineamento do texto	28
2	REVISÃO BIBLIOGRÁFICA	30
2.1	Método <i>multigrid</i>	30
2.2	Equação do calor	32
2.3	Equações da poroelasticidade	35
3	FUNDAMENTAÇÃO TEÓRICA	38
3.1	Método das Diferenças Finitas (MDF)	38
3.1.1	Expansão em fórmula de Taylor	40
3.2	Notação estêncil	43
3.3	Métodos iterativos	44
3.3.1	Método de Jacobi	45
3.3.2	Método de Gauss-Seidel	45
3.3.3	Suavizador Vanka	45
3.3.4	Ordenação das atualizações das incógnitas	47
3.4	Métodos para a aproximação temporal	48
3.4.1	Formulações temporais de EDPs	48
3.4.2	Método de <i>Euler</i>	49
3.4.3	Método de Crank-Nicolson	52
4	MÉTODO <i>MULTIGRID</i>	53
4.1	Caracterização do método <i>multigrid</i>	53
4.1.1	Análise de convergência	54
4.1.2	Análise de erros	56
4.2	Operadores de transferência entre malhas	57
4.2.1	Operadores de restrição	57
4.2.2	Operadores de prolongação	61

4.3	Esquema de correção e ciclos	63
5	ANÁLISE DE FOURIER LOCAL	66
5.1	Terminologia	66
5.2	Análise do suavizador	68
5.3	Análise em duas malhas	69
5.4	Análise de Fourier local para outros suavizadores	73
6	MODELOS MATEMÁTICOS E NUMÉRICOS	76
6.1	Equação do calor	76
6.1.1	Modelos matemáticos	76
6.1.2	Modelos numéricos	77
6.2	Equações da poroelasticidade	79
6.2.1	Modelos matemáticos	80
6.2.2	Modelos numéricos	82
6.3	Ordenação colorida para o suavizador Vanka	90
7	MÉTODOS DE SOLUÇÃO	94
7.1	Método <i>Time-Stepping</i>	94
7.2	Método <i>Waveform Relaxation</i>	96
7.3	Método <i>Space-Time</i>	99
8	VERIFICAÇÃO DOS CÓDIGOS COMPUTACIONAIS	103
8.1	Equação do calor	104
8.1.1	Problema unidimensional	104
8.1.2	Problema bidimensional	108
8.2	Equações da poroelasticidade	112
8.2.1	Problema unidimensional	113
8.2.2	Problema bidimensional	119
9	RESULTADOS PRELIMINARES	129
9.1	Testes de robustez	132
9.2	Otimização dos parâmetros <i>multigrid</i>	139
9.3	Comparação dos métodos estudados	149
10	RESULTADOS	152
10.1	Equação do calor	152
10.1.1	Descrição do método <i>Space-Time - multigrid</i> com engrossamento padrão	153
10.1.2	Robustez	158
10.1.3	Otimização dos parâmetros <i>multigrid</i>	160
10.1.4	Fatores de convergência	166

10.1.5	Método da dupla discretização	168
10.2	Equações da poroelasticidade	173
10.2.1	Robustez	175
10.2.2	Otimização dos parâmetros <i>multigrid</i>	177
11	CONCLUSÃO	185
11.1	Contribuições	186
11.2	Proposta de trabalhos futuros	187
	REFERÊNCIAS	188
	APÊNDICES	194
	APÊNDICE A – PROGRAMAÇÃO DO SUAVIZADOR VANKA . .	195

1 Introdução

Nesta tese, buscam-se métodos eficientes para a solução de sistemas de equações grandes e esparsos decorrentes da discretização de equações diferenciais parciais (EDPs) transientes que modelam matematicamente muitos problemas encontrados em diversas áreas, de aplicações industriais, econômica, de biomedicina, entre outros.

Neste capítulo introduzem-se os conceitos gerais sobre a Dinâmica dos Fluidos Computacional (em inglês, *Computational Fluid Dynamics*, CFD), sobre o acelerador de convergência *multigrid*, sobre a motivação e os objetivos deste trabalho.

1.1 Generalidades em CFD

A área do conhecimento denominada CFD trata dos estudos de métodos computacionais para simulação de fenômenos que envolvem fluidos em movimento com ou sem trocas de calor, cujo interesse principal é obter grandezas físicas, como velocidade, temperatura e pressão, na região do escoamento (FORTUNA, 2000). Estes modelos matemáticos, em geral, não têm soluções analíticas conhecidas. Buscam-se então soluções numéricas transformando-se o modelo contínuo em um modelo discreto.

De acordo com Maliska (2004), o engenheiro ou projetista tem à sua disposição, fundamentalmente, três ferramentas para desenvolver seu projeto ou analisar seu problema: os métodos analíticos, os métodos numéricos e os métodos de experimentação em laboratório.

A experimentação em laboratório trata a configuração real do objeto de estudo, porém apresenta, em geral, um alto custo para sua elaboração e, muitas vezes não pode ser realizada por questões de segurança ou pelas dificuldades de reprodução das condições reais (MALISKA, 2004).

Os métodos analíticos e numéricos formam a classe dos métodos teóricos, pois objetivam resolver as equações diferenciais que modelam fenômenos físicos reais. A diferença entre eles está apenas na complexidade das equações que cada método pode resolver. Os métodos analíticos são aplicáveis a uma classe de problemas que simplificam muito as hipóteses do fenômeno real. Além disso, são aplicados a geometrias e condições de contornos mais simples. Obviamente, as soluções analíticas não devem ser descartadas porque auxiliam na validação de modelos numéricos e no desenvolvimento de métodos numéricos mais robustos (MALISKA, 2004).

As simulações numéricas podem resolver problemas com condições de contorno gerais e definidos em praticamente todos os tipos de geometrias. Para isso, transforma-

se o domínio contínuo (com um número infinito de pontos) em um domínio discreto (com um número finito de pontos) através do uso de uma malha. Para aproximar as derivadas, os métodos tradicionais de discretização do modelo matemático são os Métodos das Diferenças Finitas (MDF) (TANNEHILL et al., 1997; GOLUB; ORTEGA, 1992; FERZIGER; PERIC, 2002; SAAD, 2003), Método dos Elementos Finitos (SAAD, 2003), Método dos Volumes Finitos (GOLUB; ORTEGA, 1992; MALISKA, 2004), entre outros. Neste trabalho emprega-se o MDF.

Os fenômenos físicos geralmente podem ser modelados por Equações Diferenciais Parciais (EDPs). As EDPs que não dependem da variável temporal dão origem ao regime denominado de permanente (ou estacionário), enquanto que as EDPs que dependem da variável temporal dão origem ao regime transiente.

Após a discretização das EDPs, tem-se um sistema linear a ser resolvido. A resolução desse sistema linear através de métodos diretos não é recomendável, visto que na prática, a matriz dos coeficientes é muito grande e o custo da inversão da matriz é alto (GOLUB; LOAN, 1989). Para problemas de grande porte os métodos iterativos são mais adequados (BURDEN; FAIRES, 2016).

Os métodos iterativos mais usados para aproximar a solução (variáveis espaciais) são o método de Jacobi ponderado e o método de Gauss-Seidel. Para aproximar a variável temporal, tem-se os métodos de Euler, Crank-Nicolson (CN), entre outros (BURDEN; FAIRES, 2016; LENT, 2006). Porém, esses métodos iterativos geralmente perdem a eficiência quando o número de iterações aumenta ou a malha é refinada (BRIGGS et al., 2000).

Para o caso de sistemas de equações diferenciais, como os decorrentes das equações da poroelasticidade, os métodos que suavizam as variáveis ponto a ponto de forma desacoplada não são eficientes, sendo necessário o uso de métodos que suavizam todas as variáveis em um ponto de forma acoplada, como por exemplo o suavizador Vanka, ver Oosterlee e Gaspar (2008) e Rodrigo (2010).

1.2 Método *multigrid*

O método *multigrid*, proposto originalmente por Fedorenko (1964), é apresentado como uma técnica numérica alternativa para resolver iterativamente sistemas de equações obtidos com a discretização de uma equação diferencial que modela algum fenômeno físico. Suas conclusões mostraram que a velocidade de convergência com o uso da técnica *multigrid* é melhor que a dos métodos iterativos puros (sem o uso de *multigrid*). A ideia básica deste método é usar um conjunto de malhas e alternar suavizações em cada nível de malha e as aproximações destas soluções em uma malha mais grossa através de operadores que transferem informações da malha fina para a malha imediatamente mais

grossa (operador de restrição), ou transferir informação da malha grossa para a malha imediatamente mais fina (operador de prolongação) (WESSELING, 1992; BRIGGS et al., 2000; TROTTEMBERG et al., 2001).

A sequência com que as diversas malhas são visitadas caracteriza um ciclo *multigrid*, que pode ser do tipo V , W , F , entre outros. Os sistemas lineares em cada malha são suavizados com um método iterativo que goza das propriedades de reduzir rapidamente os erros oscilatórios (propriedades de suavização). Podem ser usados dois tipos de esquemas (BRIGGS et al., 2000): o esquema de correção (em inglês, *Correction Scheme*, CS) e o esquema de aproximação completa (em inglês, *Full Approximation Scheme*, FAS). De acordo com Briggs et al. (2000) e Trottenberg et al. (2001), o esquema CS é geralmente recomendado a problemas lineares e o FAS, a problemas não-lineares. Trottenberg et al. (2001) comentam que uma simples modificação no algoritmo do *multigrid* pode resultar em uma grande variação no tempo computacional (ou tempo de CPU, t_{CPU}).

O objetivo do método *multigrid* é acelerar a convergência de um esquema iterativo (TANNEHILL et al., 1997). Os melhores desempenhos do método *multigrid* são obtidos em problemas totalmente dominados pela difusão, ou seja, elípticos (WESSELING, 1992). Ferziger e Peric (2002) comentam que para problemas dominados pela advecção, os resultados ainda não se mostram totalmente satisfatórios e Stüben (2001), Brandt (1977), Wesseling e Oosterlee (2001) relatam que a eficiência ideal do método *multigrid* não tem sido totalmente alcançada em aplicações realísticas em CFD.

1.3 Motivação

Se por um lado tem-se a grande importância de modelos matemáticos para equações transientes, como a equação do calor (eq. de Fourier) e as equações da poroelasticidade, por outro lado, têm-se as dificuldades em se resolver numericamente esses tipos de problemas. Sabe-se ainda que a eficiência de certas ferramentas numéricas, como o método *multigrid*, ainda não tem sido totalmente alcançada em aplicações realísticas da Engenharia (BRIGGS et al., 2000; TROTTEMBERG et al., 2001), em especial para problemas transientes. Diante desta situação, pensou-se em buscar métodos que ajudem a suprir tal necessidade. A equação do calor, por exemplo, é uma equação bastante referenciada na literatura dos métodos numéricos aplicados na resolução de EDPs (FORTUNA, 2000; INCROPERA et al., 2008). Para tal equação, usam-se principalmente como metodologias de solução: o método *Time-Stepping* (TANNEHILL et al., 1997; STRIKWERDA, 1989; LENT, 2006), o método *Waveform Relaxation* (VANDEWALLE, 1993; JANSSEN, 1997; LENT, 2006) e o método *Space-Time* (HORTON; VANDEWALLE, 1995), contudo ainda sem atingir a plena eficiência, principalmente no que diz respeito à paralelização. As equações da poroelasticidade (dadas por um sistema de equações em regime transiente), por

exemplo, têm uma ampla área de aplicações em escoamentos em meios porosos, como reservatório de petróleo, processamento de alimentos, medicina, etc. Para tal equação, usa-se principalmente a metodologia *Time-Stepping* (GASPAR et al., 2003; WIENANDS et al., 2004; RODRIGO et al., 2016), também sem atingir a plena eficiência. Devido à grande importância de tais equações, suas aplicações e também à dificuldade em suas resoluções numéricas, pretende-se desenvolver métodos com baixo erro de discretização, baixo consumo de memória, alta velocidade de resolução e paralelizáveis.

1.4 Objetivos

O objetivo principal deste trabalho é melhorar ao menos um dos métodos para solução de problemas transientes, como os governados pelas equações do calor e da poroelasticidade, com o uso do método *multigrid*. Os principais métodos estudados para resolver tais problemas são: *Time-Stepping*, *Waveform Relaxation* e *Space-Time*. A partir desse estudo pretende-se aplicar o método melhorado para resolver a equação do calor e usar o método *Waveform Relaxation* para resolver as equações da poroelasticidade.

Os objetivos específicos estão resumidos como:

- Desenvolver códigos computacionais na linguagem Fortran 90 para resolver a equação do calor, com *multigrid*, e os métodos *Time-Stepping*, *Waveform Relaxation* e *Space-Time*;
- Verificar as vantagens e desvantagens de cada um dos respectivos métodos estudados;
- Otimizar parâmetros do método *multigrid* para cada caso estudado;
- Estudar o problema da poroelasticidade e desenvolver um código em Fortran 90 usando o método *multigrid* e o método *Waveform Relaxation* para resolvê-lo.

1.5 Delineamento do texto

Este texto é delineado da seguinte forma: o Cap. 2 apresenta a revisão bibliográfica; o Cap. 3 descreve a fundamentação teórica necessária para o embasamento deste trabalho e a teoria sobre o método *multigrid* é descrita no Cap. 4. O Cap. 5, descreve-se a análise de Fourier local (em inglês, *Local Fourier Analysis*, LFA) para um problema modelo dado pela equação de Poisson e os modelos matemático e numérico. As aproximações para as equações do calor e da poroelasticidade são apresentadas no Cap. 6. O Cap. 7 descreve os métodos *Time-Stepping*, *Waveform Relaxation* e *Space-Time* usados na solução dos problemas discretizados e com o uso do método *multigrid*. No Cap. 8 é apresentada uma breve verificação dos códigos computacionais desenvolvidos para resolver as equações

do calor e da poroelasticidade. Os resultados preliminares, considerando o estudo dos métodos *Time-Stepping*, *Waveform Relaxation* e *Space-Time* para a equação do calor são mostrados no Cap. 9. Os resultados, comparações e discussões estão descritos no Cap. 10 e as conclusões finais e as sugestões para trabalhos futuros estão descritas no Cap. 11.

2 Revisão bibliográfica

Os métodos numéricos usados para aproximar a solução de problemas oriundos da discretização de EDPs podem ser empregados em casos bem gerais, com geometrias arbitrárias e condições de contorno complicadas, por exemplo. Mesmo assim, na maioria das vezes apresentam resultados com maior rapidez quando comparados a métodos analíticos (MALISKA, 2004). De acordo com Fortuna (2000), em muitas situações os métodos numéricos consistem na forma mais prática, ou até mesmo única, de se obter informações sobre um determinado problema físico. Porém, mesmo com o uso de métodos numéricos, encontrar uma aproximação para a solução muitas vezes é difícil, pois esse processo depende de algumas propriedades das equações que modelam o problema físico, da malha onde as equações serão discretizadas, do suavizador utilizado, entre outros fatores. Neste contexto, o método *multigrid* é considerado muito bom para acelerar a convergência dos sistemas de equações obtidos a partir da discretização dos modelos matemáticos.

Neste capítulo será descrito o referencial teórico usado para os estudos do método *multigrid* e principalmente em suas aplicações para acelerar a convergência dos problemas dados pela equação do calor e pelas equações da poroelasticidade.

2.1 Método *multigrid*

Os primeiros estudos sobre o método *multigrid* foram realizados por Fedorenko (1964), que investigou a convergência de problemas de valor de fronteira de segunda ordem utilizando a equação de Poisson. Pouco depois, Bakhvalov (1966) fez um estudo da convergência para as equações elípticas usando a equação de advecção-difusão.

Porém, o reconhecimento do método *multigrid* ocorreu somente na década de 70, quando Brandt (1977) apresentou uma análise teórica e numérica que, além do método *multigrid* usado para equações lineares chamado esquema de correção, CS, incluiu uma introdução do esquema específico para problemas não lineares: o FAS. O trabalho também apresenta estudos sobre razões de engrossamento, análise local de Fourier e taxa de convergência.

Wesseling (1992) e Briggs et al. (2000) comentam que o método *multigrid* pode ser geométrico ou algébrico. O *multigrid* geométrico utiliza-se de informações das malhas consideradas na discretização do problema, portanto, ele é recomendado para problemas em malhas estruturadas. Por outro lado, o *multigrid* algébrico não faz uso desta estrutura, ele usa apenas dados da matriz do sistema, por isto, ele é recomendado para problemas em malhas não estruturadas. O método *multigrid* algébrico pode ser usado para vários tipos

de problemas em que a aplicação do método *multigrid* geométrico é difícil ou impossível. Ruge e Stuben (1986) citam que, por exemplo, um problema em que a discretização na malha mais fina não permite engrossamento uniforme para todos os pontos, pode ser resolvido pelo método *multigrid* algébrico. Neste trabalho, como são empregadas apenas malhas estruturadas, utiliza-se o método *multigrid* geométrico.

No método *multigrid* podem ser usados dois tipos de esquemas: o esquema CS e o esquema FAS. O método *multigrid* com esquema CS é indicado para a resolução de problemas lineares e o método *multigrid* com esquema FAS é indicado para problemas não lineares (BRIGGS et al., 2000; TROTTEBERG et al., 2001).

A forma com que as malhas são percorridas no método *multigrid* geométrico é chamada de ciclo. Os ciclos mais utilizados são o ciclo V , F e W . Uma forma de acelerar a convergência do método *multigrid* é iniciá-lo na malha mais grossa, caracterizando o denominado *multigrid* completo (em inglês, *Full Multigrid*, FMG) (WESSELING, 1992; BRIGGS et al., 2000; TROTTEBERG et al., 2001)

Trottenberg et al. (2001) afirmam que os métodos *multigrid* são talvez, os mais eficientes métodos para resolução de sistemas lineares obtidos com a discretização de EDPs elípticas, como a equação de Poisson. Esse método foi desenvolvido inicialmente para equações elípticas, mas tem sido aplicado com grande sucesso a uma variedade de problemas, como as equações de Euler e Navier-Stokes discretizadas (GHIA et al., 1982).

Wesseling e Oosterlee (2001) fizeram uma revisão do desenvolvimento do *multigrid* geométrico na década de 90, enfatizando aplicações em CFD e apresentando o estado da arte para escoamentos compressíveis e incompressíveis. Stüben (2001) fez uma análise semelhante, relativa ao mesmo período, enfocando o método *multigrid* algébrico. Neste trabalho, Stüben afirma que apesar do grande número de métodos desenvolvidos, ainda nenhum deles seria capaz de tratar com eficiência todos os problemas práticos em CFD. Apesar de não ter alcançado a plenitude da eficiência, esses autores apontaram o método *multigrid* como um dos mais importantes desenvolvimentos em análise numérica na segunda metade do século XX.

Santiago et al. (2015) analisaram o desempenho e os parâmetros que otimizam o método *multigrid* geométrico para as equações de Laplace, Navier, Burgers e duas formulações da equação de Navier-Stokes (função corrente – vorticidade $(\psi - \omega)$ e função corrente – velocidade $(\psi - u, v)$) usando discretização através do Método das Diferenças Finitas em malhas uniformes com aproximações numéricas de primeira e segunda ordem acurácia. Eles concluem que o desempenho do método *multigrid* parece estar relacionado à física do problema e não a formulação ou acoplamento entre as equações.

2.2 Equação do calor

A equação do calor (ou de Fourier) modela um problema de transferência de calor por condução. Existem porém, outras formas de transferência de calor, que são por convecção e por radiação. Um exemplo de transferência de calor por convecção pode ser observado em Nguyen et al. (2016) que investigaram a convecção térmica livre em meios porosos heterogêneos e sua dependência nas discretizações da malha para realizar experimentos numéricos. Mossi (2011) estudou a transferência de calor por radiação térmica e analisou seus efeitos em uma chama laminar resultante da combustão de metano com o ar.

O primeiro método usado para resolver numericamente a equação do calor foi o método *Time-Stepping*. Esse método resolve o sistema discretizado em cada passo de tempo de forma subsequente até o passo de tempo final, e considera-se a solução encontrada no passo de tempo anterior como condição inicial para o passo de tempo atual. Vandewalle (1993), Lent (2006), Falgout et al. (2017) e muitos outros autores falam da eficiência em usar o método *multigrid* associado ao método *Time-Stepping*. Esta eficiência se dá pelo fato de se resolver um sistema discretizado em cada passo de tempo (VANDEWALLE, 1993). Com isso, esse sistema se comporta como um sistema oriundo da discretização de EDPs elípticas, para as quais o método *multigrid* é comprovadamente eficiente (BRANDT, 1977; WESSELING, 1992; BRIGGS et al., 2000; TROTTEBERG et al., 2001).

Vargas (2013) resolveu a equação do calor uni e bidimensional usando o método *Time-Stepping* e fez também uma análise *a priori* e *a posteriori* das ordens efetivas, aparentes e assintóticas para o erro numérico, usando os métodos de Euler implícito e CN.

Devido ao avanço computacional, o desenvolvimento de computadores com muitos núcleos de processadores e o fato do algoritmo do método *Time-Stepping* não permitir paralelização no tempo, são realizadas pesquisas de algoritmos que sejam eficientes e permitam o uso de paralelização de computadores. Dentre esses métodos pode-se citar o método *Waveform Relaxation* (LELARASMEE et al., 1982), *multigrid* parabólico (VANDEWALLE; HORTON, 1993), *multigrid* paralelo no tempo (VANDEWALLE; HORTON, 1995), *multigrid* no espaço e tempo (em inglês, *space-time multigrid*) (HORTON; VANDEWALLE, 1995), *multigrid Waveform Relaxation* com redução cíclica (WRMG-CR) e *multigrid* com redução no tempo (MGRIT) (FALGOUT et al., 2017), novo *multigrid* espaço-tempo em paralelo (em inglês, *New Space-Time parallel multigrid*) (GANDER; NEUMÜLLER, 2016), entre outros.

De acordo com Vandewalle (1993), Tannehill et al. (1997), Strikwerda (1989) e Lent (2006), verifica-se que o método *Time-Stepping* considera a equação diferencial parabólica como uma sequência de equações elípticas e soluciona-se uma equação elíptica a cada passo de tempo até alcançar uma aproximação para a solução no passo de tempo desejado.

O método *Waveform Relaxation*, também chamado de método de iteração dinâmica (em inglês, *Dynamic Iteration*) (MIEKKALA; NEVANLINNA, 1997), é uma técnica para resolver um sistema de equações diferenciais ordinárias com condição inicial. O método *Waveform Relaxation* foi inicialmente estudada por Lelarasmee et al. (1982) como um método de solução prática no contexto de simulação de circuitos integrados. Sua convergência foi descrita por Miekkala e Nevanlinna (1997). A combinação desse método com a técnica *multigrid* foi inicialmente estudado por Lubich e Ostermann (1987), Lent e Vandewalle (2002). Lubich e Ostermann (1987) descreveram o potencial desse método para o paralelismo e ilustraram seus resultados teóricos para um problema modelo unidimensional.

Vandewalle e Horton (1993) trataram a variável temporal como uma variável espacial, caracterizando o método *Waveform Relaxation*. Consideraram este problema envolvendo variáveis espaciais e temporal como um problema anisotrópico, trabalharam com paralelismo e utilizaram a análise local de Fourier (LFA) para prever os resultados numéricos.

Jansen et al. (1994) investigaram a estabilidade e convergência do método *Waveform Relaxation* aplicado à equação do calor 1D e 2D de forma teórica e compararam com experimentos numéricos.

Vandewalle e Horton (1995) fizeram uma comparação das propriedades de convergência entre o método *Waveform Relaxation* e o método *multigrid* paralelo no tempo (em inglês, *Time-parallel multigrid*) por meio da LFA e mostraram que o método *Waveform Relaxation* é robusto em relação ao tamanho de passos no espaço e tempo, enquanto o método paralelo no tempo depende do fator de anisotropia associado à discretização espacial e temporal para garantir convergência.

Horton e Vandewalle (1995) apresentam um método que resolve a equação do calor no espaço e tempo simultaneamente, o método *Space-Time multigrid*. Esses autores apresentaram os suavizadores com o método *multigrid* e os operadores de restrição e prolongação. Para a estratégia de engrossamento, os operadores de restrição e prolongação dependem, na discretização em cada nível de malha, do grau de anisotropia entre as variáveis espaciais e temporal. Os resultados para a equação do calor unidimensional e bidimensional mostraram-se concordantes com os resultados previstos pela análise dos modos de Fourier – LFA.

Janssen (1997) fez um estudo sobre a forma de se acelerar a convergência do método *Waveform Relaxation*. Entre os métodos descritos, está o método *multigrid*. Nessa tese foram comparados os fatores de convergência da equação do calor unidimensional e bidimensional obtidos numericamente e com a análise de Fourier. Para isso, foram usados os métodos de discretização das diferenças finitas e dos elementos finitos com diferentes suavizadores para a variável temporal.

Lent e Vandewalle (2002) estenderam o caso do método *multigrid* aplicado em problemas estacionários ou elípticos ao método *Waveform Relaxation*, para problemas dependentes do tempo. Nesse artigo, fez-se um estudo de vários tipos de suavizadores e ordem de atualização das incógnitas, do algoritmo para o método *Waveform Relaxation*, de algumas estratégias de engrossamento e da análise de Fourier-Laplace usando modos locais de Fourier. Os resultados obtidos numericamente para os casos isotrópicos e anisotrópicos foram comparados com a teoria e apresentaram boa concordância.

Lent e Vandewalle (2005) e Lent (2006) fizeram um estudo sobre os métodos iterativos para problemas dependentes do tempo usando como base a equação do calor. Mostraram também resultados com métodos implícitos para a aproximação temporal. Em sua tese, Lent (2006) usou os métodos *Time-Stepping* e *Waveform Relaxation* aplicados a problemas isotrópicos e anisotrópicos com esquemas de discretização ou aproximação temporal de altas ordens.

Gander e Vandewalle (2007) estudaram o método *Time-Parallel Time-Integration*, que aproxima partes da solução no tempo subsequente simultaneamente com partes da solução no tempo anterior e fizeram um estudo dos tipos de convergência com intervalos de tempo limitados e ilimitados..

Oliveira et al. (2012) fizeram um estudo sobre o método *multigrid* aplicado em malhas anisotrópicas. Propuseram os seguintes esquemas de restrição para malhas anisotrópicas: meia ponderação geométrica, ponderação completa geométrica e ponderação completa e parcial ponderadas.

Falgout et al. (2014) descrevem e demonstram com experimentos numéricos a eficiência do algoritmo *multigrid* com redução no tempo (MGRIT) para resolver a equação da difusão em duas e três dimensões e usando paralelismo também na direção temporal. Para o caso de problemas não lineares, Emmett e Minion (2012) apresentaram um novo método para a paralelização dessas EDPs (não lineares) na direção temporal. Essa abordagem usa o esquema FAS e permite uma nova forma de paralelização no tempo, em que o método iterativo consiste em suavizar alternadamente nas malhas fina e na malha grossa de uma discretização espacial e temporal. Os resultados numéricos e em uma, duas e três dimensões espaciais demonstram o potencial da abordagem para proporcionar paralelização eficiente na direção temporal.

Gander (2015) faz uma abordagem sobre as técnicas existentes para paralelização do tempo nos últimos 50 anos e comenta que estas podem ser classificadas em quatro grupos: métodos baseados em múltiplos tiros (em inglês, *multiple shooting*), método *waveform relaxation*, técnicas de *multigrid* em espaço e tempo (em inglês, *space-time methods*) e métodos de paralelização direta no tempo.

Gander e Neumüller (2016) apresentaram e analisaram um novo método no espaço

e tempo (em inglês, *new space-time*) para equações parabólicas. Esse método usa uma discretização de alta ordem de Galerkin no tempo e o método de elementos finitos no espaço. A ideia chave desse novo método consiste no uso do suavizador Jacobi por blocos, em que para cada plano espacial é usado um ciclo *multigrid*.

Falgout et al. (2017) compararam os métodos *multigrid* para EDP's parabólicas que permitem engrossamento no espaço, no tempo e no espaço e tempo simultaneamente. Ao invés de buscar o aumento da velocidade de *clock* dos processadores, busca-se pelo desenvolvimento de métodos paralelizáveis no tempo e que permitem a solução simultânea em vários passos de tempo. Neste trabalho comparou-se o tradicional método *Time-Stepping* com os métodos que abordam o espaço e tempo simultaneamente, que são: *Space-Time*, *Waveform Relaxation* com redução cíclica e *multigrid* com redução em tempo. Verifica-se que usando o paralelismo o desempenho de todos os três métodos que resolvem simultaneamente nas direções espaciais e temporal (*Space-Time*) são melhores que o método *Time-Stepping*. Entre os métodos espaço-tempo observa-se que com o número de processadores utilizados, o método chamado *multigrid Space-Time* apresenta melhores resultados em relação ao tempo computacional e os métodos *multigrid* com redução no tempo e o *new space-time* com suavização por blocos são mais eficientes em relação ao paralelismo.

2.3 Equações da poroelasticidade

Os meios porosos são assumidos como sendo um composto que contém uma matriz sólida e uma rede de poros. A presença de um fluido em movimento em tal rede pode afetar a resposta mecânica do esqueleto sólido e, ao mesmo tempo, as mudanças mecânicas influenciam o comportamento do fluido dentro dos poros. Assim, as equações da poroelasticidade modelam matematicamente a interação entre a deformação de um material elástico poroso e o fluxo de fluido dentro dele. As primeiras explicações sobre a influência dos fluidos na deformação dos sólidos encontram-se em Terzaghi (1943), que descreveu seus estudos a partir de experimentos unidimensionais. A teoria geral tridimensional da poroelasticidade foi formulada por Biot em vários trabalhos pioneiros nesta área, descritos em Biot (1941) e Biot (1955) e recebem o seu nome, sendo até hoje chamada de modelo de consolidação de Biot. Hoje em dia, a análise e a simulação numérica do modelo de Biot tornaram-se cada vez mais populares devido à ampla gama de aplicações da teoria da poroelasticidade em diferentes ramos de pesquisa como, medicina, biomecânica, engenharia de petróleo, processamento de alimentos e outros campos da ciência e engenharia.

Gaspar et al. (2003) apresentaram as estimativas de estabilidade e análise de convergência para o modelo de consolidação da Biot discretizado pelo Métodos das Diferenças Finitas. Inicialmente, são analisadas as diferenças centrais para a discretização

do espaço e um esquema ponderado de dois níveis para o tempo. Para melhorar algumas limitações de estabilidade e convergência para este esquema, foi considerado discretizações espaciais em malhas deslocadas (em que os valores para os deslocamentos e pressão são calculados em diferentes nós). Estes autores apresentaram resultados numéricos para ilustrar os resultados teóricos obtidos.

Wienands et al. (2003) mostraram um método de suavização distributiva para o sistema de equações da poroelasticidade em que também se usaram malhas deslocadas. As propriedades de suavização foram melhoradas com a ajuda da análise de Fourier e em conexão com o suavizador e aliado a bons operadores de transferência e correção na malha grossa, obteve-se um método *multigrid* eficiente.

Em Wienands et al. (2004) foi apresentado um suavizador distributivo e robusto com o método *multigrid* para o sistema de equações da poroelasticidade. Dentro da estrutura distributiva, usou-se um sistema desacoplado, que pôde ser suavizado com métodos iterativos básicos, como o método de Jacobi *red-black*. As propriedades de suavização foram otimizadas com a ajuda da análise de Fourier. Os resultados obtidos com a análise de duas malhas de Fourier e por experimentos numéricos mostraram um método *multigrid* altamente eficiente para esse tipo de problema.

Gaspar et al. (2006) usaram o Método das Diferenças Finitas para discretizar o problema de consolidação de Biot quasi-estático. A discretização espacial foi feita com aproximações em malhas deslocadas, e para aproximar as variáveis no tempo foi usado o método de Euler implícito. Foram obtidas estimativas *a priori* dos deslocamentos e da pressão e os correspondentes resultados de convergência foram comprovados através de experimentos numéricos.

Gaspar et al. (2007) apresentaram um suavizador robusto e eficiente para uma versão transformada do sistema de equações da poroelasticidade com *multigrid*. Essa transformação apresentada permitiu tratar o sistema de forma desacoplada. Os autores mostraram que a referida transformação se resume a um termo adicional de estabilização no esquema iterativo com a solução do problema original idêntica à solução do problema transformado. Um método *multigrid* altamente eficiente pôde ser desenvolvido, confirmado por experiências numéricas.

Rodrigo (2010) descreve em sua tese a estrutura e a eficiência de um método *multigrid* geométrico em malhas triangulares semiestruturadas para uma discretização através do método de elementos finitos do problema da poroelasticidade, usando o suavizador Vanka. Essa autora mostra que as oscilações não-físicas na pressão são eliminadas completamente adicionando um termo de estabilização (descrito em Gaspar et al. (2007)). Devido ao fato das equações da poroelasticidade apresentarem pontos de sela, em que o laplaciano da pressão é quase nulo, o esquema de suavização usado para essas equações é o esquema de suavização por blocos (*box-relaxation*) e parece ser a

melhor opção. Esta classe de suavizadores foi introduzida por Vanka (1986) para resolver a equação de Navier-Stokes discretizada pelo Método das Diferenças Finitas em uma malha retangular. Desde então, pôde-se encontrar muitas referências sobre a aplicação deste tipo de suavizador, principalmente na área de CFD, por exemplo em John (1999) e Turek (1999). Rodrigo (2010) também usou a análise de Fourier para obter as propriedades de convergência do suavizador Vanka e comparou os resultados teóricos com resultados obtidos através de experimentos numéricos.

3 Fundamentação teórica

Neste capítulo faz-se uma breve descrição dos principais temas abordados nesta tese. Primeiramente, faz-se uma abordagem sobre o Método das Diferenças Finitas (MDF), que é um método clássico e que apesar da base matemática não ser nova, pode apresentar diferentes formulações de aproximação numérica (TANNEHILL et al., 1997; STRIKWERDA, 1989). Depois disso, mostra-se uma das notações usadas neste trabalho, que é a notação estêncil. Também faz-se uma abordagem dos métodos iterativos e os métodos usados para aproximar a variável temporal, como o método de Euler, o método de Crank-Nicolson (CN), entre outros, veja por exemplo, Strikwerda (1989), Hirsch (2007) e Burden e Faires (2016). O método *multigrid* está exposto no capítulo seguinte.

3.1 Método das Diferenças Finitas (MDF)

De acordo com Ferziger e Peric (2002), o primeiro passo para a obtenção de uma solução numérica é discretizar o domínio geométrico, isto é, definir ao longo do domínio em quais pontos se deseja conhecer o valor da variável dependente através do auxílio de uma malha geométrica discreta. O princípio fundamental do MDF é aproximar, através de expressões algébricas, cada termo do modelo matemático em cada ponto (nó) dessa malha discretizada. Para isso, no MDF a malha geométrica é geralmente estruturada localmente, ou seja, cada ponto (nó) pode ser considerado como a origem de um sistema de coordenadas local, cujos eixos coincidem com as linhas da malha. A Fig. 1 e a Fig. 2 mostram exemplos de malhas cartesianas unidimensional (1D) e bidimensional (2D) usadas no MDF, respectivamente. Nessas figuras e nas demais figuras desta seção, o símbolo (●) denota os nós sobre os contornos e o símbolo (○) denota os nós internos da malha. No caso 1D, cada nó é representado pela posição i . No caso 2D, cada nó é unicamente definido pela interseção das linhas de malhas na posição (i, j) . Os nós vizinhos são definidos aumentando-se ou diminuindo-se uma unidade de cada índice i ou j . Assim o nó $(i - 1, j)$ representa a posição Oeste, $(i + 1, j)$ representa o nó na posição Leste, $(i, j - 1)$ representa o nó na posição Sul e $(i, j + 1)$ representa o nó na posição Norte.

A Fig. 3 mostra o MDF aplicado a um modelo 1D usando a discretização uniforme, isto é, uma malha de nós igualmente espaçados, com espaço entre os nós dado por h .

A filosofia do MDF é fazer aproximações diretamente da definição de derivada através da série de Taylor. Uma interpretação geométrica das aproximações central, adiantada e atrasada é mostrada na Fig. 4 com o auxílio de uma função contínua Φ . A primeira derivada $\frac{d\Phi}{dx}$ em um ponto x_i indica a inclinação da reta tangente à curva no ponto $(x_i, \Phi(x_i))$ e isso é mostrado com a reta marcada "Exata" na Fig. 4. Essa inclinação pode

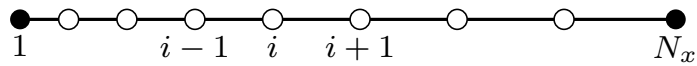


Figura 1 – Exemplo de uma malha unidimensional não uniforme.

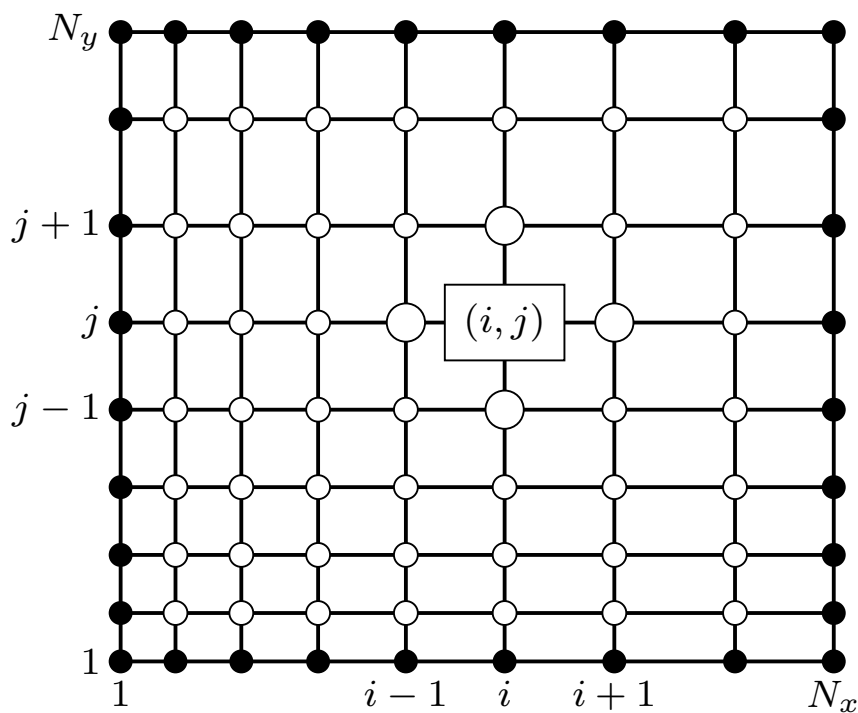


Figura 2 – Exemplo de uma malha bidimensional não uniforme.

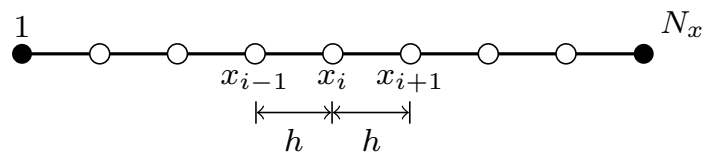


Figura 3 – Malha unidimensional uniforme com tamanho dos elementos de malha h .

ser aproximada pela inclinação da reta que passa por dois pontos vizinhos na curva. A reta marcada como "Adiantada" mostra que a derivada no ponto $(x_i, \Phi(x_i))$ é aproximada pela inclinação de uma reta que passa pelos pontos $(x_i, \Phi(x_i))$ e $(x_{i+1}, \Phi(x_{i+1}))$. A reta denotada "Atrasada" ilustra que a derivada no ponto $(x_i, \Phi(x_i))$ é aproximada pela inclinação de

uma reta que passa pelos pontos $(x_i, \Phi(x_i))$ e $(x_{i-1}, \Phi(x_{i-1}))$. A reta denominada "Central" representa aproximação por uma diferença central, ou seja, a inclinação da reta tangente no ponto $(x_i, \Phi(x_i))$ é aproximada pela inclinação de uma reta que passa pelos pontos $(x_{i-1}, \Phi(x_{i-1}))$ e $(x_{i+1}, \Phi(x_{i+1}))$, que são pontos situados em lados opostos do ponto em que a derivada é aproximada.

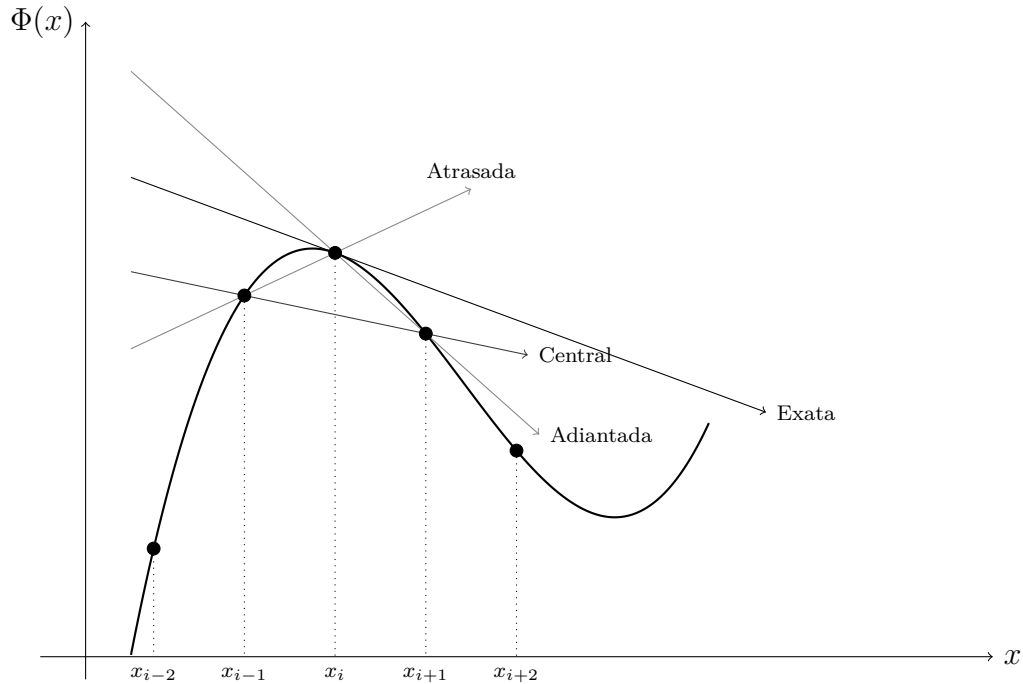


Figura 4 – Reta tangente (Exata) à curva Φ no ponto P e as aproximações Atrasada, Adiantada e Central através da discretização pelo MDF.

Na Fig. 4 observa-se que algumas aproximações para a derivada são melhores que outras (em relação a inclinação com o eixo x). Para a reta "Central", verifica-se que ocorre uma melhor aproximação em relação a reta "Exata". Verifica-se também que se h tende a zero ($h \rightarrow 0$), todas as aproximações (Atrasada, Adiantada e Central) se aproximam da solução "Exata".

Definição 3.1. Uma função $\Phi : I \rightarrow \mathbb{R}$, definida num intervalo aberto I , chama-se *analítica* quando, para cada $x \in I$ existe um $\varepsilon > 0$ tal que a série de Taylor $\sum_{n=0}^{+\infty} \frac{\Phi^{(n)}(a)}{n!} h^n$ converge para $\Phi(a + h)$ desde que $|h| < \varepsilon$.

3.1.1 Expansão em fórmula de Taylor

Para expressar cada tipo de aproximação e a respectiva ordem do erro, usa-se a série de Taylor (KREYSZIG, 1999; FERZIGER; PERIC, 2002; MARCHI, 2001). Qualquer função $\Phi(x)$ analítica na vizinhança de x_i , pode ser expressa como uma série de Taylor:

$$\Phi_x = \Phi_i + (x - x_i) \left(\frac{d\Phi}{dx} \right)_i + \frac{(x - x_i)^2}{2!} \left(\frac{d^2\Phi}{dx^2} \right)_i + \frac{(x - x_i)^3}{3!} \left(\frac{d^3\Phi}{dx^3} \right)_i + \dots \quad (3.1)$$

em que Φ_x representa $\Phi(x)$, Φ_i representa $\Phi(x_i)$ e $\left(\frac{d\Phi}{dx}\right)_i$ representa $\frac{d\Phi}{dx}(x_i)$.

O valor de Φ_x é exato se forem considerados todos os termos da série de Taylor. Para aplicar a Eq. (3.1) aos nós $i - 1$ e $i + 1$, a partir do ponto i , como é mostrado na Fig. 3 e com o valor de h sendo constante, pode-se fazer

$$\begin{aligned} x = x_{i+1} &\implies (x - x_i) = (x_{i+1} - x_i) = h, \\ \Phi_{i+1} &= \Phi_i + h \left(\frac{d\Phi}{dx}\right)_i + \frac{h^2}{2} \left(\frac{d^2\Phi}{dx^2}\right)_i + \frac{h^3}{6} \left(\frac{d^3\Phi}{dx^3}\right)_i + \dots, \end{aligned} \quad (3.2)$$

$$\begin{aligned} x = x_{i-1} &\implies (x - x_i) = (x_{i-1} - x_i) = -h, \\ \Phi_{i-1} &= \Phi_i - h \left(\frac{d\Phi}{dx}\right)_i + \frac{h^2}{2} \left(\frac{d^2\Phi}{dx^2}\right)_i - \frac{h^3}{6} \left(\frac{d^3\Phi}{dx^3}\right)_i + \dots \end{aligned} \quad (3.3)$$

De acordo com Tannehill et al. (1997), Marchi (2001) e Burden e Faires (2016) as expressões para a derivada de primeira ordem podem ser obtidas por combinação das Eqs. (3.2) e (3.3). A escolha de quais equações usar define o tipo de esquema e a ordem do erro de truncamento cometido. A seguir serão apresentados alguns dos esquemas mais comuns e que serão usados no decorrer deste texto. Para outras aproximações, veja por exemplo Burden e Faires (2016) e Ferziger e Peric (2002).

Isolando-se a derivada de primeira ordem, $\left(\frac{d\Phi}{dx}\right)_i$, na Eq. (3.2), tem-se

$$\left(\frac{d\Phi}{dx}\right)_i = \underbrace{\frac{\Phi_{i+1} - \Phi_i}{h}}_{\text{considerado}} - \underbrace{\frac{h}{2} \left(\frac{d^2\Phi}{dx^2}\right)_i - \frac{h^2}{6} \left(\frac{d^3\Phi}{dx^3}\right)_i + \dots}_{\text{desprezado (erro de truncamento)}} \quad (3.4)$$

Assim, a derivada de primeira ordem de Φ é aproximada de forma atrasada (em inglês, *Downstream Difference Scheme*, DDS) a partir do ponto i através de

$$\left(\frac{d\Phi}{dx}\right)_i^{DDS} = \frac{\Phi_{i+1} - \Phi_i}{h} + O(h), \quad (3.5)$$

e o erro de truncamento $\left(\varepsilon \left(\frac{d\Phi}{dx}\right)_i^{DDS}\right)$ é dado por

$$\varepsilon \left(\frac{d\Phi}{dx}\right)_i^{DDS} = -\frac{h}{2} \left(\frac{d^2\Phi}{dx^2}\right)_i - \frac{h^2}{6} \left(\frac{d^3\Phi}{dx^3}\right)_i - \frac{h^3}{24} \left(\frac{d^4\Phi}{dx^4}\right)_i - \dots = O(h). \quad (3.6)$$

Isolando-se a primeira derivada na Eq. (3.3), tem-se

$$\left(\frac{d\Phi}{dx}\right)_i = \underbrace{\frac{\Phi_i - \Phi_{i-1}}{h}}_{\text{considerado}} + \underbrace{\frac{h}{2} \left(\frac{d^2\Phi}{dx^2}\right)_i - \frac{h^2}{6} \left(\frac{d^3\Phi}{dx^3}\right)_i + \dots}_{\text{desprezado (erro de truncamento)}} \quad (3.7)$$

Assim, a derivada de primeira ordem de Φ é aproximada de forma adiantada (em inglês, *Upstream Difference Scheme*, UDS) a partir do ponto i , através de

$$\left(\frac{d\Phi}{dx}\right)_i^{UDS} = \frac{\Phi_{i+1} - \Phi_i}{h} + O(h), \quad (3.8)$$

e o erro de truncamento é dado por

$$\varepsilon \left(\frac{d\Phi}{dx}\right)_i^{UDS} = \frac{h}{2} \left(\frac{d^2\Phi}{dx^2}\right)_i - \frac{h^2}{6} \left(\frac{d^3\Phi}{dx^3}\right)_i + \frac{h^3}{24} \left(\frac{d^4\Phi}{dx^4}\right)_i - \dots = O(h). \quad (3.9)$$

Subtraindo-se a Eq. (3.3) da Eq. (3.2), tem-se que

$$\Phi_{i+1} - \Phi_{i-1} = 2h \left(\frac{d\Phi}{dx}\right)_i + \frac{2h^3}{6} \left(\frac{d^3\Phi}{dx^3}\right)_i + \frac{2h^5}{120} \left(\frac{d^5\Phi}{dx^5}\right)_i + \dots, \quad (3.10)$$

ou

$$\left(\frac{d\Phi}{dx}\right)_i = \underbrace{\frac{\Phi_{i+1} - \Phi_{i-1}}{2h}}_{\text{considerado}} - \underbrace{\frac{h^2}{6} \left(\frac{d^3\Phi}{dx^3}\right)_i - \frac{h^4}{120} \left(\frac{d^5\Phi}{dx^5}\right)_i - \dots}_{\text{desprezado (erro de truncamento)}}. \quad (3.11)$$

Assim, a derivada de primeira ordem de Φ é aproximada de forma centrada (em inglês, *Central Difference Scheme*, CDS) para o ponto i , através de

$$\left(\frac{d\Phi}{dx}\right)_i^{CDS} = \frac{\Phi_{i+1} - \Phi_{i-1}}{2h} + O(h^2), \quad (3.12)$$

e o erro de truncamento é dado por

$$\varepsilon \left(\frac{d\Phi}{dx}\right)_i^{CDS} = -\frac{h^2}{6} \left(\frac{d^3\Phi}{dx^3}\right)_i - \frac{h^4}{120} \left(\frac{d^5\Phi}{dx^5}\right)_i - \dots = O(h^2). \quad (3.13)$$

De forma semelhante ao que foi feito na aproximação da derivada de primeira ordem, seguem-se de Tannehill et al. (1997), Marchi (2001) e Burden e Faires (2016), as relações para expressar as aproximações da derivada de segunda ordem.

Somando-se a Eq. (3.2) e a Eq. (3.3), tem-se

$$\Phi_{i-1} + \Phi_{i+1} = 2\Phi_i + h^2 \left(\frac{d^2\Phi}{dx^2}\right)_i + \frac{h^4}{12} \left(\frac{d^4\Phi}{dx^4}\right)_i + \frac{h^6}{360} \left(\frac{d^6\Phi}{dx^6}\right)_i + \dots, \quad (3.14)$$

ou

$$\left(\frac{d^2\Phi}{dx^2}\right)_i = \underbrace{\frac{\Phi_{i-1} - 2\Phi_i + \Phi_{i+1}}{h^2}}_{\text{considerado}} - \underbrace{\frac{h^2}{12} \left(\frac{d^4\Phi}{dx^4}\right)_i - \frac{h^4}{360} \left(\frac{d^6\Phi}{dx^6}\right)_i - \dots}_{\text{desprezado (erro de truncamento)}} \quad (3.15)$$

Assim, a derivada de segunda ordem de Φ é aproximada de forma centrada com 3 pontos (CDS) para o ponto P , através de

$$\left(\frac{d^2\Phi}{dx^2}\right)_i^{CDS} = \frac{\Phi_{i-1} - 2\Phi_i + \Phi_{i+1}}{h^2} + O(h^2), \quad (3.16)$$

e o erro de truncamento é dado por

$$\varepsilon \left(\frac{d^2\Phi}{dx^2}\right)_i^{CDS-2} = -\frac{h^2}{12} \left(\frac{d^4\Phi}{dx^4}\right)_i - \frac{h^4}{360} \left(\frac{d^6\Phi}{dx^6}\right)_i - \frac{h^6}{20160} \left(\frac{d^8\Phi}{dx^8}\right)_i - \dots = O(h^2). \quad (3.17)$$

3.2 Notação estêncil

A notação estêncil é apropriada para definir um operador diferencial discreto L_h em uma malha cartesiana retangular Ω^h . Trottenberg et al. (2001) e Wienands e Joppich (2005) afirmam que é conveniente usar a terminologia estêncil para descrever funções de malhas do tipo $\omega_h : \Omega^h \rightarrow \mathbb{R}$ e também para facilitar o uso da análise de Fourier local. Assim, considera-se a função de malha

$$\begin{aligned} \omega_h : \Omega^h &\longrightarrow \mathbb{R} \\ (x, y) &\longmapsto \omega_h(x, y). \end{aligned}$$

Um estêncil geral $[S_{\kappa_1\kappa_2}]_h$ dado por

$$[S_{\kappa_1\kappa_2}]_h = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cdots & S_{-1,1} & S_{0,1} & S_{1,1} & \cdots \\ \cdots & S_{-1,0} & S_{0,0} & S_{1,0} & \cdots \\ \cdots & S_{-1,-1} & S_{0,-1} & S_{1,-1} & \cdots \\ \vdots & \vdots & \vdots \end{bmatrix}, \quad S_{\kappa_1\kappa_2} \in \mathbb{R},$$

em que $\kappa_1, \kappa_2 \in \mathbb{Z}$ indicam as posições no estêncil, define um conjunto de funções de malha dadas por

$$[S_{\kappa_1\kappa_2}]_h \omega_h(x, y) = \sum_{\kappa_1, \kappa_2} \omega_h(x + \kappa_1 h_x, y + \kappa_2 h_y), \quad (3.18)$$

em que h_x e h_y indicam as distâncias entre os nós espaciais nas direções x e y , respectivamente.

Aqui, assume-se que somente um número finito de coeficientes $S_{\kappa_1\kappa_2}$ são diferentes de zero.

Os estêncis mais comuns são os compactos de cinco pontos e de nove pontos, dados na malha h , respectivamente por

$$\begin{bmatrix} & S_{0,1} & \\ S_{-1,0} & S_{0,0} & S_{1,0} \\ & S_{0,-1} & \end{bmatrix}_h \quad \text{e} \quad \begin{bmatrix} S_{-1,1} & S_{0,1} & S_{1,1} \\ S_{-1,0} & S_{0,0} & S_{1,0} \\ S_{-1,-1} & S_{0,-1} & S_{1,-1} \end{bmatrix}_h. \quad (3.19)$$

Para ilustrar este procedimento, considera-se o operador Laplaciano $Lu = -\Delta u = -(u_{xx} + u_{yy})$. A discretização em uma malha $\Omega^h : [0, 1] \times [0, 1]$ através do Método das Diferenças Finitas usando aproximação de segunda ordem do tipo CDS, conduz a

$$\begin{aligned} L_h u_h(x, y) &= -\Delta_h u_h(x, y) \\ &= \frac{1}{h^2} [4u_h(x, y) - u_h(x - h, y) - u_h(x + h, y) - u_h(x, y - h) - u_h(x, y + h)] \\ &= \frac{1}{h^2} \begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}_h u_h(x, y). \end{aligned} \quad (3.20)$$

Assim

$$L_h = \frac{1}{h^2} \begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}_h$$

representa o estêncil de cinco pontos do operador Laplaciano discreto $-\Delta_h$.

3.3 Métodos iterativos

Os métodos mais comuns usados para resolver sistemas de equações lineares são os métodos diretos e os métodos iterativos. Os métodos diretos são mais usados para matrizes de pequeno porte e densas, enquanto que os métodos iterativos são mais eficazes quando aplicados a matrizes de grande porte e esparsas (muitos elementos iguais a zero) (BURDEN; FAIRES, 2016). Como neste caso a discretização é feita através do MDF, as matrizes obtidas com as discretizações possuem como característica a esparsidade.

Os métodos iterativos, ao contrário dos métodos diretos, fornecem a resposta como uma aproximação da solução, e isso pode ser variável. De acordo com Saad (2003), os métodos iterativos começam com uma solução aproximada dada e modificam as componentes da aproximação, uma variável ou um bloco de variáveis, em uma determinada ordem, até que um critério de convergência seja atingido. Cada uma dessas modificações, chamadas de passos de relaxamento, destina-se a suavizar as componentes do erro.

Considerando o sistema de equações lineares dado por

$$Au = f, \quad (3.21)$$

com A sendo uma matriz quadrada de ordem $n \times n$ e u e f vetores de ordem n , tem-se

$$\begin{cases} a_{11}u_1 + a_{12}u_2 + \dots + a_{1n}u_n = f_1 \\ a_{21}u_1 + a_{22}u_2 + \dots + a_{2n}u_n = f_2 \\ \vdots \\ a_{n1}u_1 + a_{n2}u_2 + \dots + a_{nn}u_n = f_n \end{cases}. \quad (3.22)$$

A matriz dos coeficientes A do sistema (3.22) pode ser dividida na forma:

$$A = -L + D - U, \quad (3.23)$$

em que $-L$ é uma matriz triangular inferior formada pela parte inferior da matriz A , D é a matriz formada pela diagonal de A e $-U$ é uma matriz triangular superior formada pela parte superior da matriz A .

3.3.1 Método de Jacobi

Seguindo Ruggiero e Lopes (1996), Saad (2003) e Ferziger e Peric (2002) e considerando a matriz D não singular, ou seja $d_{ii} \neq 0 \forall i$, pode-se reescrever o sistema linear dado na Eq. (3.21) na forma

$$\begin{aligned} (-L + D - U)u &= f \\ Du &= (L + U)v + f. \end{aligned} \quad (3.24)$$

Isolando-se u na iteração atual e considerando-se v como sendo sua aproximação, o procedimento que caracteriza o método de Jacobi na forma matricial é dado por

$$v^{\nu+1} = D^{-1}(L + U)v^{\nu} + D^{-1}f. \quad (3.25)$$

3.3.2 Método de Gauss-Seidel

Da mesma forma que o método de Jacobi, considerando-se a matriz D não singular, pode-se escrever o sistema linear dado na Eq. (3.21) na forma

$$\begin{aligned} (-L + D - U)u &= f, \\ (-L + D)u &= Uu + f. \end{aligned}$$

Isolando-se u na iteração atual e considerando-se v como sendo sua aproximação, o procedimento que caracteriza o método de Gauss-Seidel na forma matricial é dado por

$$v^{\nu+1} = (-L + D)^{-1}Uv^{\nu} + (-L + D)^{-1}f. \quad (3.26)$$

Nos procedimentos descritos anteriormente, para Jacobi e Gauss-Seidel, a matriz de iteração é constante ao longo do processo iterativo. Os métodos iterativos que possuem esta propriedade são chamados métodos iterativos básicos ou estacionários (TROTTEBERG et al., 2001).

3.3.3 Suavizador Vanka

O suavizador chamado de Vanka faz parte de uma classe de suavizadores que realizam suavização por blocos (em inglês, *box-relaxation*) e é indicado para resolver

problemas que são modelados por sistemas de equações e possuem pontos de sela ($K\Delta p \approx 0$). De acordo com Rodrigo (2010), esse suavizador Vanka consiste em um algoritmo iterativo na forma de Gauss-Seidel com um sistema pontual e que atualiza simultaneamente todas as incógnitas do deslocamento que estão ao redor de um nó, em que também é atualizado o valor da pressão. Um esboço dos blocos de incógnitas é mostrado na Fig. 5. Portanto, todas as incógnitas no sistema são consideradas acopladas, e um sistema de equações deve ser resolvido para cada ponto da malha.

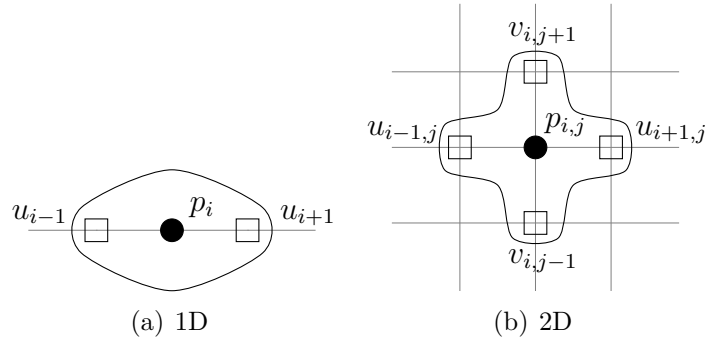


Figura 5 – Suavizador Vanka (usado para o problema da poroelasticidade).

Para resolver o sistema de equações em cada ponto da malha com as incógnitas dadas pelos deslocamentos e pressão, como mostra a Fig. 5, tem-se o sistema de equações descrito na Eq. (3.27)

$$\begin{pmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{d}^t & c \end{pmatrix} \begin{pmatrix} \mathbf{x}^u \\ x^p \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ f \end{pmatrix}, \quad (3.27)$$

em que se considera \mathbf{A} uma matriz contendo os coeficientes associados aos deslocamentos desconhecidos nas equações correspondentes ao deslocamento, \mathbf{b} o vetor dos coeficientes correspondentes à pressão desconhecida nas equações correspondentes ao deslocamento, \mathbf{d} o vetor consistindo dos coeficientes associados aos deslocamentos desconhecidos na equação correspondente a pressão e c o coeficiente do ponto relacionado à pressão na respectiva equação da pressão. Dessa forma \mathbf{x}^u são as incógnitas relacionadas ao deslocamento e x^p é a incógnita da pressão. Finalmente \mathbf{f} e f são os correspondentes lados direitos.

Para a solução do sistema de equações descrito na Eq. (3.27) pode-se usar qualquer método direto.

Para o uso do suavizador Vanka aplicado às equações da poroelasticidade, por exemplo, Rodrigo (2010), Gaspar et al. (2004) e Gaspar e Rodrigo (2015) comentam que seu uso se dá pelo fato de existirem pontos de sela no problema da poroelasticidade. A iteração com o Vanka ocorre sobre todos os pontos da malha seguindo uma ordem (lexicográfica ou colorida) e em cada um dos pontos, o correspondente sistema de equações é resolvido (de forma direta). Para o caso unidimensional é resolvido um sistema de 3×3 equações simultaneamente em cada um dos pontos da malha, centrado em torno do ponto

da pressão, ou seja, são resolvidos de forma acoplada para cada ponto, a pressão p_i e os deslocamentos adjacentes u_{i-1} e u_{i+1} , ver a Fig. 5(a). Para o caso bidimensional é resolvido um sistema de 5×5 equações simultaneamente em cada ponto da discretização, também centrado em torno do ponto da pressão, ou seja, são resolvidos de forma acoplada para cada ponto, a pressão $p_{i,j}$ e os deslocamentos adjacentes $u_{i-1,j}$, $u_{i+1,j}$, $v_{i,j-1}$ e $v_{i,j+1}$, ver a Fig. 5(b). Dessa forma pode-se perceber que para cada iteração com o suavizador Vanka, a variável relacionada à pressão é atualizada uma vez, enquanto que as variáveis relacionadas aos deslocamentos são atualizadas duas vezes.

3.3.4 Ordenação das atualizações das incógnitas

A eficiência destes métodos iterativos está fortemente ligada à ordenação em que as incógnitas são determinadas ou atualizadas. As ordens mais comuns são a ordenação lexicográfica (Fig. 6), a ordenação *red-black* (Fig. 7) e algumas variantes, como a ordenação de varredura para os métodos de suavização por blocos gerando os métodos por linhas (Fig. 8), zebra (Fig. 9), entre outros (WESSELING, 1992; TROTTEBERG et al., 2001; WIENANDS; JOPPICH, 2005). Um dos objetivos das ordens de atualização coloridas (ou zebras) é possibilitar o desenvolvimento de algoritmos paralelizáveis.

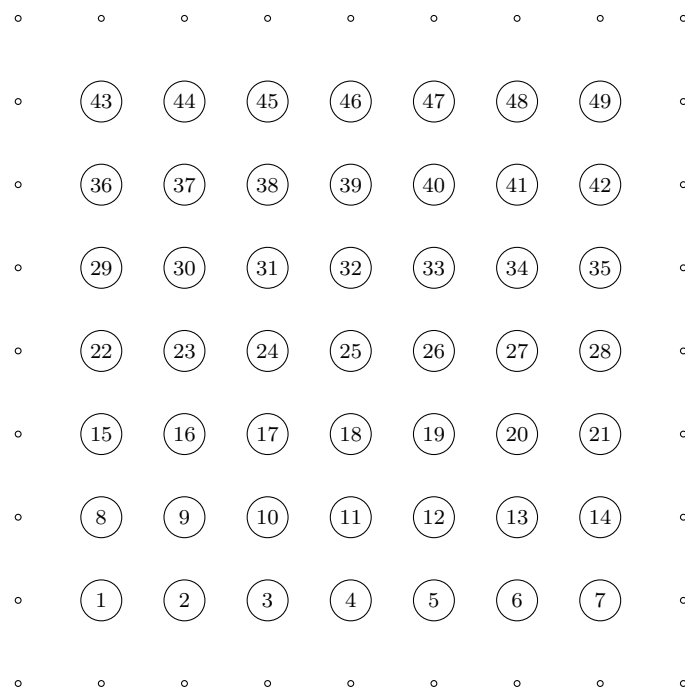


Figura 6 – Ordenação lexicográfica para atualização das incógnitas.

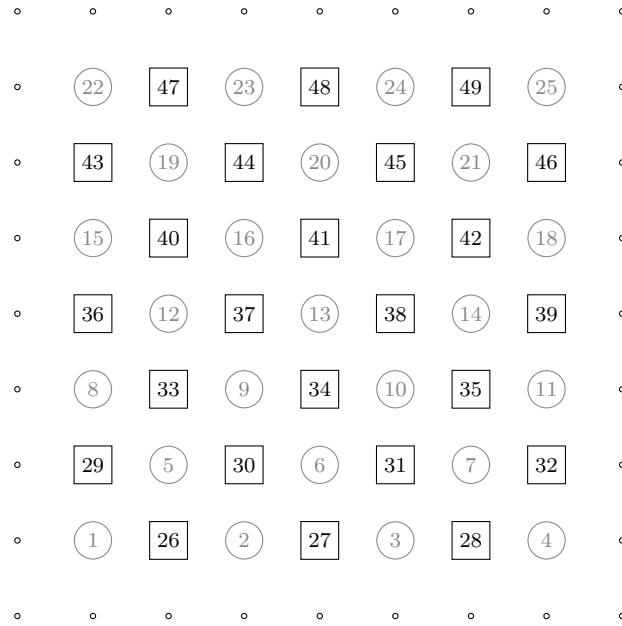


Figura 7 – Ordenação *red-black* para atualização das incógnitas: (○) *red* e (□) *black*.

3.4 Métodos para a aproximação temporal

Dado um Problema de Valor Inicial (PVI)

$$\begin{cases} \frac{du}{dt} = f \\ u(x, y, t_0) = u^0 \end{cases} \quad (3.28)$$

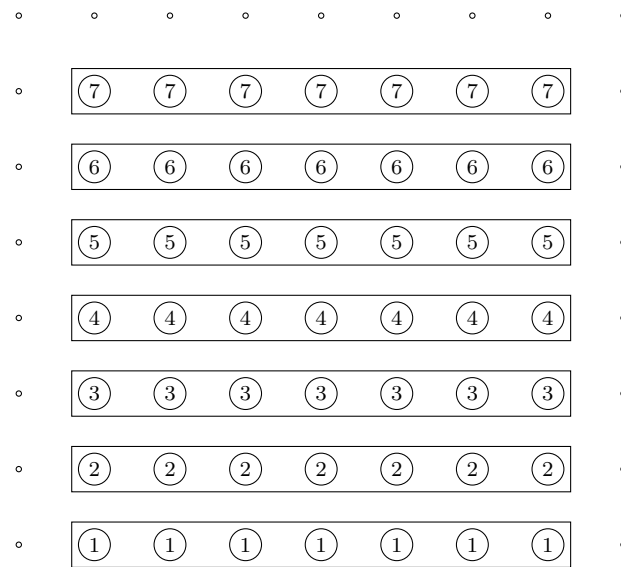
com u e f funções das variáveis (x, y, t) , em que $[x, y]$ está contido em um domínio espacial, $t \in [t_0, t_f]$, em que t_0 representa o tempo inicial e t_f o tempo final, u^0 é a condição inicial (função das variáveis espaciais x e y). Nesta tese, as condições para existência e unicidade da solução são garantidas, ver (BURDEN; FAIRES, 2016). Observa-se que a derivada temporal é ordinária, pois apesar de depender dos valores espaciais (x, y) , consideram-nos conhecidos durante o cálculo da variável temporal.

3.4.1 Formulações temporais de EDPs

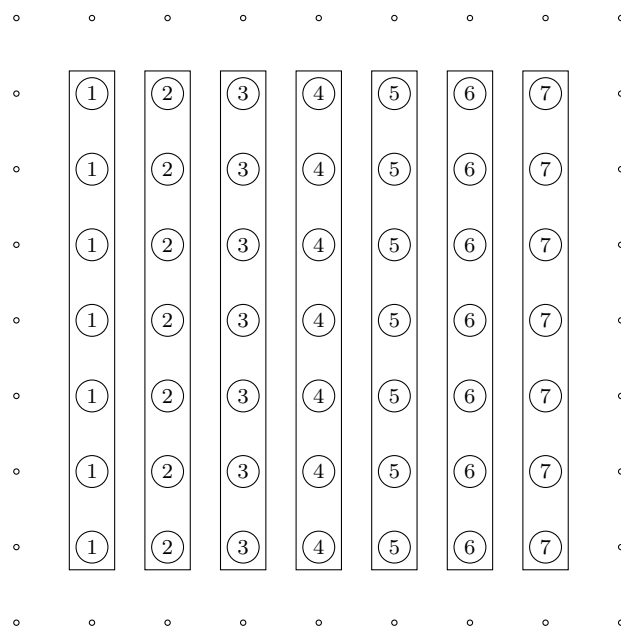
Maliska (2004) relata que uma formulação é dita explícita a um passo de tempo quando todas as incógnitas vizinhas ao ponto i são avaliadas nos passos de tempo anteriores e, portanto, já são conhecidas, como mostra a Fig. 10(a).

Uma formulação é dita totalmente Implícita a um passo de tempo quando todas as incógnitas vizinhas ao ponto i são avaliadas no passo de tempo atual e, portanto, não são conhecidas em sua totalidade (depende da ordenação de atualização das incógnitas), como mostra a Fig. 10(b).

A formulação implícita a um passo de tempo ocorre quando as incógnitas vizinhas ao ponto i são avaliadas nos passos de tempo atual e anterior, como mostra a Fig. 10(c).



(a) Linha horizontal.



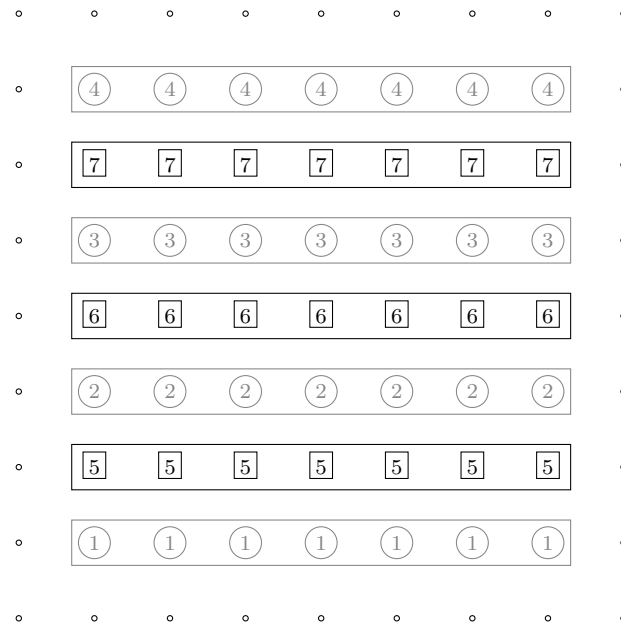
(b) Linha Vertical.

Figura 8 – Ordenação para atualização das incógnitas com um método linha.

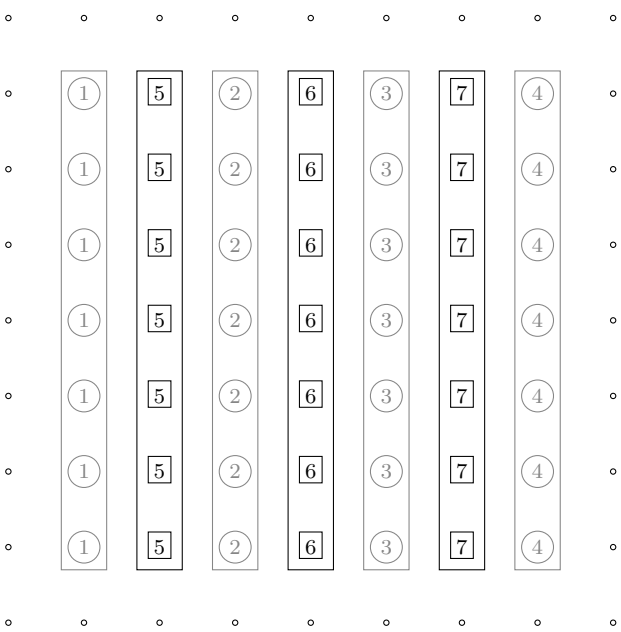
3.4.2 Método de *Euler*

O método de *Euler* consiste em obter uma aproximação para a solução $u(t)$ de (3.28) de forma discreta em diversos valores no intervalo $[t_0, t_f]$, chamados pontos de malha. Para isso, estipula-se que os pontos de malha têm distribuição uniforme no intervalo $[t_0, t_f]$, seleciona-se um número positivo inteiro N_t e os pontos de malha (em t) são dados por

$$t_{m+1} = t_0 + (m + 1)\tau \quad \text{para cada } m = 0, 1, \dots, N_t - 1. \quad (3.29)$$



(a) Zebra horizontal.



(b) Zebra Vertical.

Figura 9 – Ordenação para atualização das incógnitas com um método zebra.

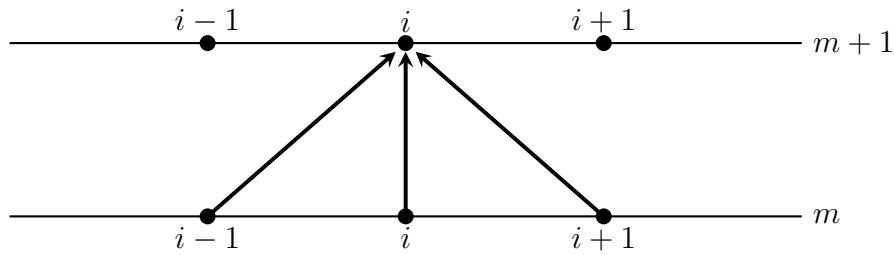
A distância entre os pontos $\tau = \frac{t_f - t_0}{N_t}$ é chamada de tamanho do passo de tempo.

Seja $u(t)$ a única solução da Eq. (3.28), com duas derivadas contínuas em $[t_0, t_f]$, de modo que sua expansão em série de Taylor para cada $m = 0, 1, \dots, N_t - 1$ é dada por

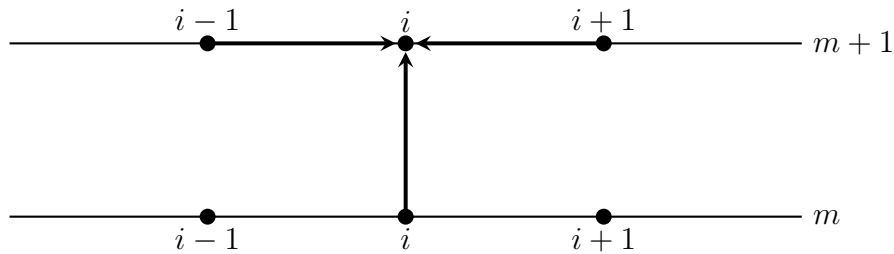
$$u(t_{m+1}) = u(t_m) + \tau \frac{d}{dt} u(t_m) + \frac{\tau^2}{2} \frac{d^2}{dt^2} u(\xi_k), \quad (3.30)$$

para algum $\xi_m \in (t_0, t_f)$.

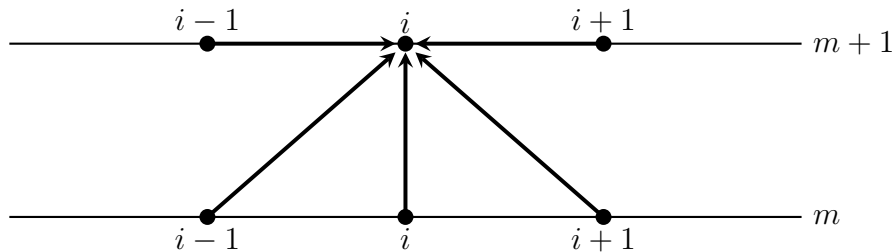
Considerando-se v uma aproximação para u , o método de Euler constrói $v^m \approx u(t_m)$



(a) Formulação Explícita.



(b) Formulação Totalmente Implícita.



(c) Formulação Implícita.

Figura 10 – Conexão espacial e temporal em um ponto i com formulações Explícita, Totalmente Implícita e Implícita (MALISKA, 2004, p. 39).

para cada $m = 0, 1, \dots, N_t - 1$ ao desconsiderar os termos $\frac{\tau^2}{2} \frac{d^2}{dt^2} u(\xi_k)$, de segunda ordem de aproximação em τ . Dessa forma, o método de *Euler Implícito* (primeira ordem de aproximação no tempo) é dado por

$$\begin{cases} v^0 = y^0 \\ v^{m+1} = v^m + \tau f(v^{m+1}, t_{m+1}) \end{cases} \quad \text{para cada } m = 0, 1, \dots, N_t - 1. \quad (3.31)$$

Esse método é chamado de método de *Euler Implícito*, pois para obter o valor de v^{m+1} são necessários valores relacionados aos passos de tempo m e $m + 1$. Se para obter

o valor de v^{m+1} utilizar apenas valores relacionados ao passo de tempo $m + 1$ o método de Euler é dito método de Euler Totalmente Implícito e se são necessários apenas valores relacionados ao passo de tempo m o método de Euler é dito método de Euler Explícito (BURDEN; FAIRES, 2016; HIRSCH, 2007).

O método Explícito é condicionalmente estável, ou seja, converge quando satisfaz um critério de convergência (BURDEN; FAIRES, 2016) que relaciona o tamanho da discretização espacial ao tamanho do passo temporal. Os métodos Implícitos e Totalmente Implícitos são incondicionalmente estáveis, ou seja, independem do tamanho da discretização espacial e do tamanho do passo temporal (BURDEN; FAIRES, 2016; STRIKWERDA, 1989).

3.4.3 Método de Crank-Nicolson

De acordo com Tannehill et al. (1997), Burden e Faires (2016), Strikwerda (1989) e Hirsch (2007), o método de Crank-Nicolson (CN) é um método incondicionalmente estável, ou seja, a sua convergência não depende da relação entre o tamanho do passo de tempo e o tamanho da discretização espacial e possui segunda ordem de aproximação temporal ($O(\tau^2)$).

Considerando por exemplo a Eq. (3.28), o método de CN consiste em fazer

$$\begin{cases} v^0 = y^0 \\ v^{m+1} = v^m + \frac{\tau}{2} [f(v^{m+1}, t_{m+1}) + f(v^m, t_m)] \end{cases} \quad m = 0, 1, \dots, N_t - 1, \quad (3.32)$$

em que os subíndices $m+1$ e m indicam os passos de tempo atual e anterior, respectivamente.

4 Método *multigrid*

Quando o problema proposto é discretizado e resolvido em apenas uma malha, diz-se que este problema está sendo resolvido com o método *singlegrid*. Se usar duas malhas para resolver o problema, diz-se que está sendo usado o método de duas malhas (ou *two-grid*). Porém, com o uso de três ou mais malhas para a solução do problema, diz-se que está sendo usado o método *multigrid*.

O método *multigrid* é uma técnica eficiente para acelerar a suavização das fortes oscilações do resíduo usando um método de relaxação (método iterativo básico). Proposto originalmente por Fedorenko (1964), esse método é atualmente muito utilizado para acelerar a convergência de sistemas de equações do tipo $Au = f$. O princípio de funcionamento desse método está fortemente relacionado ao comportamento de convergência dos métodos iterativos básicos (Gauss-Seidel e Jacobi, por exemplo). Esses métodos apresentam propriedades de suavização dos erros de alta frequência (componentes oscilatórias), enquanto as baixas frequências são mantidas praticamente inalteradas. Assim, nas primeiras iterações reduzem-se rapidamente as componentes oscilatórias do erro e à medida que esse número vai aumentando, o erro começa a cair mais lentamente, indicando a predominância das componentes suaves (BRANDT, 1977; STÜBEN; TROTTEBERG, 1981; WESSELING, 1992; BRIGGS et al., 2000; TROTTEBERG et al., 2001). Assim, o método *multigrid* consiste no uso de discretização do problema em diversas malhas, ou seja, após a suavização das componentes oscilatórias em uma malha fina, as componentes dos erros (suaves) são transferidas para uma malha mais grossa, tornando-se oscilatórias. Esse processo pode ser repetido até a malha mais grossa possível e faz com que o método iterativo usado não perca a eficiência, pois agirá sempre nas componentes oscilatória dos erros.

Como dito anteriormente, o método *multigrid* geométrico utiliza-se de informações das malhas consideradas na discretização do problema, o que significa que ele é recomendado para problemas em malhas estruturadas. Como neste trabalho serão usadas apenas malhas estruturadas, o método *multigrid* geométrico será a ferramenta de trabalho adotada.

4.1 Caracterização do método *multigrid*

A ideia principal do método *multigrid* consiste em usar uma discretização de malha adequada para que, usando um suavizador (Gauss-Seidel, Jacobi, ...), as componentes oscilatórias do erro sejam suavizadas rapidamente nas primeiras iterações. Após a suavização, o método *multigrid* procura trabalhar com uma sequência de malhas cada vez mais grossas, de tal forma que os comprimentos de ondas do erro, que são longos em

malhas finas, sejam transformados em curtos em malhas mais grossas, tornando-se mais oscilatórios, situação na qual as componentes do erro são suavizadas mais rapidamente.

A razão de engrossamento entre as malhas q é dada por

$$q = \frac{h_2}{h_1}, \quad (4.1)$$

em que h_1 e h_2 são o tamanhos dos elementos da malha consecutivamente mais grossa e fina, respectivamente. A Fig. 11 mostra uma sequência de três malhas bidimensionais (Ω^h , Ω^{2h} e Ω^{4h}) com razão de engrossamento padrão $q = 2$.

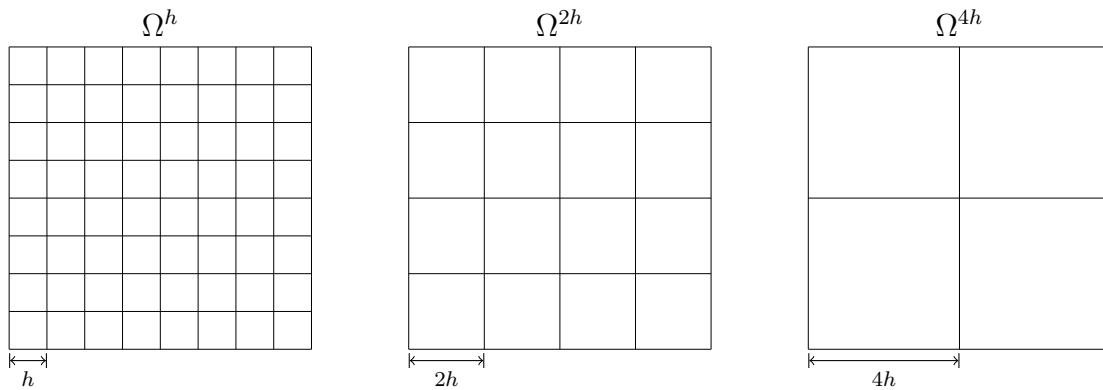


Figura 11 – Sequência de 3 malhas bidimensionais e uniformes com engrossamento padrão.

4.1.1 Análise de convergência

Seja o sistema linear na forma

$$Au = f. \quad (4.2)$$

Supondo que este sistema tenha uma única solução e que v é uma aproximação para u , pode-se definir duas medidas importantes de v como aproximação de u .

Definição 4.1. O erro (ou erro algébrico) é dado por:

$$e = u - v. \quad (4.3)$$

Porém, o erro de uma solução numérica é inacessível quando não se conhece a solução analítica. Assim, uma medida calculável para verificar o quanto v se aproxima de u é o resíduo.

Definição 4.2. O resíduo da Eq. (4.2) para uma aproximação v é dado por

$$r = f - Av. \quad (4.4)$$

Como a solução aproximada v é obtida através de um processo iterativo, pode-se escrever o erro e o resíduo em função da iterada atual v^ν , respectivamente, como

$$e^\nu = u - v^\nu, \quad (4.5)$$

$$r^\nu = f - Av^\nu, \quad (4.6)$$

sendo que a magnitude de ambos pode ser medida pelas normas vetoriais conhecidas (BURDEN; FAIRES, 2016).

Uma importante relação entre erro e resíduo é mostrado em Briggs et al. (2000), através da expressão

$$\frac{1}{\text{cond}(A)} \frac{\|r\|}{\|f\|} \leq \frac{\|e\|}{\|u\|} \leq \text{cond}(A) \frac{\|r\|}{\|f\|}, \quad (4.7)$$

com $\text{cond}(A) = \|A\| \cdot \|A^{-1}\|$ sendo o número de condicionamento da matriz A .

Observe que, pelas inequações (4.7), $\|r\| \approx 0$ implica em $\|e\| \approx 0$ somente se $\text{cond}(A)$ é pequeno, ou seja, próximo da unidade ($\text{cond}(A) \approx 1$). Neste caso, diz-se que a matriz A é bem condicionada.

A partir das Eqs. (3.25) e (3.26) pode-se construir o método iterativo dado por

$$v^{\nu+1} = Sv^\nu + g, \quad (4.8)$$

em que v é uma aproximação para a solução desejada. Nota-se que S dependerá do método iterativo utilizado.

Definição 4.3. O método iterativo $v^{\nu+1} = Sv^\nu + g$ é chamado de convergente se e somente se $\lim_{\nu \rightarrow \infty} \|S\|^\nu = 0$.

Definição 4.4. O fator de convergência do método iterativo dado na Eq. (4.8) é

$$\rho(S) = \max |\lambda(S)|.$$

Na definição 4.4, $\lambda(S)$ representa os autovalores da matriz S e o fator de convergência ρ (máximo autovalor, em módulo, de S) é também chamado de raio espectral da matriz S . Este ρ indica a pior redução do erro com o passo iterativo (BRIGGS et al., 2000).

Teorema 4.1. O método iterativo $v^{\nu+1} = Sv^\nu + g$ é convergente se e somente se $\rho(S) \leq 1$.

Prova: ver Burden e Faires (2016).

O raio espectral ou fator de convergência pode ser aproximado numericamente como descrito em Janssen (1997, p. 43) por

$$\rho^{\nu+1} = \frac{\|e^{\nu+1}\|}{\|e^\nu\|}. \quad (4.9)$$

O mesmo autor comenta que se usar o resíduo ao invés do erro na Eq. (4.9), os resultados serão semelhantes.

O fator de convergência médio pode ser definido como a média geométrica dos fatores de convergência de (4.9). Assim,

$$\rho_m = \sqrt[n]{\rho^{(1)} \cdot \rho^{(2)} \cdot \dots \cdot \rho^{(n)}} , \quad (4.10)$$

em que n representa o número de iteradas ou ciclos *multigrid* realizados.

4.1.2 Análise de erros

Para uma análise do comportamento dos erros, Briggs et al. (2000) afirmam que é suficiente utilizar o sistema homogêneo $Au = 0$. Neste caso, tem-se a vantagem da solução exata ($u = 0$) ser conhecida e o erro da aproximação dada por v é simplesmente $-v$.

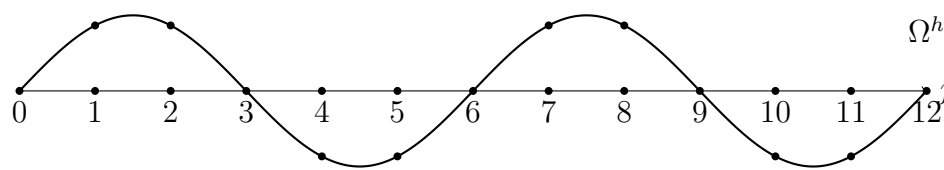
Para facilitar os cálculos, considera-se aqui o caso unidimensional da equação de Poisson com condições de contorno de Dirichlet com estimativa inicial, ou modos de Fourier, dada por $v_j = \text{sen}\left(\frac{jk\pi}{N}\right)$, $1 \leq j \leq N - 1$ e $k \leq N - 1$, em que v é uma aproximação da solução u , N é o número de pontos da discretização, k é o número de em que ou modos de Fourier e j é a componente do vetor v .

Definição 4.5. Os modos de Fourier localizados na metade inferior do espectro, com $1 \leq k < \frac{N}{2}$, são chamados de modos de Fourier de baixa frequência ou modos suaves. Os modos de Fourier localizados na metade superior do espectro, com $\frac{N}{2} \leq k \leq N - 1$, são chamados de modos de Fourier de alta frequência ou modos oscilatórios.

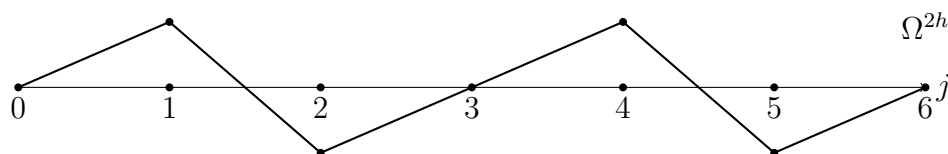
A Fig. 12(a) e a Fig. 12(b) mostram os efeitos do engrossamento de malhas para suavizar os modos de Fourier. Observa-se que o número de ondas k permanece inalterado, enquanto que o número de pontos é reduzido, fazendo com que a onda torne-se mais oscilatória ($\frac{N}{2} \leq k \leq N - 1$).

Ao se projetar um modo suave com número de ondas $1 \leq k < \frac{N}{2}$ da malha fina Ω^h para a malha grossa Ω^{2h} , transforma-se o modo em oscilatório (BRIGGS et al., 2000; TROTTEBERG et al., 2001; WESSELING, 1992). Por isso é recomendável transferir o problema de relaxação para a malha grossa após a suavização dos erros oscilatórios, pois lá os modos de erros suaves se apresentarão mais oscilatórios, e o processo de relaxação será mais eficiente.

Com essa filosofia para suavizar os erros de baixa frequência com o uso de malhas mais grossas, o método *multigrid* possui uma taxa (ou fator) de convergência ideal (teórica) que independe do tamanho da malha, isto é, independe do número de pontos da discretização da malha (FERZIGER; PERIC, 2002; ROACHE, 1998). Para obter um bom desempenho do método *multigrid*, devem-se usar diversos níveis de malhas (TANNEHILL et al., 1997; PINTO; MARCHI, 2007).



(a) $k = 4$ ondas sobre a malha com $N = 12$ pontos.



(b) $k = 4$ ondas sobre a malha com $N = 6$ pontos.

Figura 12 – Número de ondas $k = 4$ sobre uma malha fina com $N = 12$ pontos e sobre uma malha grossa com $N = 6$ pontos (BRIGGS et al., 2000, p. 32).

4.2 Operadores de transferência entre malhas

Os elementos essenciais do método *multigrid* são os operadores de restrição (I_h^{2h}) e de prolongação (I_{2h}^h) . Com o problema representado em uma malha fina (Ω^h) é necessário transferir as informações para as malhas mais grossas ($\Omega^{2h}, \Omega^{4h}, \dots$). Isto é feito por um operador chamado *operador de restrição*. Para fazer o caminho contrário e transferir as informações das malhas grossas para as malhas mais finas, faz-se necessário um operador chamado *operador de prolongação*.

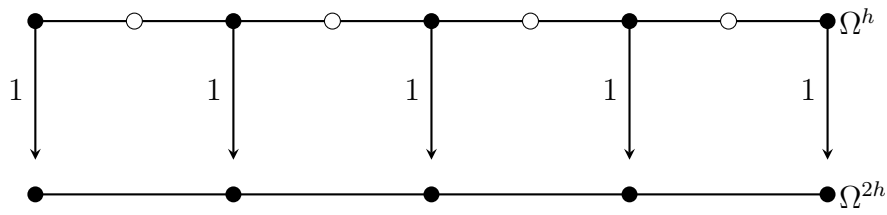
Nas figuras deste capítulo, o símbolo (\bullet) representa os nós que pertencem a ambas as malhas, o símbolo (\circ) representa os nós que pertencem apenas à malha fina e os números (ao lado das setas) representam os pesos atribuídos a cada tipo de restrição ou prolongação.

4.2.1 Operadores de restrição

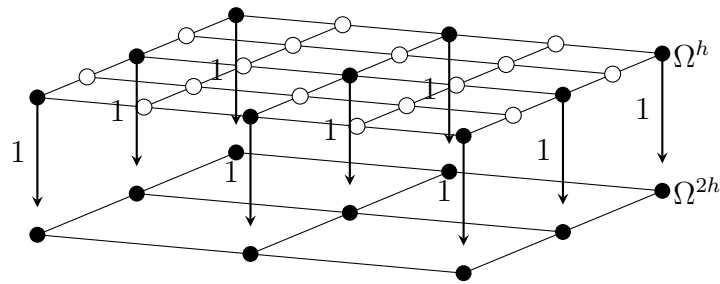
O operador de restrição (I_h^{2h}) é responsável por transferir informações (resíduo e/ou solução) de uma determinada malha Ω^h , com discretização de malhas de tamanho h (em todas as direções espaciais), para a malha mais grossa Ω^{2h} com $2h$. A restrição pode ser por Injeção (Inj), por meia ponderação (HW), por ponderação completa (FW), entre outros (BRIGGS et al., 2000; TROTTEBERG et al., 2001; HORTON; VANDEWALLE, 1995).

O operador de restrição por injeção $(I_h^{2h})_{Inj}$ apenas transfere a informação de um ponto na malha fina em seu correspondente ponto na malha mais grossa. A Fig. 13(a) e a Fig. 13(b) ilustram este procedimento para os casos unidimensional (1D) e bidimensional (2D), respectivamente.

Por exemplo, no caso 2D, o resíduo calculado pelo operador $(I_h^{2h})_{Inj}$ na malha



(a) Problema 1D.



(b) Problema 2D.

Figura 13 – Operadores de restrição por injeção 1D e 2D.

grossa, é dado por

$$\begin{aligned} r_{2h}(x, y) &= \left(I_h^{2h} \right)_{Inj} r_h(x, y) \\ &= r_h(x, y). \end{aligned} \tag{4.11}$$

Em notação estêncil, tem-se

$$\left(I_h^{2h} \right)_{Inj} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}_h^{2h}. \tag{4.12}$$

O operador de restrição por meia ponderação $\left(I_h^{2h} \right)_{HW}$, no caso 2D, utiliza as informações dos 4 pontos mais próximos ao ponto calculado na malha fina para o correspondente ponto na malha mais grossa. A Fig. 14 ilustra esse procedimento.

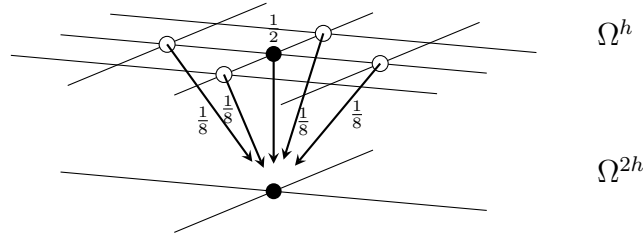


Figura 14 – Operador de restrição por meia ponderação.

O resíduo para o caso 2D, por exemplo, calculado pelo operador $(I_h^{2h})_{HW}$ na malha grossa é dado por

$$\begin{aligned} r_{2h}(x, y) &= (I_h^{2h})_{HW} r_h(x, y) \\ &= \frac{1}{8} [4r_h(x, y) + r_h(x, y - h) + r_h(x - h, y) + r_h(x + h, y) + r_h(x, y + h)]. \end{aligned} \quad (4.13)$$

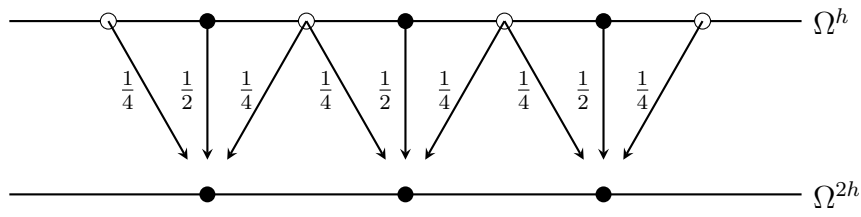
Em notação estêncil, tem-se

$$(I_h^{2h})_{HW} = \frac{1}{8} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}_h^{2h}. \quad (4.14)$$

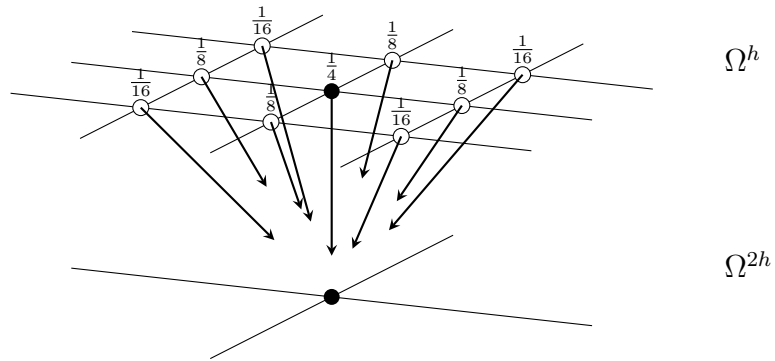
O operador de restrição por ponderação completa $(I_h^{2h})_{FW}$ utiliza informações de todos os pontos vizinhos ao ponto calculado na malha fina para o correspondente ponto na malha grossa. A Fig. 15(a) e a Fig. 15(b) ilustram este procedimento para os casos unidimensional (1D) e bidimensional (2D), respectivamente.

Por exemplo, o resíduo para o caso 2D calculado pelo operador $(I_h^{2h})_{FW}$ na malha grossa é dado por

$$\begin{aligned} r_{2h}(x, y) &= (I_h^{2h})_{FW} r_h(x, y) \\ &= \frac{1}{16} \{4r_h(x, y) + 2[r_h(x, y - h) + r_h(x - h, y) + r_h(x + h, y) + r_h(x, y + h)] + \\ &\quad + r_h(x - h, y - h) + r_h(x + h, y - h) + r_h(x - h, y + h) + r_h(x + h, y + h)\}. \end{aligned} \quad (4.15)$$



(a) Problema 1D.



(b) Problema 2D.

Figura 15 – Operadores de restrição por ponderação completa para os casos 1D e 2D.

Em notação estêncil, tem-se

$$\left(I_h^{2h}\right)_{FW} = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h^{2h}. \quad (4.16)$$

4.2.2 Operadores de prolongação

O operador de prolongação (ou interpolação) I_{2h}^h é responsável por transferir informação (correção) de uma determinada malha grossa Ω^{2h} , com discretização de malhas de tamanho $2h$, para a malha fina Ω^h com malha de tamanho h . A prolongação pode ser por interpolação linear (para o caso 1D), bilinear (para o caso 2D), entre outras. Para esses operadores serão empregados Hackbusch (1985), Wesseling (1992), Briggs et al. (2000) e Trottenberg et al. (2001).

A interpolação linear da correção pelo operador I_{2h}^h da malha grossa para a malha fina, no caso 1D (Fig. 16), é dado por

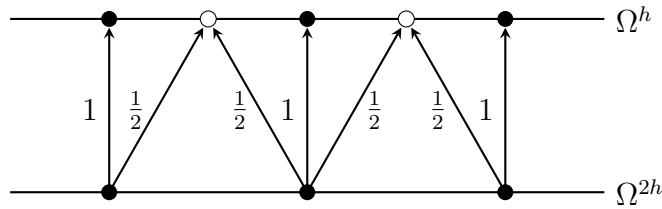


Figura 16 – Operador de interpolação linear para o caso 1D.

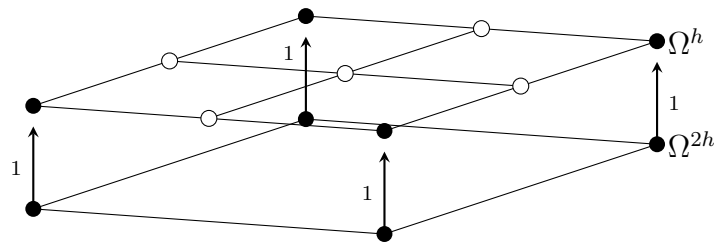
$$\begin{aligned}
 v_h(x) &= I_{2h}^h v_{2h}(x) \\
 &= \begin{cases} \frac{1}{2}[v_{2h}(x-h) + v_{2h}(x+h)] & \text{para } \circ \\ v_{2h}(x) & \text{para } \bullet \end{cases} .
 \end{aligned} \tag{4.17}$$

Em notação estêncil, tem-se

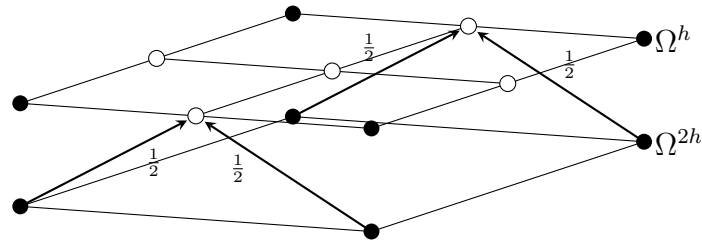
$$I_{2h}^h = \left[\begin{array}{ccc} 1 & 2 & 1 \\ \hline & & \end{array} \right]_{2h}^h . \tag{4.18}$$

Para o caso 2D, usa-se um operador de interpolação bilinear I_{2h}^h (Fig. 17). Nesse caso, a interpolação bilinear da correção pelo operador I_{2h}^h da malha Ω^{2h} para a malha Ω^h (Fig. 17(a) a 17(d)) é dado por

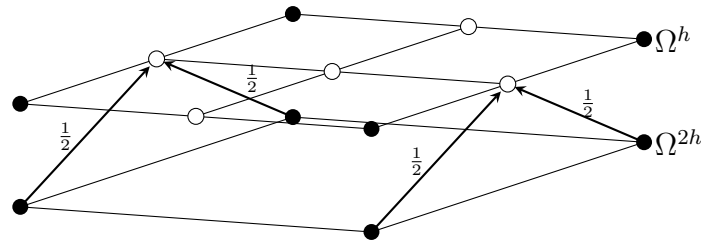
$$\begin{aligned}
 v_h(x, y) &= I_{2h}^h v_{2h}(x, y), \\
 v_h(x, y) &= \begin{cases} v_{2h}(x, y) \\ \frac{1}{2}[v_{2h}(x-h, y) + v_{2h}(x+h, y)] \\ \frac{1}{2}[v_{2h}(x, y-h) + v_{2h}(x, y+h)] \\ \frac{1}{4}[v_{2h}(x-h, y) + v_{2h}(x+h, y) + v_{2h}(x, y-h) + v_{2h}(x, y+h)] \end{cases}
 \end{aligned} \tag{4.19}$$



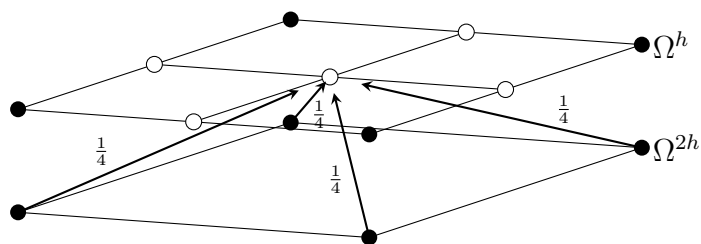
(a) Injeção direta.



(b) Interpolação linear na direção x .



(c) Interpolação linear na direção y .



(d) Interpolação bilinear.

Figura 17 – Operador de Interpolação bilinear para o caso 2D.

Em notação estêncil, tem-se

$$I_h^{2h} = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \begin{matrix} \left[\right. \\ \\ \left. \right]_{2h} \end{matrix} \quad (4.20)$$

4.3 Esquema de correção e ciclos

Neste trabalho o método *multigrid* foi implementado utilizando o esquema de correção (em inglês, *Correction Scheme*, CS). De acordo com Briggs et al. (2000), o esquema CS é mais indicado para problemas lineares.

A filosofia do esquema CS pode ser representada pelo seguinte procedimento:

- Suavize $Au = f$ na malha mais fina Ω^h para obter uma aproximação v^h .
- Calcule o resíduo $r = f - Av^h$.

Suavize a equação residual $Ae = r$ em Ω^{2h} com estimativa inicial $e = 0$ para obter uma aproximação para o erro e^{2h} .

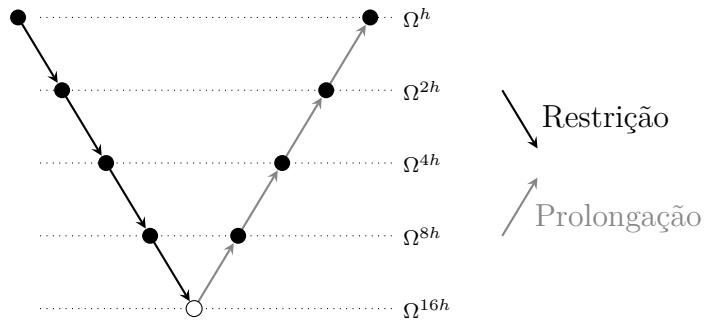
- Corrija a aproximação obtida em Ω^h com o erro estimado em Ω^{2h} : $v^h \leftarrow v^h + e^{2h}$.

Esse procedimento expõe um esquema CS para o caso de duas malhas, porém a mesma ideia pode ser estendida para o número de malhas que se queira.

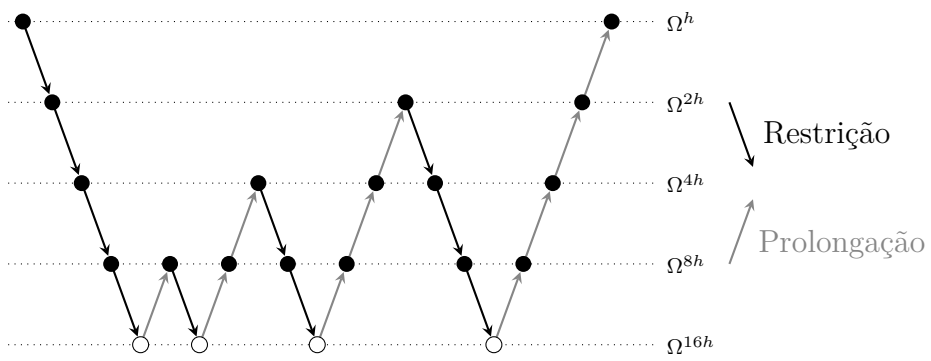
A sequência com que as malhas são percorridas dá origem aos chamados ciclos de suavização. Wesseling (1992), Briggs et al. (2000) e Trottenberg et al. (2001) apresentaram a generalização dos ciclos V , F e W como parte de uma família ciclos chamada de μ -ciclo. As Fig. 18(a), Fig. 18(b) e Fig. 18(c) ilustram cada um desses ciclos V , F e W , respectivamente.

O número de suavizações, ou iterações do suavizador (ν), geralmente depende do suavizador e não precisa ser necessariamente o mesmo em todos os níveis de malha, tanto no processo de restrição quanto na prolongação. Define-se como ν_1 o número de suavizações realizadas quando ocorre o processo de restrição (pré-suavização) e ν_2 o número de suavizações realizadas quando ocorre o processo de prolongação (pós-suavização). Briggs et al. (2000) afirmam que para problemas que não apresentam dificuldades de convergência, são empregadas entre 1 e 3 suavizações.

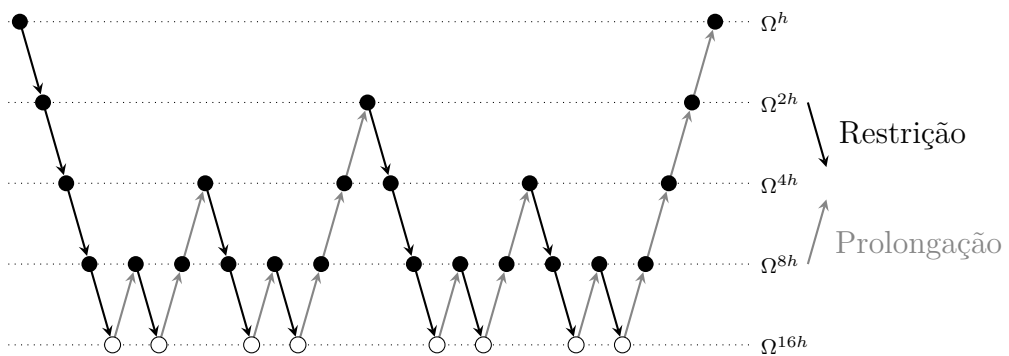
O algoritmo 1 mostra um esquema para o ciclo V ou o ciclo W . Se $\mu = 1$ o algoritmo 1 realiza um ciclo V e se $\mu = 2$ realiza um ciclo W . O algoritmo 2 mostra um esquema para o ciclo F .



(a) Ciclo V .



(b) Ciclo F .



(c) Ciclo W .

Figura 18 – Estrutura dos ciclos V , F e W . O símbolo (\bullet) representa suavização e o símbolo (\circ) a solução exata.

Algoritmo 1: MG- μ -ciclo (l)

```

if  $l = L_{max}$  é o nível de malha mais grossa then
  Resolva o sistema  $A_l v^{(l)} = f^{(l)}$  na malha grossa  $\Omega^{2^{l-1}h}$ .
else
  Suavize  $\nu_1$  vezes  $A_l v^{(l)} = f^{(l)}$  na malha  $\Omega^{2^{l-1}h}$ .
  Calcule e restrinja o resíduo:  $f^{(l+1)} = I_{2^{l-1}h}^{2^l h} (f^{(l)} - A_l v^{(l)})$ .
  for  $ciclo = 1 : \mu$  do
    Resolva no próximo nível: MG- $\mu$ -ciclo ( $l + 1$ ).
  end for
  Corrija usando interpolação:  $v^{(l)} \leftarrow v^{(l)} + I_{2^{l-1}h}^{2^l h} v^{(l+1)}$ .
  Suavize  $\nu_2$  vezes  $A_l v^{(l)} = f^{(l)}$  na malha  $\Omega^{2^{l-1}h}$ .
end if

```

Algoritmo 2: MG-Ciclos-F (l)

```

if  $l = L_{max}$  é o nível de malha mais grossa then
  Resolva o sistema  $A_l v^{(l)} = f^{(l)}$  na malha grossa  $\Omega^{2^{l-1}h}$ .
else
  Suavize  $\nu_1$  vezes  $A_l v^{(l)} = f^{(l)}$  na malha  $\Omega^{2^{l-1}h}$ .
  Calcule e restrinja o resíduo:  $f^{(l+1)} = I_{2^{l-1}h}^{2^l h} (f^{(l)} - A_l v^{(l)})$ .
  Resolva no próximo nível: MG-Ciclos-F ( $l + 1$ ).
  Corrija usando interpolação:  $v^{(l)} \leftarrow v^{(l)} + I_{2^{l-1}h}^{2^l h} v^{(l+1)}$ .
  Suavize  $\nu_2$  vezes  $A_l v^{(l)} = f^{(l)}$  na malha  $\Omega^{2^{l-1}h}$ .
  if  $l \neq 1$  then
    Resolva no próximo nível: MG- $\mu$ -ciclo ( $l$ ) usando o algoritmo 1 e  $\mu = 1$ .
  end if
end if

```

5 Análise de Fourier local

A análise de Fourier local (em inglês, *Local Fourier Analysis*, LFA) é a mais poderosa ferramenta para análise quantitativa e para a construção de projetos com métodos *multigrid* eficientes (TROTTEMBERG et al., 2001). Essa análise permite prever o desempenho do método *multigrid* através de estimativas das taxas de convergência dos métodos iterativos e do processo *multigrid*.

Segundo Wienands e Joppich (2005), o principal objetivo da LFA é estimar o raio espectral ou certas normas do operador *multigrid*, que são medidas quantitativas para verificar a redução do erro (ou resíduo). Para um cálculo exato destas quantidades, é necessário que exista uma base unitária de autofunções periódicas do operador discreto do problema diferencial L_h que geram todo o espaço de funções de malha e que sejam compatíveis com as condições de contorno. Em seguida, é possível expandir o erro (ou resíduo) após a l -ésima malha de um ciclo em uma série de Fourier por uma transformação de base unitária.

Neste capítulo serão expostas a terminologia usada em LFA, a análise em duas malhas para os fatores de suavização e de convergência usando uma ordenação lexicográfica e também para o caso de se usar outra ordenação, como por exemplo a ordenação *red-black*.

5.1 Terminologia

Para este trabalho, seguiram-se Stüben e Trottenberg (1981), Trottenberg et al. (2001) e Wienands e Joppich (2005). Esses autores afirmam que qualquer operador discreto pode ser linearizado e pode ser substituído localmente por um operador com coeficientes constantes. Formalmente, esses operadores serão definidos em uma malha infinita, em que as fronteiras não serão levadas em consideração.

Define-se nesta seção, para o caso bidimensional, $x = (x_1, x_2)$ como sendo os pontos da discretização espacial e $h = (h_1, h_2)$ como as respectivas distância entre os pontos. A malha mais fina é dada por

$$\Omega^h = \{(x_j)_h = (x_1, x_2)_h = jx = (j_1 h_1, j_2 h_2)\}, \quad j = (j_1, j_2) \in \mathbb{Z}^2, \quad (5.1)$$

enquanto que a malha imediatamente mais grossa é dada por

$$\Omega^{2h} = \{(x_j)_{2h} = (x_{2j})_h = (2x_1, 2x_2)_h\}, \quad j = (j_1, j_2) \in \mathbb{Z}^2. \quad (5.2)$$

O operador discreto L_h na forma de estêncil, como descrito na seção 3.2, é dado por

$$L_h \hat{=} [s_\kappa]_h \quad (\kappa = (\kappa_1, \kappa_2) \in \mathbb{Z}^2), \quad (5.3)$$

isto é,

$$L_h \omega_h(x) = \sum_{\kappa} s_{\kappa} \omega(x + \kappa h), \quad (5.4)$$

em que κ representa os índices do estêncil e as constantes $s_{\kappa} \in \mathbb{R}$ (ou \mathbb{C}) são os valores de L_h referentes às posições κ na malha Ω^h .

Todas as considerações no contexto de LFA são baseadas em funções de malha da forma

$$\varphi_h(\theta, x) = e^{i\theta \cdot \frac{x}{h}} = e^{i\theta_1 \frac{x_1}{h_1}} e^{i\theta_2 \frac{x_2}{h_2}}, \quad (5.5)$$

em que $x \in \Omega^h$ e $\theta = (\theta_1, \theta_2)$ é um parâmetro contínuo que caracteriza a frequência da função de malha. Além disso,

$$\varphi_h(\theta, x) = \varphi_h(\theta', x) \quad x \in \Omega^h, \quad (5.6)$$

se, e somente se,

$$\theta_1 \equiv \theta'_1 \pmod{2\pi} \quad \text{e} \quad \theta_2 \equiv \theta'_2 \pmod{2\pi}. \quad (5.7)$$

Portanto, é suficiente considerar $\varphi(\theta, \cdot)$, $\theta \in [-\pi, \pi) \times [-\pi, \pi) = [-\pi, \pi)^2$, ou ainda, $-\pi \leq \theta < \pi$.

As funções de malha $\varphi(\theta, \cdot)$ para $-\pi \leq \theta < \pi$ são linearmente independentes em Ω^h . Para melhorar a notação, às vezes escreve-se $\varphi_h(\theta, \cdot)$ para definir a dependência da malha Ω^h .

Lema 5.1. Para $-\pi \leq \theta < \pi$, todas as funções de malha $\varphi(\theta, \cdot)$ são autofunções de algum operador discreto L_h na forma estêncil como nas Eqs. (5.3) e (5.4). Assim

$$L_h \varphi(\theta, x) = \tilde{L}_h(\theta) \varphi(\theta, x) \quad \forall x \in \Omega^h,$$

com os autovalores $\tilde{L}_h(\theta)$ dados por

$$\tilde{L}_h(\theta) = \sum_{\kappa} s_{\kappa} e^{i\theta \cdot \kappa}. \quad (5.8)$$

Prova: ver Trottenberg et al. (2001).

Para suavização e análise em duas malhas, deve-se distinguir as componentes de altas e baixas frequências de Ω^h com relação a Ω^{2h} . Para pontos na malha Ω^{2h} tem-se

$$\varphi_h(\theta, x_{2j}) = \varphi_{2h}(2\theta, x_j), \quad (5.9)$$

ou seja, as funções com número de onda θ relativo a Ω^h têm número de onda 2θ relativo a Ω^{2h} . Assim, somente as funções

$$\varphi_h(\theta, \cdot) \quad \text{com} \quad -\frac{\pi}{2} \leq \theta < \frac{\pi}{2},$$

são distinguíveis (visíveis) em Ω^{2h} .

Para cada $\theta' \in [-\frac{\pi}{2}, \frac{\pi}{2})^2$, três outras componentes de frequência $\varphi_h(\theta, \cdot)$ com $\theta \in [-\pi, \pi)^2$ coincidem em Ω^{2h} com $\varphi_{2h}(\theta', \cdot)$ e não são visíveis em Ω^{2h} .

Definição 5.1. Altas e baixas frequências para engrossamento padrão.

- φ é uma componente de baixa frequência $\iff \theta \in T^{\text{baixa}} = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]^2$;
- φ é uma componente de alta frequência $\iff \theta \in T^{\text{alta}} = [-\pi, \pi]^2 \setminus \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]^2$.

Essas componentes podem ser visualizadas na Fig. 19, retirada de Trottenberg et al. (2001, p. 102). Nesta, a região em branco representa as baixas frequências, com as componentes representadas por (o), e a região hachurada representa as altas frequências, com as componentes representadas por (•).

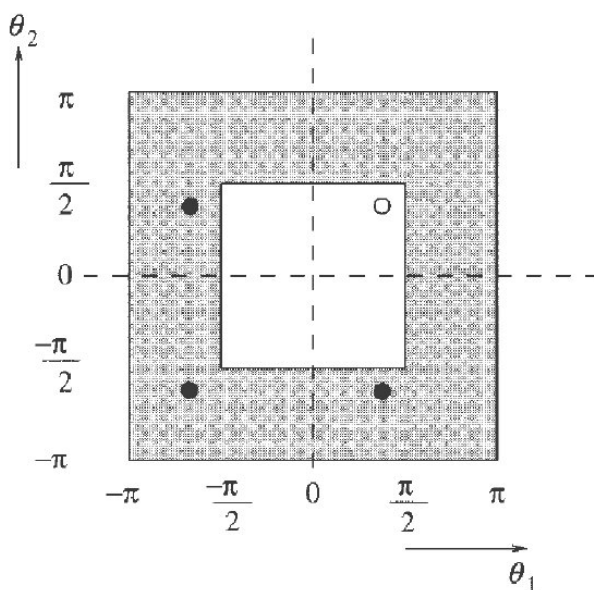


Figura 19 – Regiões de componentes de altas e baixas frequências (TROTTEBERG et al., 2001, p. 102).

5.2 Análise do suavizador

Considerando-se uma Equação Diferencial Parcial (EDP) discretizada, da forma

$$L_h u_h = f_h, \tag{5.10}$$

e supondo-se w_h como uma aproximação de u_h , além da separação

$$L_h = L_h^+ + L_h^-, \tag{5.11}$$

um método de relaxação pode ser escrito localmente como

$$L_h^+ \bar{w}_h + L_h^- w_h = f_h, \tag{5.12}$$

em que \bar{w}_h é a aproximação anterior de u_h e w_h corresponde à nova aproximação de u_h (TROTTEBERG et al., 2001; WIENANDS; JOPPICH, 2005).

Subtraindo a Eq. (5.12) da Eq. (5.10), obtém-se

$$L_h^+ \bar{v}_h + L_h^- v_h = 0, \quad (5.13)$$

ou

$$\bar{v}_h = S_h v_h, \quad (5.14)$$

para os erros $\bar{v}_h = u_h - \bar{w}_h$, $v_h = u_h - w_h$. Neste caso, S_h é o operador de suavização resultante.

Aplicando L_h^+ e L_h^- às autofunções $\varphi(\theta, x)$, obtém-se

$$L_h^- e^{i\theta \cdot x/h} = \tilde{L}_h^-(\theta) e^{i\theta \cdot x/h}, \quad (5.15)$$

$$L_h^+ e^{i\theta \cdot x/h} = \tilde{L}_h^+(\theta) e^{i\theta \cdot x/h}, \quad (5.16)$$

em que \tilde{L}_h^- e \tilde{L}_h^+ representam os autovalores dos operadores L_h^- e L_h^+ , respectivamente.

Lema 5.2. Sob as hipóteses das Eqs. (5.11) e (5.12), todas as funções $\varphi(\theta, \cdot)$ com $\tilde{L}_h^+(\theta) \neq 0$ são autofunções de S_h :

$$S_h \varphi(\theta, x) = \tilde{S}_h(\theta) \varphi(\theta, x) \quad (-\pi \leq \theta < \pi), \quad (5.17)$$

com o fator de amplificação dado por

$$\tilde{S}_h(\theta) = -\frac{\tilde{L}_h^-(\theta)}{\tilde{L}_h^+(\theta)}. \quad (5.18)$$

Prova: ver Trottenberg et al. (2001).

Definição 5.2. O fator de suavização é definido como:

$$\mu_{loc} = \mu_{loc}(S_h) = \sup \{ |\tilde{S}_h(\theta)| : \theta \in T^{\text{alta}} \}. \quad (5.19)$$

O algoritmo 3 descreve o procedimento adotado para encontrar o fator de suavização de um método iterativo.

5.3 Análise em duas malhas

Para aplicar LFA a um operador em duas malhas, faz-se

$$M_h^{2h} = S_h^{\nu_2} K_h^{2h} S_h^{\nu_1}, \quad (5.20)$$

em que $K_h^{2h} = I_h - I_{2h}^h [L_{2h}]^{-1} I_h^{2h} L_h$ é o operador de correção na malha grossa, I_h é o estêncil do operador identidade, ν_1 e ν_2 representam os números de pré e pós suavizações,

Algoritmo 3: LFA para análise do fator de suavização

Leia os dados de entrada: h e o suavizador a ser usado (se necessário, escolha ω).

for Cada um dos modos oscilatórios **do**

 Calcule a matriz diagonal $\hat{S}_h(4 \times 4)$ com os autovalores de S_h .

 Calcule os autovalores da matriz \hat{S}_h e escolha seu valor máximo em módulo.

 Guarde o autovalor máximo em cada modo de Fourier oscilatório.

end for

Escolha o máximo dos autovalores, μ (este representa o pior fator de suavização).

respectivamente. Para calcular o fator de convergência e outras quantidades de M_h^{2h} , analisa-se como os operadores L_h , I_h^{2h} , L_{2h} , I_{2h}^h e S_h agem nas componentes $\varphi(\theta, \cdot)$.

Para alguma baixa frequência $\theta = (\theta_1, \theta_2) \in T^{\text{baixa}} = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]^2$, consideram-se as frequências

$$\theta^{(0,0)} = (\theta_1, \theta_2), \quad (5.21)$$

$$\theta^{(1,1)} = (\bar{\theta}_1, \bar{\theta}_2), \quad (5.22)$$

$$\theta^{(1,0)} = (\bar{\theta}_1, \theta_2), \quad (5.23)$$

$$\theta^{(0,1)} = (\theta_1, \bar{\theta}_2), \quad (5.24)$$

em que

$$\bar{\theta}_i = \begin{cases} \theta_i + \pi & \text{se } \theta_i < 0 \\ \theta_i - \pi & \text{se } \theta_i \geq 0 \end{cases}. \quad (5.25)$$

Lema 5.3.

1. Para alguma baixa frequência $\theta^{(0,0)} \in T^{\text{baixa}}$, tem-se

$$\varphi(\theta^{(0,0)}, x) \equiv \varphi(\theta^{(1,1)}, x) \equiv \varphi(\theta^{(1,0)}, x) \equiv \varphi(\theta^{(0,1)}, x), \quad x \in \Omega^{2h}. \quad (5.26)$$

2. Cada uma dessas quatro componentes de Fourier $\varphi(\theta^\alpha, \cdot) = \varphi_h(\theta^\alpha, \cdot)$, com $\alpha \in \{(0,0), (1,1), (1,0), (0,1)\}$, coincide em Ω^{2h} com a respectiva função de malha $\varphi_{2h}(2\theta^{(0,0)}, \cdot)$:

$$\varphi_h(\theta^\alpha, x) \equiv \varphi_{2h}(2\theta^{(0,0)}, x), \quad x \in \Omega^{2h}. \quad (5.27)$$

Prova: ver Trottenberg et al. (2001).

Definição 5.3. O conjunto de funções de malha de Ω^{2h} obtido através de funções de malha em Ω^h é chamado de harmônico para engrossamento padrão em Ω^h . O espaço gerado por esse harmônico quadridimensional é dado por

$$\mathcal{F}_h(\theta) = \text{span}[\varphi(\theta^{(0,0)}, \cdot), \varphi(\theta^{(1,1)}, \cdot), \varphi(\theta^{(1,0)}, \cdot), \varphi(\theta^{(0,1)}, \cdot)], \quad (5.28)$$

em que *span* representa o espaço gerado (STEINBRUCH; WINTERLE, 1997).

Observa-se que o espaço $\mathcal{F}_h(\theta)$ é invariante sob o operador de duas malhas $M_h^{2h} = S_h^{\nu_2} K_h^{2h} S_h^{\nu_1}$ com hipóteses gerais.

Considerando-se $\Psi \in \mathcal{F}_h(\theta)$, pode se escrever

$$\Psi = A^{(0,0)}\varphi(\theta^{(0,0)}, \cdot) + A^{(1,1)}\varphi(\theta^{(1,1)}, \cdot) + A^{(1,0)}\varphi(\theta^{(1,0)}, \cdot) + A^{(0,1)}\varphi(\theta^{(0,1)}, \cdot), \quad (5.29)$$

com coeficientes A^α unicamente definidos.

Com as hipóteses que L_h, I_h^{2h}, L_{2h} e I_h^h são representados por estêncis em Ω^h e Ω^{2h} , e assumindo-se que na formação de K_h^{2h} e M_h^{2h} , dados na Eq. (5.20), existe $(L_{2h})^{-1}$, pode-se enunciar os teoremas a seguir

Teorema 5.1. O operador de correção na malha grossa K_h^{2h} é representado no espaço dos harmônicos $\mathcal{F}_h(\theta)$ por uma matriz $\hat{K}_h^{2h}(\theta)$ de ordem 4×4 , como

$$\hat{K}_h^{2h}(\theta) = \hat{I}_h - \hat{I}_{2h}^h(\theta)[\hat{L}_{2h}(2\theta)]^{-1}\hat{I}_h^{2h}(\theta)\hat{L}_h(\theta), \quad (5.30)$$

para cada $\theta \in T^{\text{baixa}}$.

Prova: ver Trottenberg et al. (2001) e Wienands e Joppich (2005).

Em outras palavras, ao se aplicar K_h^{2h} em qualquer $\Psi \in \mathcal{F}_h(\theta)$, os coeficientes A^α , dados na Eq. (5.29), são transformados da seguinte maneira:

$$\begin{pmatrix} A^{(0,0)} \\ A^{(1,1)} \\ A^{(1,0)} \\ A^{(0,1)} \end{pmatrix} \Leftarrow \hat{K}_h^{2h}(\theta) \begin{pmatrix} A^{(0,0)} \\ A^{(1,1)} \\ A^{(1,0)} \\ A^{(0,1)} \end{pmatrix}. \quad (5.31)$$

Teorema 5.2. Se os espaços $\mathcal{F}_h(\theta)$ são invariantes sob o operador de suavização S_h , isto é,

$$S_h : \mathcal{F}_h(\theta) \longrightarrow \mathcal{F}_h(\theta), \quad \forall \theta \in T^{\text{baixa}}, \quad (5.32)$$

pode-se obter representação para M_h^{2h} no espaço $\mathcal{F}_h(\theta)$ por uma matriz de ordem 4×4 da forma $\hat{M}_h^{2h}(\theta)$ no espaço $\mathcal{F}_h(\theta)$ dada por

$$\hat{M}_h^{2h}(\theta) = \hat{S}_h(\theta)^{\nu_2} \hat{K}_h^{2h}(\theta) \hat{S}_h(\theta)^{\nu_1}, \quad (5.33)$$

sendo $\hat{K}_h^{2h}(\theta)$ como na Eq. (5.30) e a matriz $\hat{S}_h(\theta)$ representando a matriz do estêncil do suavizador S_h .

Prova: ver Trottenberg et al. (2001) e Wienands e Joppich (2005).

Com isso, $M_h^{2h}\Psi$ pode ser escrita como:

$$M_h^{2h}\Psi = B^{(0,0)}\varphi(\theta^{(0,0)}, \cdot) + B^{(1,1)}\varphi(\theta^{(1,1)}, \cdot) + B^{(1,0)}\varphi(\theta^{(1,0)}, \cdot) + B^{(0,1)}\varphi(\theta^{(0,1)}, \cdot), \quad (5.34)$$

em que

$$\begin{pmatrix} B^{(0,0)} \\ B^{(1,1)} \\ B^{(1,0)} \\ B^{(0,1)} \end{pmatrix} = \hat{M}_h^{2h}(\theta) \begin{pmatrix} A^{(0,0)} \\ A^{(1,1)} \\ A^{(1,0)} \\ A^{(0,1)} \end{pmatrix}. \quad (5.35)$$

Para garantir a existência de M_h^{2h} , deve-se excluir todos os θ cujos autovalores $\tilde{L}_h(\theta) = \tilde{L}_{2h}(\theta) = 0$, ou seja, exclui-se o conjunto

$$\Lambda = \left\{ \theta \in T^{\text{baixa}} = [-\pi, \pi) \times [-\pi, \pi) \mid \tilde{L}_h(\theta) = 0 \text{ ou } \tilde{L}_{2h}(\theta) = 0 \right\}. \quad (5.36)$$

Os autovalores (que também são chamados de símbolos) dos operadores de restrição I_h^{2h} , por injeção (Inj), meia ponderação (HW) e ponderação completa (FW), usados na Eq. (5.30) são mostrados na Tab. 1.

Tabela 1 – Autovalores dos operadores de restrição usados na LFA.

Restrição	$\tilde{I}_h^{2h}(\theta^\alpha)$
Injeção	1
Meia Ponderação	$\frac{1}{4}(2 + \cos \bar{\theta}_1 + \cos \bar{\theta}_2)$
Ponderação Completa	$\frac{1}{4}(1 + \cos \bar{\theta}_1)(1 + \cos \bar{\theta}_2)$

O operador de suavização na malha grossa L_{2h} é dado pelo estêncil

$$L_{2h} = [S_{\kappa, 2h}]_{2h}. \quad (5.37)$$

Para $\theta = \theta^{(0,0)}$ e $\theta \in T^{\text{baixa}}$, tem-se

$$L_{2h}\varphi_{2h}(2\theta, \cdot) = \tilde{L}_{2h}(2\theta)\varphi_{2h}(2\theta, \cdot), \quad (5.38)$$

com o autovalor (ou símbolo)

$$\tilde{L}_{2h}(2\theta) = \sum_{\kappa \in V} S_{\kappa, 2h} e^{i2\theta \cdot \kappa}. \quad (5.39)$$

A solução na malha Ω^{2h} (para $2\theta \notin \Lambda$) para os respectivos coeficientes é dada por

$$A_{2h} \Leftarrow [\tilde{L}_{2h}(2\theta)]^{-1} A_{2h}. \quad (5.40)$$

Os operadores de prolongação por interpolação bilinear I_{2h}^h , são descritos como

$$\tilde{I}_{2h}^h(\theta^\alpha) = \frac{1}{4}(1 + \cos \bar{\theta}_1)(1 + \cos \bar{\theta}_2). \quad (5.41)$$

Com os operadores já definidos, é possível calcular o fator de convergência ρ associado à matriz M , que representa o processo de suavização em duas malhas.

Definição 5.4. Considerando-se que M_h^{2h} é representado pela matriz 4×4 \hat{M}_h^{2h} , o fator de convergência assintótico pode ser calculado por

$$\rho_{\text{loc}}(M_h^{2h}) = \sup \left\{ \rho_{\text{loc}}(\hat{M}_h^{2h}(\theta)) \mid \theta \in T^{\text{baixa}}, \theta \notin \Lambda \right\}, \quad (5.42)$$

em que $\rho_{\text{loc}}(\hat{M}_h^{2h}(\theta))$ é o raio espectral da matriz \hat{M}_h^{2h} .

O algoritmo 4 descreve o procedimento adotado para encontrar o fator de convergência de um operador de duas malhas de um método *multigrid* com esquema CS e engrossamento padrão.

Algoritmo 4: LFA para o fator de convergência em duas malhas

Leia os dados de entrada: h, ν_1, ν_2 .

Escolha o suavizador, os operadores de restrição e prolongação.

for Cada um dos modos suaves, exceto $(0, 0)$ **do**

Calcule θ^α usando Eqs. (5.21) – (5.25).

Crie a matriz diagonal $\hat{L}_h(4 \times 4)$ com os autovalores de L_h .

Calcule a matriz $\hat{I}_h^{2h}(1 \times 4)$ do operador de restrição I_h^{2h} .

Calcule $\hat{L}_{2h}(1 \times 1)$ na malha Ω^{2h} .

Calcule a matriz $\hat{I}_{2h}^h(4 \times 1)$ do operador de prolongação I_{2h}^h .

Calcule a matriz $\hat{K}_h^{2h}(\theta) = \hat{I}_h - \hat{I}_{2h}^h(\theta)[\hat{L}_{2h}(2\theta)]^{-1}\hat{I}_h^{2h}(\theta)\hat{L}_h(\theta)$.

Calcule a matriz diagonal $\hat{S}_h(4 \times 4)$ com os autovalores de S_h .

Calcule os autovalores da matriz \hat{M}_h^{2h} e escolha seu valor máximo em módulo.

Guarde o autovalor máximo em cada modo de Fourier suave.

end for

Escolha o máximo dos autovalores, ρ (este representa o pior fator de convergência).

5.4 Análise de Fourier local para outros suavizadores

Uma análise similar é feita se o operador S_h não atender as hipóteses do Lema 5.2. Neste caso, assume-se um espaço E com a propriedade de invariância, tal que

$$S_h : E_h^\theta \rightarrow E_h^\theta, \quad \forall \theta \in T^{\text{baixa}} \quad (5.43)$$

seja válida para o operador de suavização S_h . Esta hipótese atende uma grande classe de suavizadores, incluindo Gauss-Seidel *red-black*.

Se o operador de suavização tem a propriedade de invariância dada na Eq. (5.43), as altas e baixas frequências podem ser mescladas por S_h . Para medir a propriedade de suavização de S_h é necessário assumir um operador na malha grossa "ideal" Q_h^{2h} (ao invés de K_h^{2h}) que aniquila as componentes de baixa frequência e mantém as componentes de alta frequência inalteradas. Mais precisamente, Q_h^{2h} é um operador de projeção, definido em E_h^θ por

$$Q_h^{2h}\varphi(\theta, \cdot) = \begin{cases} 0 & \text{se } \theta = \theta^{(0,0)} \in T^{\text{baixa}} \\ \varphi(\theta, \cdot) & \text{se } \theta \in \{\theta^{(1,1)}, \theta^{(1,0)}, \theta^{(0,1)}\} \equiv T^{\text{alta}} \end{cases} \quad (5.44)$$

Como consequência, a matriz K_h^{2h} dada na Eq. (5.30) é substituída por

$$\hat{Q}_h^{2h}(\theta) = \hat{Q}_h^{2h} = \begin{pmatrix} 0 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{pmatrix}, \quad \theta \in T^{\text{baixa}} \quad (5.45)$$

para a análise de suavização. Além disso, pode-se substituir M_h^{2h} dado na Eq. (5.33) por

$$M_h^{2h} = S_h^{\nu_1} Q_h^{2h} S_h^{\nu_2} = Q_h^{2h} S_h^\nu, \quad (5.46)$$

em que $\nu = \nu_1 + \nu_2$, e $\rho_{\text{loc}}(M_h^{2h})$ dado na Eq. (5.42) por

$$\rho_{\text{loc}}(M_h^{2h}) = \sup \left\{ \rho(\hat{Q}_h^{2h} \hat{S}_h^\nu) : \theta \in T^{\text{baixa}} \right\}. \quad (5.47)$$

Definição 5.5. Com a hipótese que S_h tem a propriedade de invariância dada na Eq. (5.43), o fator de suavização $\mu_{\text{loc}}(S_h, \nu)$ de S_h é dado por

$$\mu_{\text{loc}}(S_h, \nu) = \sup \left\{ \sqrt[\nu]{\rho_{\text{loc}}(\hat{Q}_h^{2h} \hat{S}_h^\nu)}, \theta \in T^{\text{baixa}} \right\}, \quad (5.48)$$

que generaliza a definição de μ_{loc} dada na definição 5.2 (Eq. (5.19)).

A fim de generalizar as hipóteses do procedimento de suavização, dado nas seções 5.2 e 5.3 para os suavizadores do tipo *red-lack* ou outros suavizadores que seguem esse "padrão" de atualização das incógnitas, como por exemplo, o Gauss-Seidel zebra (TROTTEBERG et al., 2001), é necessário um correspondente operador S_h^{Parcial} , que é dado por

$$S_h^{\text{Parcial}} v_h(x) = \begin{cases} -(L_h^+)^{-1} L_h^- v_h(x) & \text{para } x \in \tilde{\Omega}_h \\ v_h(x) & \text{para } x \in \Omega^h \setminus \tilde{\Omega}_h \end{cases}, \quad (5.49)$$

assumindo-se que $(L_h^+)^{-1}$ existe e $\tilde{\Omega}_h$ é um subconjunto de Ω^h que depende das características do operador de suavização. Obviamente, somente os pontos de $\tilde{\Omega}_h$ são processados no passo de suavização acima. O restante dos pontos $\Omega^h \setminus \tilde{\Omega}_h$ (que pertencem ao conjunto Ω^h e não pertencem ao conjunto $\tilde{\Omega}_h$) não são suavizados. Estes serão atualizados em outras parciais do processo de suavização.

Por exemplo, para os operadores de suavização denotados por S_h^{Red} e S_h^{Black} , ambos têm a forma da Eq. (5.49) com $\tilde{\Omega}_h$ começando a leitura dos pontos de Ω^h de *red* e *black*, respectivamente. O operador de suavização (completo) GS-RB é o produto

$$S_h^{\text{RB}} = S_h^{\text{Black}} S_h^{\text{Red}}. \quad (5.50)$$

Considerando-se a propriedade de invariância dada na Eq. (5.43) em $E_h^\theta \left(-\frac{\pi}{2} \leq \theta < \frac{\pi}{2}\right)$, as matrizes \hat{S}_h^{Red} e \hat{S}_h^{Black} para S_h^{RB} são dadas por

$$\hat{S}_h^{Red}(\theta, \omega) = \frac{1}{2} \begin{pmatrix} A^{(0,0)} + 1 & A^{(1,1)} - 1 & 0 & 0 \\ A^{(0,0)} - 1 & A^{(1,1)} + 1 & 0 & 0 \\ 0 & 0 & A^{(1,0)} + 1 & A^{(0,1)} - 1 \\ 0 & 0 & A^{(1,0)} - 1 & A^{(0,1)} + 1 \end{pmatrix} \quad (5.51)$$

e

$$\hat{S}_h^{Black}(\theta, \omega) = \frac{1}{2} \begin{pmatrix} A^{(0,0)} + 1 & -A^{(1,1)} + 1 & 0 & 0 \\ -A^{(0,0)} + 1 & A^{(1,1)} + 1 & 0 & 0 \\ 0 & 0 & A^{(1,0)} + 1 & -A^{(0,1)} + 1 \\ 0 & 0 & -A^{(1,0)} + 1 & A^{(0,1)} + 1 \end{pmatrix}, \quad (5.52)$$

em que

$$A^\alpha = 1 - \frac{\omega h^2}{4} \tilde{L}_h(\theta^\alpha). \quad (5.53)$$

Substituindo-se a matriz \hat{S}_h na Eq. (5.33) por S_h^{RB} dado na Eq. (5.50) é possível calcular o fator de convergência assintótico usando a análise em duas malhas com o suavizador GS-RB, por exemplo. Para o suavizador zebra, ver Wienands e Joppich (2005).

6 Modelos matemáticos e numéricos

Neste capítulo apresentam-se os modelos matemáticos e numéricos para os problemas da difusão de calor (equação de Fourier) e da poroelasticidade.

Para os modelos numéricos, as aproximações espaciais são feitas através do Método das Diferenças Finitas (MDF) e a aproximação temporal é feita com os métodos de Euler ou Crank-Nicolson (CN).

6.1 Equação do calor

6.1.1 Modelos matemáticos

O modelo matemático para o problema da difusão de calor dependente do tempo (regime transiente) é representado pela equação do calor (EDP parabólica)

$$\frac{\partial u}{\partial t} - \Delta u = f, \quad (6.1)$$

no domínio espacial dado por $\Omega \subset \mathbb{R}^d$, com $d \in \{1, 2\}$. Nesta notação $x_1 = x$ e $x_2 = y$, o operador Laplaciano d -dimensional é dado por $\Delta = \sum_{i=1}^d \frac{\partial^2}{\partial x_i^2}$ e o intervalo de tempo (t) considerado é $(0, t_f]$.

Supõe-se a condição inicial dada por

$$u(\mathbf{x}, 0) = \prod_{i=1}^d \sin(\pi x_i), \quad \mathbf{x} \in \Omega, \quad (6.2)$$

com $\mathbf{x} = x_1$ e $\Omega = [0, 1]$ se $d = 1$ ou $\mathbf{x} = (x_1, x_2)$ e $\Omega = [0, 1] \times [0, 1]$ se $d = 2$. As condições de contornos são do tipo *Dirichlet*, ou seja,

$$u(\mathbf{x}, t) = 0, \quad \mathbf{x} \in \partial\Omega, \quad 0 < t \leq t_f. \quad (6.3)$$

O termo fonte é dado por

$$f(\mathbf{x}, t) = (d\pi^2 - 1)e^{-t} \prod_{i=1}^d \sin(\pi x_i), \quad \mathbf{x} \in \Omega, \quad 0 < t \leq t_f, \quad (6.4)$$

e a solução analítica é

$$u(\mathbf{x}, t) = e^{-t} \prod_{i=1}^d \sin(\pi x_i), \quad \mathbf{x} \in \Omega, \quad 0 \leq t \leq t_f. \quad (6.5)$$

6.1.2 Modelos numéricos

Para o modelo numérico, considera-se o modelo matemático apresentado na seção 6.1.1 e v como sendo uma aproximação da solução analítica u . Discretiza-se o domínio espacial através do MDF com aproximação espacial do tipo CDS, como descrito na seção 3.1. Faz-se a aproximação temporal e a conexão espacial e temporal usando um método de discretização implícita (seção 3.4), como por exemplo neste caso, os métodos de Euler Implícito e de Crank-Nicolson.

Para o caso do método de Euler, considerando-se $d = 1$ e discretizando a Eq. (6.1) tem-se

$$\frac{v_i^{m+1} - v_i^m}{\tau} = \frac{v_{i-1}^{m+1} - 2v_i^{m+1} + v_{i+1}^{m+1}}{h^2} + f_i^{m+1}, \quad (6.6)$$

ou

$$v_i^{m+1} = v_i^m + \frac{\tau}{h^2} [v_{i-1}^{m+1} - 2v_i^{m+1} + v_{i+1}^{m+1}] + \tau f_i^{m+1}, \quad (6.7)$$

em que i , $i - 1$ e $i + 1$ indicam a discretização espacial, $m + 1$ e m indicam os passos de tempo atual e anterior, respectivamente. O tamanho do passo de tempo é dado por τ e o comprimento da discretização espacial é dado por h .

Observa-se que os coeficientes da Eq. (6.7) dependem de um parâmetro que leva em consideração o tamanho do passo do tempo (τ) e o quadrado do comprimento da discretização espacial (h). Esse parâmetro é chamado de *fator de anisotropia* (HORTON; VANDEWALLE, 1995) e é dado por

$$\lambda = \frac{\tau}{h^2}. \quad (6.8)$$

Substituindo λ e reorganizando os termos semelhantes na Eq. (6.7), tem-se com o método de Euler

$$(1 + 2\lambda) v_i^{m+1} = v_i^m + \lambda (v_{i-1}^{m+1} + v_{i+1}^{m+1}) + \tau f_i^{m+1}. \quad (6.9)$$

Para o Método de Crank-Nicolson (CN), a Eq. (6.1) é discretizada da seguinte forma

$$\frac{v_i^{m+1} - v_i^m}{\tau} = \frac{1}{2} \left(\frac{v_{i-1}^{m+1} - 2v_i^{m+1} + v_{i+1}^{m+1}}{h^2} + f_i^{m+1} \right) + \frac{1}{2} \left(\frac{v_{i-1}^m - 2v_i^m + v_{i+1}^m}{h^2} + f_i^m \right), \quad (6.10)$$

reorganizando os termos, tem-se com o método de CN

$$(1 + \lambda) v_i^{m+1} = \frac{\lambda}{2} (v_{i-1}^{m+1} + v_{i+1}^{m+1} + v_{i-1}^m + v_{i+1}^m) + (1 - \lambda) v_i^m + \frac{\tau}{2} (f_i^{m+1} + f_i^m). \quad (6.11)$$

Para associar os métodos de Euler Implícito e o método de Crank-Nicolson usou-se um esquema chamado de esquema θ . Esse esquema associa o parâmetro de mistura (θ)

aos esquemas de Euler Implícito e de Crank-Nicolson conforme indicado na Eq. (6.13) e é definido na forma

$$\frac{v_i^{m+1} - v_i^m}{\tau} = \theta \left(\frac{v_{i-1}^{m+1} - 2v_i^{m+1} + v_{i+1}^{m+1}}{h^2} + f_i^{m+1} \right) + (1 - \theta) \left(\frac{v_{i-1}^m - 2v_i^m + v_{i+1}^m}{h^2} + f_i^m \right), \quad (6.12)$$

em que $0 \leq \theta \leq 1$ e

$$\theta = \begin{cases} \frac{1}{2}, & \text{método de Crank-Nicolson} \\ 1, & \text{método de Euler} \end{cases}. \quad (6.13)$$

Assim sendo, a Eq. (6.12) pode ser reescrita como

$$(1 + 2\lambda\theta) v_i^{m+1} = \lambda\theta (v_{i-1}^{m+1} + v_{i+1}^{m+1}) + \tau\theta f_i^{m+1} + \lambda(1-\theta) (v_{i-1}^m - 2v_i^m + v_{i+1}^m) + v_i^m + \tau(1-\theta) f_i^m. \quad (6.14)$$

O operador discreto para a equação do calor 1D usando o esquema θ pode ser escrito em notação estêncil da seguinte forma

$$\begin{bmatrix} 0 & 0 & 0 \\ -\lambda\theta & 1 + 2\lambda\theta & -\lambda\theta \\ -\lambda(1-\theta) & 2\lambda(1-\theta) - 1 & -\lambda(1-\theta) \end{bmatrix}. \quad (6.15)$$

Nota-se que com $\theta = 1$ esse estêncil representa o método de Euler implícito e se $\theta = \frac{1}{2}$ representa o método de Crank-Nicolson. Como esse é um estêncil espaço/tempo e o problema é unidimensional, em cada linha representa-se a evolução no tempo (de baixo para cima) e em cada coluna, a discretização no espaço (da esquerda para a direita), por isso e pelo fato de não usar passos de tempos futuros, a primeira linha é nula.

Considerando-se $d = 2$ e discretizando a Eq. (6.1) considerando o comprimento das discretizações espaciais nas direções x e y iguais a $h_x = h_y = h$, tem-se

$$\frac{v_{i,j}^{m+1} - v_{i,j}^m}{\tau} = \frac{v_{i-1,j}^{m+1} - 2v_{i,j}^{m+1} + v_{i+1,j}^{m+1}}{h^2} + \frac{v_{i,j-1}^{m+1} - 2v_{i,j}^{m+1} + v_{i,j+1}^{m+1}}{h^2} + f_{i,j}^{m+1}, \quad (6.16)$$

considerando o fator de anisotropia λ dado na Eq. (6.8) e reorganizando os termos, tem-se

$$(1 + 4\lambda)v_{i,j}^{m+1} = v_{i,j}^m + \lambda [v_{i,j-1}^{m+1} + v_{i-1,j}^{m+1} + v_{i+1,j}^{m+1} + v_{i,j+1}^{m+1}] + \tau f_{i,j}^{m+1}. \quad (6.17)$$

Usando o método de Crank-Nicolson, a Eq. (6.1) para o caso bidimensional é discretizada da seguinte forma

$$\begin{aligned} \frac{v_{i,j}^{m+1} - v_{i,j}^m}{\tau} &= \frac{1}{2} \left[\frac{v_{i-1,j}^{m+1} - 2v_{i,j}^{m+1} + v_{i+1,j}^{m+1}}{h^2} + \frac{v_{i,j-1}^{m+1} - 2v_{i,j}^{m+1} + v_{i,j+1}^{m+1}}{h^2} + f_{i,j}^{m+1} \right] + \\ &\frac{1}{2} \left[\frac{v_{i-1,j}^m - 2v_{i,j}^m + v_{i+1,j}^m}{h^2} + \frac{v_{i,j-1}^m - 2v_{i,j}^m + v_{i,j+1}^m}{h^2} + f_{i,j}^m \right]. \end{aligned} \quad (6.18)$$

Reorganizando os termos, fica

$$\begin{aligned}
(1 + 2\lambda) v_{i,j}^{m+1} &= \frac{\lambda}{2} [v_{i,j-1}^{m+1} + v_{i-1,j}^{m+1} + v_{i+1,j}^{m+1} + v_{i,j+1}^{m+1}] + \\
&+ \frac{\lambda}{2} [v_{i,j-1}^m + v_{i-1,j}^m + v_{i+1,j}^m + v_{i,j+1}^m] + \\
&+ [1 - 2\lambda] v_{i,j}^m + \frac{\tau}{2} [f_{i,j}^{m+1} + f_{i,j}^m].
\end{aligned} \tag{6.19}$$

Para associar os métodos de Euler Implícito e o método de Crank-Nicolson usando o esquema θ , usam-se os valores de θ descritos na Eq. (6.13) e faz-se na Eq. (6.18) a seguinte alteração

$$\begin{aligned}
\frac{v_{i,j}^{m+1} - v_{i,j}^m}{\tau} &= \theta \left[\frac{v_{i-1,j}^{m+1} - 2v_{i,j}^{m+1} + v_{i+1,j}^{m+1}}{h^2} + \frac{v_{i,j-1}^{m+1} - 2v_{i,j}^{m+1} + v_{i,j+1}^{m+1}}{h^2} + f_{i,j}^{m+1} \right] + \\
&+ (1 - \theta) \left[\frac{v_{i-1,j}^m - 2v_{i,j}^m + v_{i+1,j}^m}{h^2} + \frac{v_{i,j-1}^m - 2v_{i,j}^m + v_{i,j+1}^m}{h^2} + f_{i,j}^m \right].
\end{aligned} \tag{6.20}$$

Reorganizando os termos, tem-se

$$\begin{aligned}
(1 + 4\lambda\theta) v_{i,j}^{m+1} &= \lambda\theta [v_{i,j-1}^{m+1} + v_{i-1,j}^{m+1} + v_{i+1,j}^{m+1} + v_{i,j+1}^{m+1}] + \\
&+ \lambda(1 - \theta) [v_{i,j-1}^m + v_{i-1,j}^m + v_{i+1,j}^m + v_{i,j+1}^m] + \\
&+ [1 - 4\lambda(1 - \theta)] v_{i,j}^m + \tau [\theta f_{i,j}^{m+1} + (1 - \theta) f_{i,j}^m].
\end{aligned} \tag{6.21}$$

O operador discreto para a equação do calor 2D usando o esquema θ pode ser escrito em notação estêncil da seguinte forma

$$\left[\begin{bmatrix} 0 & -\lambda(1 - \theta) & 0 \\ -\lambda(1 - \theta) & 4\lambda(1 - \theta) - 1 & -\lambda(1 - \theta) \\ 0 & -\lambda(1 - \theta) & 0 \end{bmatrix} \begin{bmatrix} 0 & -\lambda\theta & 0 \\ -\lambda\theta & 1 + 4\lambda\theta & -\lambda\theta \\ 0 & -\lambda\theta & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right]. \tag{6.22}$$

Nota-se que com $\theta = 1$ esse estêncil representa o método de Euler implícito e se $\theta = \frac{1}{2}$ representa o método de Crank-Nicolson. Neste caso, cada um dos estêncis representa as posições espaciais para um passo de tempo, por isso o estêncil da direita é nulo. Assim, em cada estêncil os termos da esquerda para a direita representam a direção x e de baixo para cima representam a direção y .

6.2 Equações da poroelasticidade

A teoria da poroelasticidade, na qual estudam-se materiais porosos preenchidos por fluidos, é de grande de grande interesse e essencial em diversas áreas de engenharia. Biot(1941, 1955) descreveu e analisou a teoria mais conhecida sobre poroelasticidade, atualmente conhecida como Teoria de Biot.

Essa teoria descreve o comportamento mecânico de meios porosos saturados com fluidos. O meio poroso natural é composto por uma fase sólida (grãos) e espaços entre esta fase sólida, que podem conter fluidos. Na Fig. 20 é mostrado um exemplo desse meio poroso para o caso 1D com domínio $\left[0, \frac{1}{2}\right]$. Os modelos da poroelasticidade são especificamente mais usados para o estudo de problemas em geomecânica, hidrogeologia, engenharia do petróleo e biomecânica (MOW; LAI, 1980; EHLERS; BLUHM, 2002).

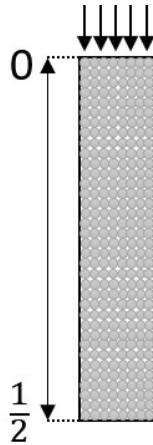


Figura 20 – Modelo do problema da poroelasticidade unidimensional.

6.2.1 Modelos matemáticos

O problema da poroelasticidade pode ser formulado como um sistema acoplado de equações diferenciais parciais dependentes do tempo. Existem formulações de dois, três ou quatro campos do problema de Biot. Aqui, considera-se a formulação de dois campos que inclui a pressão do fluido e os deslocamentos da matriz sólida como incógnitas. Em problemas mais complexos, pode-se incluir: erosão, sedimentação, entre outros efeitos físicos.

Para essas equação considera-se o problema clássico da consolidação de Biot para um meio poroso saturado, homogêneo, isotrópico e incompressível que seguem os modelos descrito por Gaspar et al.(2003, 2006) e Rodrigo (2010).

Para o caso unidimensional, considerando o domínio espacial $\Omega = \left(0, \frac{1}{2}\right)$ tem-se

$$\left\{ \begin{array}{l} -E \frac{\partial^2 u}{\partial x^2} + \frac{\partial p}{\partial x} = \mathcal{U} \\ \frac{\partial}{\partial t} \left(\frac{\partial u}{\partial x} \right) - K \frac{\partial^2 p}{\partial x^2} = \mathcal{P} \end{array} \right. , \quad (6.23)$$

em que E denota o módulo de Young e K guarda informações das propriedades físicas relacionadas à porosidade e permeabilidade do meio e é chamada de condutividade hidráulica. O termo \mathcal{U} representa a densidade da força aplicada no corpo, e \mathcal{P} a força de

injeção ou extração do fluido no meio poroso. A componente de deslocamento na direção espacial x é dada por $u(x, t)$ e $p(x, t)$ denota a pressão. O intervalo de tempo considerado nos cálculos é $(0, t_f]$. A primeira equação modela os deslocamentos na direção u e a segunda equação modela a pressão p .

As condições de contornos assumem fronteira esquerda sem variação do deslocamento e permeável (drenagem livre); e fronteira direita rígida (deslocamento zero) e sem variação de pressão, ou seja,

$$\begin{cases} E \frac{\partial u(0, t)}{\partial x} = 0 \\ p(0, t) = 0 \end{cases} \quad (6.24)$$

e

$$\begin{cases} u\left(\frac{1}{2}, t\right) = 0 \\ K \frac{\partial p\left(\frac{1}{2}, t\right)}{\partial x} = 0 \end{cases} . \quad (6.25)$$

Supondo-se a solução analítica dada por

$$u(x, t) = \cos(\pi x) e^{-t} \quad (6.26)$$

e

$$p(x, t) = \sin(\pi x) e^{-t}, \quad (6.27)$$

e que satisfazem às condições de contornos dadas nas Eq. (6.24) e Eq. (6.25), pode-se definir os termos forçantes \mathcal{U} e \mathcal{P} como

$$\mathcal{U} = (E\pi + 1)\pi \cos(\pi x) e^{-t} \quad (6.28)$$

e

$$\mathcal{P} = (1 + K\pi) \pi \sin(\pi x) e^{-t}. \quad (6.29)$$

O problema da poroelasticidade bidimensional é dado com $u(x, y, t)$ e $v(x, y, t)$ que denotam o deslocamento e $p(x, y, t)$ que representa a pressão no plano espacial (x, y) . O modelo matemático, considerando o domínio espacial $\Omega = (0, 1) \times (0, 1)$ é dado por

$$\begin{cases} -(\lambda + 2\mu) \frac{\partial^2 u}{\partial x^2} - \mu \frac{\partial^2 u}{\partial y^2} - (\lambda + \mu) \frac{\partial}{\partial y} \left(\frac{\partial v}{\partial x} \right) + \frac{\partial p}{\partial x} = \mathcal{U} \\ -(\lambda + \mu) \frac{\partial}{\partial y} \left(\frac{\partial u}{\partial x} \right) - \mu \frac{\partial^2 v}{\partial x^2} - (\lambda + 2\mu) \frac{\partial^2 v}{\partial y^2} + \frac{\partial p}{\partial y} = \mathcal{V} \\ \frac{\partial}{\partial t} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) - K \left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} \right) = \mathcal{P} \end{cases} , \quad (6.30)$$

em $(x, y) \in \Omega$ e $t \in (0, t_f]$. As constantes λ e μ são coeficientes de Lamé enquanto que K guarda informações das propriedades físicas relacionadas à porosidade e permeabilidade do meio e é chamada de condutividade hidráulica. As variáveis u e v indicam o deslocamento enquanto p indica a pressão. Os termos \mathcal{U} e \mathcal{V} representam a densidade da força aplicada no corpo, e \mathcal{P} a força de injeção ou extração do fluido no meio poroso. Neste sistema de equações, a primeira equação é a equação para o deslocamento u , a segunda é a equação para o deslocamento v e a terceira é a equação para a pressão p .

Neste trabalho supõe-se que, no caso bidimensional, $\partial\Omega$ (a fronteira de Ω) é rígida (deslocamento zero) e permeável (drenagem livre), tal que tem-se a condição de contorno de Dirichlet,

$$u(x, y, t) = 0, \quad v(x, y, t) = 0, \quad p(x, y, t) = 0, \quad (x, y) \in \partial\Omega. \quad (6.31)$$

Através do método das soluções fabricadas, considerando-se a solução analítica

$$u(x, y, t) = v(x, y, t) = p(x, y, t) = \sin(\pi x) \sin(\pi y) e^{-t}, \quad (6.32)$$

tem-se as respectivas expressões para \mathcal{U} , \mathcal{V} e \mathcal{P}

$$\mathcal{U} = (\lambda + 3\mu)\pi^2 \sin(\pi x) \sin(\pi y) e^{-t} - (\lambda + \mu)\pi^2 \cos(\pi x) \cos(\pi y) e^{-t} + \pi \cos(\pi x) \sin(\pi y) e^{-t}, \quad (6.33)$$

$$\mathcal{V} = (\lambda + 3\mu)\pi^2 \sin(\pi x) \sin(\pi y) e^{-t} - (\lambda + \mu)\pi^2 \cos(\pi x) \cos(\pi y) e^{-t} + \pi \sin(\pi x) \cos(\pi y) e^{-t} \quad (6.34)$$

e

$$\mathcal{P} = 2K\pi^2 \sin(\pi x) \sin(\pi y) e^{-t} - \pi \sin(\pi(x + y)) e^{-t}. \quad (6.35)$$

6.2.2 Modelos numéricos

Para o modelo numérico, considera-se o modelo matemático apresentado na seção 6.2.1 e neste caso consideram-se as próprias variáveis u e v como sendo aproximações das soluções de u e v , respectivamente. Assim como na equação do calor, o domínio espacial é discretizado através do MDF considerando uma malha uniforme e com aproximação espacial do tipo CDS, como descrito na seção 3.1. A aproximação temporal e a conexão espacial e temporal é feita usando um método de discretização implícita (seção 3.4), como por exemplo neste caso, os métodos de Euler Implícito e de Crank-Nicolson.

Gaspar et al. (2007) apresentaram uma versão reformulada para o sistema da poroelasticidade. Essa nova versão (usada também por Rodrigo (2010)) consiste no uso adicional de um termo de suavização no lado esquerdo da igualdade da equação da pressão.

Esse termo não altera o resultado final e melhora a estabilidade do sistema para a solução numérica. O termo de suavização é dado por

$$-\frac{h^2}{4(\lambda + 2\mu)} \frac{\partial \Delta p}{\partial t}. \quad (6.36)$$

Caso unidimensional

Para o caso unidimensional, considera-se $E = \lambda + 2\mu$ e a Eq. (6.23) com o termo de suavização dado pela Eq. (6.36) pode ser discretizada nos pontos internos, i.e., $i = 2, 3, \dots, N - 1$, usando os métodos de Euler implícito e o método de Crank-Nicolson.

Método de Euler

Usando o método de Euler para aproximação temporal, tem-se

$$\left\{ \begin{array}{l} -E \frac{u_{i+1}^{m+1} - 2u_i^{m+1} + u_{i-1}^{m+1}}{h^2} + \frac{p_{i+1}^{m+1} - p_{i-1}^{m+1}}{2h} = \mathcal{U}_i^{m+1} \\ \frac{\frac{u_{i+1}^{m+1} - u_{i-1}^{m+1}}{2h} - \frac{u_{i+1}^m - u_{i-1}^m}{2h}}{\tau} - K \frac{p_{i+1}^{m+1} - 2p_i^{m+1} + p_{i-1}^{m+1}}{h^2} - \\ - \frac{h^2}{4E\tau} \left[\frac{p_{i+1}^{m+1} - 2p_i^{m+1} + p_{i-1}^{m+1}}{h^2} - \frac{p_{i+1}^m - 2p_i^m + p_{i-1}^m}{h^2} \right] = \mathcal{P}_i^{m+1} \end{array} \right., \quad (6.37)$$

reorganizando os termos, tem-se

$$\left\{ \begin{array}{l} \frac{2E}{h^2} u_i^{m+1} - \frac{E}{h^2} (u_{i+1}^{m+1} + u_{i-1}^{m+1}) + \frac{1}{2h} (p_{i+1}^{m+1} - p_{i-1}^{m+1}) = \mathcal{U}_i^{m+1} \\ \frac{u_{i+1}^{m+1} - u_{i-1}^{m+1}}{2\tau h} - \frac{u_{i+1}^m - u_{i-1}^m}{2\tau h} - \frac{K}{h^2} (p_{i+1}^{m+1} - 2p_i^{m+1} + p_{i-1}^{m+1}) - \\ - \frac{1}{4E\tau} [p_{i+1}^{m+1} - 2p_i^{m+1} + p_{i-1}^{m+1} - (p_{i+1}^m - 2p_i^m + p_{i-1}^m)] = \mathcal{P}_i^{m+1} \end{array} \right., \quad (6.38)$$

ou

$$\left\{ \begin{array}{l} \frac{2E}{h^2} u_i^{m+1} = \frac{E}{h^2} (u_{i-1}^{m+1} + u_{i+1}^{m+1}) + \frac{1}{2h} (p_{i-1}^{m+1} - p_{i+1}^{m+1}) + \mathcal{U}_i^{m+1} \\ \left(\frac{2K}{h^2} + \frac{1}{2E\tau} \right) p_i^{m+1} + \frac{1}{2\tau h} (u_{i+1}^{m+1} - u_{i-1}^{m+1}) - \left(\frac{K}{h^2} + \frac{1}{4E\tau} \right) (p_{i+1}^{m+1} + p_{i-1}^{m+1}) - \\ - \frac{1}{2\tau h} (u_{i+1}^m - u_{i-1}^m) + \frac{1}{4E\tau} (p_{i+1}^m - 2p_i^m + p_{i-1}^m) = \mathcal{P}_i^{m+1} \end{array} \right. , \quad (6.39)$$

ou ainda

$$\left\{ \begin{array}{l} \frac{2E}{h^2} u_i^{m+1} = \frac{E}{h^2} (u_{i-1}^{m+1} + u_{i+1}^{m+1}) + \frac{1}{2h} (p_{i-1}^{m+1} - p_{i+1}^{m+1}) + \mathcal{U}_i^{m+1} \\ \left(\frac{h^2 + 4E\tau K}{2E\tau h^2} \right) p_i^{m+1} = \frac{1}{2\tau h} (u_{i-1}^{m+1} - u_{i+1}^{m+1}) + \left(\frac{h^2 + 4E\tau K}{4E\tau h^2} \right) (p_{i-1}^{m+1} + p_{i+1}^{m+1}) + \\ - \frac{1}{2\tau h} (u_{i-1}^m - u_{i+1}^m) - \frac{1}{4E\tau} (p_{i-1}^m - 2p_i^m + p_{i+1}^m) + \mathcal{P}_i^{m+1} \end{array} \right. \quad (6.40)$$

Em $x = 0$ ($i = 1$), tem-se a condição de contorno $E \frac{\partial u(0,t)}{\partial x} = 0$, ou

$$\begin{aligned} E \frac{u_2^{m+1} - u_0^{m+1}}{2h} &= 0 \\ u_0^{m+1} &= u_2^{m+1} \end{aligned} \quad (6.41)$$

e considerando a equação

$$-E \frac{\partial^2 u}{\partial x^2} + \frac{\partial p}{\partial x} = \mathcal{U}, \quad (6.42)$$

tem-se

$$-E \frac{u_2^{m+1} - 2u_1^{m+1} + u_0^{m+1}}{h^2} + \frac{p_2^{m+1} - p_0^{m+1}}{2h} = \mathcal{U}_i^{m+1}, \quad (6.43)$$

considerando $p(0,t) = 0$, $p_1 = 0$ e $\frac{p_2 + p_0}{2} = p_1 = 0 \implies p_0 = -p_2$, na Eq. (6.43) e substituindo a Eq. (6.41) na Eq. (6.43), tem-se

$$\begin{aligned} -E \frac{u_2^{m+1} - 2u_1^{m+1} + u_2^{m+1}}{h^2} + \frac{p_2^{m+1} + p_2^{m+1}}{2h} &= \mathcal{U}_1^{m+1}, \\ -E \frac{u_2^{m+1} - 2u_1^{m+1} + u_2^{m+1}}{h^2} + \frac{2p_2^{m+1}}{2h} &= \mathcal{U}_1^{m+1}, \\ \frac{-2E}{h^2} (u_2^{m+1} - u_1^{m+1}) + \frac{1}{h} p_2^{m+1} &= \mathcal{U}_1^{m+1}, \end{aligned} \quad (6.44)$$

reorganizando os termos na Eq. (6.44), tem-se

$$\frac{2E}{h^2}u_1^{m+1} = \frac{2E}{h^2}u_2^{m+1} - \frac{1}{h}p_2^{m+1} + \mathcal{U}_1^{m+1}. \quad (6.45)$$

Em $x = \frac{1}{2}$ ($i = N$), tem-se

$$\frac{\partial}{\partial t} \left(\frac{\partial u}{\partial x} \right) - K \frac{\partial^2 p}{\partial x^2} - \frac{\partial}{\partial t} \left(\frac{h^2}{4E} \frac{\partial^2 p}{\partial x^2} \right) = \mathcal{P}, \quad (6.46)$$

na forma discretizada é

$$\begin{aligned} \left(\frac{h^2 + 4E\tau K}{2E\tau h^2} \right) p_N^{m+1} &= \frac{1}{2\tau h} (u_{N-1}^{m+1} - u_{N+1}^{m+1}) + \left(\frac{h^2 + 4E\tau K}{4E\tau h^2} \right) (p_{N-1}^{m+1} + p_{N+1}^{m+1}) + \\ &- \frac{1}{2\tau h} (u_{N-1}^m - u_{N+1}^m) - \frac{1}{4E\tau} (p_{N-1}^m - 2p_N^m + p_{N+1}^m) + \mathcal{P}_N^{m+1}, \end{aligned} \quad (6.47)$$

e considerando a condição de contorno $u\left(\frac{1}{2}, t\right) = 0$, tem-se

$$\frac{u_{N+1} + u_{N-1}}{2} = u_N = 0 \quad \implies \quad u_{N+1} = -u_{N-1} \quad (6.48)$$

e com a condição $K \frac{\partial p(1,t)}{\partial x} = 0$ tem-se

$$\begin{aligned} K \frac{p_{N+1} - p_{N-1}}{2h} &= 0 \\ p_{N+1} &= p_{N-1}. \end{aligned} \quad (6.49)$$

Substituindo as Eqs. (6.48) e (6.49) na Eq. (6.47) tem-se

$$\begin{aligned} \left(\frac{h^2 + 4E\tau K}{2E\tau h^2} \right) p_N^{m+1} &= \frac{1}{\tau h} u_{N-1}^{m+1} + \left(\frac{h^2 + 4E\tau K}{2E\tau h^2} \right) p_{N-1}^{m+1} + \\ &- \frac{1}{\tau h} u_{N-1}^m - \frac{1}{2E\tau} (p_{N-1}^m - p_N^m) + \mathcal{P}_N^{m+1}. \end{aligned} \quad (6.50)$$

Método de Crank-Nicolson

Considerando-se o método de CN para efetuar a aproximação temporal, reescreve-se a equação da pressão descrita na Eq. (6.23) e adiciona-se o termo de suavização dado pela Eq. (6.36) da seguinte forma

$$\frac{\partial}{\partial t} \left(\frac{\partial u}{\partial x} - \frac{h^2}{4E} \frac{\partial^2 p}{\partial x^2} \right) = K \frac{\partial^2 p}{\partial x^2} + \mathcal{P}, \quad (6.51)$$

assim, tem-se

$$\begin{aligned} \frac{\frac{u_{i+1}^{m+1} - u_{i-1}^{m+1}}{2h} - \frac{u_{i+1}^m - u_{i-1}^m}{2h}}{\tau} - \frac{h^2}{4E\tau} \left[\frac{p_{i+1}^{m+1} - 2p_i^{m+1} + p_{i-1}^{m+1}}{h^2} - \frac{p_{i+1}^m - 2p_i^m + p_{i-1}^m}{h^2} \right] &= \\ &= \frac{K}{2} \left(\frac{p_{i+1}^{m+1} - 2p_i^{m+1} + p_{i-1}^{m+1}}{h^2} + \frac{p_{i+1}^m - 2p_i^m + p_{i-1}^m}{h^2} \right) + \frac{\mathcal{P}_i^{m+1} + \mathcal{P}_i^m}{2}, \end{aligned} \quad (6.52)$$

reorganizando os termos, tem-se

$$\begin{aligned} \frac{u_{i+1}^{m+1} - u_{i-1}^{m+1}}{2\tau h} - \frac{u_{i+1}^m - u_{i-1}^m}{2\tau h} - \frac{1}{4E\tau} \left[p_{i+1}^{m+1} - 2p_i^{m+1} + p_{i-1}^{m+1} - (p_{i+1}^m - 2p_i^m + p_{i-1}^m) \right] = \\ = \frac{K}{2h^2} \left(p_{i+1}^{m+1} - 2p_i^{m+1} + p_{i-1}^{m+1} + p_{i+1}^m - 2p_i^m + p_{i-1}^m \right) + \frac{\mathcal{P}_i^{m+1} + \mathcal{P}_i^m}{2}, \end{aligned} \quad (6.53)$$

ou

$$\begin{aligned} \left(\frac{K}{h^2} + \frac{1}{2E\tau} \right) p_i^{m+1} = -\frac{1}{2\tau h} \left(u_{i+1}^{m+1} - u_{i-1}^{m+1} \right) + \left(\frac{K}{2h^2} + \frac{1}{4E\tau} \right) \left(p_{i+1}^{m+1} + p_{i-1}^{m+1} \right) + \\ + \frac{1}{2\tau h} \left(u_{i+1}^m - u_{i-1}^m \right) + \left(\frac{K}{2h^2} - \frac{1}{4E\tau} \right) \left(p_{i+1}^m - 2p_i^m + p_{i-1}^m \right) + \frac{\mathcal{P}_i^{m+1} + \mathcal{P}_i^m}{2}, \end{aligned} \quad (6.54)$$

ou ainda

$$\begin{aligned} \left(\frac{h^2 + 2E\tau K}{2E\tau h^2} \right) p_i^{m+1} = \frac{1}{2\tau h} \left(u_{i-1}^{m+1} - u_{i+1}^{m+1} \right) + \left(\frac{h^2 + 2E\tau K}{4E\tau h^2} \right) \left(p_{i+1}^{m+1} + p_{i-1}^{m+1} \right) - \\ - \frac{1}{2\tau h} \left(u_{i-1}^m - u_{i+1}^m \right) + \left(\frac{2E\tau K - h^2}{4E\tau h^2} \right) \left(p_{i-1}^m - 2p_i^m + p_{i+1}^m \right) + \frac{\mathcal{P}_i^{m+1} + \mathcal{P}_i^m}{2}. \end{aligned} \quad (6.55)$$

Para os contornos, as aproximações são similares às apresentadas para o método de Euler e não serão expostas aqui.

Caso bidimensional

Para o caso bidimensional, considera-se a Eq. (6.30) e adiciona-se o termo de suavização dado pela Eq. (6.36) na equação da pressão.

Método de Euler

Usando o método de Euler implícito para a aproximação temporal, as equações

discretizadas no pontos internos, $i = 2, 3, \dots, N - 1$, são escritas da seguinte forma

$$\left\{ \begin{array}{l} -(\lambda + 2\mu) \frac{u_{i-1,j}^{m+1} - 2u_{i,j}^{m+1} + u_{i+1,j}^{m+1}}{h^2} - \mu \frac{u_{i,j-1}^{m+1} - 2u_{i,j}^{m+1} + u_{i,j+1}^{m+1}}{h^2} + \\ -(\lambda + \mu) \frac{\partial}{\partial y} \left(\frac{v_{i+1,j}^{m+1} - v_{i-1,j}^{m+1}}{2h} \right) + \frac{p_{i+1,j}^{m+1} - p_{i-1,j}^{m+1}}{2h} = \mathcal{U}_{i,j}^{m+1} \\ \\ -(\lambda + \mu) \frac{\partial}{\partial y} \left(\frac{u_{i+1,j}^{m+1} - u_{i-1,j}^{m+1}}{2h} \right) - \mu \frac{v_{i-1,j}^{m+1} - 2v_{i,j}^{m+1} + v_{i+1,j}^{m+1}}{h^2} - \\ -(\lambda + 2\mu) \frac{v_{i,j-1}^{m+1} - 2v_{i,j}^{m+1} + v_{i,j+1}^{m+1}}{h^2} + \frac{p_{i,j+1}^{m+1} - p_{i,j-1}^{m+1}}{2h} = \mathcal{V}_{i,j}^{m+1} \end{array} \right. , \quad (6.56)$$

$$\left\{ \begin{array}{l} \frac{\partial}{\partial t} \left(\frac{u_{i+1,j}^{m+1} - u_{i-1,j}^{m+1}}{2h} + \frac{v_{i,j+1}^{m+1} - v_{i,j-1}^{m+1}}{2h} \right) - \\ -K \left(\frac{p_{i-1,j}^{m+1} - 2p_{i,j}^{m+1} + p_{i+1,j}^{m+1}}{h^2} + \frac{p_{i,j-1}^{m+1} - 2p_{i,j}^{m+1} + p_{i,j+1}^{m+1}}{h^2} \right) - \\ - \frac{h^2}{4(\lambda + 2\mu)} \frac{\partial}{\partial t} \left(\frac{p_{i-1,j}^{m+1} - 2p_{i,j}^{m+1} + p_{i+1,j}^{m+1}}{h^2} + \frac{p_{i,j-1}^{m+1} - 2p_{i,j}^{m+1} + p_{i,j+1}^{m+1}}{h^2} \right) = \mathcal{P}_{i,j}^{m+1} \end{array} \right.$$

ou

$$\left\{ \begin{array}{l} \frac{2(\lambda + 3\mu)}{h^2} u_{i,j}^{m+1} - \frac{\lambda + 2\mu}{h^2} (u_{i-1,j}^{m+1} + u_{i+1,j}^{m+1}) - \frac{\mu}{h^2} (u_{i,j-1}^{m+1} + u_{i,j+1}^{m+1}) - \frac{(\lambda + \mu)}{4h^2} \cdot \\ \cdot (v_{i+1,j+1}^{m+1} - v_{i+1,j-1}^{m+1} - v_{i-1,j+1}^{m+1} + v_{i-1,j-1}^{m+1}) + \frac{1}{2h} (p_{i+1,j}^{m+1} - p_{i-1,j}^{m+1}) = \mathcal{U}_{i,j}^{m+1} \\ \\ - \frac{\lambda + \mu}{4h^2} (u_{i+1,j+1}^{m+1} - u_{i+1,j-1}^{m+1} - u_{i-1,j+1}^{m+1} + u_{i-1,j-1}^{m+1}) + \frac{2(\lambda + 3\mu)}{h^2} v_{i,j}^{m+1} - \\ - \frac{\mu}{h^2} (v_{i-1,j}^{m+1} + v_{i+1,j}^{m+1}) - \frac{\lambda + 2\mu}{h^2} (v_{i,j-1}^{m+1} + v_{i,j+1}^{m+1}) + \frac{1}{2h} (p_{i,j+1}^{m+1} - p_{i,j-1}^{m+1}) = \mathcal{V}_{i,j}^{m+1} \end{array} \right. , \quad (6.57)$$

$$\left\{ \begin{array}{l} \frac{1}{2h\tau} [u_{i+1,j}^{m+1} - u_{i-1,j}^{m+1} + v_{i,j+1}^{m+1} - v_{i,j-1}^{m+1} - (u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m)] - \\ -K \left(\frac{p_{i-1,j}^{m+1} - 2p_{i,j}^{m+1} + p_{i+1,j}^{m+1}}{h^2} + \frac{p_{i,j-1}^{m+1} - 2p_{i,j}^{m+1} + p_{i,j+1}^{m+1}}{h^2} \right) - \\ - \frac{1}{4(\lambda + 2\mu)\tau} (p_{i-1,j}^{m+1} - 2p_{i,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} - 2p_{i,j}^{m+1} + p_{i,j+1}^{m+1}) + \\ + \frac{1}{4(\lambda + 2\mu)\tau} (p_{i-1,j}^m - 2p_{i,j}^m + p_{i+1,j}^m + p_{i,j-1}^m - 2p_{i,j}^m + p_{i,j+1}^m) = \mathcal{P}_{i,j}^{m+1} \end{array} \right.$$

ou ainda

$$\left\{ \begin{array}{l}
 \frac{2(\lambda + 3\mu)}{h^2} u_{i,j}^{m+1} = \frac{\lambda + 2\mu}{h^2} (u_{i-1,j}^{m+1} + u_{i+1,j}^{m+1}) + \frac{\mu}{h^2} (u_{i,j-1}^{m+1} + u_{i,j+1}^{m+1}) + \frac{(\lambda + \mu)}{4h^2} \cdot \\
 \cdot (v_{i+1,j+1}^{m+1} - v_{i+1,j-1}^{m+1} - v_{i-1,j+1}^{m+1} + v_{i-1,j-1}^{m+1}) + \frac{1}{2h} (p_{i-1,j}^{m+1} - p_{i+1,j}^{m+1}) + \mathcal{U}_{i,j}^{m+1} \\
 \\
 \frac{2(\lambda + 3\mu)}{h^2} v_{i,j}^{m+1} = \frac{\lambda + \mu}{4h^2} (u_{i+1,j+1}^{m+1} - u_{i+1,j-1}^{m+1} - u_{i-1,j+1}^{m+1} + u_{i-1,j-1}^{m+1}) + \\
 + \frac{\mu}{h^2} (v_{i-1,j}^{m+1} + v_{i+1,j}^{m+1}) + \frac{\lambda+2\mu}{h^2} (v_{i,j-1}^{m+1} + v_{i,j+1}^{m+1}) + \frac{1}{2h} (p_{i,j-1}^{m+1} - p_{i,j+1}^{m+1}) + \mathcal{V}_{i,j}^{m+1} \\
 \\
 \frac{1}{2h\tau} (u_{i+1,j}^{m+1} - u_{i-1,j}^{m+1} + v_{i,j+1}^{m+1} - v_{i,j-1}^{m+1}) + \left(\frac{4k}{h^2} + \frac{1}{(\lambda + 2\mu)\tau} \right) p_{i,j}^{m+1} - \\
 - \left(\frac{K}{h^2} + \frac{1}{4(\lambda + 2\mu)\tau} \right) (p_{i-1,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1}) - \\
 - \frac{1}{2h\tau} (u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m) + \\
 + \frac{1}{4(\lambda + 2\mu)\tau} (p_{i-1,j}^m - 2p_{i,j}^m + p_{i+1,j}^m + p_{i,j-1}^m - 2p_{i,j}^m + p_{i,j+1}^m) = \mathcal{P}_{i,j}^{m+1}
 \end{array} \right. , \quad (6.58)$$

e reorganizando os termos, tem-se

$$\left\{ \begin{array}{l}
 \frac{2(\lambda + 3\mu)}{h^2} u_{i,j}^{m+1} = \frac{\lambda + 2\mu}{h^2} (u_{i-1,j}^{m+1} + u_{i+1,j}^{m+1}) + \frac{\mu}{h^2} (u_{i,j-1}^{m+1} + u_{i,j+1}^{m+1}) + \frac{(\lambda + \mu)}{4h^2} \cdot \\
 \cdot (v_{i+1,j+1}^{m+1} - v_{i+1,j-1}^{m+1} - v_{i-1,j+1}^{m+1} + v_{i-1,j-1}^{m+1}) + \frac{1}{2h} (p_{i-1,j}^{m+1} - p_{i+1,j}^{m+1}) + \mathcal{U}_{i,j}^{m+1} \\
 \\
 \frac{2(\lambda + 3\mu)}{h^2} v_{i,j}^{m+1} = \frac{\lambda + \mu}{4h^2} (u_{i+1,j+1}^{m+1} - u_{i+1,j-1}^{m+1} - u_{i-1,j+1}^{m+1} + u_{i-1,j-1}^{m+1}) + \\
 + \frac{\mu}{h^2} (v_{i-1,j}^{m+1} + v_{i+1,j}^{m+1}) + \frac{\lambda+2\mu}{h^2} (v_{i,j-1}^{m+1} + v_{i,j+1}^{m+1}) + \frac{1}{2h} (p_{i,j-1}^{m+1} - p_{i,j+1}^{m+1}) + \mathcal{V}_{i,j}^{m+1} \\
 \\
 \left(\frac{h^2 + 4k(\lambda + 2\mu)\tau}{h^2(\lambda + 2\mu)\tau} \right) p_{i,j}^{m+1} = \frac{1}{2h\tau} (u_{i-1,j}^{m+1} - u_{i+1,j}^{m+1} + v_{i,j-1}^{m+1} - v_{i,j+1}^{m+1}) + \\
 + \left(\frac{h^2 + 4k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \right) (p_{i-1,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1}) + \\
 + \frac{1}{2h\tau} (u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m) + \\
 + \frac{1}{4(\lambda + 2\mu)\tau} (4p_{i,j}^m - p_{i-1,j}^m - p_{i+1,j}^m - p_{i,j-1}^m - p_{i,j+1}^m) + \mathcal{P}_{i,j}^{m+1}
 \end{array} \right. . \quad (6.59)$$

Método de Crank-Nicolson

Considerando-se o método de CN para efetuar a aproximação temporal, reescreve-se a equação da pressão descrita na Eq. (6.30) e adiciona-se o termo de suavização dado pela Eq. (6.36) da seguinte forma

$$\frac{\partial}{\partial t} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) - K \left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} \right) - \frac{h^2}{4(\lambda + 2\mu)} \frac{\partial}{\partial t} \left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} \right) = \mathcal{P},$$

ou

$$\frac{\partial}{\partial t} \left[\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} - \frac{h^2}{4(\lambda + 2\mu)} \left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} \right) \right] = K \left(\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} \right) + \mathcal{P},$$

com a discretização das variáveis espaciais pode-se escrever

$$\begin{aligned} & \frac{\partial}{\partial t} \left[\frac{u_{i+1,j}^{m+1} - u_{i-1,j}^{m+1}}{2h} + \frac{v_{i,j+1}^{m+1} - v_{i,j-1}^{m+1}}{2h} - \right. \\ & \left. - \frac{h^2}{4(\lambda + 2\mu)} \left(\frac{p_{i-1,j}^{m+1} - 2p_{i,j}^{m+1} + p_{i+1,j}^{m+1}}{h^2} + \frac{p_{i,j-1}^{m+1} - 2p_{i,j}^{m+1} + p_{i,j+1}^{m+1}}{h^2} \right) \right] = \\ = & \frac{K}{2} \left(\frac{p_{i-1,j}^{m+1} - 2p_{i,j}^{m+1} + p_{i+1,j}^{m+1}}{h^2} + \frac{p_{i,j-1}^{m+1} - 2p_{i,j}^{m+1} + p_{i,j+1}^{m+1}}{h^2} + \frac{p_{i-1,j}^m - 2p_{i,j}^m + p_{i+1,j}^m}{h^2} + \right. \\ & \left. + \frac{p_{i,j-1}^m - 2p_{i,j}^m + p_{i,j+1}^m}{h^2} \right) + \frac{\mathcal{P}_{i,j}^{m+1} + \mathcal{P}_{i,j}^m}{2}, \end{aligned}$$

e com a aproximação temporal, fica

$$\begin{aligned} & \frac{1}{2h\tau} \left[u_{i+1,j}^{m+1} - u_{i-1,j}^{m+1} + v_{i,j+1}^{m+1} - v_{i,j-1}^{m+1} - \left(u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m \right) \right] \\ & - \frac{1}{4(\lambda + 2\mu)\tau} \left(p_{i-1,j}^{m+1} - 2p_{i,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} - 2p_{i,j}^{m+1} + p_{i,j+1}^{m+1} \right) \\ & + \frac{1}{4(\lambda + 2\mu)\tau} \left(p_{i-1,j}^m - 2p_{i,j}^m + p_{i+1,j}^m + p_{i,j-1}^m - 2p_{i,j}^m + p_{i,j+1}^m \right) = \\ = & \frac{K}{2} \left(\frac{p_{i-1,j}^{m+1} - 4p_{i,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1}}{h^2} + \frac{p_{i-1,j}^m - 4p_{i,j}^m + p_{i+1,j}^m + p_{i,j-1}^m + p_{i,j+1}^m}{h^2} \right) + \\ & + \frac{\mathcal{P}_{i,j}^{m+1} + \mathcal{P}_{i,j}^m}{2}, \end{aligned}$$

reorganizando os termos, tem-se

$$\begin{aligned} & \frac{1}{2h\tau} \left(u_{i+1,j}^{m+1} - u_{i-1,j}^{m+1} + v_{i,j+1}^{m+1} - v_{i,j-1}^{m+1} \right) + \left(\frac{2k}{h^2} + \frac{1}{(\lambda + 2\mu)\tau} \right) p_{i,j}^{m+1} \\ & - \left(\frac{K}{2h^2} + \frac{1}{4(\lambda + 2\mu)\tau} \right) \left(p_{i-1,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1} \right) \\ & - \frac{1}{2h\tau} \left(u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m \right) \\ & + \left(\frac{1}{4(\lambda + 2\mu)\tau} - \frac{K}{2h^2} \right) \left(p_{i-1,j}^m - 4p_{i,j}^m + p_{i+1,j}^m + p_{i,j-1}^m + p_{i,j+1}^m \right) = \frac{\mathcal{P}_{i,j}^{m+1} + \mathcal{P}_{i,j}^m}{2}, \end{aligned}$$

ou ainda,

$$\begin{aligned}
& \left(\frac{h^2 + 2k(\lambda + 2\mu)\tau}{h^2(\lambda + 2\mu)\tau} \right) p_{i,j}^{m+1} = \frac{1}{2h\tau} \left(u_{i-1,j}^{m+1} - u_{i+1,j}^{m+1} + v_{i,j-1}^{m+1} - v_{i,j+1}^{m+1} \right) + \\
& \left(\frac{h^2 + 2k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \right) \left(p_{i-1,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1} \right) + \frac{1}{2h\tau} \left(u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m \right) \\
& + \left(\frac{h^2 - 2k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \right) \left(4p_{i,j}^m - p_{i-1,j}^m - p_{i+1,j}^m - p_{i,j-1}^m - p_{i,j+1}^m \right) + \frac{\mathcal{P}_{i,j}^{m+1} + \mathcal{P}_{i,j}^m}{2}. \quad (6.60)
\end{aligned}$$

Uma ideia para a programação do suavizador Vanka de 3 e de 5 pontos está descrita no apêndice A.

6.3 Ordenação colorida para o suavizador Vanka

Sabe-se que a solução dos grandes sistemas lineares resultantes da discretização do modelo de Biot é a parte que representa maior custo computacional em simulações reais. Por isso, a estrutura de suavizadores eficientes para este tipo de problemas atrai muita atenção. Além disso, as arquiteturas de computadores atuais tendem a aumentar as capacidades de multiprocessamento ao apresentar mais núcleos de processamentos. Para explorar essas características, é preciso desenvolver algoritmos que sejam capazes de usar um grande número de núcleos, tornando possível um maior paralelismo. De acordo com Rodrigo (2010), com o método *multigrid* não é diferente, sendo muito importante a escolha de um suavizador adequado. Além disso, para suavizar um sistema de equações como o problema da poroelasticidade existem algumas dificuldades adicionais, pois a matriz discretizada apresenta pontos de sela, em que o laplaciano da pressão é quase nulo. Isso faz com que os métodos que utilizam suavização por pontos não sejam satisfatórios, tornando o suavizador Vanka muito atrativo para esse caso.

Nesta seção será proposta uma formulação para a ordenação colorida a ser usada no suavizador Vanka, de forma que este suavizador possa ser completamente paralelizável no espaço. Para que um suavizador seja paralelizável é necessário que nas atualizações das variáveis de uma mesma cor não se utilizem valores que foram atualizados na mesma iteração. Após o estudo da formulação do suavizador Vanka usado nas equações da poroelasticidade discretizadas através do Método das Diferenças Finitas (MDF), percebeu-se que o número mínimo de cores possíveis para possibilitar a paralelização do referido suavizador são 4 cores para o caso 1D e 12 cores para o caso 2D.

No caso unidimensional, após a discretização através do MDF obtém-se a célula descrita na Fig. 21. Nessa figura, as variáveis que estão em um círculo (○) serão atualizadas na iteração atual e as demais variáveis serão apenas usadas na iterada atual para os respectivos cálculos.

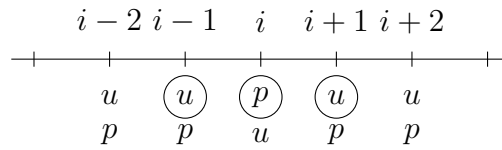


Figura 21 – Variáveis usadas pelo suavizador Vanka de 3 pontos para equação da poroelasticidade 1D. As variáveis que estão em um círculo (○) serão atualizadas na iteração atual.

Com o auxílio da Fig. 21 é possível notar que após a suavização e atualização das variáveis que se encontram nas posições u_{i-1}, p_i, u_{i+1} , as próximas variáveis a serem atualizadas devem ser as variáveis que se encontram nas posições $u_{i+3}, p_{i+4}, u_{i+5}$ (ver a Eq. (A.3)). Observa-se que no segundo passo de atualização, usam-se as variáveis u_{i+2} e p_{i+2} , que também foram usadas (mas não atualizadas) no primeiro passo da suavização, ver por exemplo a Fig. 22. Na Fig. 22 descrevem-se dois passos feitos em uma mesma iteração usando-se tal ordenação, que estão com as variáveis relacionadas ao primeiro e ao segundo passo de atualização, abaixo e acima do eixo principal, respectivamente. A numeração dos nós desta figura seguem a numeração descrita na Fig. 21. Com isso, pode-se garantir que para o suavizador Vanka de 3 pontos ser paralelizável, devem-se usar no mínimo 4 cores.

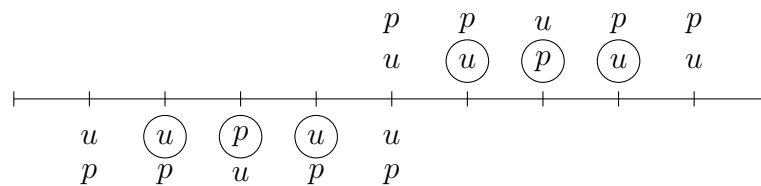


Figura 22 – Suavizador Vanka de 3 pontos para equação da poroelasticidade 1D. Os valores abaixo da linha reta pertencem a um passo de suavização e os valores acima da linha pertencem ao passo subsequente. Apenas as variáveis que estão em um círculo (○) serão atualizadas na iteração atual.

Sendo assim, pode-se mostrar na Fig. 23 a ordenação de 4 cores usada no problema da poroelasticidade 1D.



Figura 23 – Ordenação de 4 cores para o suavizador Vanka: (○) *vermelho*; (□) *preto*; (△) *azul* e (▽) *verde*.

Para o caso bidimensional, em primeiro momento pensou-se em usar um esquema com 4 cores em cada direção espacial, ou seja 16 cores no total. Porém, com a análise da célula do suavizador Vanka descrito na Fig. 24, pode-se observar que o ponto \textcircled{p} (que está em um círculo (○) possui vizinhos em 2 pontos distintos nas posições Oeste, Leste,

Norte e Sul. Porém, este ponto \textcircled{p} possui apenas 1 vizinho nas posições Nordeste, Sudeste, Sudoeste e Noroeste (para interpretar melhor, ver a Eq. (A.20) e a Fig. (96)).

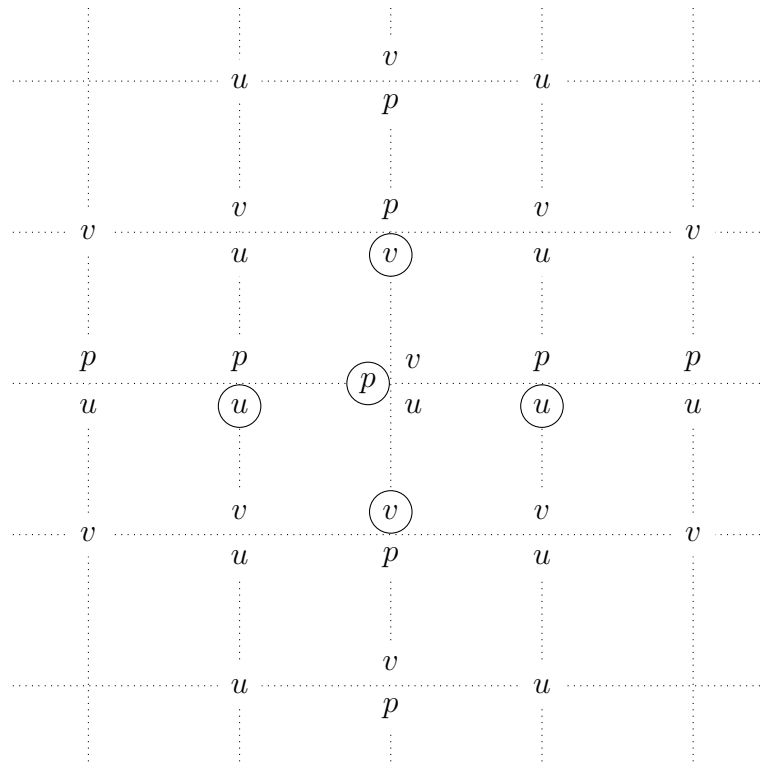


Figura 24 – Variáveis usadas pelo suavizador Vanka de 5 pontos para equação da poroelasticidade 2D. As variáveis que estão em um círculo ($\textcircled{}$) são atualizadas na iteração atual.

Considerando-se os argumentos citados, propôs-se uma ordenação que utiliza 6 cores em uma direção espacial (no caso, na direção x) e apenas 2 cores na outra direção espacial, totalizando o número mínimo de 12 cores e garantindo que o algoritmo seja paralelizável. Essa descrição é feita, por exemplo em uma malha com 17×17 pontos como mostra a Fig. 25.

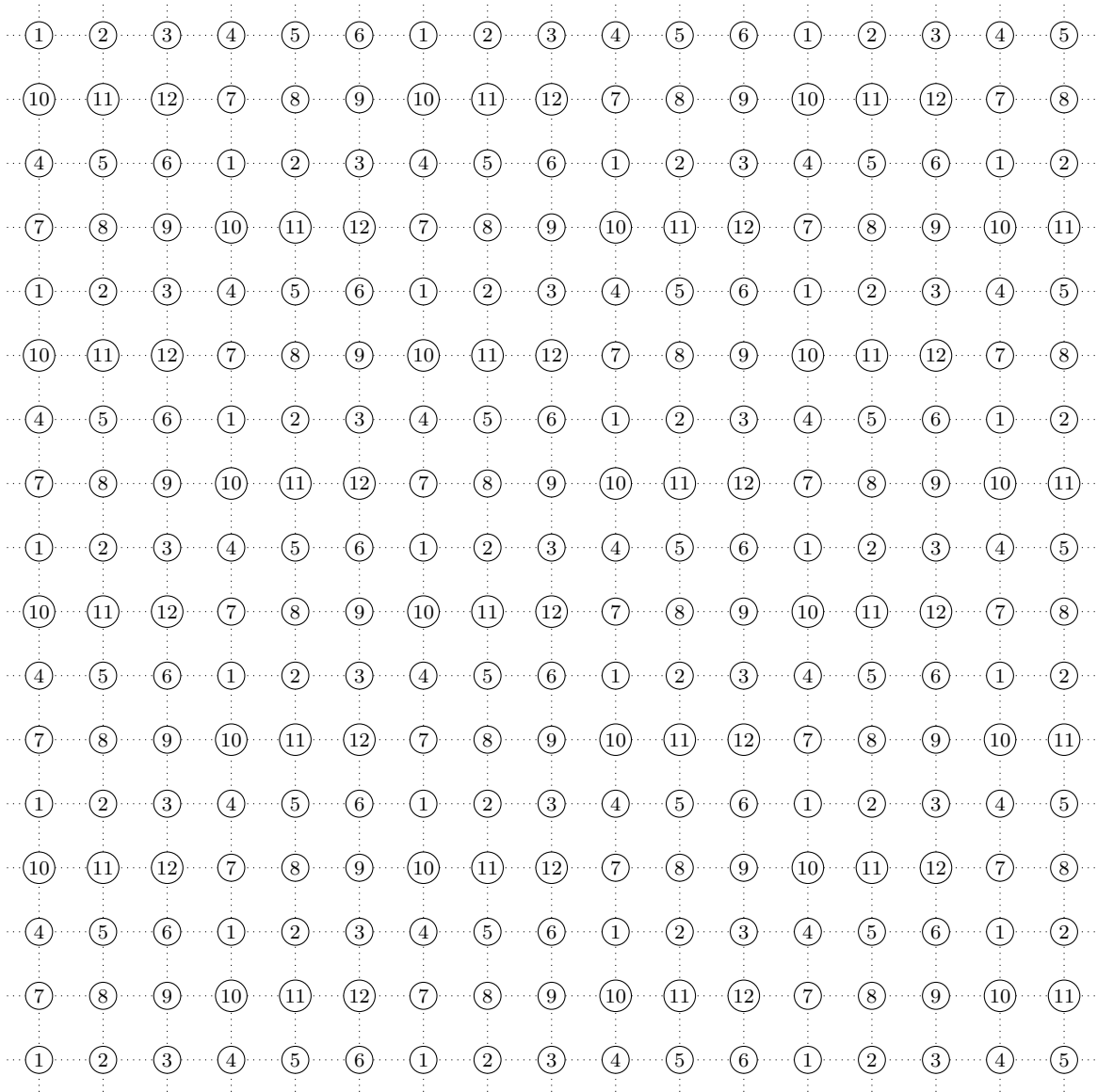


Figura 25 – Ordenação de 12 cores usada para o problema da poroelasticidade 2D com o suavizador Vanka de 5 pontos.

7 Métodos de solução

Neste capítulo serão descritos os métodos (ou técnicas) *Time-Stepping*, *Waveform Relaxation* e *Space-Time*, usados para a solução de EDPs parabólicas.

Por simplicidade, para a descrição dessas técnicas, segue-se como base o problema modelo padrão para classe de EDPs parabólicas, dado pela equação do calor (ou equação de Fourier) bidimensional descrita na Eq. (6.1) como

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + f. \quad (7.1)$$

Este problema combina características de EDPs elípticas estacionárias para o caso de se considerar um passo de tempo fixo (gerando o método *Time-Stepping*), equações diferenciais ordinárias (EDOs) transientes para o caso de se considerar um ponto espacial fixo e todos os passos de tempo (gerando o método *Waveform Relaxation*) e uma descrição espaço-tempo com suavizador por pontos (gerando o método *Space-Time*). Será exposta também a descrição do método *Waveform Relaxation* para o problema da poroelasticidade.

7.1 Método *Time-Stepping*

Segundo Tannehill et al. (1997) alguns estudos têm dado ênfase para algoritmos que tratam a equação parabólica como uma sequência de equações elípticas em cada passo de tempo. Este método é chamado de método *Time-Stepping*.

Nesse método, resolve-se o sistema discretizado em cada passo de tempo de forma consecutiva até o passo de tempo final. Assim, em cada passo de tempo, pode-se usar algum método para o problema estacionário. Com isso, a estrutura do sistema a ser resolvido é correspondente à discretização de EDPs elípticas (TANNEHILL et al., 1997; BURDEN; FAIRES, 2016; LENT, 2006; HIRSCH, 2007).

A principal característica do método *Time-Stepping* é mostrado na Fig. 26, retirada de (VANDEWALLE, 1993, p. 18). Em cada passo de tempo é usada a solução do passo de tempo anterior como estimativa inicial e resolve-se o sistema de equações correspondente no passo de tempo atual, considerando-se um sistema estacionário. Com isso, esse sistema de equações diferenciais parciais parabólicas pode ser comparado às equações diferenciais parciais elípticas, para as quais o método *multigrid* é comprovadamente eficiente (WESSELING, 1992; BRIGGS et al., 2000; TROTTEBERG et al., 2001). Dessa forma, na programação do método *Time-Stepping* deve-se fazer um laço externo para os passos de tempo ($m + 1$), até o número de passos de tempo N_t , e um laço interno para as variáveis espaciais (h ou i, j). Esse procedimento é especificado através do algoritmo 5.

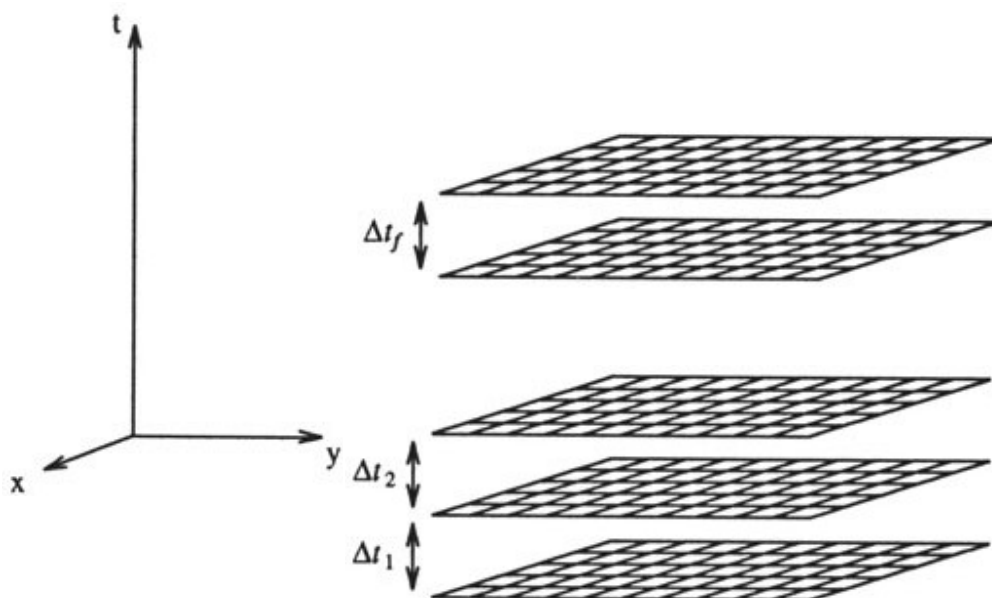


Figura 26 – Procedimento de atualização das incógnitas no método *Time-Stepping* (VANDEWALLE, 1993, p. 18) para o caso 2D.

Algoritmo 5: Método *Time-Stepping* em uma malha

Entre com os dados de entrada, condição inicial e condições de contornos.

for $m = 1 : N_t$ **do**

while Não alcançar algum critério de parada **do**

 Suavize o sistema correspondente no passo de tempo $m + 1$.

end while

end for

Devido à característica do método *Time-Stepping*, para os casos em que se usa a técnica *multigrid*, além da malha fina Ω^h , as malhas mais grossas $\Omega^{2h}, \Omega^{4h}, \dots, \Omega^{lh}$, com $l = 2^{(L_{\max}-1)}$, em que L_{\max} é o número máximo de malhas a serem utilizadas, são discretizadas apenas nas variáveis espaciais, ou seja, em cada passo de tempo fixo. Por exemplo, considerando-se a discretização das variáveis espaciais x, y na malha Ω^h com 33×33 pontos, as malhas mais grossas $\Omega^{2h}, \Omega^{4h}, \Omega^{8h}$ e Ω^{16h} terão $17 \times 17, 9 \times 9, 5 \times 5$ e 3×3 pontos, respectivamente.

Pode-se especificar o método *multigrid* com os ciclos $V(\nu_1, \nu_2), F(\nu_1, \nu_2)$ ou $W(\nu_1, \nu_2)$ para o método *Time-Stepping* através do algoritmo 6. A sigla **MG-ciclo** representa qualquer um dos ciclos (V, W ou F) definidos anteriormente nos Alg. 1 e Alg. 2. A partir de agora, tal símbolo será usado nesses algoritmos.

Algoritmo 6: Método *Time-Stepping - multigrid*

Entre com os dados de entrada, condição inicial e condições de contornos.
for $m = 1 : N_t$ **do**
 while Não alcançar algum critério de parada **do**
 Um ciclo *multigrid* com **MG-ciclo** (1).
 end while
end for

7.2 Método *Waveform Relaxation*

O método *Waveform Relaxation* é um método iterativo contínuo no tempo e foi desenvolvido para resolver numericamente grandes sistemas de equações diferenciais ordinárias (ODEs) (LELARASMEE et al., 1982; LUBICH; OSTERMANN, 1987; VANDEWALLE, 1993). No entanto, esse método também pode ser aplicado para resolver equações diferenciais parciais (EDPs) dependentes do tempo, em que as EDPs são transformadas em um grande conjunto de equações diferenciais ordinárias (EDOs). Inicialmente é feita uma discretização do problema apenas espacialmente, transformando a Eq. (7.1) em um conjunto de EDOs da forma

$$\frac{dv_h}{dt} = G_h(v_h, f_h), \quad (7.2)$$

em que G_h está em função dos valores de v_h e f_h que são funções ou vetores contendo informações do tempo para cada posição espacial h . Por isso, assumindo-se conhecidos os valores das posições espaciais, pode-se considerar G_h um sistema de EDOs em t . Esse método iterativo, calculado em função do tempo, é conhecido como MOL (em inglês, *Method of Lines*). O método *Waveform Relaxation* é a forma de esse tipo de sistema por relaxação (VANDEWALLE, 1993; LENT, 2006; FALGOUT et al., 2017).

Cada componente do sistema dado na Eq. (7.2) pode ser escrito como uma EDO, da seguinte forma

$$\left\{ \begin{array}{ll} \frac{d}{dt} v_1 = G_1(v_1, v_2, \dots, v_d, f_1) & v_1(0) = v_1^0 \\ \frac{d}{dt} v_2 = G_2(v_1, v_2, \dots, v_d, f_2) & \text{com } v_2(0) = v_2^0 \\ \vdots & \\ \frac{d}{dt} v_d = G_d(v_1, v_2, \dots, v_d, f_d) & v_d(0) = v_d^0 \end{array} \right., \quad (7.3)$$

em que d é a dimensão do sistema linear. A notação $v_i(0) = v_i^0$, $1 \leq i \leq d$, indica as condições iniciais ($t=0$) em cada ponto da discretização espacial. Esse procedimento de solução pode ser observado na Fig. 27, conforme Lent (2006, p. 94) para o caso 1D.

Considerando-se o método de Gauss-Seidel (GS) e a ordem Lexicográfica, o sistema

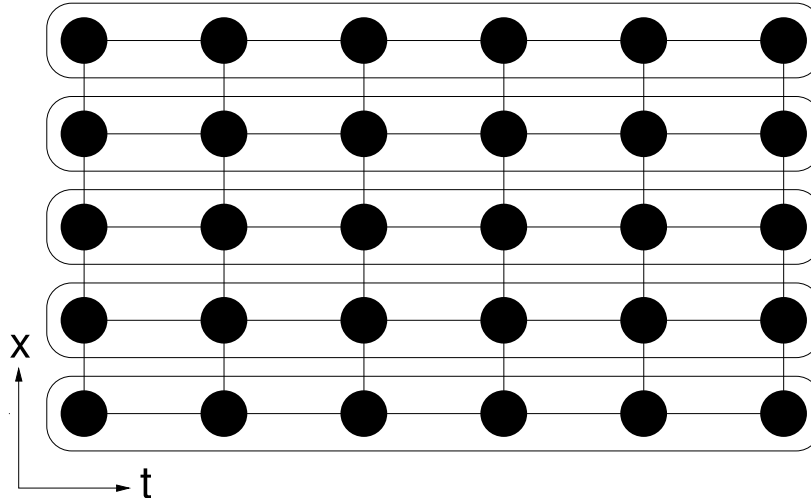


Figura 27 – Procedimento de atualização das incógnitas no método *Waveform Relaxation* (LENT, 2006, p. 94) para o caso 1D.

dado pelas Eqs. (7.3) pode ser escrito como

$$\begin{cases} \frac{d}{dt}v_1^{\nu+1} = G_1(v_1^{\nu+1}, v_2^\nu, \dots, v_d^\nu, f_1) & v_1^\nu(0) = v_1^{0,\nu} \\ \frac{d}{dt}v_2^{\nu+1} = G_2(v_1^{\nu+1}, v_2^{\nu+1}, v_3^\nu, \dots, v_d^\nu, f_2) & \text{com } v_2^\nu(0) = v_2^{0,\nu} \\ \vdots & \\ \frac{d}{dt}v_d^{\nu+1} = G_d(v_1^{\nu+1}, v_2^{\nu+1}, \dots, v_d^{\nu+1}, f_d) & v_d^\nu(0) = v_d^{0,\nu} \end{cases}. \quad (7.4)$$

Observa-se a relação entre as equações do sistema (7.4) e as linhas do tempo na Fig. 27, ou seja, para cada ponto da discretização espacial é resolvida uma EDO temporal. Os métodos de aproximação da derivada temporal podem ser o método de Euler implícito e de Crank-Nicolson, entre outros. Após realizar esse procedimento em todos os pontos espaciais, considera-se realizada uma iteração ν .

Considerando-se que a principal característica do método *Waveform Relaxation*, em que para cada ponto da discretização espacial resolve-se uma EDO temporal, como mostra a Fig. 27, na programação computacional deve-se fazer um laço externo para as iterações ν , um laço intermediário para percorrer os pontos da discretização espacial h e um laço interno para os passos de tempo m (ou para resolver a EDO temporal). Esse procedimento é especificado através do algoritmo 7, baseado em Vandewalle (1993, p. 26) para o método *Waveform Relaxation* com *singlegrid* (método de malha única).

Se considerar o uso do método *Waveform Relaxation* com *multigrid*, deve-se seguir a teoria descrita no capítulo 4. Vandewalle (1993) comenta que o princípio do método *multigrid* para problemas dependentes do tempo é essencialmente da mesma forma que para os métodos clássicos de relaxamento.

Algoritmo 7: Método *Waveform Relaxation* em uma malha.

Entre com os dados de entrada, condição inicial e condições de contornos.
while Não alcançar algum critério de parada **do**
 for $i = 1 : d$ **do**
 Resolva a EDO discretizada correspondente à posição espacial i no sistema de Eq. (7.4).
 end for
end while

Devido à filosofia do método *Waveform Relaxation*, no caso de se usar a técnica *multigrid*, a malha fina Ω^h e as malhas mais grossas $\Omega^{2h}, \Omega^{4h}, \dots, \Omega^{lh}$, com $l = 2^{(L_{\max}-1)}$, são simultaneamente discretizadas no espaço e aproximadas no tempo, porém, o processo de engrossamento das malhas ocorre somente nas direções espaciais. Por exemplo, considerando-se engrossamento padrão $q = 2$, a discretização das variáveis espaciais x, y e o número de passos no tempo t na malha Ω^h com $x \times y \times t$ com $33 \times 33 \times 100$ pontos, as malhas mais grossas $\Omega^{2h}, \Omega^{4h}, \Omega^{8h}$ e Ω^{16h} terão $17 \times 17 \times 100$, $9 \times 9 \times 100$, $5 \times 5 \times 100$ e $3 \times 3 \times 100$ pontos, respectivamente. Com isso, os valores dos resíduos são calculados e restritos e a correção é prolongada e suavizada em todos os pontos espaciais e passos de tempo, simultaneamente.

No caso da equação do calor, os suavizadores espaciais mais comuns nesse método são Gauss-Seidel (lexicográfico ou *red-black*), o método de Jacobi ponderado e os métodos de relaxação por linhas. Como já foi mencionado, os métodos de aproximação da derivada temporal podem ser os métodos de Euler implícito e de Crank-Nicolson, entre outros. Pode-se especificar o método *multigrid* para o esquema CS com ciclos $V(\nu_1, \nu_2)$, $F(\nu_1, \nu_2)$ ou $W(\nu_1, \nu_2)$ através do algoritmo 8 (VANDEWALLE, 1993; FALGOUT et al., 2017).

Algoritmo 8: Método *Waveform Relaxation - multigrid*

Entre com os dados de entrada, condição inicial e condições de contornos.
while Não alcançar algum critério de parada **do**
 Um ciclo *multigrid* com **MG-ciclo** (1), e considerando engrossamento apenas no espaço em todos os passos de tempo. Para o processo de suavização, em cada nível de malha use o método descrito no algoritmo 7.
end while

Para as equações da poroelasticidade, devido à presença de pontos de sela no sistema de equações, considerou-se o uso do suavizador Vanka de três pontos. Esse suavizador atualiza simultaneamente em cada passo de iteração um bloco de três incógnitas, na forma de Gauss-Seidel. Dessa maneira, todas as incógnitas do deslocamento são calculadas juntamente com a incógnita da pressão, posicionada no ponto central, de forma simultânea em cada passo de tempo. Esse cálculo se dá consecutivamente em todos os passos de tempo através do método *Waveform Relaxation*. Para o caso unidimensional, as incógnitas, u_{i-1} , p_i e u_{i+1} são resolvidas simultaneamente. Esse procedimento será chamado de suavização

Waveform Relaxation Vanka de três pontos, e a ideia geométrica está descrita na Fig. 28. No caso bidimensional, as incógnitas, $u_{i,j-1}$, $u_{i-1,j}$, $p_{i,j}$, $u_{i+1,j}$ e $u_{i,j+1}$ são resolvidas simultaneamente. Esse procedimento será chamado de suavização *Waveform Relaxation* Vanka de cinco pontos, e a ideia geométrica está descrita na Fig. 29.

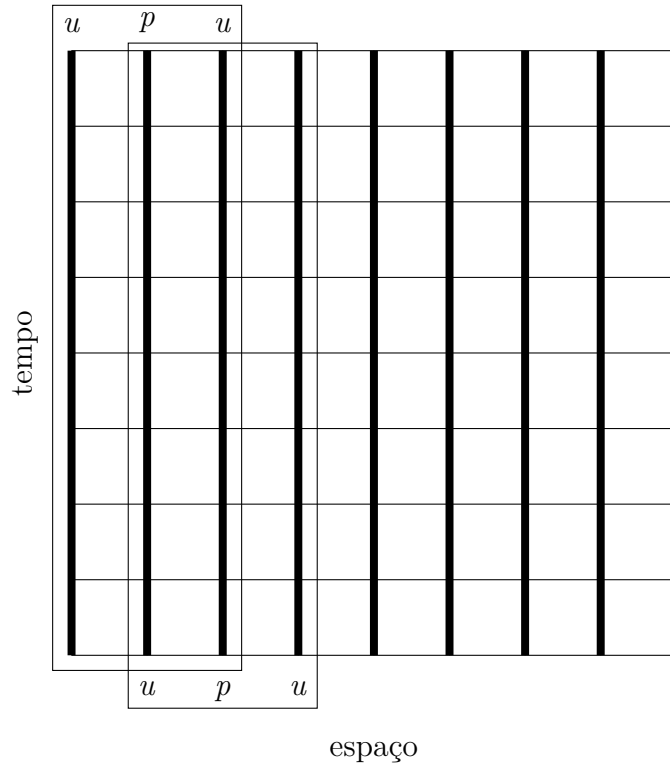


Figura 28 – Método *Waveform Relaxation* Vanka de três pontos, usada para o problema da poroelasticidade 1D.

O algoritmo do método *multigrid* com o suavizador Vanka para as equações da poroelasticidade 1D e 2D segue a mesma ideia do algoritmo mostrado para a equação do calor, algoritmo 8.

7.3 Método *Space-Time*

O *Space-Time* é um método que se caracteriza por usar um suavizador por pontos, como o Gauss-Seidel *red-black*, sendo que em cada iterada atualizam-se todos os pontos no espaço e no tempo. O algoritmo 9 descreve o uso desse método em uma malha (*singlegrid*).

Algoritmo 9: Método *Space-Time* em uma malha

Entre com os dados de entrada, condição inicial e condições de contornos.

while Não alcançar algum critério de parada **do**

 Suavize (ponto-a-ponto) o sistema correspondente em todos os pontos no espaço e no tempo.

end while

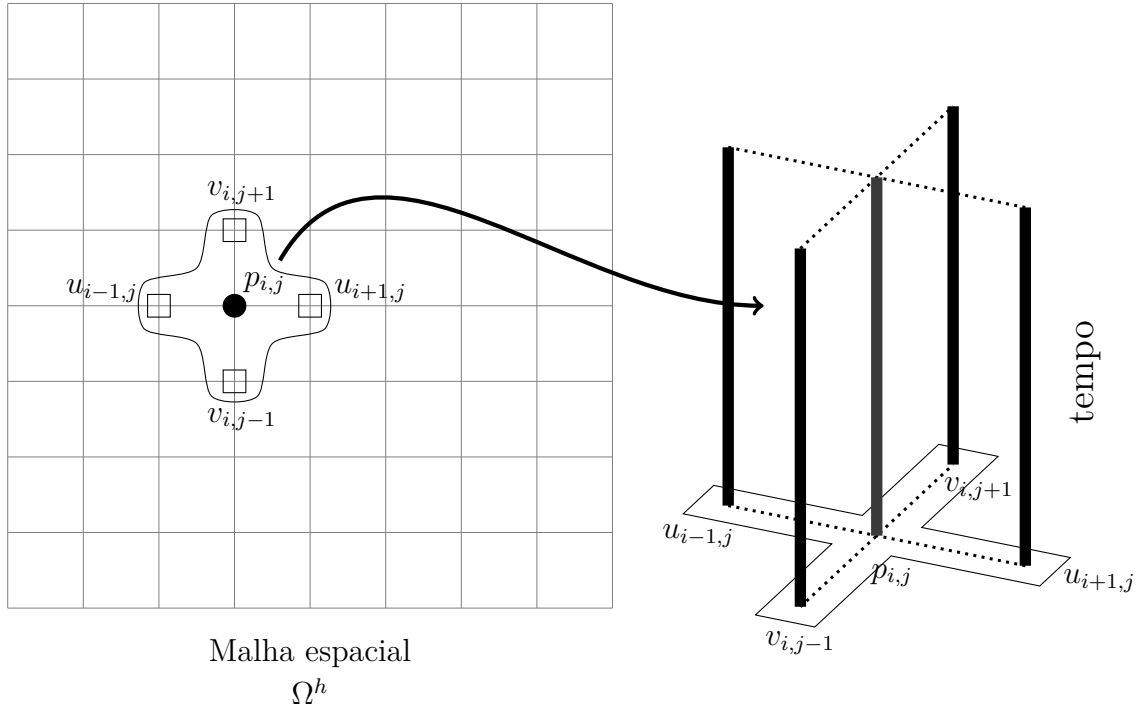


Figura 29 – Método *Waveform Relaxation* Vanka de cinco pontos, usada para o problema da poroelasticidade 2D.

Em particular, o suavizador frequentemente usado no algoritmo 9 é o Gauss-Seidel com ordenação *red-black* nas direções espaciais.

Com o método *multigrid*, Horton e Vandewalle (1995) comentam que o *Space-Time* se caracteriza por adotar uma estratégia de semiengrossamento no espaço ou no tempo, e que tal estratégia depende de um parâmetro chamado *fator de anisotropia* (λ_l) que relaciona o tamanho do passo temporal e espacial em cada nível de malha l . A razão de engrossamento usada para ambos os semiengrossamentos é $q = 2$. O valor de λ que define o momento do semiengrossamento espacial ou temporal é denominado λ_{crit} e em Horton e Vandewalle (1995) e no presente trabalho foi determinado através da LFA. Segundo Horton e Vandewalle (1995) e Falgout et al. (2017) seu valor é $\lambda_{crit} = 0,60$.

A estratégia de semiengrossamentos adotada no método *Space-Time* está descrita no algoritmo 10.

Algoritmo 10: Estratégia de engrossamento no método *Space-Time* - *multigrid*

```

if  $\lambda < \lambda_{crit}$  then
    Faça semiengrossamento na direção temporal.
else
    Faça semiengrossamento nas direções espaciais.
end if
    
```

Como $\lambda_l = \frac{\tau_l}{h_l^2}$, quando é usado semiengrossamento espacial ($h_{l+1} = 2h_l$), o valor

de λ_{l+1} na malha imediatamente mais grossa é $\lambda_{l+1} = \frac{1}{4}\lambda_l$. Se o engrossamento ocorrer na direção temporal ($\tau_{l+1} = 2\tau_l$), o valor de λ_{l+1} na malha imediatamente mais grossa é $\lambda_{l+1} = 2\lambda_l$. Dessa forma, pode-se observar que o valor de λ_l pode aumentar ou diminuir dependendo do tipo de semiengrossamento escolhido. Após restar apenas uma incógnita (ou 3 pontos) em uma direção, o engrossamento ocorrerá na outra direção até restar apenas uma incógnita ou 3 pontos naquela direção. Com isso, o número máximo de malhas possíveis nesse método é dado por $L_{\max} = nm_x + nm_t - 1$, em que nm_x representa o número máximo de malhas na direção espacial e nm_t é o número máximo de malhas na direção temporal.

Os operadores de prolongação (I_{2h}^h) e restrição (I_h^{2h}) são escolhidos de acordo com Horton e Vandewalle (1995). Para o caso 1D, se o engrossamento ocorrer na direção espacial, os operadores são dados por

$$I_{2h}^h = \frac{1}{2} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix}_{2h}^h, \quad I_h^{2h} = \frac{1}{4} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix}_h^{2h}. \quad (7.5)$$

e se o engrossamento ocorrer na direção temporal, os operadores são dados por

$$I_{2h}^h = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{2h}^h, \quad I_h^{2h} = \frac{1}{2} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}_h^{2h}. \quad (7.6)$$

Como esses estêncis são do tipo espaço/tempo e o problema é unidimensional, como já mencionado na seção 6.1.2, em cada linha representa-se a evolução no tempo (de baixo para cima) e em cada coluna, a discretização no espaço (da esquerda para a direita). Por isso, os estêncis descritos na Eq. (7.5) apresentam valores apenas na linha do meio, ou seja, apenas nas posições espaciais e no passo de tempo atual, enquanto que os estêncis descritos na Eq. (7.6) apresentam valores apenas na coluna do meio, ou seja, apenas nas posições temporal e para uma posição espacial. Observa-se que como não se usam valores relacionados a tempos futuros, os estêncis descritos na Eq. (7.6) têm uma linha nula.

Para o caso 2D, se o engrossamento ocorrer na direção espacial, os operadores são dados por

$$I_{2h}^h = \begin{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{2h}^h & \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_{2h}^h & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{2h}^h \end{bmatrix}, \quad (7.7)$$

$$I_h^{2h} = \begin{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_h^{2h} & \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h^{2h} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_h^{2h} \end{bmatrix},$$

e se o engrossamento ocorrer na direção temporal, os operadores são dados por

$$I_{2h}^h = \begin{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{2h} & \begin{bmatrix} 1 \\ 4 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{2h} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{2h} \end{bmatrix}, \quad (7.8)$$

$$I_h^{2h} = \frac{1}{2} \begin{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}_h & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}_h & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_h \end{bmatrix}^{2h},$$

Observa-se que a notação de estêncil nas Eq. (7.7) e Eq. (7.8) corresponde a uma sequência de estêncis aplicados em sucessivos passos de tempo, do mais baixo (estêncil esquerdo) para o mais alto (estêncil direito).

O fato desse método trabalhar com o espaço e o tempo simultaneamente possibilita o desenvolvimento de algoritmos paralelizáveis mais eficientes, diminuindo assim o tempo necessário para encontrar uma solução (PRIETO et al., 2001; FALGOUT et al., 2017).

O algoritmo 11 expressa os ciclos $V(\nu_1, \nu_2)$, $W(\nu_1, \nu_2)$ ou $F(\nu_1, \nu_2)$ para o método *Space-Time - multigrid*.

Algoritmo 11: Método *Space-Time - multigrid*

Entre com os dados de entrada, condição inicial e condições de contornos.

while Não alcançar algum critério de parada **do**

Um ciclo *multigrid* com **MG-ciclo** (1) e respeitando a estratégia de engrossamento descrita no algoritmo 10. Caso não seja possível engrossar em uma direção, utilize engrossamento em outra direção até se obter a malha mais grossa possível.

end while

8 Verificação dos Códigos Computacionais

A verificação dos códigos foi realizada com simulações numéricas e análise *a posteriori* dos resultados, considerando-se os seguintes testes:

1. Verificar se os erros numéricos entre a solução aproximada e a solução analítica são iguais entre os métodos *singlegrid* e *multigrid*;
2. Mostrar graficamente os comportamentos do tempo computacional (t_{CPU}) *versus* tamanho do problema (\mathcal{N}) para os métodos *singlegrid* e *multigrid*;
3. Calcular a complexidade dos métodos (ou algoritmos);
4. Verificar as ordens aparente p_U e efetiva p_E dos erros de discretização de certas variáveis de interesse e se convergem para as suas ordens assintóticas p_l quando h tende a 0.

A verificação das soluções numéricas, de acordo com Marchi (2001), permite fazer uma análise das ordens efetiva (p_E), aparente (p_U) e assintótica (p_L) do método de aproximação numérica usado. Para isso, usou-se o estimador de Richardson baseado na ordem aparente do erro numérico, dada por

$$p_U = \frac{\log\left(\frac{\phi_2 - \phi_3}{\phi_1 - \phi_2}\right)}{\log(q)}, \quad (8.1)$$

quando não se conhece a solução analítica, e na ordem efetiva, dada por

$$p_E = \frac{\log\left(\frac{e(\phi_2)}{e(\phi_1)}\right)}{\log(q)}, \quad (8.2)$$

em que ϕ_1 , ϕ_2 e ϕ_3 indicam a solução da variável de interesse nas malhas fina, grossa e super grossa, respectivamente. A variável q indica a razão de engrossamento entre os níveis de malhas relacionados à ϕ_1 , ϕ_2 e ϕ_3 . A variável e indica o erro numérico (erro de arredondamento, de truncamento e iteração) da variável ϕ relacionada aos cálculos.

Para essas simulações o processo iterativo é interrompido quando se atinge o erro de máquina, ou seja, são minimizados os erros de iteração, prevalecendo basicamente os erros de discretização.

Para determinar o efeito do número de incógnitas em relação ao t_{CPU} , chamado de complexidade do método (ou algoritmo), fez-se um ajuste não linear do tipo (BURDEN; FAIRES, 2016)

$$t_{cpu}(\mathcal{N}) = c \cdot \mathcal{N}^p, \quad (8.3)$$

em que \mathcal{N} representa o número total de incógnitas do problema a ser resolvido, p representa a ordem do algoritmo (inclinação da curva em escala bilogarítmica) e c é uma constante que depende do método. No método *multigrid* quanto mais próximo da unidade estiver o expoente p , melhor o seu desempenho; $p = 1$ é considerado o caso ideal. No método *singlegrid* o valor teórico é $p = 2$ (TROTTEMBERG et al., 2001).

Nas simulações envolvendo o tempo computacional (t_{CPU}), o processo iterativo é interrompido quando se reduz a norma $\|\cdot\|_\infty$ do resíduo em 10^{-10} , ou seja, se $\frac{\|r(it)\|_\infty}{\|r(0)\|_\infty} \leq 10^{-10}$.

8.1 Equação do calor

Para os testes de verificação considerou-se o método *Waveform Relaxation* com os suavizadores zebra com linha no tempo para os casos *singlegrid* e *multigrid*. Essa ordenação (colorida) permite desenvolver algoritmos paralelizáveis. As discretizações espaciais e aproximações no tempo foram feitas de acordo com o desenvolvimento descrito na seção 6.1.2. Para as simulações numéricas com *multigrid* usou-se o ciclo $W(1, 1)$.

Os resultados usando outros métodos, como o método *Time-Stepping* e o método *Space-Time* com o suavizador Gauss-Seidel *red-black*, por exemplo, fornecem resultados similares e não serão expostos aqui.

8.1.1 Problema unidimensional

O primeiro teste para verificar o código computacional é a análise do erro numérico. Para isso, fizeram-se simulações numéricas considerando os métodos *singlegrid* e *multigrid* até alcançar o erro de máquina. Na Tab. 2 mostra-se a norma $\|\cdot\|_\infty$ dos erros numéricos considerando os métodos de Euler e de Crank-Nicolson para a aproximação temporal e o tempo final sendo $t_f = 10^{-5}$ ou $t_f = 1$ em relação ao número de pontos nas discretizações espacial e temporal dada por $N = N_x = N_t$.

Os resultados mostrados na Tab. 2 mostram que os erros numéricos relacionados aos métodos *singlegrid* e *multigrid* são iguais, exceto nas malhas mais refinadas (neste caso, $N \geq 2^9 + 1$), em que começa a ocorrer a incidência de erros de arredondamento.

Para verificar a complexidade dos algoritmos considerou-se o t_{CPU} em função do número de incógnitas como mostra a Fig. 30. Verifica-se nessa figura que o método *multigrid* possui um t_{CPU} muito inferior com relação ao método *singlegrid*, o que pode ser comprovando com o cálculo do *speed-up*, sendo $speed-up = \frac{t_{CPU} \text{ do método } singlegrid}{t_{CPU} \text{ do método } multigrid}$ e é mostrado na Fig. 31.

A fim de fazer um ajuste não linear dado pela Eq. (8.3), usaram-se os dados descritos na Fig. 30, utilizando-se os 6 últimos pontos, no caso *singlegrid*, e os últimos 8

Tabela 2 – Valor da $\|\cdot\|_\infty$ dos erros numéricos para a equação do calor 1D, usando o método *Waveform Relaxation* para os casos *singlegrid* e *multigrid*.

N	tipo	$t_f = 1,0E - 5$		$t_f = 1,0$	
		Euler	CN	Euler	CN
$2^2 + 1$	<i>singlegrid</i>	4,96990E-6	4,96996E-6	3,97453E-2	5,03607E-2
	<i>multigrid</i>	4,96990E-6	4,96996E-6	3,97453E-2	5,03607E-2
$2^3 + 1$	<i>singlegrid</i>	1,26178E-6	1,26178E-6	1,31349E-2	1,02786E-2
	<i>multigrid</i>	1,26178E-6	1,26178E-6	1,31349E-2	1,02786E-2
$2^4 + 1$	<i>singlegrid</i>	3,16665E-7	3,16663E-7	4,62482E-3	2,48260E-3
	<i>multigrid</i>	3,16665E-7	3,16663E-7	4,62482E-3	2,48260E-3
$2^5 + 1$	<i>singlegrid</i>	7,92434E-8	7,92420E-8	1,78462E-3	6,15558E-4
	<i>multigrid</i>	7,92434E-8	7,92420E-8	1,78462E-3	6,15558E-4
$2^6 + 1$	<i>singlegrid</i>	1,98160E-8	1,98153E-8	7,54140E-4	1,53579E-4
	<i>multigrid</i>	1,98160E-8	1,98153E-8	7,54140E-4	1,53579E-4
$2^7 + 1$	<i>singlegrid</i>	4,95450E-9	4,95412E-9	3,41743E-4	3,83871E-5
	<i>multigrid</i>	4,95451E-9	4,95412E-9	3,41743E-4	3,83871E-5
$2^8 + 1$	<i>singlegrid</i>	1,23875E-9	1,23854E-9	1,61912E-4	9,59561E-6
	<i>multigrid</i>	1,23875E-9	1,23855E-9	1,61912E-4	9,59561E-6
$2^9 + 1$	<i>singlegrid</i>	3,09758E-10	3,09652E-10	7,87012E-5	2,39883E-6
	<i>multigrid</i>	3,09743E-10	3,09647E-10	7,87012E-5	2,39883E-6
$2^{10} + 1$	<i>singlegrid</i>	7,73979E-11	7,74579E-11	3,87849E-5	5,99698E-7
	<i>multigrid</i>	7,74618E-11	7,74258E-11	3,87850E-5	5,99703E-7
$2^{11} + 1$	<i>singlegrid</i>	1,94050E-11	1,92295E-11	1,92508E-5	1,49900E-7
	<i>multigrid</i>	1,93485E-11	1,93596E-11	1,92508E-5	1,49925E-7

pontos, no caso *multigrid*, pois neste trecho os valores são mais representativos. A Tab. 3 mostra os valores de c e p para os métodos *singlegrid* e *multigrid*. Com esses resultados pode-se observar que os valores encontrados são próximos dos descritos na literatura, na qual para *singlegrid* tem-se que $p = 2$ e para *multigrid* tem-se que $p = 1$.

Tabela 3 – Coeficientes c e p de $t_{cpu}(\mathcal{N}) = c \cdot \mathcal{N}^p$ considerando a equação do calor 1D, os métodos de Euler e de CN com *singlegrid* e *multigrid*.

<i>singlegrid</i>				<i>multigrid</i>			
Euler		CN		Euler		CN	
c	p	c	p	c	p	c	p
6,661E-9	2,06652	8,271E-9	2,03909	3,566E-7	1,05401	6,070E-7	1,07870

A análise das ordens aparente e efetiva foi feita para 3 variáveis de interesse: 1) uma variável que verifica apenas um ponto, a temperatura no ponto médio no passo de tempo final, $u(0,5;t_f)$ com $t_f = 1,0$; 2) uma variável que engloba todos os pontos no passo de tempo final, a temperatura média no passo de tempo final, $u_m(x,t_f)$; e 3) uma variável que engloba todas as variáveis em todos os passos de tempo, a norma l_∞ do erro numérico em todo o domínio, $\|e\|_\infty$.

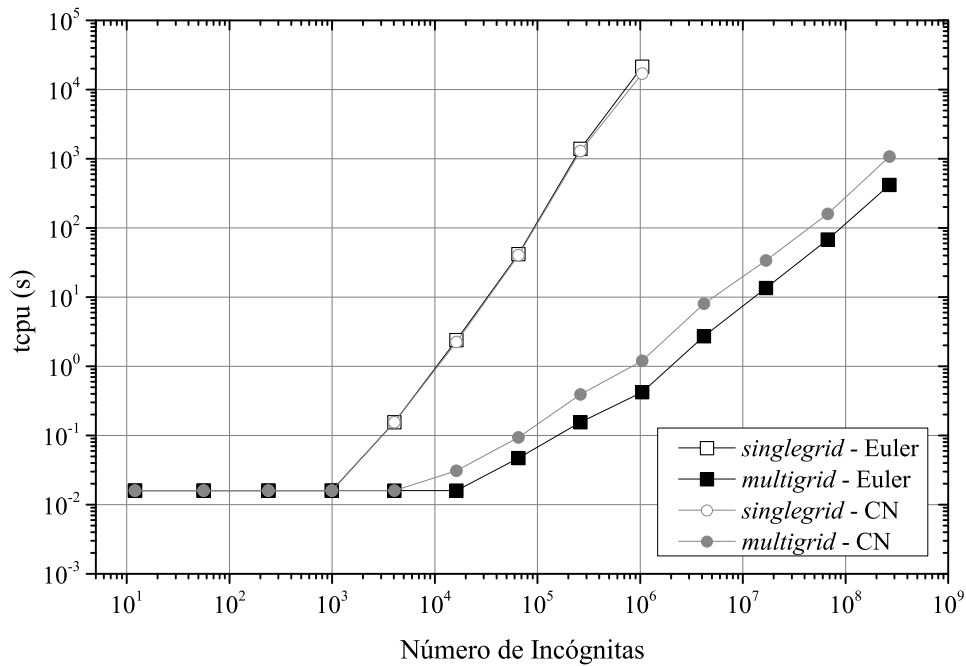


Figura 30 – Tempo computacional usado para resolver a equação do calor 1D com os métodos *singlegrid* e *multigrid* e com aproximações temporal usando Euler e CN.

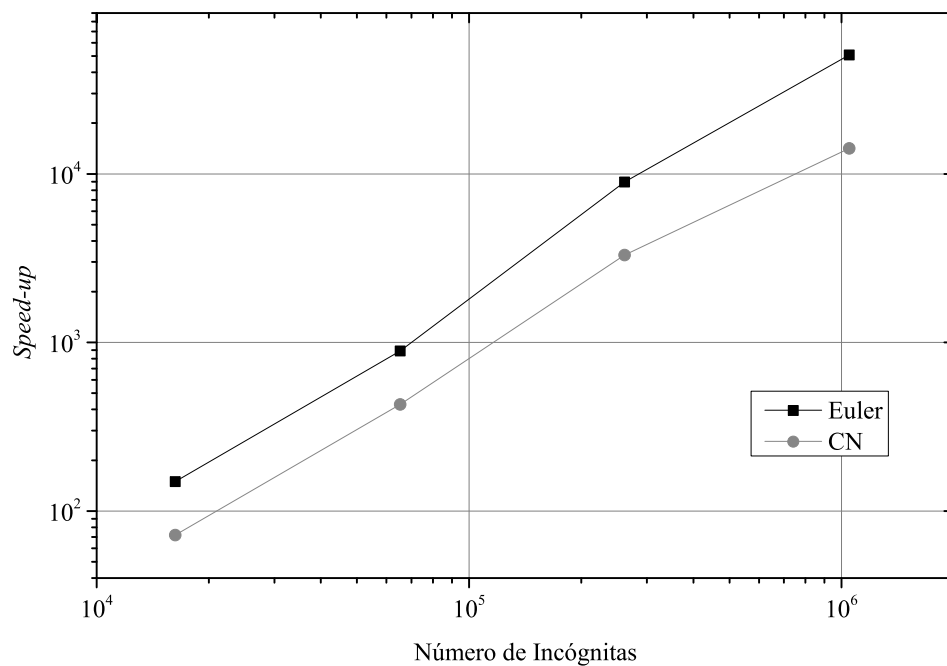


Figura 31 – *Speed-up* dos métodos *singlegrid* e *multigrid* para as aproximações temporal usando Euler e CN.

Com os resultados descritos nas Fig. 32 e Fig. 33, verifica-se que em relação a todas as variáveis de interesse consideradas, quando ocorre um refino da malha, as ordens efetiva (p_E) e aparente (p_U) tendem ao valor da ordem assintótica $p_L = 1,0$ para o método de

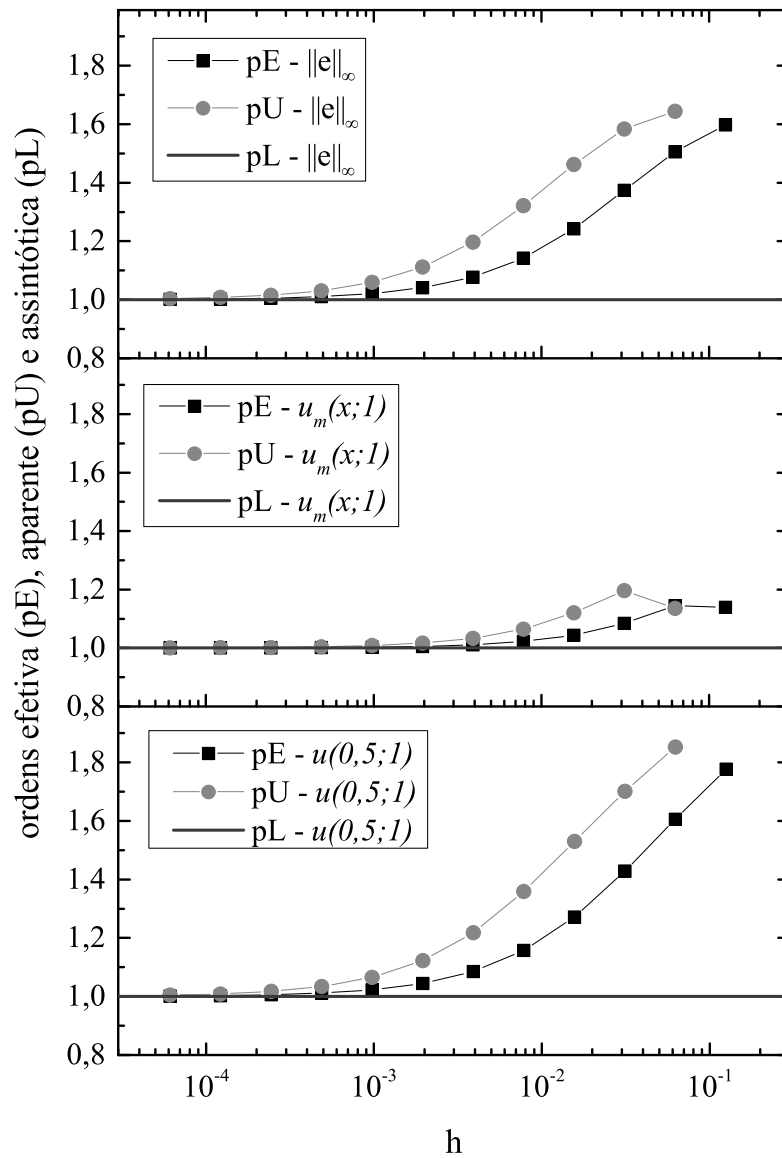


Figura 32 – Ordens efetiva, aparente e assintótica considerando a equação do calor 1D e o método de Euler.

Euler e $p_L = 2,0$ para o método de CN.

Esses testes numéricos mostram a coerência entre os resultados numéricos e os analíticos, indicando a validade do código utilizado para resolver a equação do calor 1D.

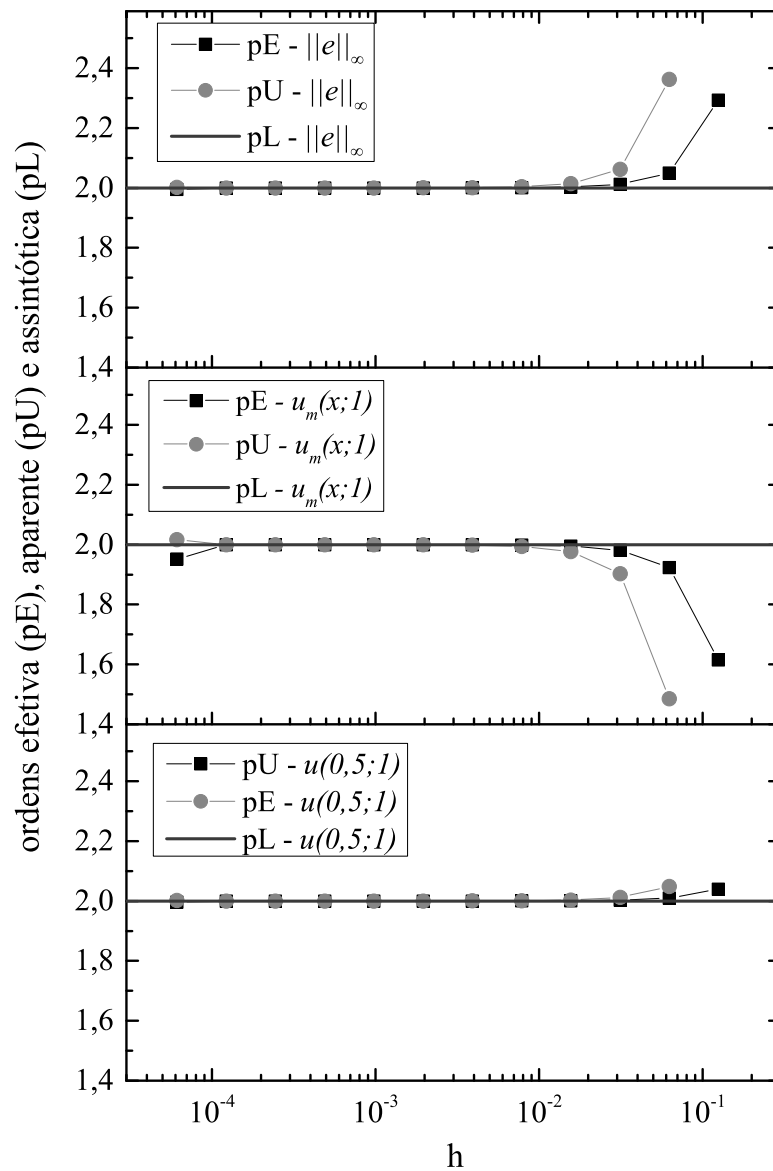


Figura 33 – Ordens efetiva, aparente e assintótica considerando a equação do calor 1D e o método de CN.

8.1.2 Problema bidimensional

Assim como no caso unidimensional, o primeiro teste para verificar o código computacional é a análise do erro numérico. A Tab. 4 mostra a norma $\|\cdot\|_\infty$ dos erros numéricos considerando os métodos de Euler e de Crank-Nicolson para a aproximação temporal e os tempos final sendo $t_f = 10^{-5}$ ou $t_f = 1$ em relação ao número de pontos nas discretizações espaciais e temporal dada por $N = N_x = N_y = N_t$.

Os resultados mostrados na Tab. 4 mostram que os erros numéricos relacionados aos métodos *singlegrid* e *multigrid* são iguais.

O t_{CPU} em função do número de incógnitas e o *speed-up* entre os metodos *singlegrid* e *multigrid* são mostrados nas Fig. 34 e Fig. 35, respectivamente. Nessas figuras pode-se

Tabela 4 – Valor da $\| \cdot \|_{\infty}$ dos erros numéricos para a equação do calor 2D, usando o método *Waveform Relaxation* para os caso *singlegrid* e *multigrid*.

N	tipo	$t_f = 1,0E - 5$		$t_f = 1,0$	
		Euler	CN	Euler	CN
$2^2 + 1$	<i>singlegrid</i>	9,93921E-6	9,93945E-6	3,86965E-2	6,57688E-2
	<i>multigrid</i>	9,93921E-6	9,93945E-6	3,86965E-2	6,57688E-2
$2^3 + 1$	<i>singlegrid</i>	2,52341E-6	2,52343E-6	1,19649E-2	1,33186E-2
	<i>multigrid</i>	2,52341E-6	2,52343E-6	1,19649E-2	1,33186E-2
$2^4 + 1$	<i>singlegrid</i>	6,33293E-7	6,33294E-7	3,77446E-3	2,78551E-3
	<i>multigrid</i>	6,33293E-7	6,33294E-7	3,77446E-3	2,78551E-3
$2^5 + 1$	<i>singlegrid</i>	1,58477E-7	1,58476E-7	1,30722E-3	6,85490E-4
	<i>multigrid</i>	1,58477E-7	1,58476E-7	1,30722E-3	6,85490E-4
$2^6 + 1$	<i>singlegrid</i>	3,96293E-8	3,96286E-8	4,98529E-4	1,70610E-4
	<i>multigrid</i>	3,96293E-8	3,96286E-8	4,98529E-4	1,70610E-4
$2^7 + 1$	<i>singlegrid</i>	9,90813E-9	9,90774E-9	2,09418E-4	4,26052E-5
	<i>multigrid</i>	9,90813E-9	9,90774E-9	2,09418E-4	4,26052E-5

observar a eficiência do método *multigrid*.

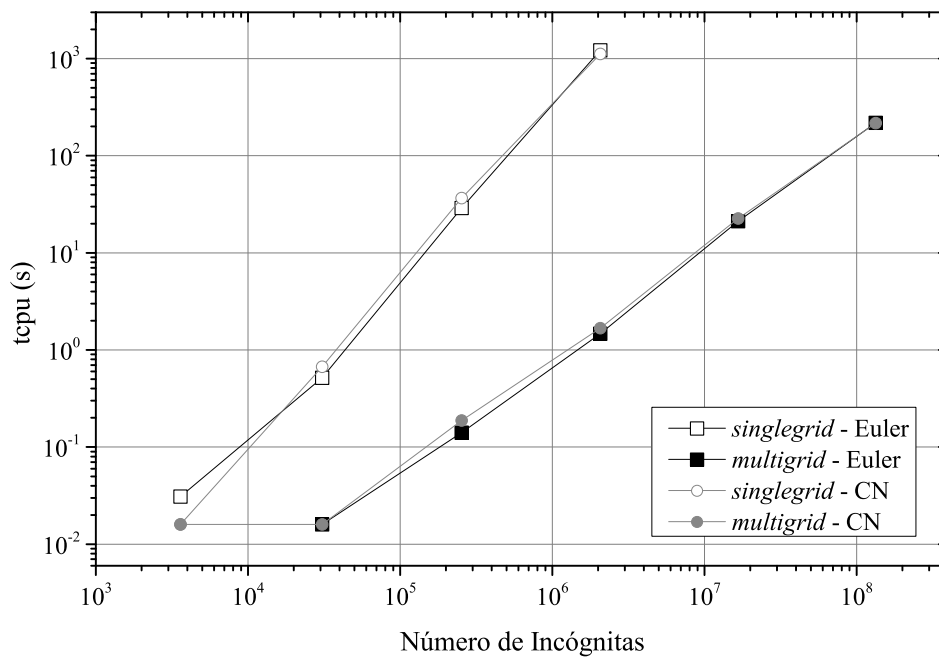


Figura 34 – Tempo computacional usado para resolver a equação do calor 2D com os métodos *singlegrid* e *multigrid* e com aproximações temporal usando Euler e CN.

Para calcular a complexidade do algoritmo, assim como no caso unidimensional, fez-se uma aproximação não linear dada pela Eq. (8.3), usaram-se os dados descritos na Fig. 34, utilizando-se os 4 últimos pontos, no caso *singlegrid*, e os últimos 5 pontos, no caso *multigrid*, por motivos já alegados. A Tab. 5 mostra os valores de c e p para os métodos

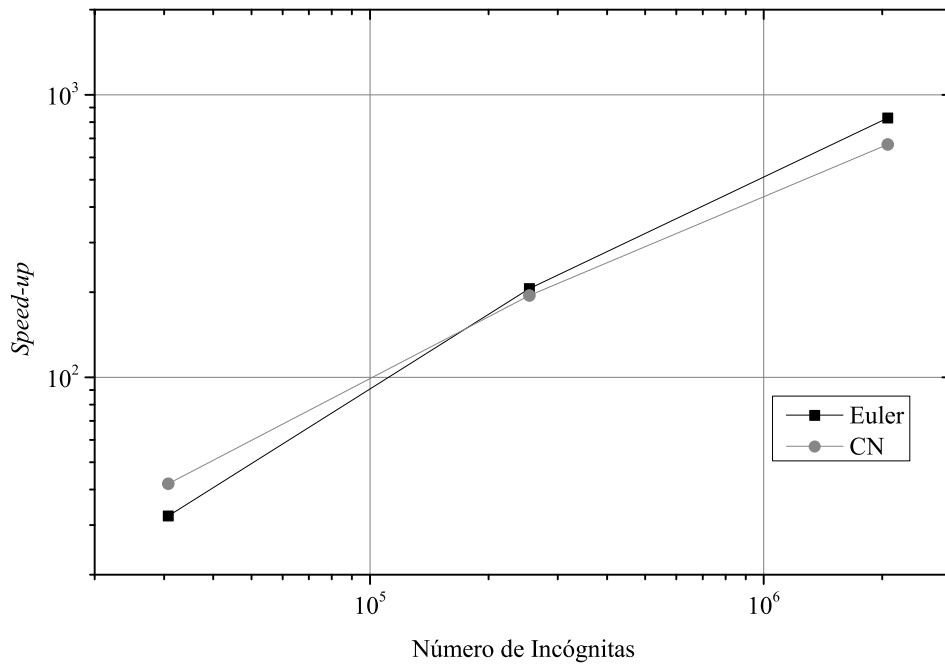


Figura 35 – *Speed-up* entre os métodos *singlegrid* e *multigrid* para as aproximações temporal usando Euler e CN aplicados na equação do calor 2D.

singlegrid e *multigrid*. Com esses resultados pode-se observar que os valores encontrados se aproximam dos valores teóricos.

Tabela 5 – Coeficientes c e p de $t_{cpu}(\mathcal{N}) = c \cdot \mathcal{N}^p$ considerando a equação do calor 2D, os métodos de Euler e de CN com *singlegrid* e *multigrid*.

<i>singlegrid</i>				<i>multigrid</i>			
Euler		CN		Euler		CN	
c	p	c	p	c	p	c	p
2,208E-8	1,68871	8,264E-9	1,76959	9,717E-8	1,14872	1,260E-7	1,13734

A análise das ordens aparente e efetiva no caso bidimensional também foi feita usando o método *multigrid* e para 3 variáveis de interesse: 1) uma variável que verifica apenas um ponto, a temperatura no ponto médio no passo de tempo final, $u(0, 5; 0, 5; t_f)$ com $t_f = 1, 0$; 2) uma variável que engloba todos os pontos no passo de tempo final, a temperatura média no passo de tempo final, $u_m(x, y, t_f)$; e 3) uma variável que engloba todas as variáveis em todos os passos de tempo, a norma l_∞ do erro numérico em todo o domínio, $\|e\|_\infty$. Os resultados são mostrados nas Fig. 36 e Fig. 37.

Com os resultados descritos nas Fig. 36 e Fig. 37, verifica-se que em relação a todas as variáveis de interesse consideradas, quando ocorre um refino da malha, as ordens efetiva (p_E) e aparente (p_U) tendem ao valor da ordem assintótica $p_L = 1, 0$ para o método de Euler e $p_L = 2, 0$ para o método de CN, concordando com os valores descritos na literatura.

Esses testes numéricos mostram a coerência entre os resultados numéricos e o

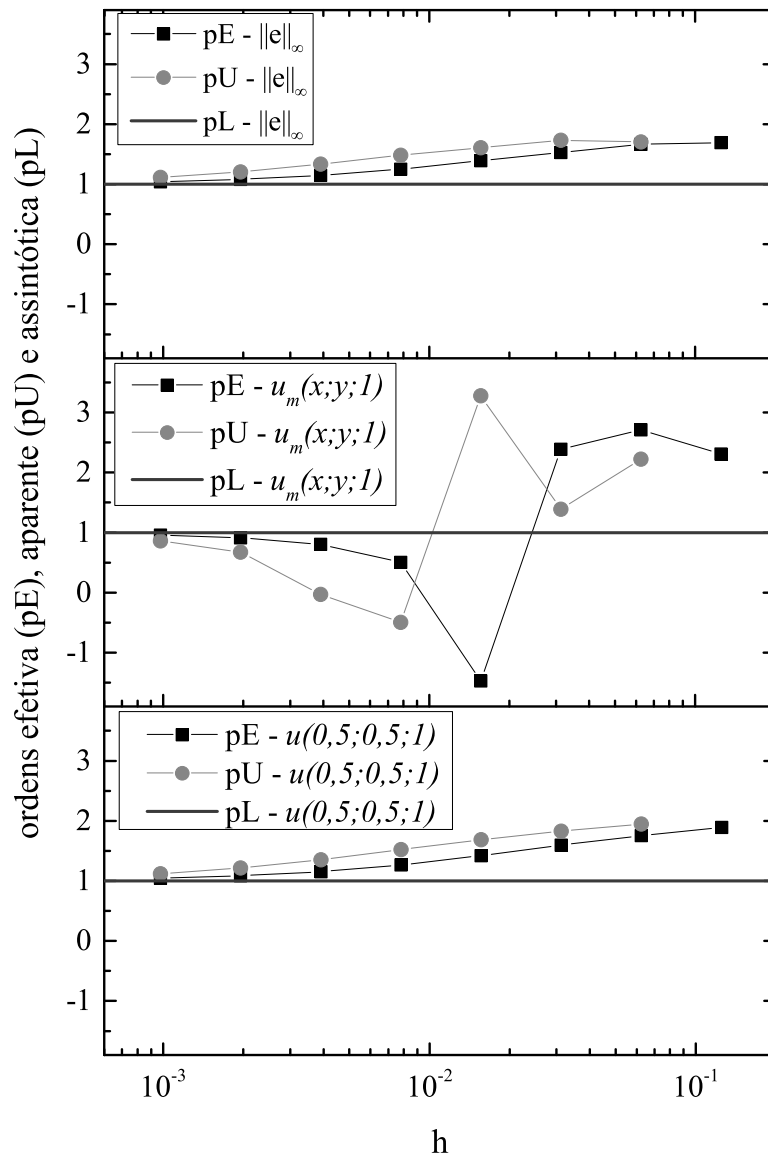


Figura 36 – Ordens efetiva, aparente e assintótica considerando a equação do calor 2D e o método de Euler.

analítico, ajudando a validar o código numérico utilizado para resolver a equação do calor 2D.

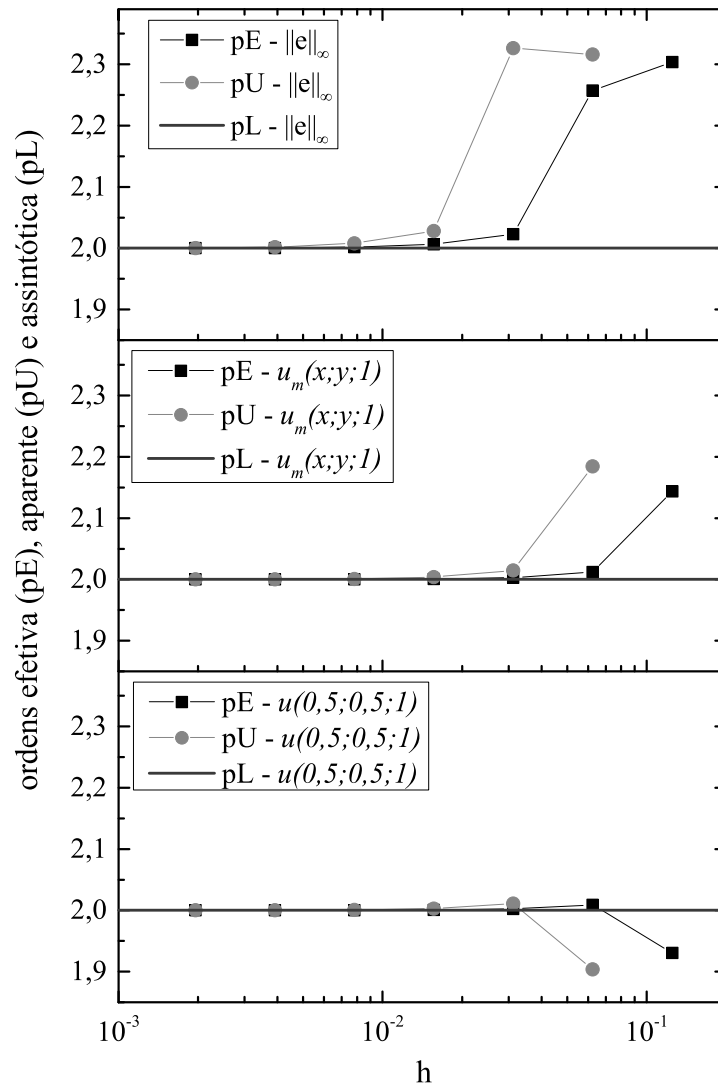


Figura 37 – Ordens efetiva, aparente e assintótica considerando a equação do calor 2D e o método de CN.

8.2 Equações da poroelasticidade

Para os testes de verificação consideraram-se os suavizadores Vanka de 3 pontos com ordenação de 4 cores para o caso 1D e Vanka de 5 pontos com ordenação de 12 cores para o caso 2D, ambos com linhas no tempo (zebras). Observa-se que o fato de usar no método *singlegrid* um suavizador zebra com linha no tempo, implica na utilização do método *Waveform Relaxation*. No caso *multigrid*, utilizou-se também o método *Waveform Relaxation* com ciclo $W(1,1)$. As discretizações espaciais e aproximações no tempo foram feitas de acordo com o desenvolvimento descrito na seção 6.2.2. O valor da condutividade hidráulica (para esses testes) foi escolhida bem pequena, com valor de $K = 10^{-9}$ que pode representar as características físicas de um meio poroso realístico, e o tempo final considerado foi $t_f = 1,0$.

8.2.1 Problema unidimensional

Nas simulações do problema 1D considerou-se o módulo de Young $E = 1$. Assim como nos testes para a equação do calor, o primeiro teste para verificar o código computacional é a análise do erro numérico. Para isso, fizeram-se simulações numéricas considerando os métodos *singlegrid* e *multigrid* até alcançar o erro de máquina. Na Tab. 6 mostra-se a norma $\|\cdot\|_\infty$ dos erros numéricos para o deslocamento u e pressão p , considerando a aproximação temporal dada pelo método de Euler e pelo método de CN. O número de pontos nas discretizações espacial e temporal é dada por $N = N_x = N_t$.

Tabela 6 – Valor da $\|\cdot\|_\infty$ dos erros numéricos para a equação da poroelasticidade 1D, usando o método *Waveform Relaxation* e o suavizador Vanka de 3 pontos com ordenação de 4 cores e linha no tempo para os métodos *singlegrid* e *multigrid*.

N	tipo	Euler		CN	
		$\ e(u)\ _\infty$	$\ e(p)\ _\infty$	$\ e(u)\ _\infty$	$\ e(p)\ _\infty$
$2^2 + 1$	<i>singlegrid</i>	5,15733E-2	1,39346E-1	2,15553E-2	9,33929E-2
	<i>multigrid</i>	5,15733E-2	1,39346E-1	2,15553E-2	9,33929E-2
$2^3 + 1$	<i>singlegrid</i>	3,26112E-2	9,66748E-2	5,53909E-3	2,35644E-2
	<i>multigrid</i>	3,26112E-2	9,66748E-2	5,53909E-3	2,35644E-2
$2^4 + 1$	<i>singlegrid</i>	1,80353E-2	5,51840E-2	1,39422E-3	5,90467E-3
	<i>multigrid</i>	1,80353E-2	5,51840E-2	1,39422E-3	5,90467E-3
$2^5 + 1$	<i>singlegrid</i>	9,44861E-3	2,93097E-2	3,49146E-4	1,47702E-3
	<i>multigrid</i>	9,44861E-3	2,93097E-2	3,49146E-4	1,47702E-3
$2^6 + 1$	<i>singlegrid</i>	4,83158E-3	1,50847E-2	8,73235E-5	3,69307E-4
	<i>multigrid</i>	4,83158E-3	1,50847E-2	8,73235E-5	3,69307E-4
$2^7 + 1$	<i>singlegrid</i>	2,44253E-3	7,64981E-3	2,18332E-5	9,23301E-5
	<i>multigrid</i>	2,44253E-3	7,64981E-3	2,18332E-5	9,23301E-5
$2^8 + 1$	<i>singlegrid</i>	1,22794E-3	3,85177E-3	5,45844E-6	2,30827E-5
	<i>multigrid</i>	1,22794E-3	3,85177E-3	5,45844E-6	2,30827E-5
$2^9 + 1$	<i>singlegrid</i>	6,15609E-4	1,94758E-3	1,39462E-6	1,67690E-5
	<i>multigrid</i>	6,15639E-4	1,93260E-3	1,36462E-6	5,77070E-6

Observa-se que os erros numéricos referentes aos métodos *singlegrid* e *multigrid* são iguais, exceto na malha mais refinada ($N = 2^9 + 1$) em que ocorreu uma pequena influência de erros de arredondamento.

As Fig. 38 e Fig. 39 mostram o t_{CPU} em função do número de incógnitas e o *speed-up* entre os metodos *singlegrid* e *multigrid*, respectivamente.

Com o cálculo do *speed-up* descrito na Fig. 39 pode-se observar a eficiência do método *multigrid* para resolver esse tipo de problema.

Para calcular a complexidade do algoritmo, assim como para a equação do calor, fez-se um ajuste não linear dado pela Eq. (8.3) e usaram-se os dados descritos na Fig. 38. A Tab. 7 mostra os valores de c e p para os métodos *singlegrid* e *multigrid*. Com esses resultados pode-se observar que os valores encontrados são próximos dos teóricos.

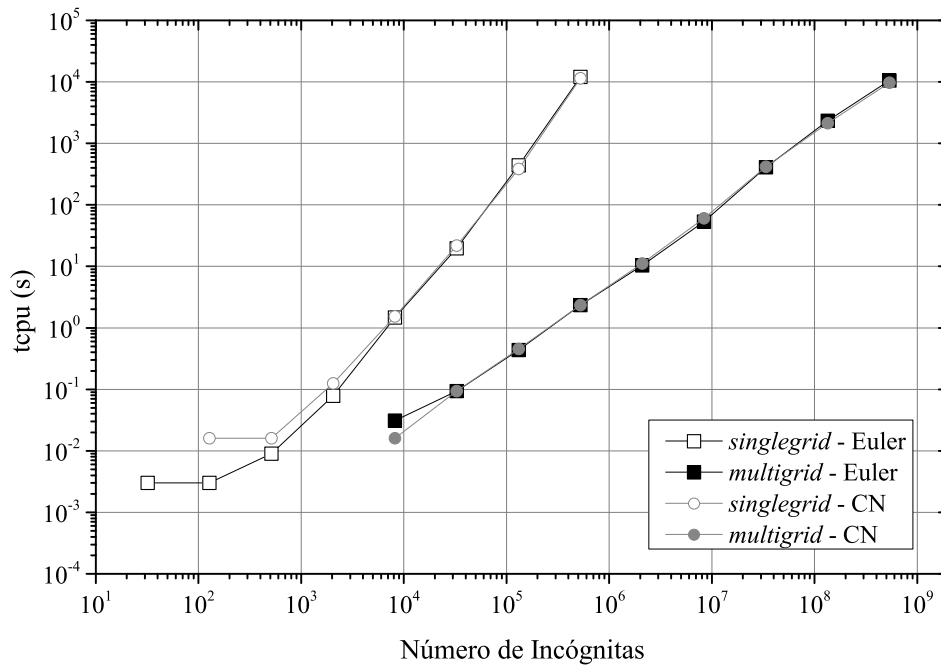


Figura 38 – Tempo computacional usado para resolver a equação da poroelasticidade 1D com os métodos *singlegrid* e *multigrid* e com aproximações temporal usando Euler e CN. $tol = 10^{-6}$

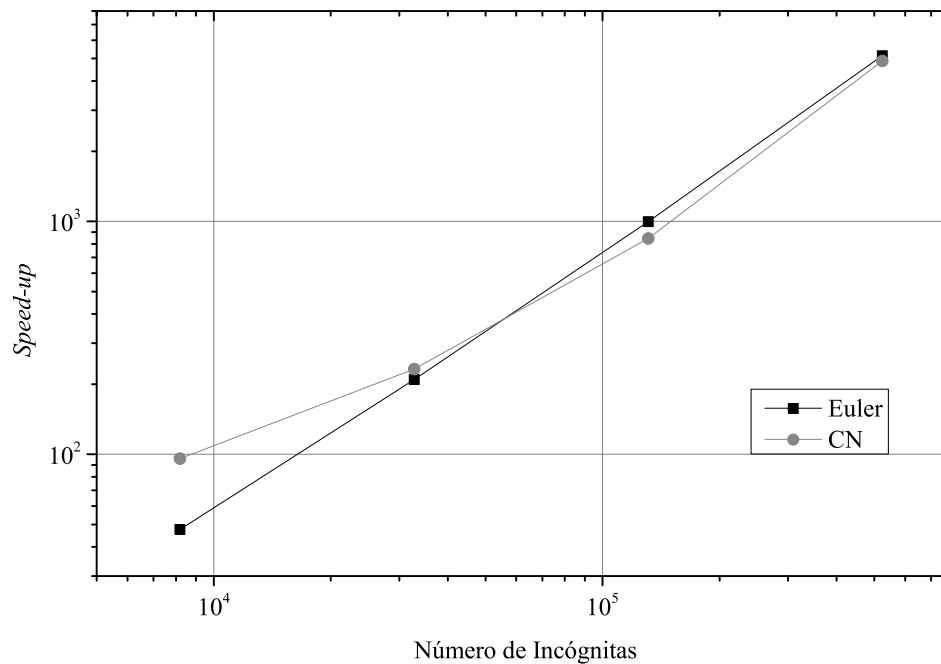


Figura 39 – *Speed-up* entre os métodos *singlegrid* e *multigrid* para as aproximações temporal usando Euler e CN.

A análise das ordens aparente e efetiva foi feita usando o método *multigrid* e com 3 variáveis de interesse: 1) uma variável que verifica apenas um ponto, o deslocamento e a pressão no ponto médio e passo de tempo final, $u(0, 25; t_f)$ e $p(0, 25; t_f)$ com $t_f = 1, 0$,

Tabela 7 – Coeficientes c e p de $t_{cpu}(\mathcal{N}) = c \cdot \mathcal{N}^p$ considerando a equação da poroelasticidade 1D, os métodos de Euler e de CN com *singlegrid* e *multigrid*.

<i>singlegrid</i>				<i>multigrid</i>			
Euler		CN		Euler		CN	
c	p	c	p	c	p	c	p
3,439E-9	2,18051	5,12E-9	2,1438	4,6879E-7	1,18051	3,0E-7	1,20669

respectivamente; 2) uma variável que engloba todos os pontos no passo de tempo final, o deslocamento médio e a pressão média no passo de tempo final, $u_m(x, t_f)$ e $p_m(x, t_f)$, respectivamente; 3) uma variável que engloba todas as variáveis em todos os passos de tempo, a norma infinita do erro numérico dos deslocamentos e das pressões $\|e(u)\|_\infty$ e $\|e(p)\|_\infty$, respectivamente.

Nas Fig. 40 e Fig. 41 é considerado o método de Euler e os resultados são expressos considerando as variáveis u e p , respectivamente. Enquanto isso, nas Fig. 42 e Fig. 43 é considerado o método de CN e os resultados também são expressos considerando as variáveis u e p , respectivamente.

Com os resultados mostrados nas Fig. 40 a Fig. 43 pode se observar a concordância entre os resultados numéricos e teóricos. Observa-se na Fig. 43 que os resultados referentes à malha mais refinada, ou seja, com o menor valor de h , são um pouco dispersos da reta que indica a ordem assintótica; isso deve-se ao fato de que para valores muito pequenos de h ocorre um aumento nos erros de arredondamento.

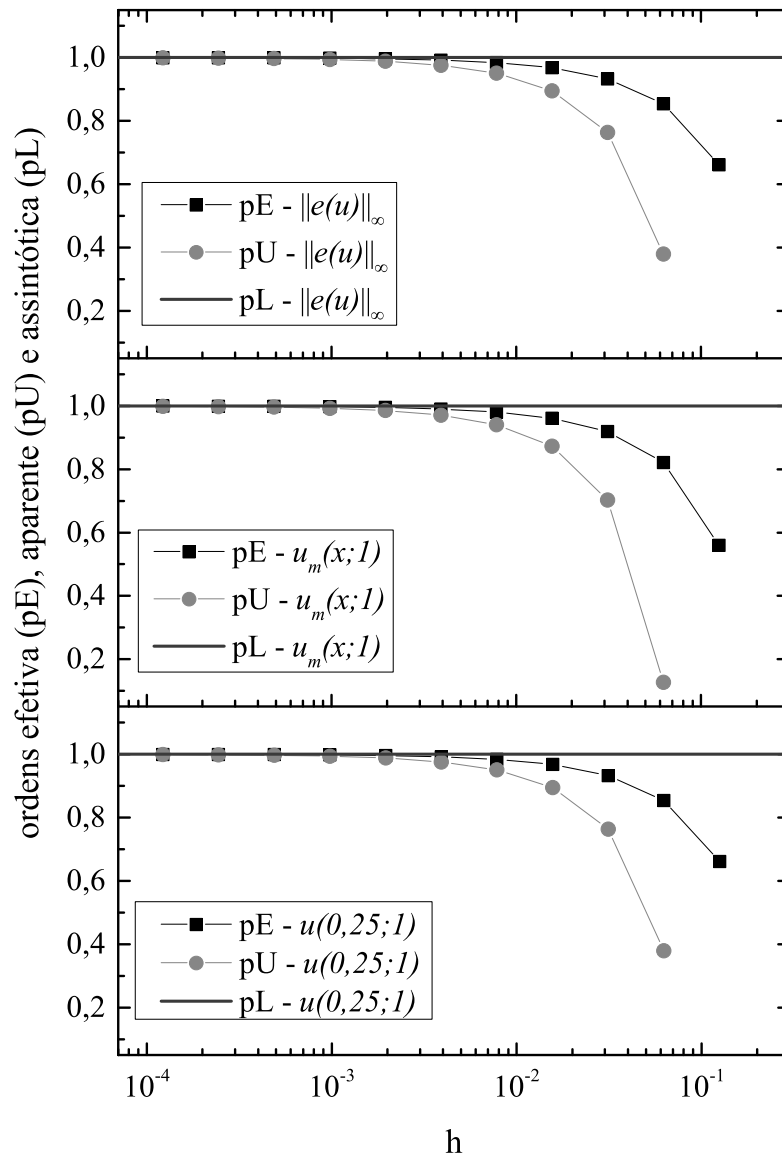


Figura 40 – Ordens efetiva, aparente e assintótica considerando a variável u da equação da poroelasticidade 1D e o método de Euler.

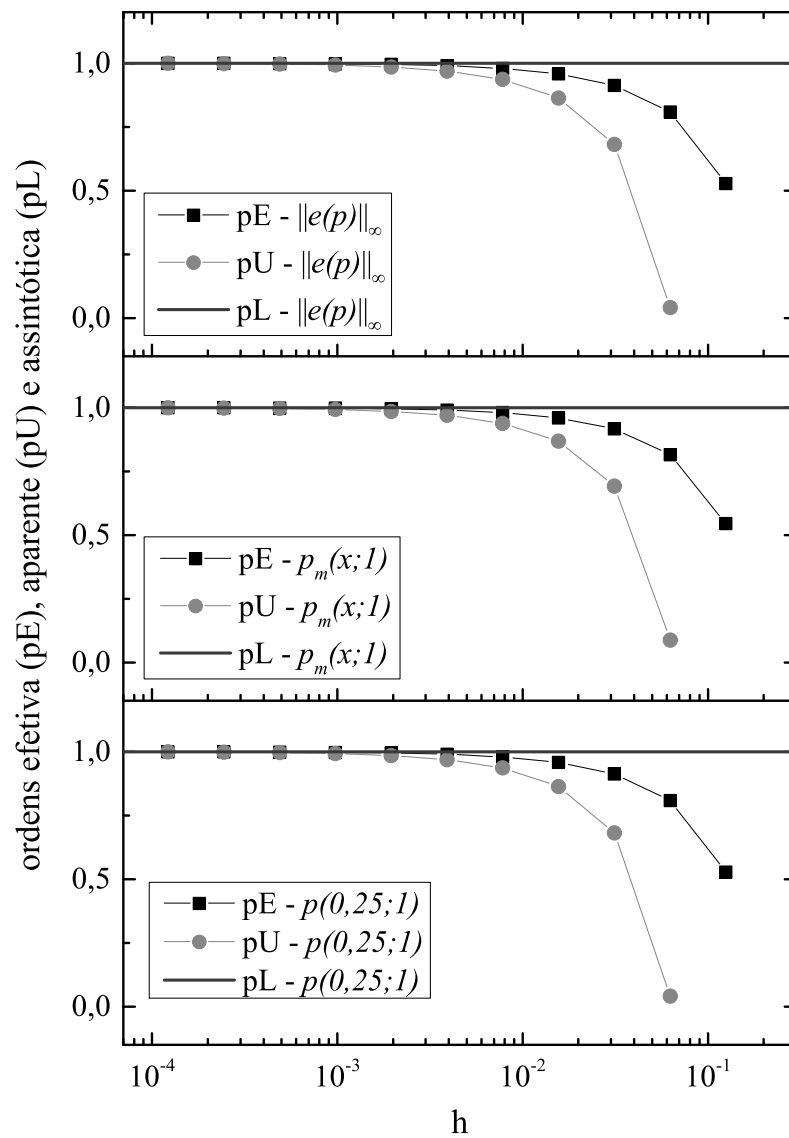


Figura 41 – Ordens efetiva, aparente e assintótica considerando a variável p da equação da poroelasticidade 1D e o método de Euler.

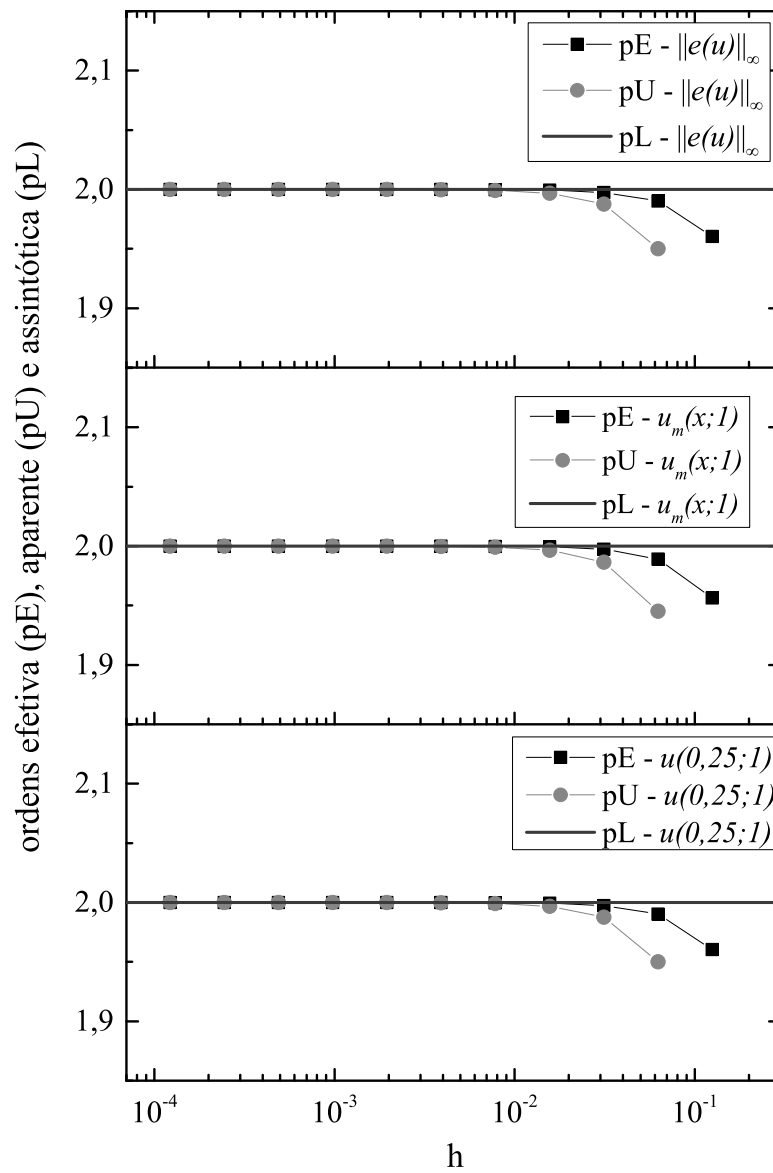


Figura 42 – Ordens efetiva, aparente e assintótica considerando a variável u da equação da poroelasticidade 1D e o método de CN.

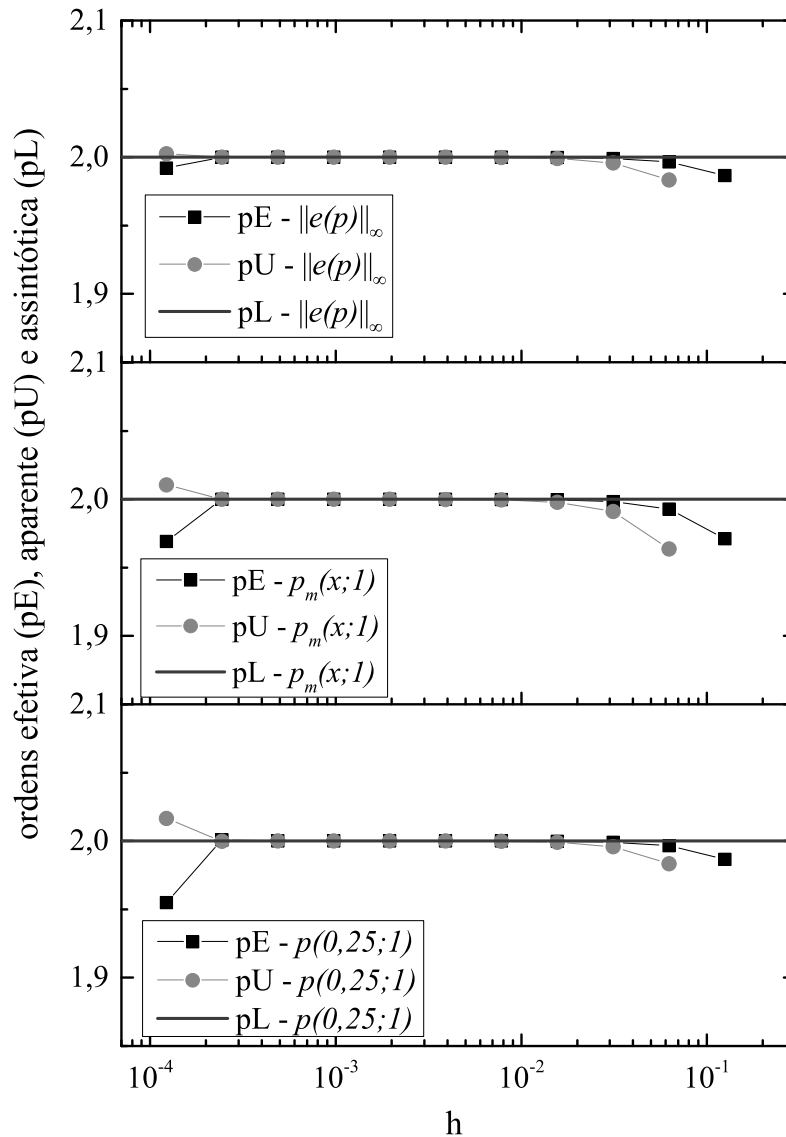


Figura 43 – Ordens efetiva, aparente e assintótica considerando a variável p da equação da poroelasticidade 1D e o método de CN.

8.2.2 Problema bidimensional

Nas simulações do problema 2D consideraram-se os parâmetros de Lamé $\lambda = 2.8 \cdot 10^4$ e $\mu = 4,2 \cdot 10^4$. Assim como nos testes para o problema 1D, o primeiro teste para verificar o código computacional é a análise do erro numérico. Para isso, fizeram-se simulações numéricas considerando os métodos *singlegrid* e *multigrid* até alcançar o erro de máquina. Na Tab. 8 mostra-se a norma $\|\cdot\|_\infty$ dos erros numéricos para os deslocamentos u e v e a pressão p , considerando a aproximação temporal dada pelo método de Euler e pelo método de CN. O número de pontos nas discretizações espaciais e temporal é dada por $N = N_x = N_y = N_t$.

Observa-se que os erros numéricos referentes aos métodos *singlegrid* e *multigrid*

Tabela 8 – Valor da $\|\cdot\|_\infty$ dos erros numéricos para a equação da poroelasticidade 2D, usando o método *Waveform Relaxation* e o suavizador Vanka de 5 pontos com ordenação de 12 cores e linha no tempo para os métodos *singlegrid* e *multigrid*.

N	tipo	Euler			CN		
		$\ e(u)\ _\infty$	$\ e(v)\ _\infty$	$\ e(p)\ _\infty$	$\ e(u)\ _\infty$	$\ e(v)\ _\infty$	$\ e(p)\ _\infty$
$2^2 + 1$	<i>singlegrid</i>	2,405E-2	2,405E-2	1,150E+4	1,735E-2	1,735E-2	2,688E+4
	<i>multigrid</i>	2,405E-2	2,405E-2	1,150E+4	1,735E-2	1,735E-2	2,688E+4
$2^3 + 1$	<i>singlegrid</i>	1,475E-2	1,475E-2	7,650E+3	8,796E-3	8,796E-3	8,545E+3
	<i>multigrid</i>	1,475E-2	1,475E-2	7,650E+3	8,796E-3	8,796E-3	8,545E+3
$2^4 + 1$	<i>singlegrid</i>	1,186E-2	1,186E-2	6,701E+3	2,993E-3	2,993E-3	2,335E+3
	<i>multigrid</i>	1,186E-2	1,186E-2	6,701E+3	2,993E-3	2,993E-3	2,335E+3
$2^5 + 1$	<i>singlegrid</i>	7,383E-3	7,383E-3	4,158E+3	8,407E-4	8,407E-4	6,067E+2
	<i>multigrid</i>	7,383E-3	7,383E-3	4,158E+3	8,407E-4	8,407E-4	6,067E+2
$2^6 + 1$	<i>singlegrid</i>	4,070E-3	4,070E-3	2,283E+3	2,195E-4	2,195E-4	1,539E+2
	<i>multigrid</i>	4,070E-3	4,070E-3	2,283E+3	2,195E-4	2,195E-4	1,539E+2
$2^7 + 1$	<i>singlegrid</i>	2,120E-3	2,120E-3	1,190E+3	5,559E-5	5,559E-5	3,867E+1
	<i>multigrid</i>	2,120E-3	2,120E-3	1,190E+3	5,559E-5	5,559E-5	3,867E+1

mostrados na Tab. 8 são iguais.

As Fig. 44 e Fig. 45 mostram o t_{CPU} em função do número de incógnitas e o *speed-up* entre os métodos *singlegrid* e *multigrid*, respectivamente.

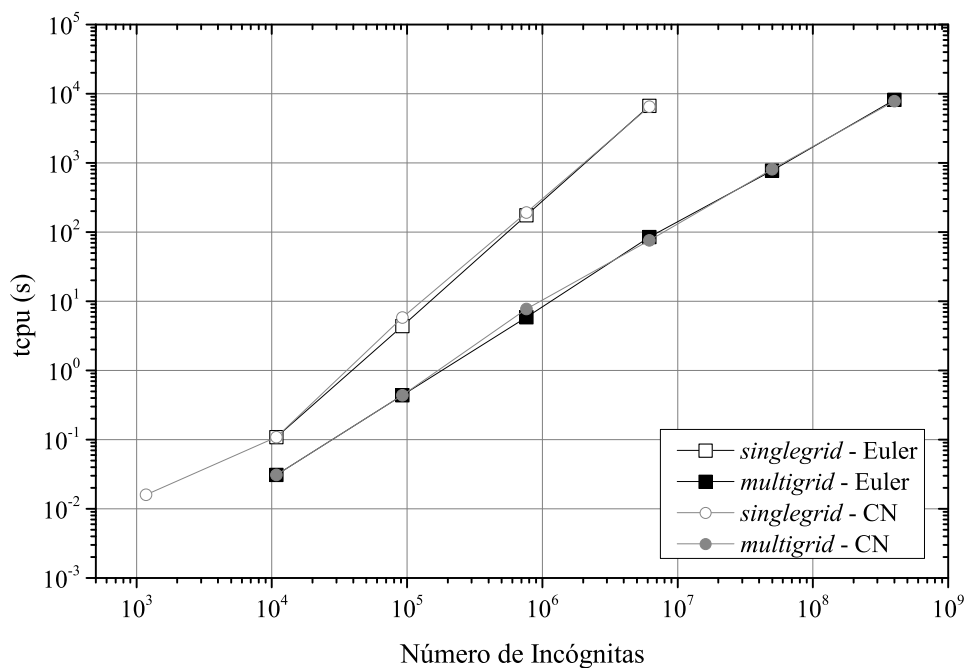


Figura 44 – Tempo computacional usado para resolver a equação da poroelasticidade 2D com os métodos *singlegrid* e *multigrid* e com aproximações temporal usando Euler e CN. $tol = 10^{-6}$

Com o cálculo do *speed-up* descrito na Fig. 45 pode-se verificar que o método

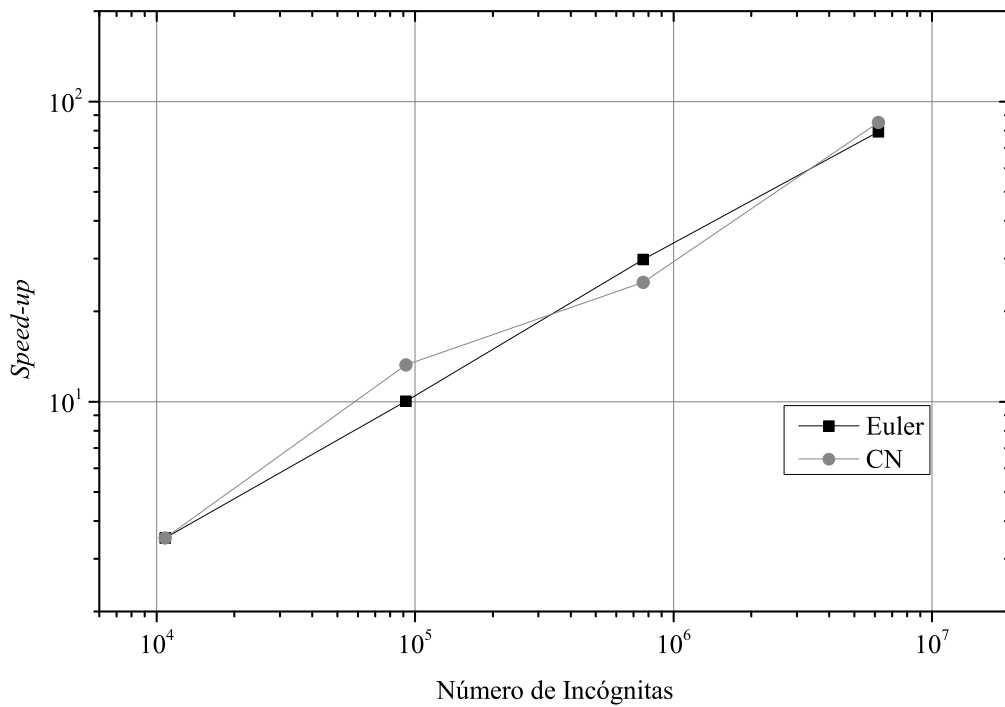


Figura 45 – *Speed-up* entre os métodos *singlegrid* e *multigrid* para as aproximações temporal usando Euler e CN na equação da poroelasticidade 2D.

multigrid é eficiente para resolver esse tipo de problema.

Para calcular a complexidade do algoritmo, assim como nos casos anteriores, fez-se um ajuste não linear dado pela Eq. (8.3) e usaram-se os dados descritos na Fig. 44. A Tab. 9 mostra os valores de c e p para os métodos *singlegrid* e *multigrid*. Com esses resultados pode-se observar que os valores encontrados são próximos dos teóricos.

Tabela 9 – Coeficientes c e p de $t_{cpu}(\mathcal{N}) = c \cdot \mathcal{N}^p$ considerando a equação da poroelasticidade 2D, os métodos de Euler e de CN com *singlegrid* e *multigrid*.

<i>singlegrid</i>				<i>multigrid</i>			
Euler		CN		Euler		CN	
c	p	c	p	c	p	c	p
1,06E-8	1,7368	1,37E-8	1,72355	5,553E-7	1,18926	6,266E-7	1,18287

As análises das ordens aparente e efetiva para este caso também foi feita usando o método *multigrid* e para 3 variáveis de interesse: 1) uma variável que verifica apenas um ponto, o deslocamento nas direções u e v e a pressão no ponto médio e passo de tempo final com $t_f = 1, 0, u(0, 5; 0, 5; t_f), v(0, 5; 0, 5; t_f)$ e $p(0, 5; 0, 5; t_f)$, respectivamente; 2) uma variável que engloba todos os pontos no passo de tempo final, o deslocamento médio nas direções u e v e a pressão média no passo de tempo final, $u_m(x, y, t_f), v_m(x, y, t_f)$ e $p_m(x, y, t_f)$, respectivamente; 3) uma variável que engloba todas as variáveis em todos os passos de tempo, a norma infinita do erro numérico dos deslocamentos nas direções u e v

e das pressões $\|e(u)\|_\infty$, $\|e(v)\|_\infty$ e $\|e(p)\|_\infty$, respectivamente.

As Figs. 46 a 48 consideram o método de Euler e as variáveis u , v e p , respectivamente. As Figs. 49 a 51 consideram o método de CN e as variáveis u , v e p , respectivamente.

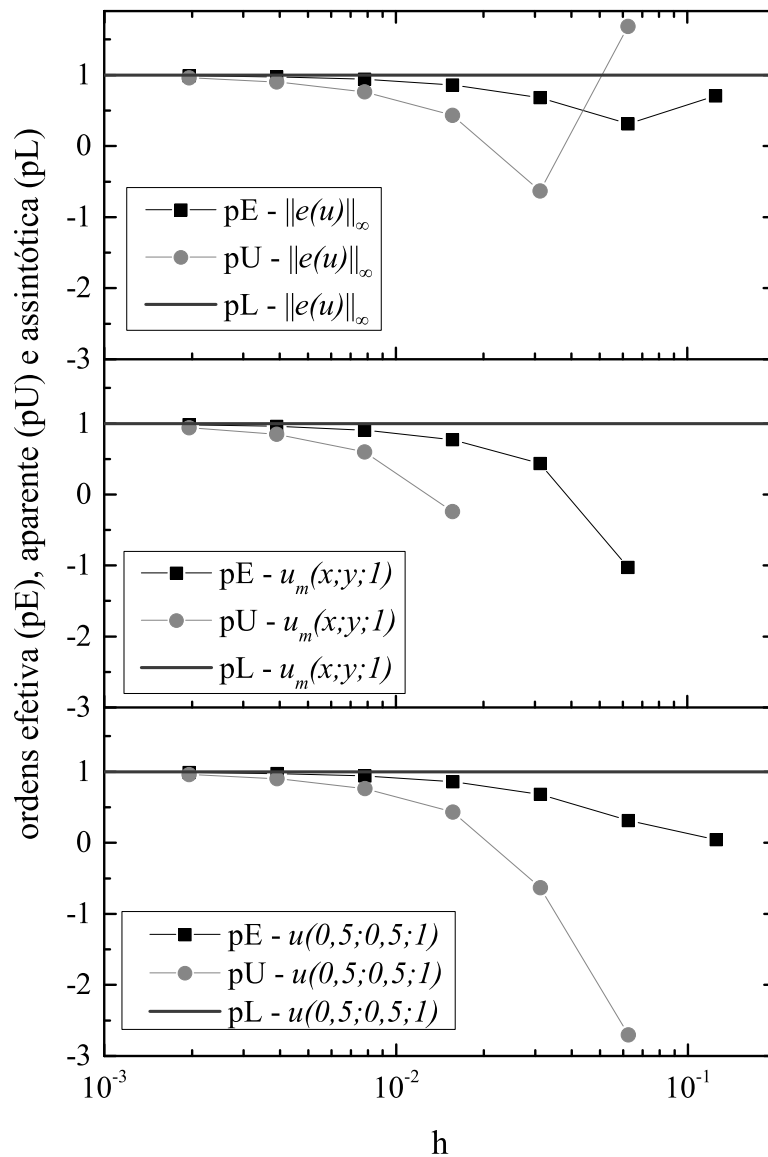


Figura 46 – Ordens efetiva, aparente e assintótica considerando a variável u da equação da poroelasticidade 2D e o método de Euler.

Nas Figs. 46 a 51 observa-se que as ordens efetiva e aparente tendem à ordem assintótica para todas as variáveis analisadas e para os métodos de Euler e CN, exceto para as variáveis $p(0, 5; 0, 5; 1)$ (pressão no ponto central no passo de tempo final) e $p_m(x; y; 1)$ (pressão média no passo de tempo final), considerando-se o método de Euler, ver a Fig. 48.

Neste caso (Fig. 48), nota-se que a tanto a ordem efetiva como a ordem aparente estão tendendo a 2, enquanto que a teoria indica que deveriam tender a 1 (método de

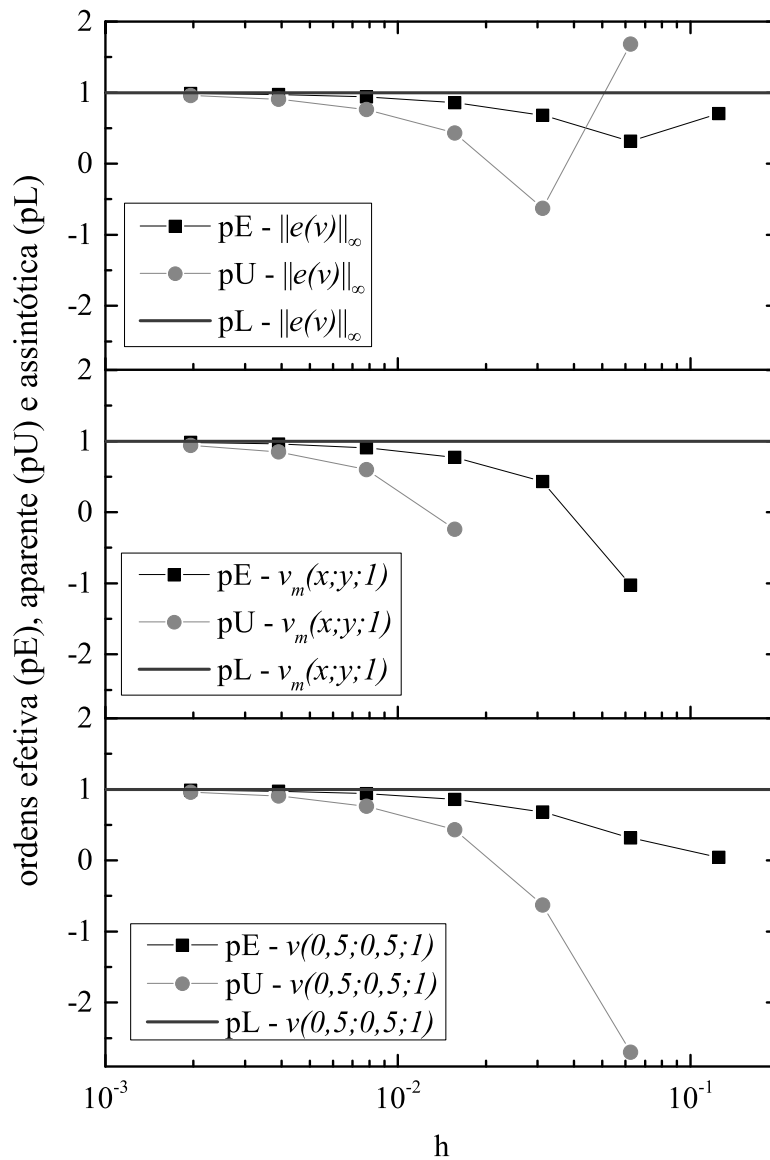


Figura 47 – Ordens efetiva, aparente e assintótica considerando a variável v da equação da poroelasticidade 2D e o método de Euler.

Euler de primeira ordem). Deste modo, são elencados alguns pontos que tentam explicar tal discordância:

- Essas variáveis (pressão no ponto central e pressão média) apresentam o comportamento adequado no caso 1D (veja Fig. 41), mas não o tiveram no caso 2D. No caso bidimensional, pode haver relações entre as componentes de erro ou estimativa de erro (incerteza) nas direções coordenadas x e y que podem favorecer o cálculo do erro ou da incerteza, tal que o estimador de Richardson não consiga captar tais influências;
- A solução analítica sugerida (Eq. (6.32)) pode ter um comportamento que favoreça

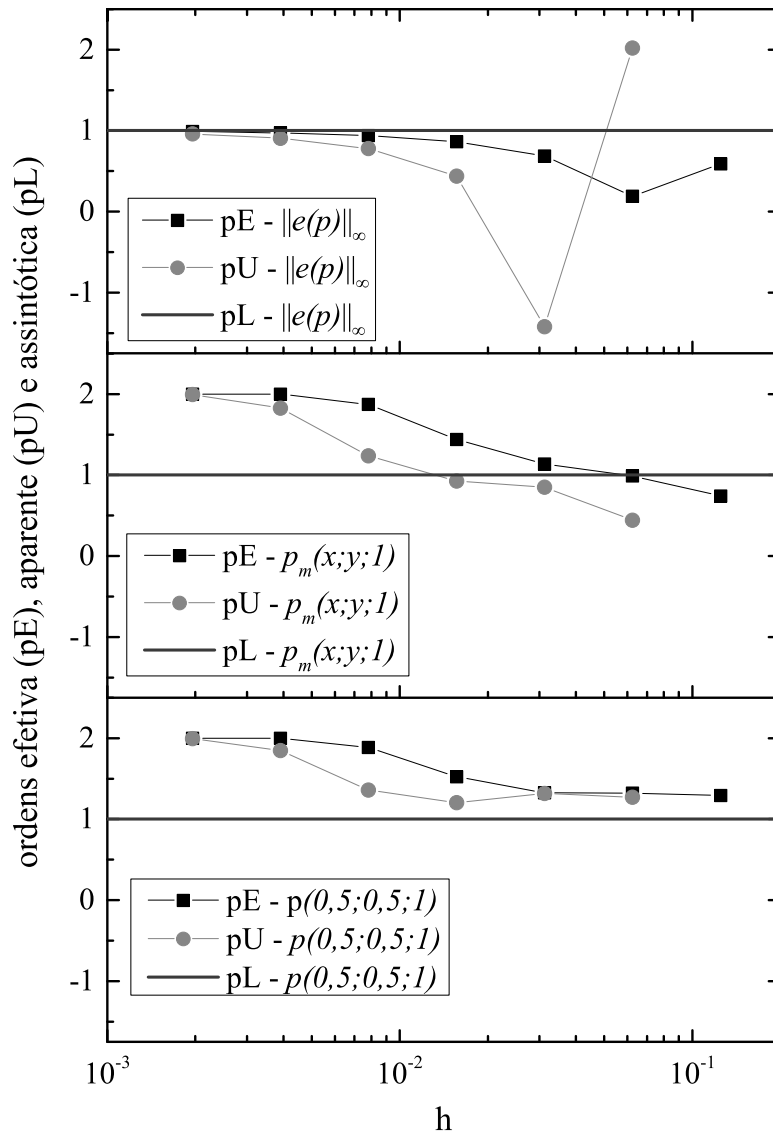


Figura 48 – Ordens efetiva, aparente e assintótica considerando a variável p da equação da poroelasticidade 2D e o método de Euler.

mais os cálculos do erro ou da incerteza e tal comportamento também não é captado pelo estimador de Richardson.

De acordo com esses apontamentos, é proposta uma análise de erros baseada em um estimador de erros mais preciso, neste caso, o estimador bicoeficiente (MARCHI, 2001).

Com isso, devido ao fato de as ordens efetiva e aparente tenderem a 2 em um método de 1ª ordem (método de Euler), buscou-se fazer testes com outros estimadores que também fornecem ordens efetiva e aparente, como por exemplo, o estimador bicoeficiente (MARCHI, 2001).

Para o estimador bicoeficiente (U_{bi}), definido por Marchi (2001), a incerteza é

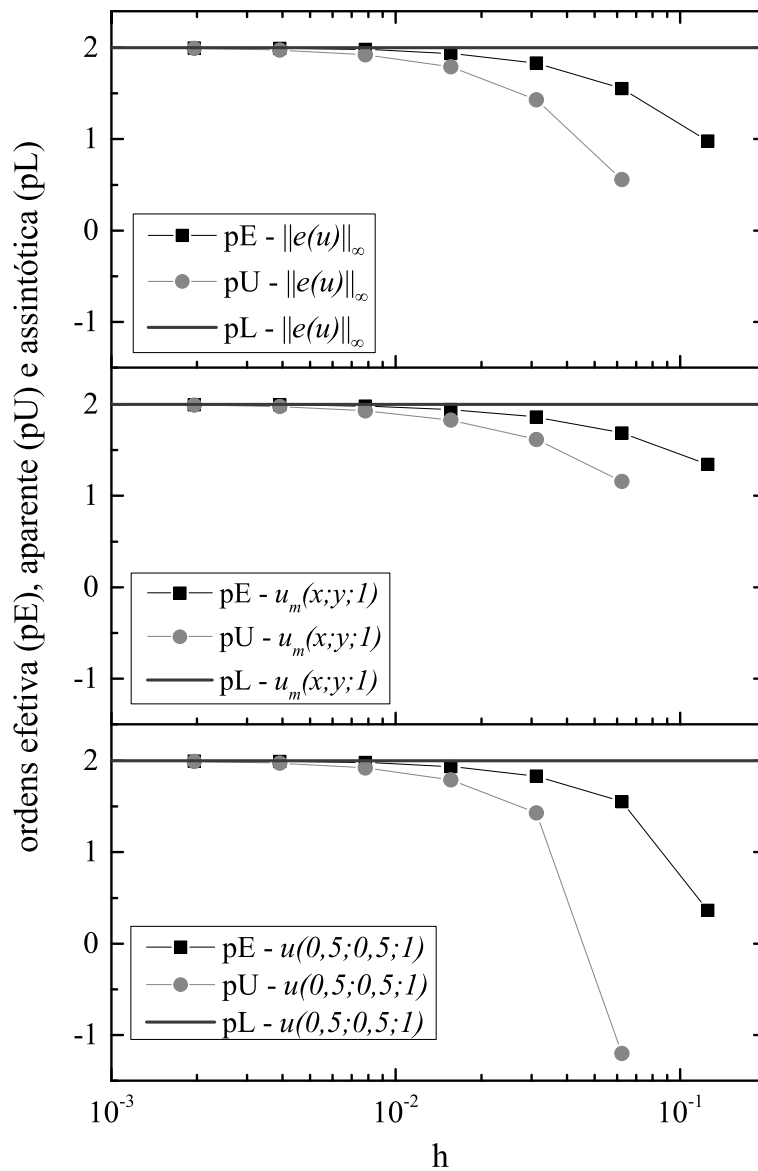


Figura 49 – Ordens efetiva, aparente e assintótica considerando a variável u da equação da poroelasticidade 2D e o método de CN.

calculada com

$$U_{bi}(\phi) = \phi_\infty - \phi, \tag{8.4}$$

em que ϕ é a solução numérica calculada com uma malha cujo tamanho de seus elementos é h e ϕ_∞ é o valor estimado para a solução analítica exata.

Considerando-se ϕ_1 a solução numérica obtida na malha mais fina, ϕ_3 a solução numérica na malha mais grossa, enquanto que ϕ_2 é a solução numérica na malha intermediária, com a razão de refinamento entre cada malha constante ($q = 2$), tem-se que o valor estimado para a solução analítica exata através do método bicoeficiente, é dado por

$$\phi_\infty = \frac{8\phi_1 - 6\phi_2 + \phi_3}{3}. \tag{8.5}$$

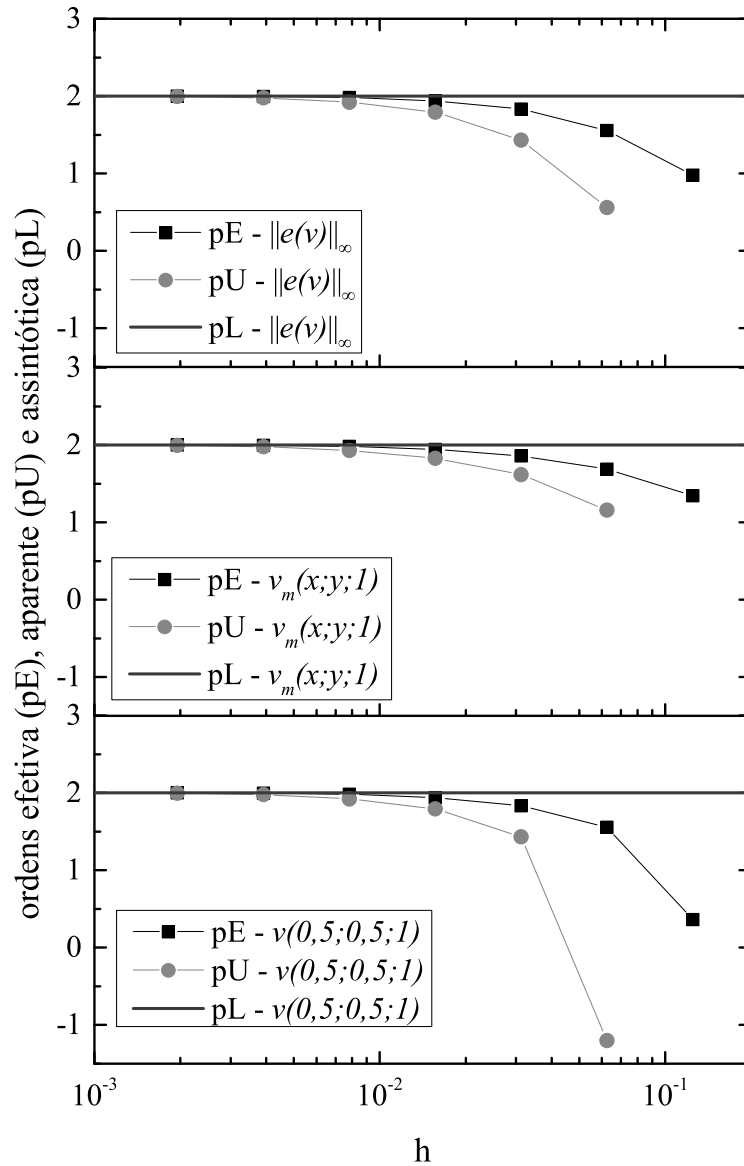


Figura 50 – Ordens efetiva, aparente e assintótica considerando a variável v da equação da poroelasticidade 2D e o método de CN.

A efetividade do estimador bicoeficiente (θ_{bi}) é definida pela razão entre a sua incerteza (U_{bi}) e o erro de discretização (e) na malha mais fina, ou seja,

$$\theta_{bi} = \frac{U_{bi}}{e(\phi_1)}. \quad (8.6)$$

A efetividade assintótica é definida para a situação limite em que $h \rightarrow 0$ e representa a ordem de aproximação dos métodos numéricos utilizados.

Na Fig. 52 é mostrada a efetividade do estimador bicoeficiente para a variável p e considerando-se os mesmos resultados usados nos cálculos das ordens efetiva e aparente mostrados na Fig. 48.

Observa-se na Fig. 52 que as variáveis $p(0,5;0,5;1)$ e $p_m(x,y;1)$ para as quais o

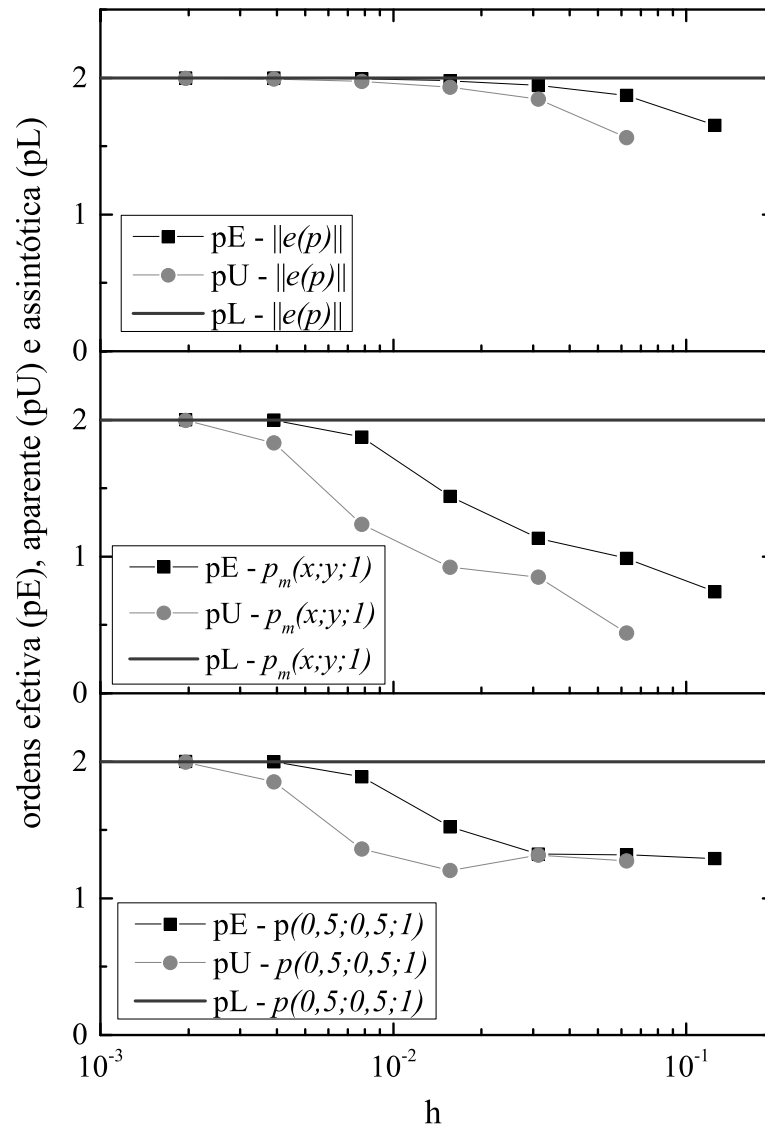


Figura 51 – Ordens efetiva, aparente e assintótica considerando a variável p da equação da poroelasticidade 2D e o método de CN.

estimador de Richardson não convergia para a ordem assintótica $pL = 1$ (veja a Fig. 48), com o estimador bicoeficiente os resultados convergem para a referida ordem assintótica $pL = 1$. Porém, observa-se nessa figura que a efetividade para os referidos valores de $p(0, 5; 0, 5; 1)$ e $p_m(x; y; 1)$ tendem a 1 lentamente quando usa-se a condutividade hidráulica $K = 10^{-9}$ (como para as demais figuras desta seção). Dessa forma, pensou-se em alguns testes adicionais, alterando somente o valor da condutividade hidráulica, ou seja, utilizando $K = 10^{-3}$ (que também é uma condutividade hidráulica realística). Os resultados usando o estimador bicoeficiente são mostrados na Fig. 53. Observa-se nessa figura, que os referidos valores de $p(0, 5; 0, 5; 1)$ e $p_m(x; y; 1)$ tendem a 1 mais rapidamente. Com isso pode-se afirmar que o método de Euler usado para o problema da poroelasticidade bilinear possui ordem 1, validando assim o esquema numérico desenvolvido e programado.

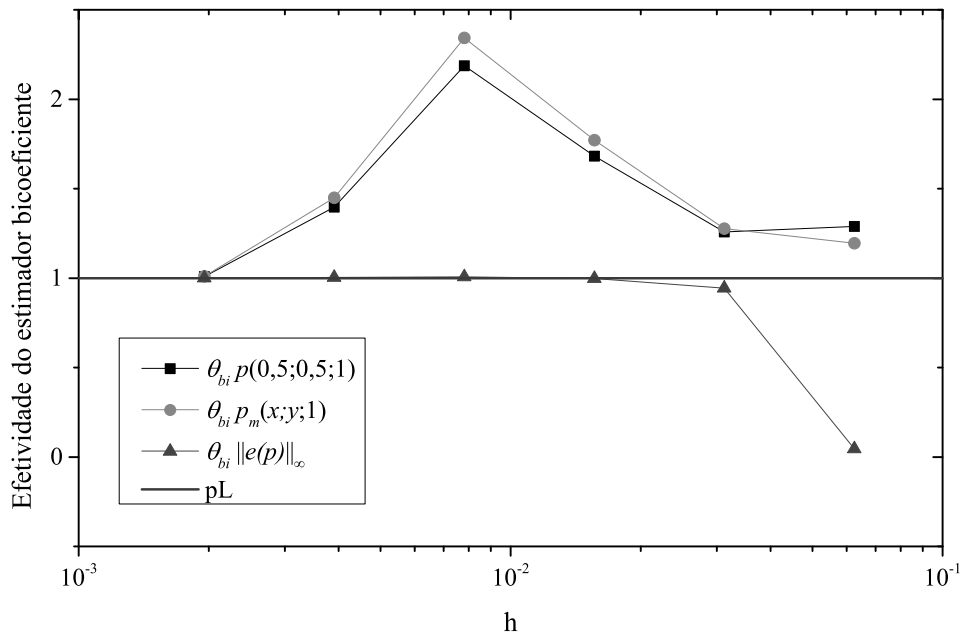


Figura 52 – Efetividade do estimador bicoeficiente considerando a variável p da equação da poroelasticidade 2D e o método de Euler e condutividade hidráulica $K = 10^{-9}$.

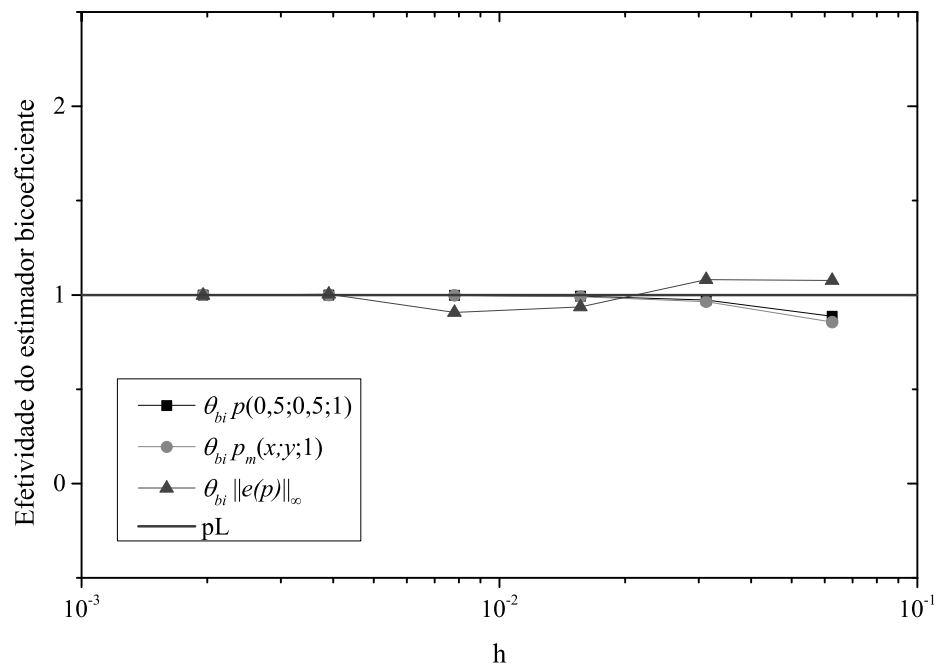


Figura 53 – Efetividade do estimador bicoeficiente considerando a variável p da equação da poroelasticidade 2D, o método de Euler e a condutividade hidráulica $K = 10^{-3}$.

9 Resultados Preliminares

Neste capítulo serão estudados e expostos os resultados obtidos através das simulações numéricas basicamente para os métodos *Time-Stepping - multigrid*, *Waveform Relaxation - multigrid* e *Space-Time - multigrid* aplicados às equações do calor 1D e 2D. A denominação deste capítulo como "resultados preliminares" se deu pelo fato que já existem alguns resultados publicados com esses métodos para a respectiva equação do calor 1D e 2D, ver por exemplo, Vandewalle (1993), Horton e Vandewalle (1995), Janssen (1997), Lent (2006), Choptuik (2008) e Falgout et al. (2017). Portanto, são resultados preliminares, já existentes na literatura, mas importantes para a continuidade e compreensão do capítulo seguinte, em que estão reportados os principais resultados desta tese.

Os algoritmos foram implementados na linguagem Fortran® 2013 da Intel® com precisão dupla, projeto Console Application com Release. As simulações foram realizadas em computador com processadores Intel core i7-7700, Clockspeed: 3.6 GHz e 32 GB de memória RAM (DDR4 – 2400 MHz). Para o cálculo do tempo computacional (t_{CPU}), usou-se a função *timef* do Fortran® 2013. Entende-se por t_{CPU} o tempo medido em segundos (s), necessário para realizar a geração de malhas, atribuir a condição inicial, calcular os coeficientes e resolver o sistema linear obtido com a discretização da EDP, até ser atingida a tolerância estabelecida com base no critério de convergência.

Os problemas resolvidos foram descritos no capítulo 6 e os métodos de solução foram descritos detalhadamente no capítulo 7.

Para verificar os intervalos e os valores das variáveis utilizadas nos cálculos, considerou-se o parâmetro λ definido por $\lambda = \frac{\tau}{h^2}$. Neste caso, λ pode ser considerado como uma medida do grau de anisotropia no operador discretizado na malha Ω (HORTON; VANDEWALLE, 1995). Para isso, foram realizados testes computacionais com o método *singlegrid*, como mostra a Fig. 54, em que para alguns valores de λ foi medido o fator de convergência médio ρ_m (ver a Eq. (4.10)) para verificar os intervalos de λ para o qual o método *singlegrid* é eficiente, ou seja, não sendo necessário o uso de técnicas *multigrid*.

De acordo com Thole e Trottenberg (1986) um suavizador apresenta melhores fatores de convergência quando é aplicado na direção de mais forte acoplamento. Sendo assim, e com o objetivo de desenvolver algoritmos paralelizáveis, testaram-se os intervalos para considerar λ usando os suavizadores zebra com linha no tempo e zebra com linha no espaço. Para o suavizador zebra com linha no tempo, basta percorrer os pontos na direção temporal, enquanto que para o suavizador zebra com linha no espaço usou-se o método TDMA (ver por exemplo, Maliska (2004)) em cada passo de tempo sucessivamente.

Nas simulações descritas na Fig. 54 usou-se o método *singlegrid* com o método de

Euler, discretização de malha com $N_x = N_t = 2^{10} + 1$, condição inicial e de contornos nula, estimativa inicial aleatória entre 0 e 1 para os pontos internos e o t_f é fixo em cada simulação e calculado para cada valor de $\log_2 \lambda$. O fator de convergência médio ρ_m foi calculado considerando as 20 primeiras iterações (ver a Eq. (4.10)).

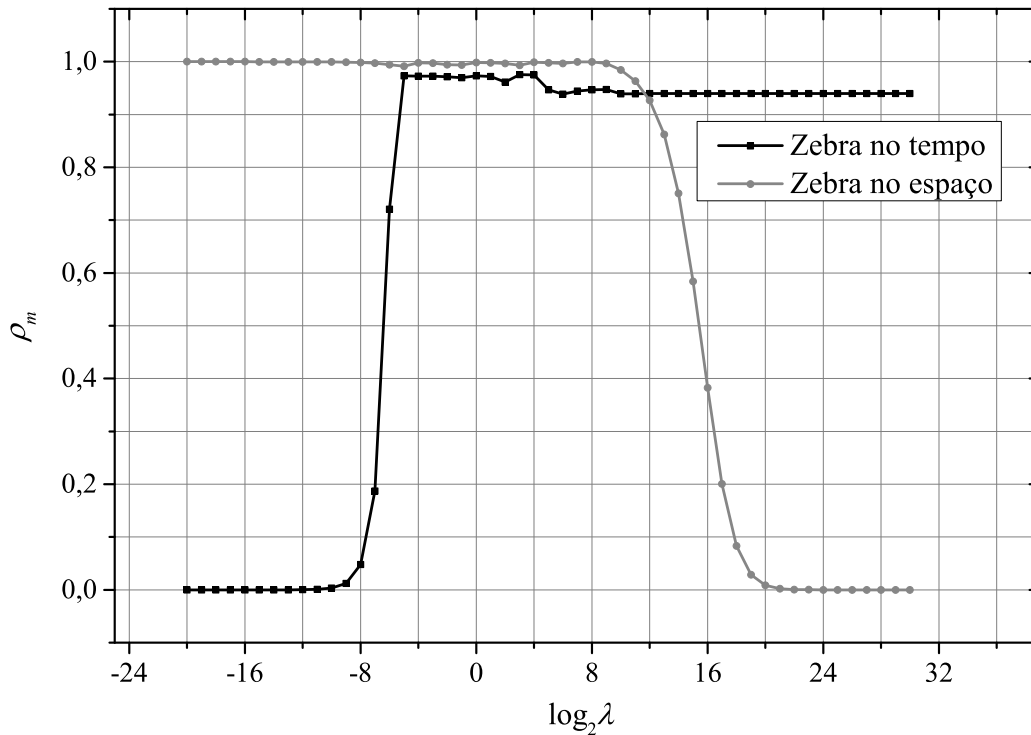


Figura 54 – Fator de convergência médio (ρ_m) do método *singlegrid* após 20 iterações com os suavizadores zebra no tempo e zebra no espaço para a equação do calor 1D.

Observa-se na Fig. 54 que com o suavizador zebra no tempo e no intervalo $\log_2 \lambda \leq -7$ ou usando-se o suavizador zebra no espaço e no intervalo $\log_2 \lambda \geq 16$, o fator de convergência do método *singlegrid* é pequeno (indicando que o método é bom). Analisando o estêncil dado na Eq. (6.15) verifica-se que quando λ é pequeno a direção dominante é a temporal, enquanto que se λ é grande, a direção dominante é a direção espacial. Portanto, os resultados obtidos com o teste numéricos são concordantes.

Desta forma, verifica-se com esse teste numérico que não há necessidades de se estudar o método *multigrid* para intervalos em que $\log_2 \lambda < -8$ ou $\log_2 \lambda > 16$, pois nestes intervalos o método *singlegrid* é eficiente. Porém, para $\log_2 \lambda > 8$, verificou-se com alguns testes numéricos que as propriedades do método *multigrid*, como o fator de convergência, permanecem inalteradas. Assim, seguindo Vandewalle e Horton (1993), Horton e Vandewalle (1995), optou-se por considerar neste trabalho o intervalo para o fator de anisotropia sendo $-8 \leq \log_2 \lambda \leq 8$.

Na Fig. 55 obtida de Vandewalle e Horton (1993, p. 103) é mostrado o fator de convergência através da LFA para a equação do calor 1D e aproximação temporal feita

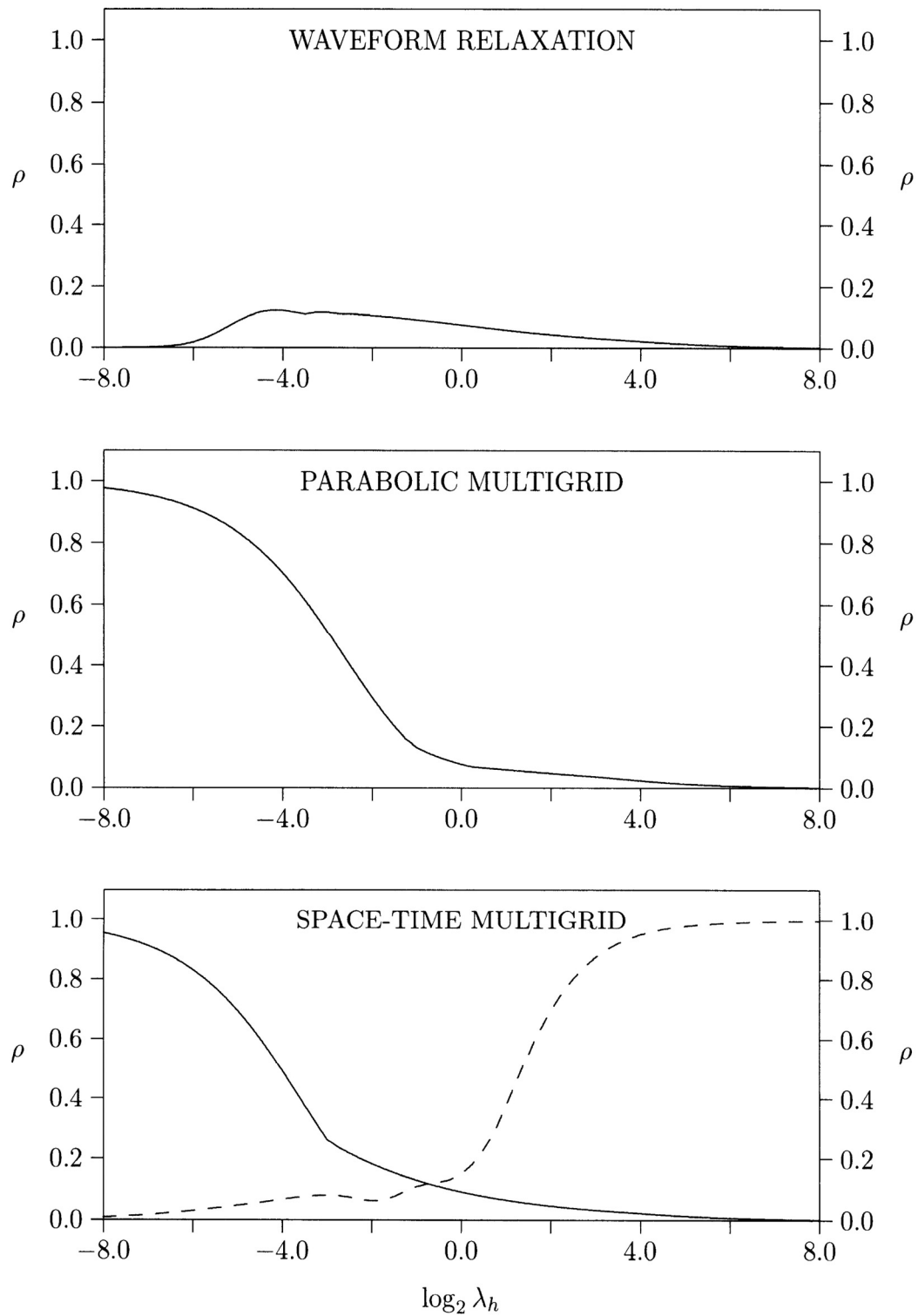


Figura 55 – Fator de convergência calculado através do método de 2 malhas (LFA) para a equação do calor 1D e aproximação temporal feita com o método de Euler, considerando $N_x = N_t = 128$, $\nu_1 = 2$ (pré) e $\nu_2 = 1$ (pós suavização). No método *Space-Time - multigrid* a linha sólida indica engrossamento apenas na direção espacial e a linha tracejada indica engrossamento apenas na direção temporal. Figura retirada de Vandewalle e Horton (1993, p. 103).

com o método de Euler.

Observa-se nessa Fig. 55 que o método *Waveform Relaxation - multigrid* apresenta bons resultados para todos os valores de $\log_2 \lambda$. Esse método utiliza engrossamento apenas no espaço e um suavizador linha no tempo. O método *Space-Time - multigrid* depende de um fator (λ_{crit}) que indica se deve ocorrer engrossamento na direção espacial ou temporal; na Fig. 55, no método *Space-Time - multigrid* a linha sólida indica engrossamento apenas na direção espacial, a linha tracejada indica engrossamento apenas na direção temporal e o λ_{crit} é definido no ponto de interseção entre as linhas sólida e tracejada. Sendo assim, esse método utiliza semiengrossamentos, na direção espacial se $\lambda \leq \lambda_{crit}$ ou na direção temporal se $\lambda > \lambda_{crit}$, e um suavizador ponto-a-ponto, o Gauss-Seidel *red-black*. Considerando essa troca de direção de engrossamentos, o método *Space-Time - multigrid* também apresenta bons resultados em relação ao fator de convergência. O método chamado *Parabolic Multigrid* utiliza engrossamento somente no espaço e usa um suavizador ponto-a-ponto no espaço (Gauss-Seidel *red-black*), replicando em todos os passos de tempo sucessivamente. Observa-se que esse método é atraente somente se $\log_2 \lambda$ é grande, portanto não será discutido nesta tese.

Para os testes numéricos na sequência, considera-se o número de pontos na aproximação temporal igual ao número de pontos da discretização espacial, ou seja, para o caso 1D tem-se $N = N_x = N_t = 2^{L_{max}} + 1$, em que L_{max} representa o número máximo de malhas que podem ser usadas no método *multigrid*. Para o caso 2D o número de pontos na discretização espacial na direção de y é igual ao número de pontos da discretização espacial na direção de x , assim, $N = N_x = N_y = N_t = 2^{L_{max}} + 1$.

Nas simulações numéricas expostas neste capítulo, consideraram-se os métodos *Time-Stepping - multigrid*, *Waveform Relaxation - multigrid* e *Space-Time - multigrid*. Nestes testes foram considerados os métodos de Euler e de CN, os ciclos V , F e W (quando for possível, pois em alguns casos o método de CN ou o ciclo V não apresentam bons resultados ou robustez, como será descrito na seção 9.1), restrição por ponderação completa e prolongação por interpolação linear para o caso 1D e bilinear para o caso 2D nos pontos espaciais para os métodos *Time-Stepping - multigrid* e *Waveform Relaxation - multigrid* (ver Trottenberg et al. (2001)) e no espaço e tempo para o método *Space-Time - multigrid*, (ver Horton e Vandewalle (1995)).

9.1 Testes de robustez

Com o objetivo de verificar a validade dos métodos usados para resolver o problema estudado, fizeram-se alguns testes para verificar sua robustez. Um código é dito robusto quando ele resolve o determinado problema independente dos dados de entrada, como o fator de anisotropia ou o tamanho de discretização de malha, além disso, para os

parâmetros de entrada fixos, seu fator de convergência assintótico se mantém constante quando ocorre refinamento da malha. Dessa forma, pode-se garantir que tal método e principalmente o suavizador associado ao método *multigrid* funciona bem quando a malha é super-refinada, ver por exemplo Llorente e Melson (1998).

Para esses testes numéricos, a condição inicial e de contornos são consideradas nulas, a estimativa inicial para os pontos internos é considerada aleatória entre 0 e 1 e o ciclo escolhido é o ciclo $V(0, 1)$ (ou $V(1, 0)$). A escolha do ciclo V com apenas 1 suavização se deu pelo fato de que se observou através de vários testes numéricos que se um método for robusto com o ciclo V e 1 suavização, este também será robusto com os ciclos F e W e com mais suavizações internas.

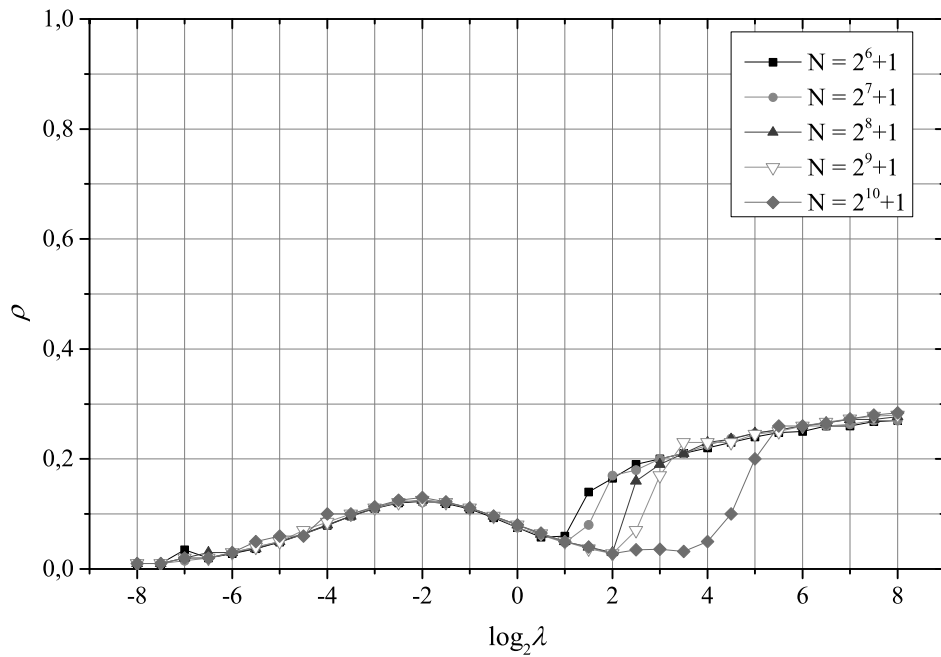
O método *Time-Stepping - multigrid* não foi testado na equação do calor 1D. Isso se dá pelo fato de ser possível utilizar o método TDMA para resolver de forma direta e eficiente esse problema 1D com o método *Time-Stepping - multigrid*, ou seja, resolve-se de forma direta em todos os pontos espaciais e em cada passo de tempo e de forma sucessiva até o passo de tempo final.

Para verificar a robustez do método *Time-Stepping - multigrid* na equação do calor 2D, fizeram-se simulações numéricas considerando o ciclo $V(0, 1)$, $N = 2^{L_{\max}} + 1$ em que $L_{\max} = 6, 7, \dots, 10$ e os métodos de Euler e de CN, mostrados nas Fig. 56(a) e Fig. 56(b), respectivamente. Excepcionalmente neste método, pelo fato de se considerar a solução obtida no passo de tempo anterior como condição inicial para a solução no passo de tempo atual (subsequente), o fator de convergência assintótico é obtido nos cálculos relacionados ao último passo de tempo e também no último ciclo *multigrid* antes de se alcançar o erro de máquina.

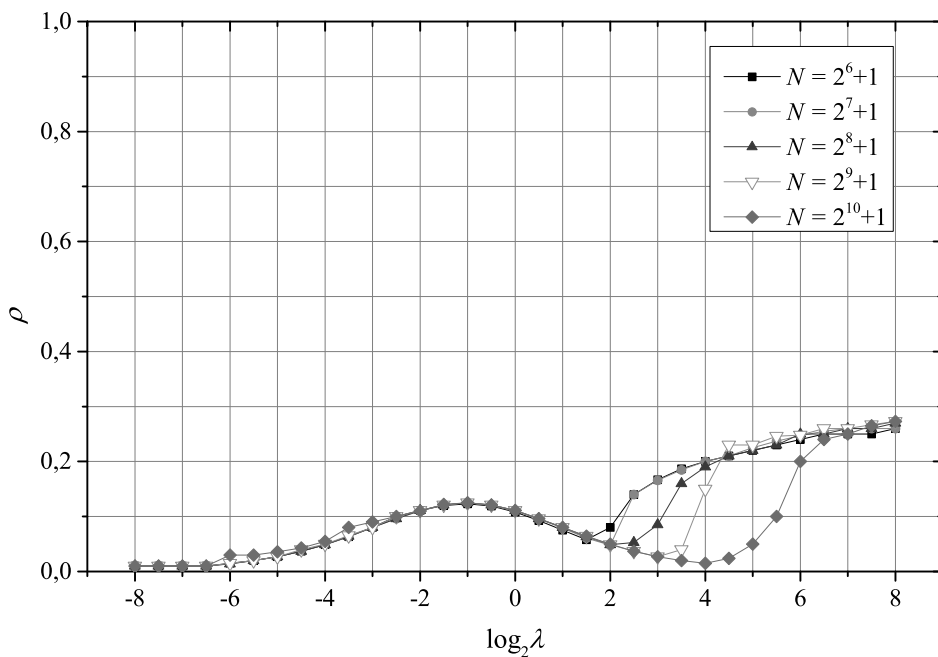
Observa-se nas curvas descritas na Fig. 56 que, quando a malha é refinada, os fatores de convergência se estabilizam. Desta forma, pode-se dizer que o método *Time-Stepping - multigrid* é robusto com relação ao fator de anisotropia e à discretização da malha.

Para verificar a robustez do método *Waveform Relaxation - multigrid* nas equações do calor 1D e 2D, fizeram-se simulações numéricas considerando o ciclo $V(0, 1)$ e $N = 2^{L_{\max}} + 1$ em que $L_{\max} = 8, 9, \dots, 13$ para o caso 1D e $L_{\max} = 5, 6, \dots, 9$ para o caso 2D. As Fig. 57 e Fig. 58 mostram as simulações para os casos 1D e 2D, respectivamente. Nestas Fig. 57(a) e Fig. 58(a) é usado o método de Euler, enquanto que nas Fig. 57(b) e Fig. 58(b) é usado o método de CN.

Com os resultados descritos nas Fig. 57(a) e Fig. 57(b) pode-se observar que com método *Waveform Relaxation - multigrid*, o fator de convergência assintótico ρ não aumenta quando a malha é refinada e/ou o fator de anisotropia λ é alterado. Com isso pode-se dizer que o método *Waveform Relaxation - multigrid* é robusto em relação ao tamanho de discretização e também em relação ao fator de anisotropia λ para o ciclo $V(0, 1)$ no



(a) Método de Euler.

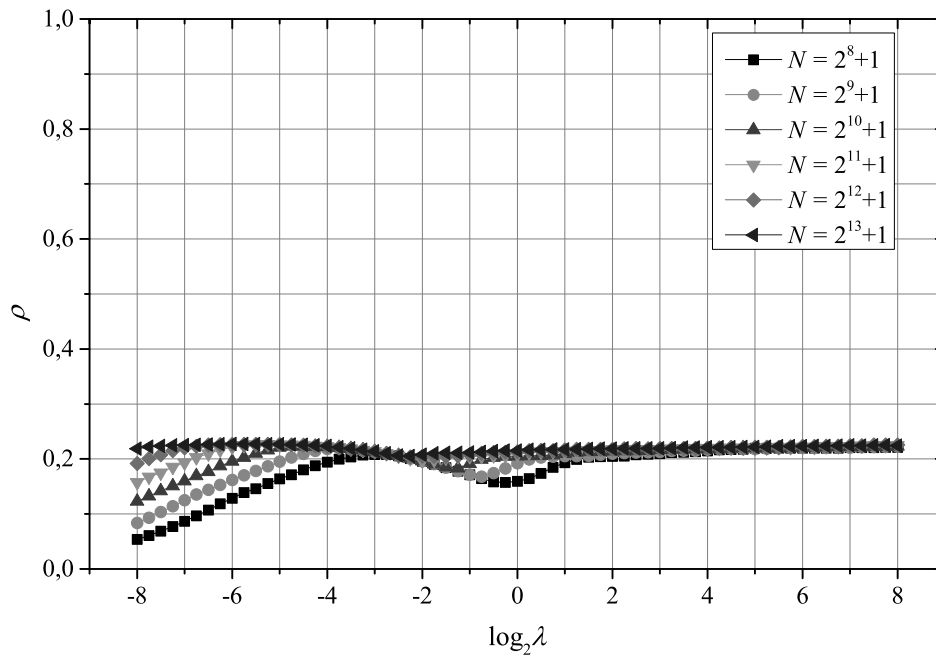


(b) Método de CN.

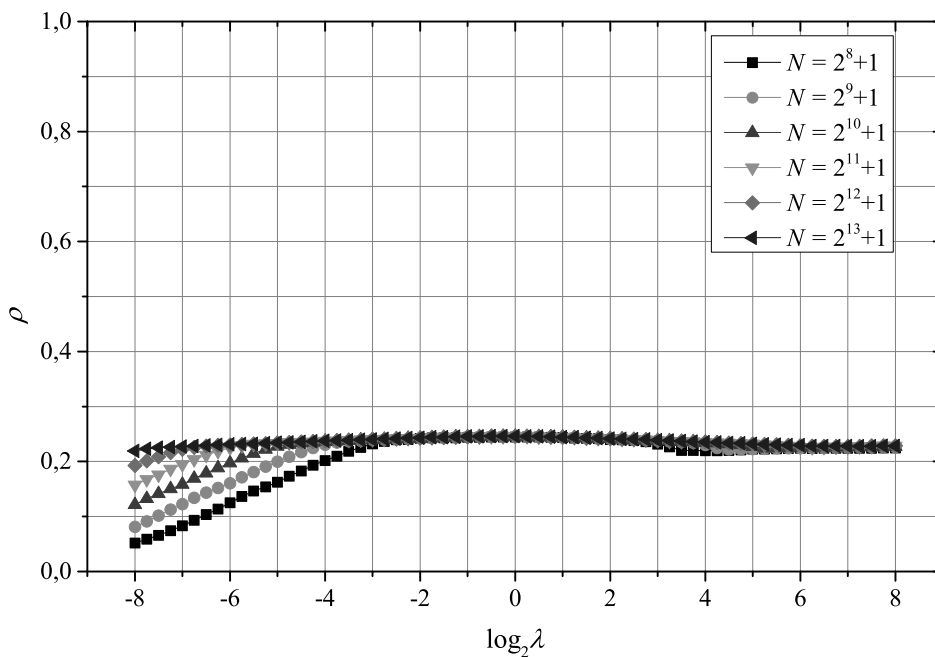
Figura 56 – Robustez do método *Time-Stepping - multigrid* e ciclo $V(0, 1)$ para resolver a equação do calor 2D.

caso unidimensional. Sendo assim, este método é robusto para qualquer ciclo *multigrid* e também para qualquer número de suavizações para aproximar a equação do calor 1D.

Observa-se nas Fig. 58(a) e Fig. 58(b) que, quando a malha é refinada, ocorre um leve aumento no valor do fator de convergência assintótico ρ . Sendo assim, fizeram-se



(a) Método de Euler.

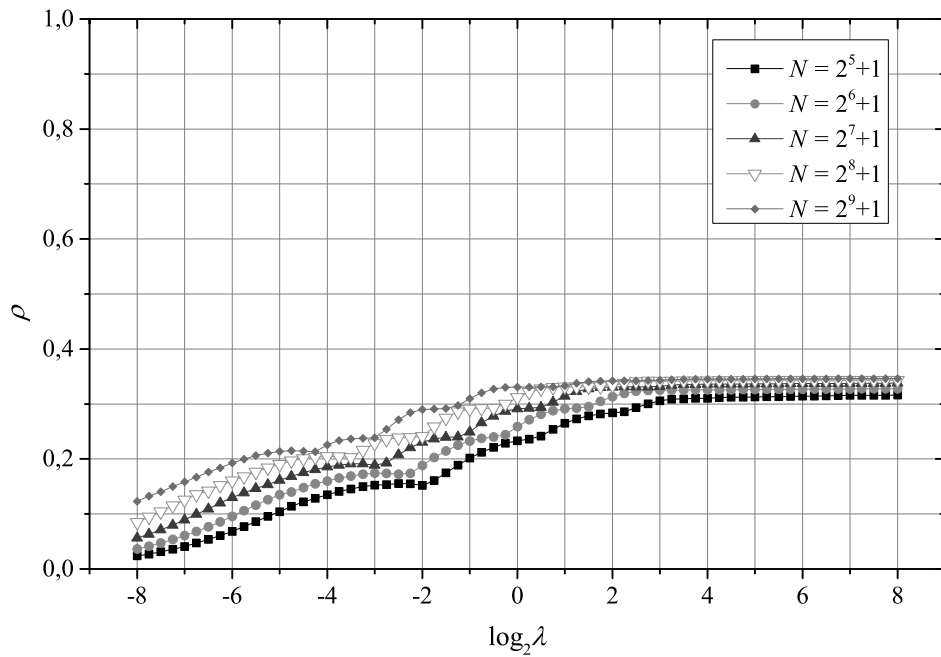


(b) Método de CN.

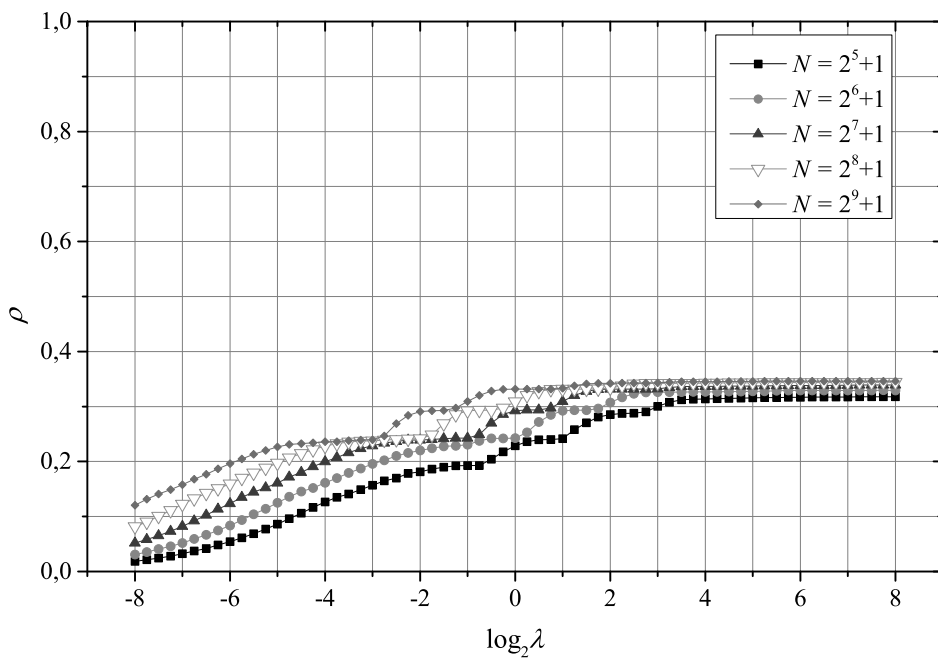
Figura 57 – Robustez do método *Waveform Relaxation - multigrid* e ciclo $V(0, 1)$ para resolver a equação do calor 1D.

testes adicionais para verificar melhor a robustez desse método, considerando $N_x = N_y$ ou N_t sempre fixos, como pode-se ver nas Fig. 59 e Fig. 60.

Com o valor de $N_x = N_y$ fixo (Fig. 59) pode-se observar que para ambos os métodos de Euler (Fig. 59(a)) e de CN (Fig. 59(b)), o fator de convergência assintótico ρ se estabiliza



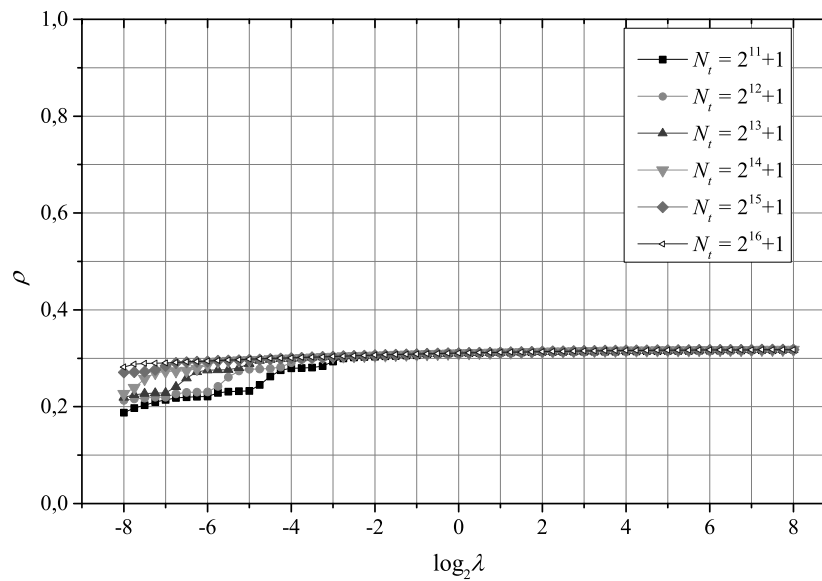
(a) Método de Euler.



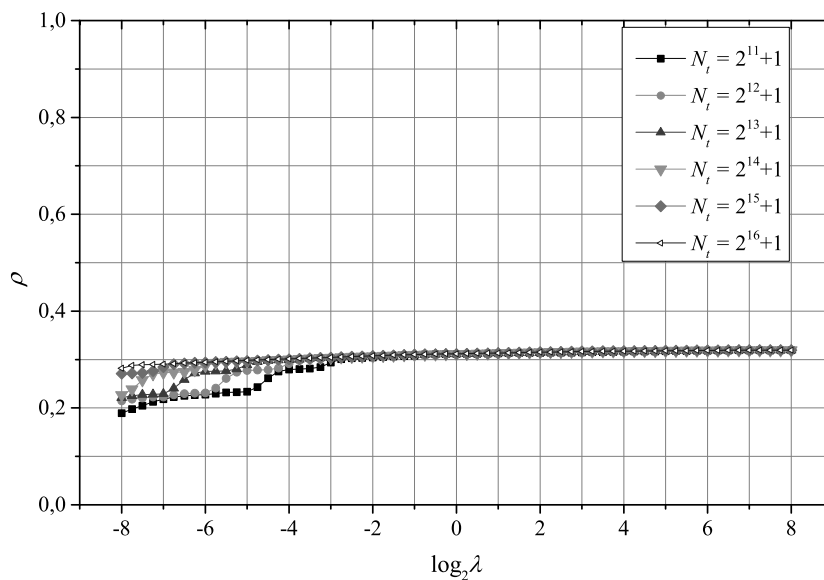
(b) Método de CN.

Figura 58 – Robustez do método *Waveform Relaxation - multigrid* e ciclo $V(0, 1)$ para resolver a equação do calor 2D.

quando o número de passos de tempo aumenta, independente do fator de anisotropia λ . Ocorre comportamento semelhante quando se fixa o número de passos de tempo e se refina a discretização da malha espacial; pode-se observar a Fig. 60(a) com o método de Euler e a Fig. 60(b) com o método de CN. Com isso, devido a essa estabilidade do fator de



(a) Método de Euler.

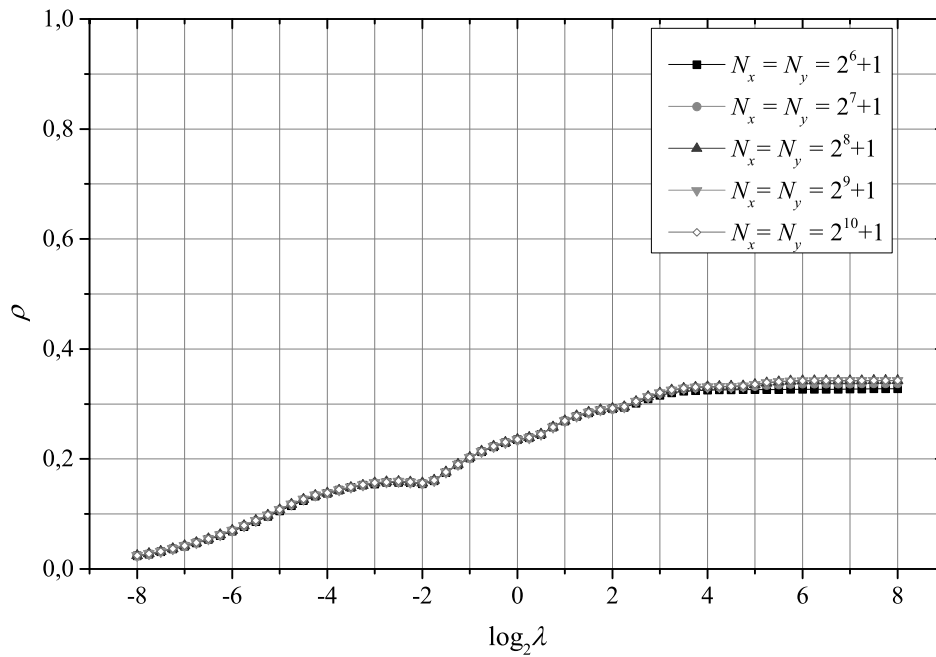


(b) Método de CN.

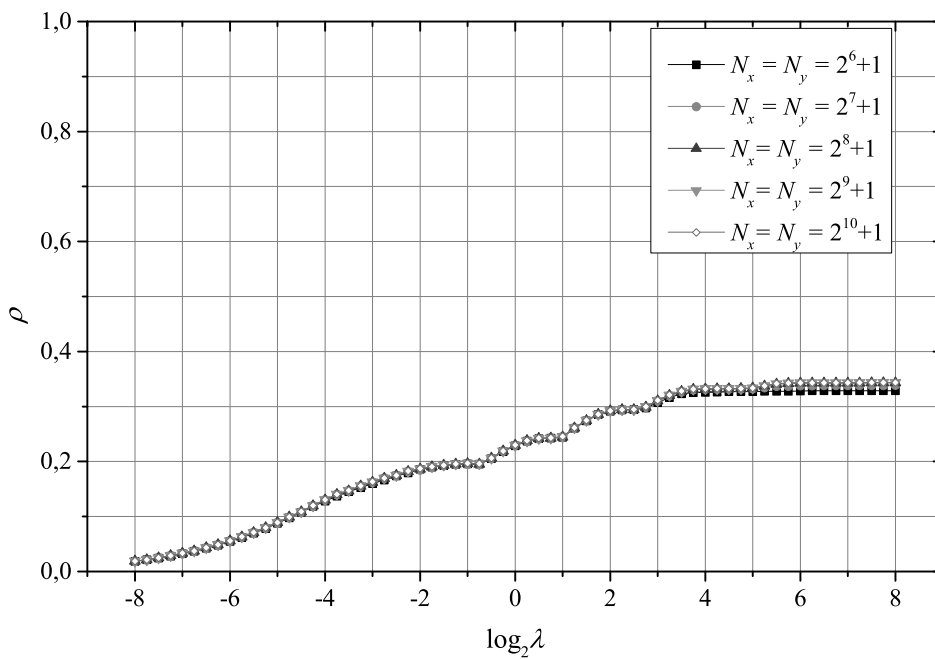
Figura 59 – Robustez do método *Waveform Relaxation - multigrid* e ciclo $V(0, 1)$ para resolver a equação do calor 2D. Neste caso considerou-se a malha espacial mais refinada fixa com $N_x = 2^5 + 1$.

convergência em relação à discretização de malha e ao fator de anisotropia, pode-se dizer que o método *Waveform Relaxation - multigrid* é robusto para qualquer ciclo *multigrid* e também para qualquer número de suavizações para aproximar a equação do calor 2D.

Para o método *Space-Time - multigrid* não serão apresentados aqui os testes de robustez. Horton e Vandewalle (1995) comentam que este método não é robusto com o ciclo V . Com alguns testes computacionais (que não são mostrados nessa tese) foi possível verificar que este método não é robusto com o ciclo F e apenas uma suavização interna.



(a) Método de Euler.



(b) Método de CN.

Figura 60 – Robustez do método *Waveform Relaxation - multigrid* e ciclo $V(0, 1)$ para resolver a equação do calor 2D. Neste caso considerou-se a aproximação temporal mais refinada fixa $N_t = 2^5 + 1$.

Com isso, pode-se concluir que esse método é robusto quando se usam os ciclos F com 2 ou mais suavizações ou o ciclo W .

9.2 Otimização dos parâmetros *multigrid*

Para verificar os parâmetros ótimos do método *multigrid* fizeram-se testes considerando o t_{CPU} em função do fator de anisotropia dado por $\log_2 \lambda$. Para esses testes usaram-se a discretização de malha com $N = 2^{13} + 1$ para o caso 1D e $N = 2^9 + 1$ para o caso 2D. O uso de apenas um valor para a discretização se deu pelo fato de que nos testes numéricos constatou-se que com diferentes discretizações de malhas os resultados qualitativos são semelhantes. Os ciclos a serem comparados são V , F e W e com um número de suavizações internas que permita verificar seu valor ótimo. O critério de parada usado foi $\frac{\|r(it)\|_\infty}{\|r(0)\|_\infty} \leq 10^{-7}$ para o método *Time-Stepping - multigrid*, em que $\|r(it)\|_\infty$ representa o máximo resíduo na iterada atual e $\|r(0)\|_\infty$ o correspondente resíduo considerando-se a estimativa inicial dada por números aleatórios entre 0 e 1, e $\frac{\|r(it)\|_\infty}{\|r(0)\|_\infty} \leq 10^{-10}$ para os demais métodos usados. O uso de um critério de parada distinto para o método *Time-Stepping - multigrid* dá-se pelo fato de que a estimativa inicial para os cálculos em um determinado passo de tempo é a solução encontrada no passo de tempo anterior. Portanto, nesse método a norma infinito do resíduo inicial $\|r(0)\|_\infty$ é menor, e não é possível alcançar o critério de parada $\frac{\|r(it)\|_\infty}{\|r(0)\|_\infty} \leq 10^{-10}$ devido à presença de possíveis erros de arredondamento (ver os tipos de erros em Marchi (2001)).

Nas Fig. 61 e Fig. 62 é mostrado, para os diversos tipos de ciclos e iterações internas, o t_{CPU} em função do fator de anisotropia λ para a equação do calor 2D com discretização espacial igual ao número de pontos da aproximação temporal dada por $N = 2^9 + 1$ e considerou-se o método *Time-Stepping - multigrid* com os métodos de Euler e de CN, respectivamente.

Com os resultados descritos nas Fig. 61 e Fig. 62 é possível observar que em ambos os métodos, Euler ou CN, em geral com 1 suavização interna, obtém-se os menores t_{CPU} . Sendo assim, para melhor visualização, esboçou-se em uma mesma figura (Fig. 63) os resultados obtidos com os ciclos V , F e W com 1 suavização interna.

Nos resultados mostrados na Fig. 63(a) para o método de *Time-Stepping - multigrid* com Euler, os parâmetros ótimos correspondem ao ciclo $V(0, 1)$ se $\log_2 \lambda \leq 1$ e $F(0, 1)$ se $\log_2 \lambda > 1$. Na Fig. 63(b) para o método *Time-Stepping - multigrid* com o método de CN, pode-se observar um comportamento semelhante ao apresentado na Fig. 63(a), porém os parâmetros ótimos correspondem ao ciclo $V(0, 1)$ se $\log_2 \lambda \leq 2$ e o ciclo $F(0, 1)$ se $\log_2 \lambda > 2$.

Os resultados dos testes numéricos que otimizam os parâmetros do método *Waveform Relaxation - multigrid*, para o caso 1D, são mostrados na Fig. 64 para o método de Euler e Fig. 65 para o método de CN. Observa-se que, em geral o ciclo V com 1 suavização interna apresenta os melhores resultados.

Os resultados referentes ao método *Waveform Relaxation - multigrid* para a equação

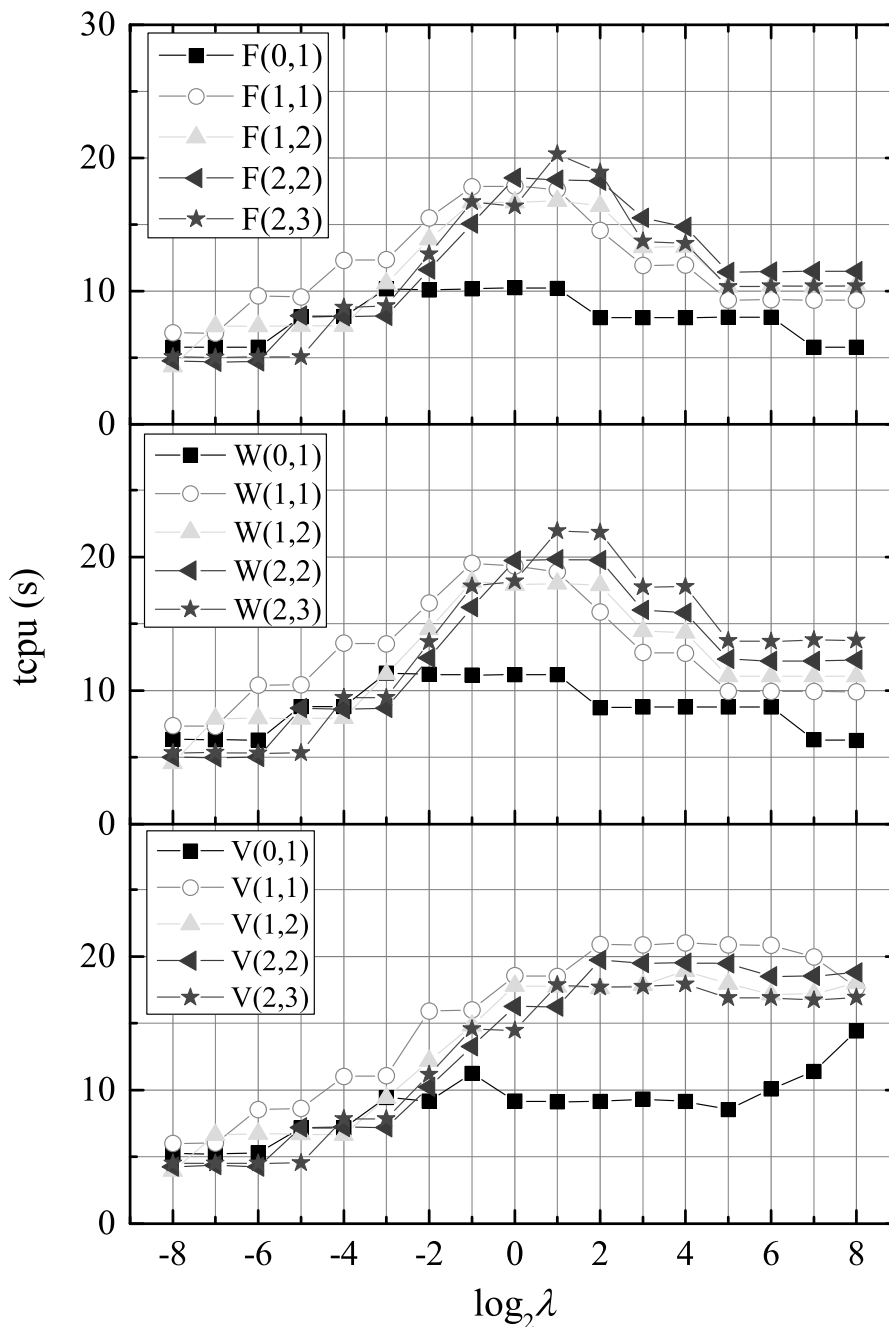


Figura 61 – Otimização dos parâmetros relacionados ao método *Time-Stepping - multigrid* para a equação do calor 2D com o método de Euler e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.

do calor 2D são expostos nas Fig. 66 e Fig. 67 com os métodos de Euler e de CN, respectivamente.

Com os resultados mostrados nas Fig. 66 e Fig. 67 observa-se que os parâmetros ótimos são: ciclo *V* com 1 ou 2 suavizações internas. Pode-se observar que para o método de *Waveform Relaxation - multigrid* com Euler, o parâmetro ótimo é o ciclo $V(0, 1)$ se $\log_2 \lambda \leq -4$ e $V(1, 1)$ se $\log_2 \lambda > -4$, enquanto que para o método de *Waveform Relaxation*

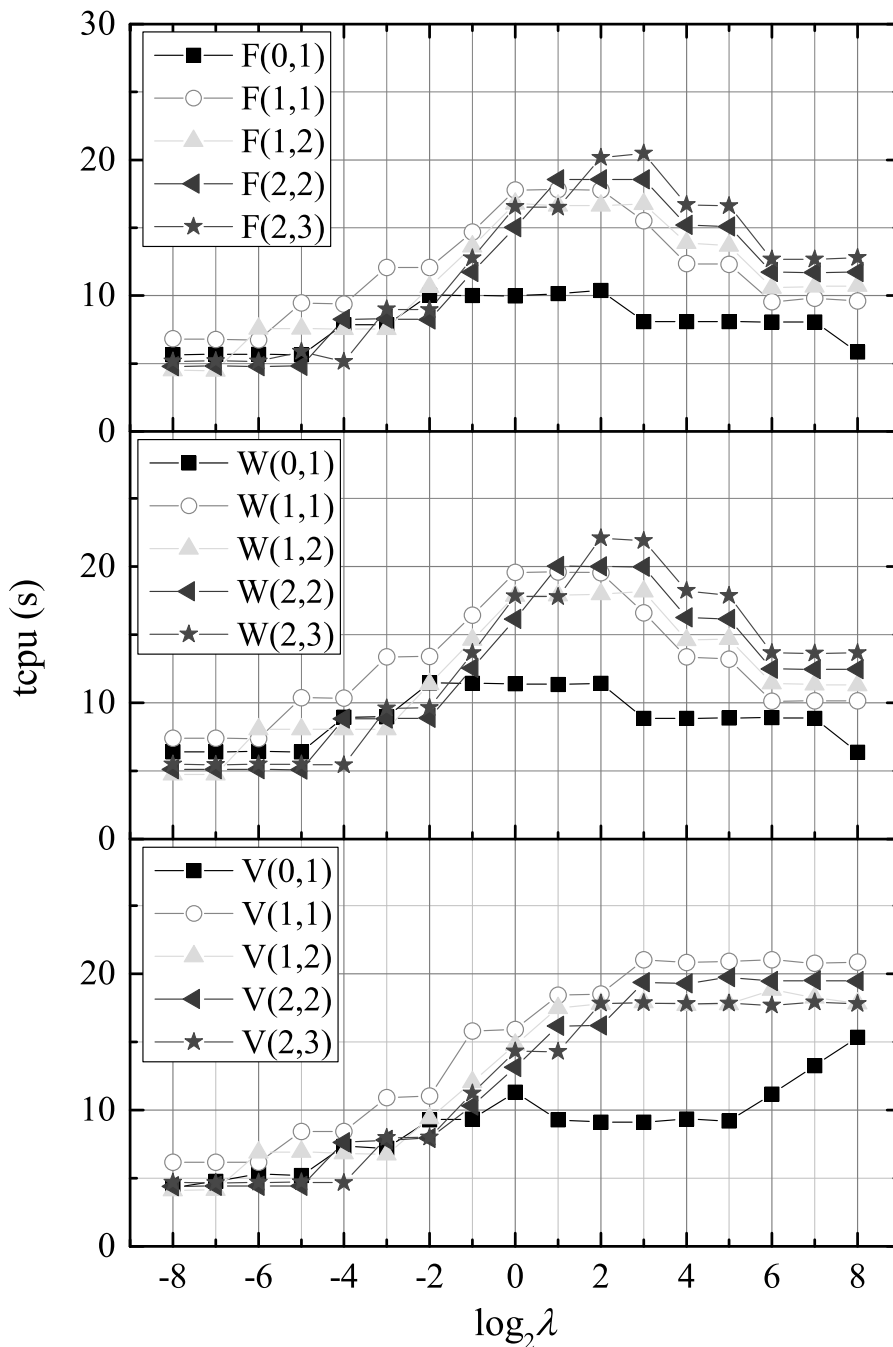
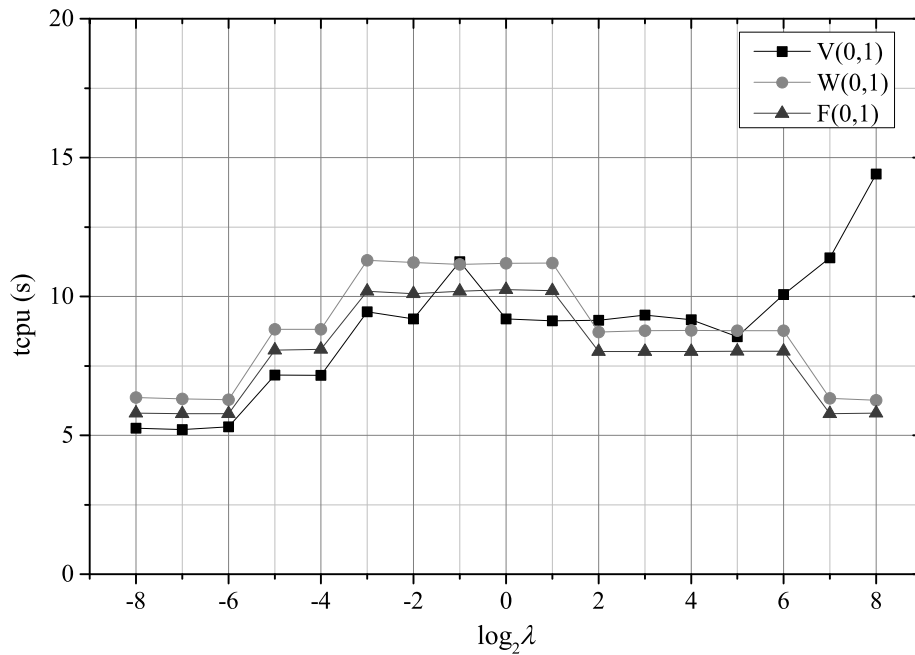


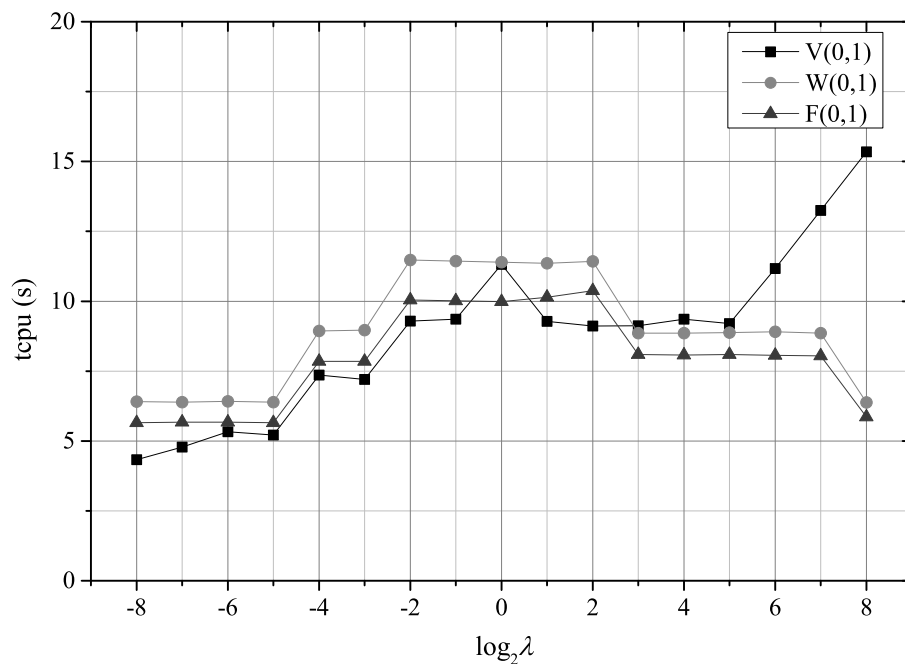
Figura 62 – Otimização dos parâmetros relacionados ao método *Time-Stepping - multigrid* para a equação do calor 2D com o método de CN e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.

- *multigrid* com CN o comportamento é muito semelhante em que o parâmetro ótimo é o ciclo $V(0,1)$ se $\log_2\lambda \leq -3$ e $V(1,1)$ se $\log_2\lambda > -3$. Uma comparação entre os métodos *Time-Stepping - multigrid* e *Waveform Relaxation - multigrid* está descrita em Franco et al. (2015).

Para o estudo do método *Space-Time - multigrid*, Horton e Vandewalle (1995) afirmam que com o ciclo V não é alcançada a robustez quando a malha é refinada



(a) Método de Euler.



(b) Método de CN.

Figura 63 – Otimização dos parâmetros relacionados ao método *Time-Stepping - multigrid* para a equação do calor 2D com os métodos de Euler e de CN, considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção e 1 suavização com os ciclos V , W ou F .

(Observou-se experimentalmente que com o ciclo F e apenas 1 suavização, a robustez também não é alcançada). A aproximação temporal com o método de Crank-Nicolson também não fornece bons resultados, como pode-se verificar na Fig. 68, retirada de Horton

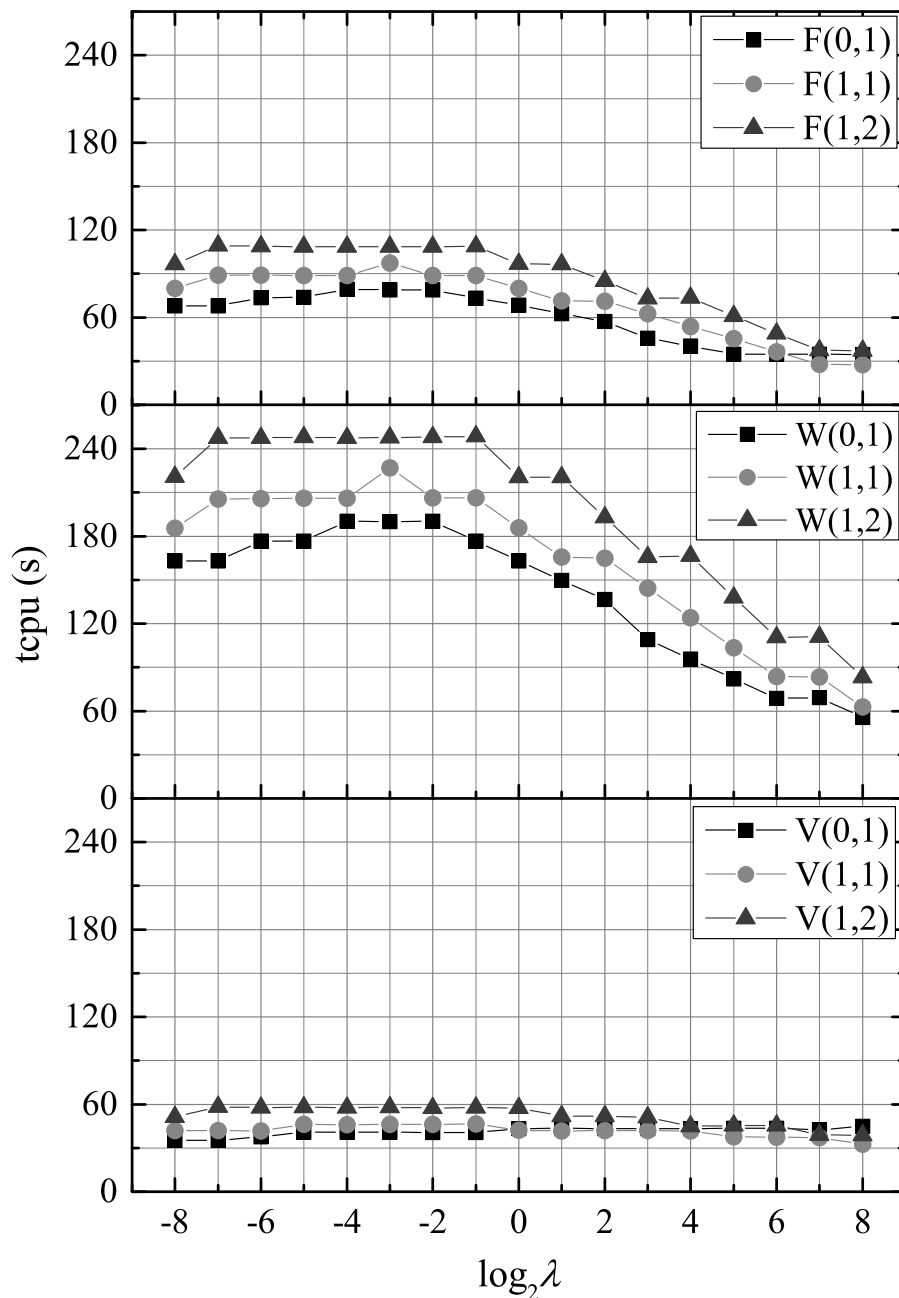


Figura 64 – Otimização dos parâmetros relacionados ao método *Waveform Relaxation - multigrid* para a equação do calor 1D com o método de Euler e considerando uma discretização com $N = 2^{13} + 1$ pontos em cada direção.

e Vandewalle (1995, p. 858). Assim, neste método será usado apenas o método de Euler para efetuar a aproximação temporal.

Observa-se na Fig. 68(a) que na interseção entre as linhas sólida e tracejada é definido λ_{crit} . O método *Space-Time - multigrid* consiste no uso de engrossamento apenas na direção temporal se $\lambda \leq \lambda_{crit}$ e engrossamento apenas na direção espacial se $\lambda > \lambda_{crit}$.

Na Fig. 68(b) pode-se verificar que com o uso do método de CN para o método

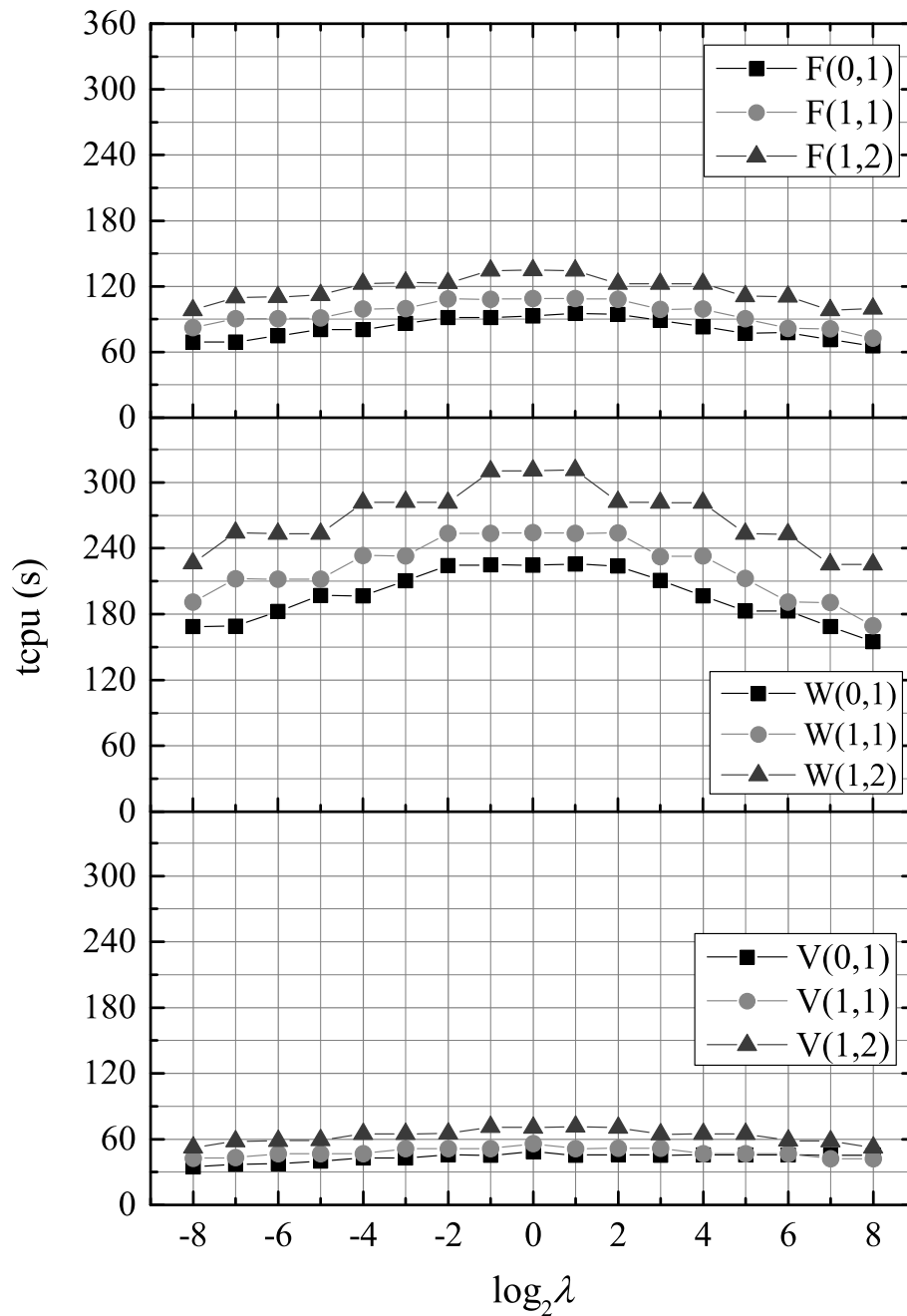


Figura 65 – Otimização dos parâmetros relacionados ao método *Waveform Relaxation - multigrid* para a equação do calor 1D com o método de CN e considerando uma discretização com $N = 2^{13} + 1$ pontos em cada direção.

Space-Time - multigrid não existe um fator de convergência pequeno se λ é grande, ou seja, quando λ cresce não existe uma forma de engrossamento para resolver de forma eficiente o referido problema.

Os resultados numéricos referentes ao método *Space-Time - multigrid* para as equações do calor 1D e 2D são expostos nas Fig. 69 e Fig. 70, respectivamente. Nestes experimentos usou-se $\lambda_{crit} = 0,6$ obtido através da LFA, ver Horton e Vandewalle (1995)

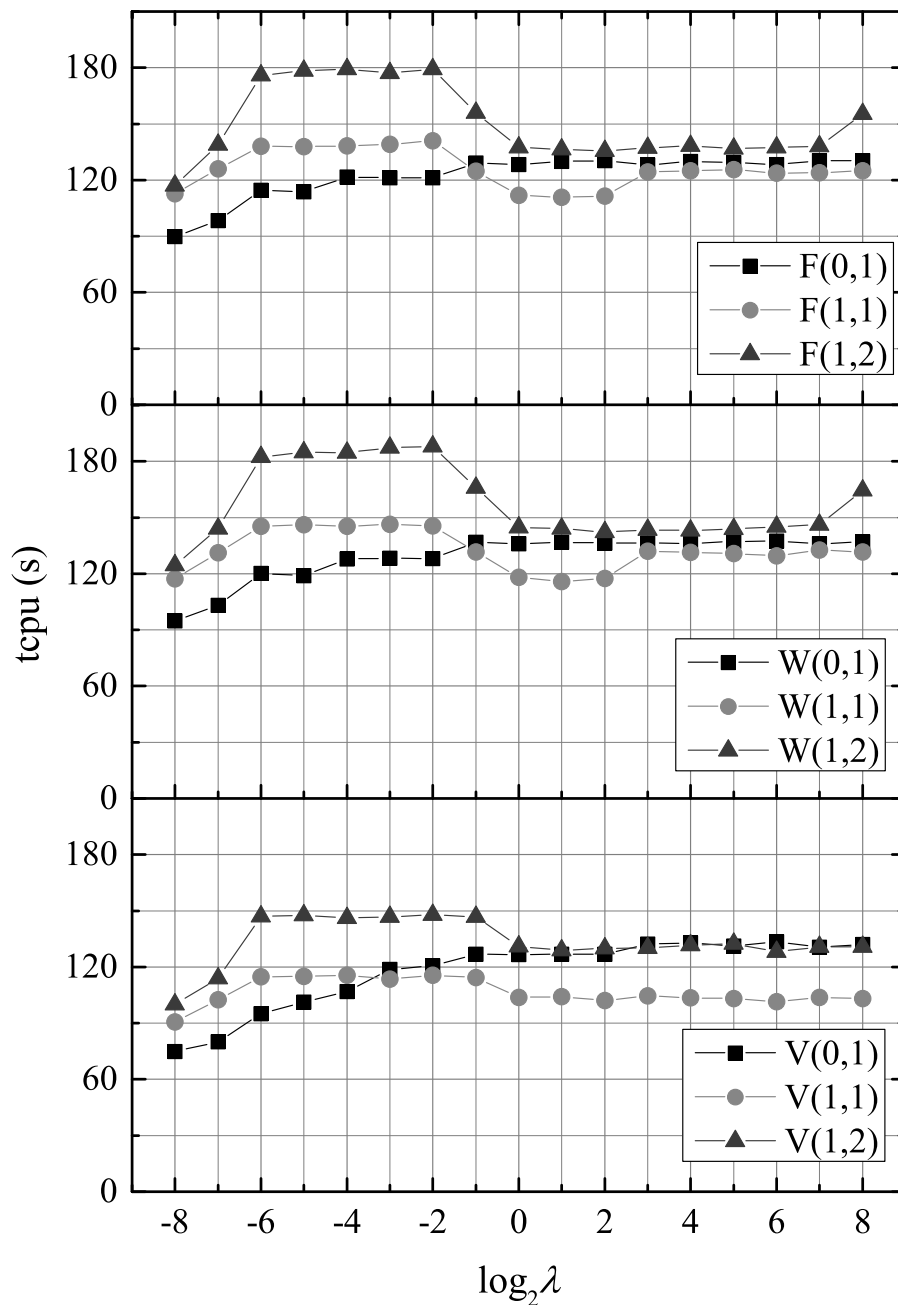


Figura 66 – Otimização dos parâmetros relacionados ao método *Waveform Relaxation - multigrid* para a equação do calor 2D com o método de Euler e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.

e Falgout et al. (2017).

Pode-se observar nas Figs. 69 e 70 que para ambas as equações do calor 1D e 2D os melhores t_{CPU} ocorrem quando é usado o ciclo F com 2 ou 3 suavizações internas. No caso 1D os resultados com os referidos ciclos $F(1,1)$ e $F(1,2)$ são muito parecidos e eles se alternam como valores ótimos. Para o caso 2D o ciclo $F(1,1)$ apresenta melhores t_{CPU} se $\log_2\lambda \leq \lambda_{crit}$ e $F(1,2)$ se $\log_2\lambda > \lambda_{crit}$. Horton e Vandewalle (1995) usaram o ciclo F

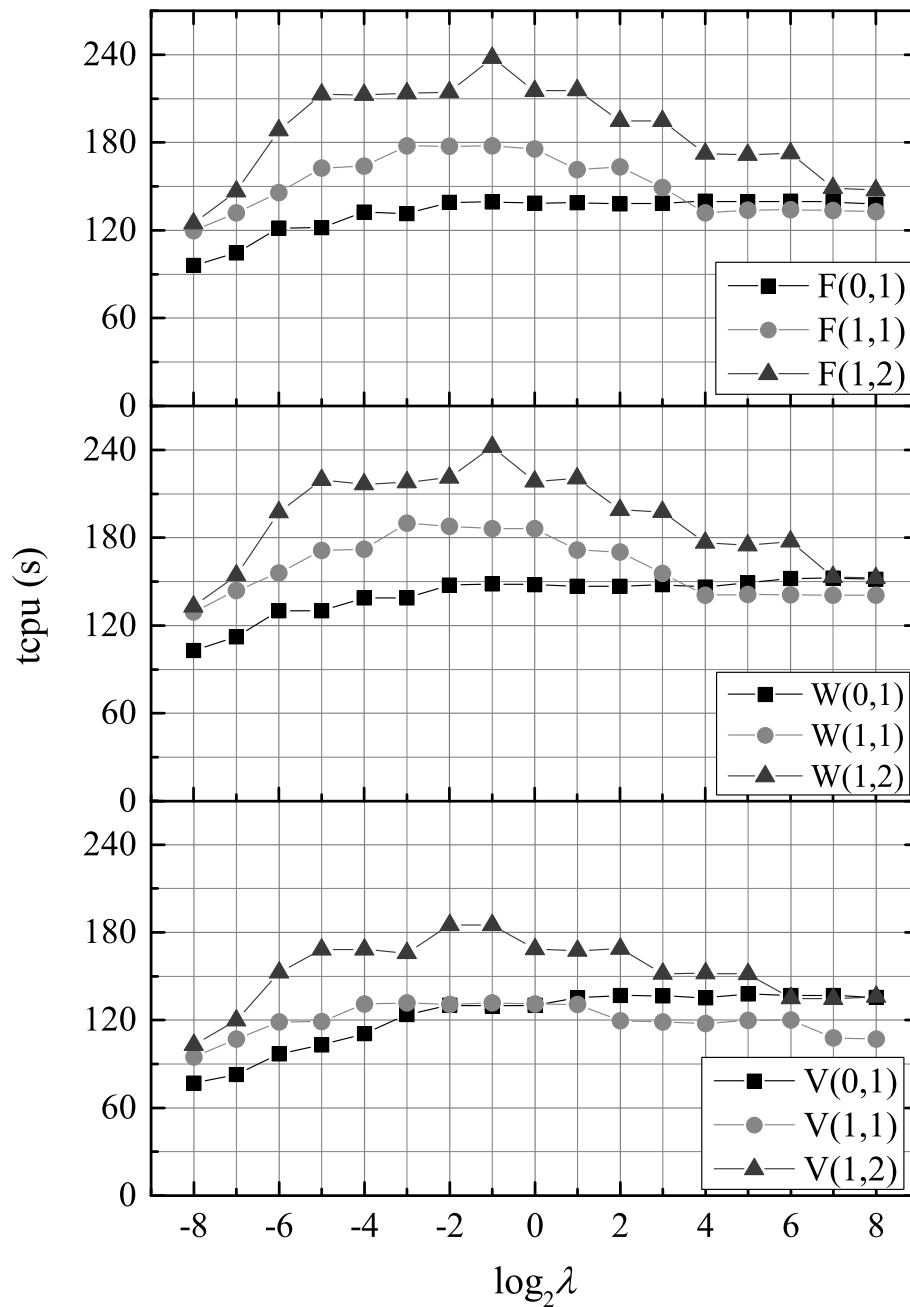
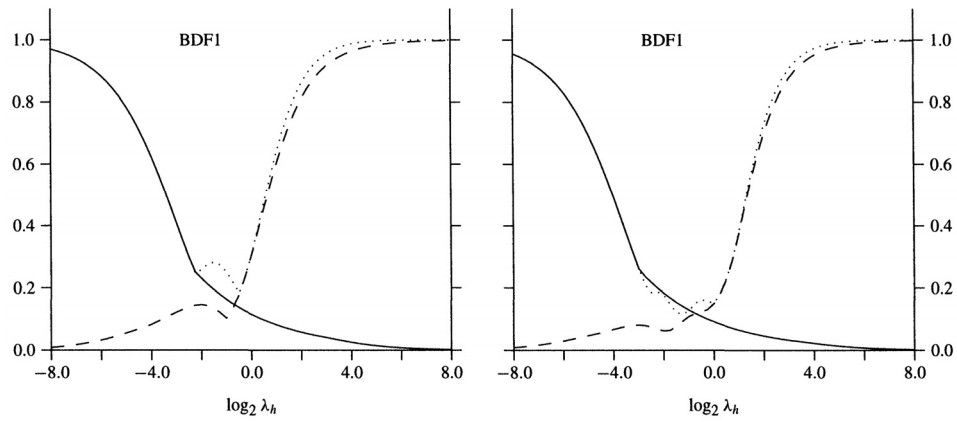
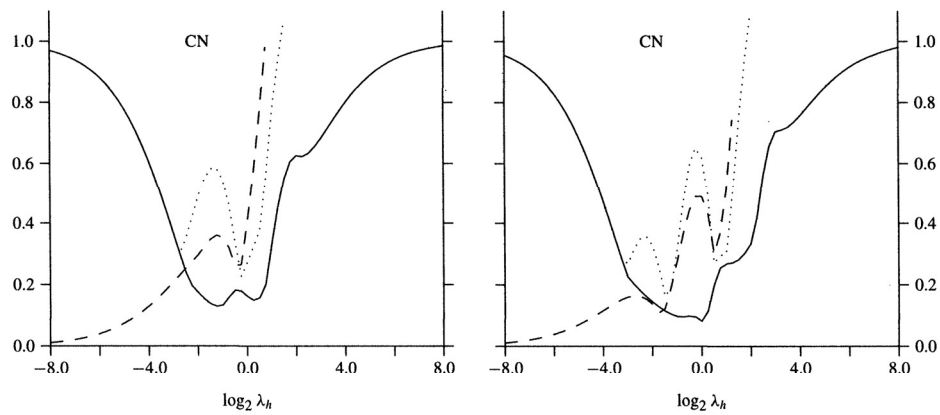


Figura 67 – Otimização dos parâmetros relacionados ao método *Waveform Relaxation - multigrid* para a equação do calor 2D com o método de CN e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.

com 3 suavizações para os cálculos numéricos.



(a) Método de Euler.



(b) Método de CN.

Figura 68 – Fator de convergência ρ obtido através da análise de duas malhas com LFA para o método de Euler (BDF1) e o método de Crank-Nicolson (CN), com 2 passos de suavização (esquerda) e 3 passos de suavização (direita). A linha sólida indica engrossamento apenas na direção espacial, a linha tracejada indica engrossamento apenas na direção temporal e a linha pontilhada indica engrossamento padrão. Figura retirada de Horton e Vandewalle (1995, p. 858).

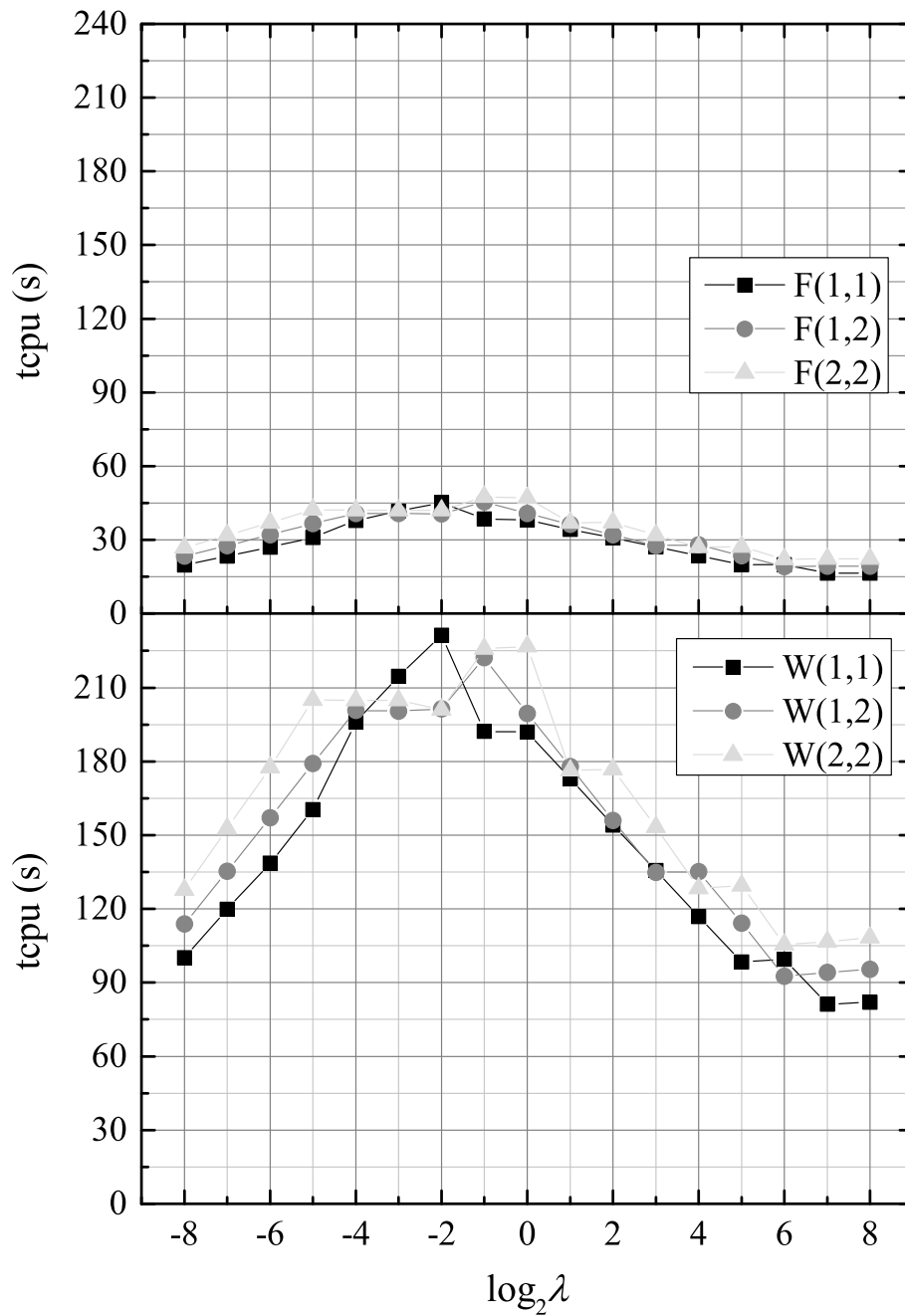


Figura 69 – Otimização dos parâmetros relacionados ao método *Space-Time - multigrid* para a equação do calor 1D com o método de Euler e considerando uma discretização com $N = 2^{13} + 1$ pontos em cada direção.

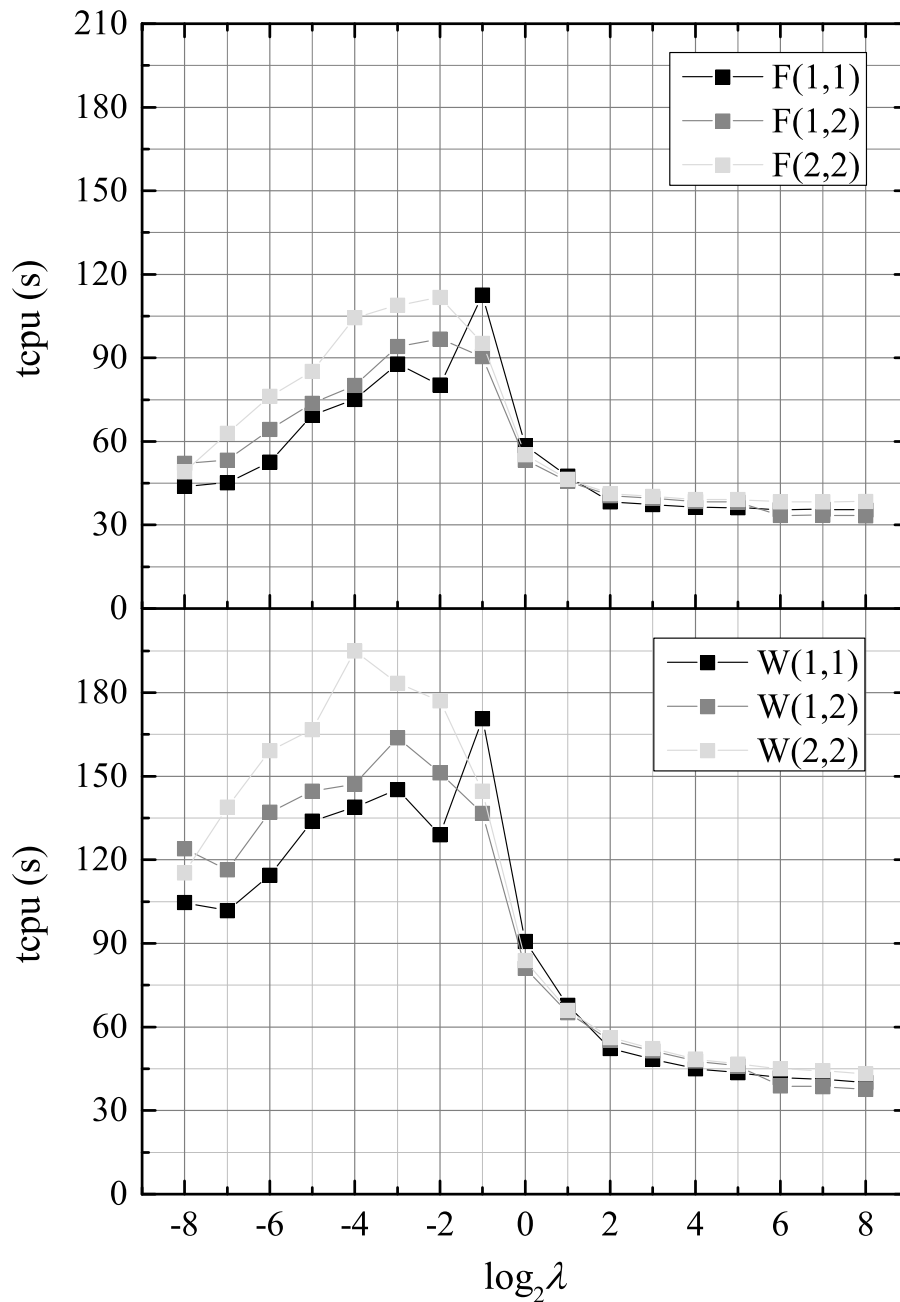


Figura 70 – Otimização dos parâmetros relacionados ao método *Space-Time - multigrid* para a equação do calor 2D com o método de Euler e considerando uma discretização com $N = 2^9 + 1$ pontos em cada direção.

9.3 Comparação dos métodos estudados

Com esse estudo e as simulações numéricas foi possível verificar os parâmetros que otimizam o uso do t_{CPU} para os casos estudados, como mostra a Tab. 10.

Com os resultados descritos nessa seção e os parâmetros ótimos indicados na Tab. 10 pode-se verificar que, fazendo as simulações com apenas um núcleo de programação

Tabela 10 – Parâmetros que otimizam o t_{CPU} para resolver as equações do calor 1D e 2D usando os métodos *Time-Stepping - multigrid*, *Waveform Relaxation - multigrid* e *Space-Time - multigrid*.

Método	Dimensão	Aprox. no tempo	Parâmetros ótimos
<i>Time-Stepping - multigrid</i>	2D	Euler	$V(0, 1)$ se $\lambda \leq 2$ $F(0, 1)$ se $\lambda > 2$
		CN	$V(0, 1)$ se $\lambda \leq 4$ $F(0, 1)$ se $\lambda > 4$
<i>Waveform Relaxation - multigrid</i>	1D	Euler	$V(0, 1)$
		CN	$V(0, 1)$
	2D	Euler	$V(0, 1)$ se $\lambda \leq 2^{-4}$ $V(1, 1)$ se $\lambda > 2^{-4}$
		CN	$V(0, 1)$ se $\lambda \leq 2^{-3}$ $V(1, 1)$ se $\lambda > 2^{-3}$
<i>Space-Time - multigrid</i>	1D	Euler	$F(1, 1)$
	2D	Euler	$F(1, 1)$ se $\lambda \leq \lambda_{crit}$ $F(1, 2)$ se $\lambda > \lambda_{crit}$

(programação sequencial, ou seja, não usando programação paralela), o método *Time-Stepping - multigrid* (mesmo com $tol = 10^{-7}$) é muito mais eficiente que os métodos *Waveform Relaxation - multigrid* e *Space-Time - multigrid*. Na comparação entre os dois últimos, o método *Space-Time - multigrid* apresenta uma leve superioridade em relação ao método *Waveform Relaxation - multigrid*.

Falgout et al. (2017) fizeram uma comparação entre os métodos *Time-Stepping - multigrid*, *waveform relaxation* com redução cíclica (WRMG-CR), *Space-Time - multigrid* e *Space-Time* com suavização por blocos (STMG-BR) usando paralelismo com muitos núcleos de processamento. O método WRMG-CR habilita o paralelismo completo na direção temporal do método *Waveform Relaxation - multigrid* (ver Horton et al. (1995)). A Fig. 71 (retirada de Falgout et al. (2017)) mostra o t_{CPU} usado para resolver a equação do calor 2D com discretização espacial e temporal dada por $128^2 \times 16.384$ (equivalente a $N = 2^7 + 1$ pontos na discretização de cada direção espacial e 16.384 pontos na direção temporal) com os métodos *Time-Stepping - multigrid*, WRMG-CR, *Space-Time - multigrid*, o método *multigrid* com redução no tempo (MGRIT), proposto em Falgout et al. (2017) e o método STMG-BR proposto em Gander e Neumüller (2016). O método MGRIT e STMG-BR não serão abordados nesta tese, apesar de apresentarem excelentes resultados ao se usar paralelismo de computadores.

Com os resultados descritos na Fig.71 pode-se afirmar que se for usado paralelismo de computadores, o método *Space-Time - multigrid* apresenta melhores resultados em

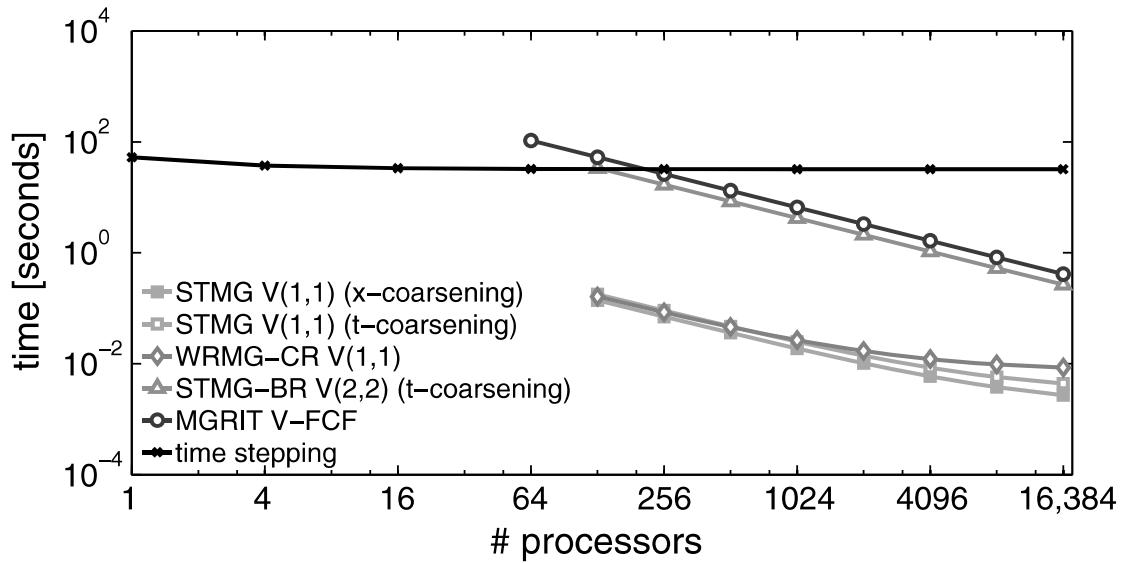


Figura 71 – Tempo computacional para resolver a equação do calor 2D usando paralelização de computadores, com o método de Euler e considerando uma discretização espacial com $N_x = N_y = 2^7 + 1$ pontos e temporal com $N_t = 16.384$ pontos. Figura retirada de Falgout et al. (2017, p. 140).

comparação com os métodos *Waveform Relaxation - multigrid* e *Time-Stepping - multigrid*. Pode-se afirmar também que o método *Time-Stepping - multigrid*, por ser paralelizável apenas nas direções espaciais não é um bom método para ser usado com paralelização de computadores. Vale lembrar que este método apresentou bons resultados quando foi resolvido com apenas um núcleo de processamento.

10 Resultados

Neste capítulo são apresentados os principais resultados obtidos no desenvolvimento desta tese, que são basicamente: 1) o desenvolvimento de um método *Space-Time - multigrid* com engrossamento padrão aplicado à equação do calor e; 2) o método *Waveform Relaxation - multigrid* usado nas equações da poroelasticidade.

10.1 Equação do calor

Com o objetivo de entender melhor os métodos usados na solução de problemas parabólicos, buscou-se auxílio adicional na literatura sobre os métodos *multigrid* para a solução de problemas elípticos com anisotropias. O problema modelo dado pela equação do calor (eq. de Fourier) é um problema parabólico, porém, para um passo de tempo fixo, esse problema pode ser considerado elíptico (no espaço). Considerando-se a relação espaço e tempo, esse problema apresenta um fator de anisotropia dado por $\lambda = \frac{h}{\tau^2}$. De acordo com os estêncis dados nas Eq. (6.15) e Eq. (6.22) verifica-se que a equação do calor apresenta fortes anisotropias para valores pequenos ou grandes de λ , e nesta tese consideraram-se $\lambda < 2^{-3}$ como pequeno e $\lambda > 2^2$ como grande. Thole e Trottenberg (1986) descrevem que para o método *multigrid* em problemas anisotrópicos são duas estratégias que dão origem a boas taxas (ou fatores) de convergência. Por um lado, o uso de suavizador padrão por pontos combinado com uma estratégia de semiengrossamento na direção da maior anisotropia (em que ocorre o maior acoplamento). Por outro lado, pode-se usar engrossamento padrão se um suavizador mais forte for considerado. Em particular, seguindo a regra fundamental de relaxação por blocos (BRANDT; LIVNE, 2011), as variáveis que estão fortemente acopladas devem ser atualizadas (suavizadas) simultaneamente, ou seja, deve-se usar um suavizador linha (ou plano) na direção da maior anisotropia.

Llorente e Melson(1998,2000) usaram as relações descritas por Thole e Trottenberg (1986) e apresentaram um suavizador robusto e um esquema usando relaxação por planos para resolver a equação de Poisson 3D com fortes anisotropias. Prieto et al. (2001) usaram relaxação por planos e o esquema de semiengrossamentos para resolver a equação da difusão 3D anisotrópica.

Com o exposto acima, verifica-se que o método *Time-Stepping - multigrid* apresenta bons resultados devido ser uma EDP elíptica em cada passo de tempo.

O método *Waveform Relaxation* por sua vez, apresenta bons resultados para todos os fatores de anisotropia devido ao uso de engrossamento na direção espacial e um suavizador linha na direção temporal, ou seja, se o forte acoplamento ocorre na direção

espacial, o engrossamento nessa direção faz com que o método *multigrid* seja eficiente para esse caso, e se o forte acoplamento ocorre na direção temporal, o suavizador linha na direção do tempo garante que o método *multigrid* continue sendo eficiente.

Observam-se os fatores de convergência para o referido método *Waveform Relaxation - multigrid* na Fig. 55. Pode-se perceber que, mesmo não se usando um suavizador linha no tempo, ao usar-se engrossamento na direção espacial e um suavizador por pontos no espaço (método parabólico), os resultados são bons apenas para λ grande. Ou seja, como não se usou um suavizador linha no tempo, quanto maior o acoplamento nesta direção, piores são os resultados.

Por fim, nessa mesma Fig. 55 pode-se observar que o método *Space-Time - multigrid* apresenta bons resultados quando o semiengrossamento ocorre na direção de maior acoplamento. Portanto, considerando-se os semiengrossamentos nas direções espaciais e temporal, o método *Space-Time - multigrid* funciona bem para todos os valores de λ .

Dessa forma, pensou-se em uma forma de resolver a equação do calor usando a ideia já existente para problemas elípticos com anisotropias. Um novo método, proposto neste trabalho, consiste em usar engrossamento padrão em todas as direções espaciais e temporal e um suavizador zebra (com linhas ou planos de relaxação) sempre na direção de mais forte acoplamento. Esse método será designado método *Space-Time - multigrid* com engrossamento padrão (ver também em Franco et al. (2018a)).

10.1.1 Descrição do método *Space-Time - multigrid* com engrossamento padrão

O método *Space-Time - multigrid* com engrossamento padrão, que será apresentado aqui, consiste em usar engrossamento padrão em todos os níveis de malha, um apropriado operador de restrição e prolongação (HORTON; VANDEWALLE, 1995), e uma estratégia de suavização baseada em um processo que depende do grau de anisotropia (λ) de cada malha hierarquicamente. O parâmetro $\lambda = \frac{\tau}{h^2}$ definido na Eq. (6.8) indica o grau de anisotropia do operador discreto, que deve ser levado em consideração na escolha das componentes do método *multigrid*. Opostamente ao método *Space-Time - multigrid* (HORTON; VANDEWALLE, 1995) em que é usada a estratégia de diferentes semiengrossamentos, aqui é proposto um método *Space-Time - multigrid* que combina diferentes suavizadores e leva em consideração as conexões de forte acoplamento dependentes do parâmetro λ nos estêncis que representam o modelo numérico discretizado (ver as Eq. (6.15) e Eq. (6.22)). Observa-se que o valor de λ varia de nível de malha para nível de malha hierarquicamente e, portanto, devem ser escolhidos suavizadores convenientes nas diferentes malhas.

Estratégias de engrossamento

O método *Space-Time - multigrid* com engrossamento padrão, como o próprio nome indica, utiliza engrossamento padrão para construir as malhas hierarquicamente. O tamanho do espaçamento entre os nós h ou τ na malha consecutivamente mais grossa possui o dobro do tamanho do espaçamento da malha fina em cada direção (em ambas as direções espaciais e temporal). De qualquer maneira, devem ser escolhidos operadores de restrição e de prolongação apropriados. Em particular segue-se Horton e Vandewalle (1995) e utiliza-se o operador de prolongação, I_{2h}^h , de forma que não se transfira informações para trás no tempo, enquanto o operador de restrição, I_h^{2h} , é assimétrico e não transfere nenhuma informação para a frente no tempo. Esses operadores, para o caso 1D, são apresentados na Eq. (10.1)

$$I_{2h}^h = \frac{1}{2} \begin{bmatrix} 1 & 2 & 1 \\ 1 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix}_{2h}^h, \quad I_h^{2h} = \frac{1}{8} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \\ 1 & 2 & 1 \end{bmatrix}_h^{2h}. \quad (10.1)$$

Nesses estêncis, em cada linha representa-se a evolução no tempo (de baixo para cima) e em cada coluna, a discretização no espaço (da esquerda para a direita).

Para o caso 2D, os estêncis são apresentados na Eq. (10.2)

$$I_{2h}^h = \frac{1}{4} \begin{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{2h}^h & \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_{2h}^h & \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_{2h}^h \end{bmatrix}, \quad (10.2)$$

$$I_h^{2h} = \frac{1}{32} \begin{bmatrix} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h^{2h} & \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h^{2h} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_h^{2h} \end{bmatrix}.$$

Observa-se que a notação de estêncil na Eq. (10.2) corresponde a uma sequência de estêncis aplicados em sucessivos passos de tempo, do mais baixo (estêncil esquerdo) para o mais alto (estêncil direito).

Estratégias de suavização

Uma estratégia de suavização adaptativa, que depende do valor do parâmetro λ , é o ingrediente chave para a proposta do método *Space-Time - multigrid* com engrossamento padrão. Essa abordagem é baseada na escolha, em cada nível de malha hierarquicamente, de um suavizador que é adequado para o valor atual de λ .

Para o caso 1D, consideram-se os seguintes suavizadores para o problema parabólico

- **Suavizador zebra com linha no tempo:** Todas as incógnitas localizadas em um mesmo ponto espacial, mas de diferentes passos de tempo são atualizadas simultaneamente. Observa-se que isso pode ser feito de forma eficiente, uma vez que a matriz resultante da discretização do sistema é bidiagonal. Além disso, as linhas na direção temporal são visitadas em forma de zebra. Dessa forma, as primeiras incógnitas suavizadas são colocadas nos nós espaciais ímpares e, depois disso, atualizam-se as linhas cujos pontos espaciais são pares.
- **Suavizador zebra com linha no espaço:** é o mesmo algoritmo citado acima, mas, desta vez, as linhas estão na direção espacial e, claro, todas as incógnitas em cada linha devem ser atualizadas simultaneamente, resolvendo o sistema de equações correspondente. Para a solução desse sistema de equações pode-se usar o método TDMA.

Para o caso 2D, os suavizadores são:

- **Suavizador linha no tempo *red-black*:** Este suavizador é semelhante ao *suavizador zebra com linha no tempo* descrito anteriormente no caso unidimensional, com a única diferença de que as linhas são visitadas seguindo a ordem *red-black* (Veja a Fig. 7) dentro da malha espacial bidimensional.
- **Suavizador zebra com planos no espaço:** Aqui todas as incógnitas localizadas no mesmo passo de tempo são atualizadas simultaneamente e os planos resultantes são visitados seguindo a forma de zebra. Teoricamente, esse processo parece ser caro devido à necessidade de resolver exatamente os problemas 2D que surgem no relaxamento por planos. No entanto, na prática, geralmente não é necessário resolvê-los exatamente, e é suficiente aplicar um ciclo *multigrid* em cada plano (ver por exemplo, Brandt e Livne (2011), Trottenberg et al. (2001) e Llorente e Melson (1998)). Aqui, verificou-se que, com apenas 1 ciclo V e com $\nu_1 = 1$ e $\nu_2 = 2$ (pré e pós-suavização) e usando restrição por ponderação completa e interpolação bilinear (TROTTEMBERG et al., 2001) para o relaxamento por planos, o ciclo *multigrid* resulta em um algoritmo *multigrid* no espaço e tempo muito eficiente (ver a seção 10.1.3).

Propõe-se nesse método um valor crítico λ_{crit} que é usado como um interruptor para escolher a estratégia de suavização usada para cada nível de malha. Essa estratégia está descrita no Algoritmo 12.

Um ponto-chave para o bom desempenho do método *Space-Time - multigrid* com engrossamento padrão é a escolha do parâmetro λ_{crit} . Esse parâmetro indica quando escolher um ou outro suavizador. Para encontrar o valor de λ_{crit} contou-se com a ajuda da

Algoritmo 12: Estratégia de suavização

if $\lambda \leq \lambda_{crit}$ **then**

 Use *Suavizador zebra com linha no tempo* para o caso 1D ou *Suavizador linha no tempo red-black* para o caso 2D.

else

 Use *Suavizador zebra com linha no espaço* para o caso 1D ou *Suavizador zebra com planos no espaço* para o caso 2D.

end if

análise de Fourier local (LFA), que é uma análise quantitativa para estudar as propriedades de suavização e o operador de duas malhas com a estrutura do método *multigrid*. A forma de como desenvolver a LFA para esse caso pode ser encontrado também em Franco et al. (2018a).

Análise de Fourier local (LFA)

A base da LFA é formalmente estendida para todas as componentes *multigrid* para uma malha infinita $\Omega_{h,\tau}$, desprezando as condições de contorno e analisando operadores discretos lineares com coeficientes constantes. Desta forma, as autofunções de tais operadores são algumas funções exponenciais complexas chamadas componentes de Fourier, dadas por

$$\phi_{h,\tau}(\boldsymbol{\theta}, x, t) = e^{i\theta_1 x} e^{i\theta_2 t}, \quad (10.3)$$

em que $(x, t) \in \Omega_{h,\tau}$ e $\boldsymbol{\theta} = (\theta_1, \theta_2) \in \Theta_{h,\tau} = (-\pi/h, \pi/h] \times (-\pi/\tau, \pi/\tau]$. Essas funções de malha formam uma base unitária para o espaço de funções delimitadas na malha infinita e no espaço de Fourier dado por

$$\mathcal{F}(\Omega_{h,\tau}) := \text{span}\{\phi_{h,\tau}(\boldsymbol{\theta}, \cdot) \mid \boldsymbol{\theta} \in \Theta_{h,\tau}\}. \quad (10.4)$$

Considerando-se que o erro é uma combinação linear dos componentes de Fourier, o comportamento do método *multigrid* pode ser analisado através da avaliação da redução do erro associado a cada componente *multigrid* particular nos modos de Fourier.

Para investigar a interação entre a relaxação e a correção na malha grossa é necessário realizar pelo menos uma análise de duas malhas. É preciso considerar o operador de propagação de erros do método de duas malhas dado por

$$M_h^{2h} = S_h^{\nu_2} (I_h - I_{2h}^h L_{2h}^{-1} I_h^{2h} L_h) S_h^{\nu_1}, \quad (10.5)$$

em que S_h é o operador de suavização, I_h é o operador identidade, L_h e L_{2h} são os operadores de correção nas malhas fina e grossa, respectivamente, e os operadores de transferência entre as malhas são o operador de restrição I_h^{2h} , e o operador de prolongação I_{2h}^h . A análise de duas malhas é a base para as estimativas de convergência assintóticas

clássicas do método *multigrid* e o raio espectral do operador de duas malhas, $\rho_{loc}(M_h^{2h})$, indica o fator de convergência assintótico do método de duas malhas. Para estimar esse valor, a observação crucial é que o operador de correção na malha grossa, bem como o suavizador, deixa os chamados espaços de harmônicos- $2h$ invariantes. Tais espaços são definidos por

$$\mathcal{F}^4(\boldsymbol{\theta}^{00}) = \text{span}\{\phi_{h,\tau}(\boldsymbol{\theta}^{\alpha_1\alpha_2}, x, t) \mid \alpha_1\alpha_2 \in \{0, 1\}\}, \quad (10.6)$$

com $\boldsymbol{\theta}^{00} \in \Theta_{baixo} = (-\pi/2h, \pi/2h] \times (-\pi/2\tau, \pi/2\tau]$, e $\boldsymbol{\theta}^{\alpha_1\alpha_2} = \boldsymbol{\theta}^{00} - (\alpha_1 \text{sign}(\theta_1^{00})\pi, \alpha_2 \text{sign}(\theta_2^{00})\pi)$. Assim, M_h^{2h} é equivalente a uma matriz diagonal por blocos, consistindo de blocos denotados por $\widetilde{M}_h^{2h}(\boldsymbol{\theta}^{00}) = M_h^{2h}|_{\mathcal{F}^4(\boldsymbol{\theta}^{00})}$, que é a representação do domínio de Fourier. Desta forma, pode-se determinar o raio espectral $\rho_{loc}(M_h^{2h})$ calculando o raio espectral dessas matrizes menores, ou seja:

$$\rho_{loc} = \rho_{loc}(M_h^{2h}) = \sup_{\boldsymbol{\theta}^{00} \in \Theta_{baixo}} \rho_{loc}(\widetilde{M}_h^{2h}(\boldsymbol{\theta}^{00})). \quad (10.7)$$

Propõe-se neste método *Space-Time - multigrid* com engrossamento padrão, combinar o uso de dois suavizadores linha em direções distintas, um na direção temporal e outro na direção espacial, ambos realizados de forma zebra. Nesse processo, analisa-se para quais valores do parâmetro λ esses suavizadores proporcionam boa taxa de convergência para o algoritmo *multigrid*. Na Fig. 72 são mostrados os fatores de convergência em duas malhas ρ_{loc} fornecidos pela LFA, correspondentes aos dois métodos *multigrid* com base nos dois suavizadores considerados (zebra com linha no tempo e zebra com linha no espaço), para diferentes valores de $\log_2 \lambda$ variando de -8 a 8 e aplicado na equação do calor 1D. Pode-se observar, como esperado, que se o suavizador zebra com linha no tempo for usado, obtém-se bons resultados para valores pequenos de λ ($\lambda \leq \lambda_{crit}$), enquanto que o suavizador zebra com linha no espaço é adequado quando λ é grande ($\lambda > \lambda_{crit}$). Para confirmar as previsões da LFA, também são apresentados os fatores de convergência médio obtidos experimentalmente (ρ_m). Em particular, esses fatores de convergência são obtidos com a média desses fatores (ver a Eq. (4.10)) após 30 iterações *multigrid* com o ciclo $W(0, 1)$.

Verifica-se que o valor de λ para o qual o fator de convergência do método *multigrid* para ambos os suavizadores são iguais, é $\lambda = 0,46$, que, portanto, é escolhido como o valor do interruptor λ_{crit} . Este parâmetro será considerado nos testes numéricos. Os casos envolvendo a equação do calor 2D e também os casos com dupla discretização (seção 10.1.5) apresentaram $\lambda_{crit} \approx 0,46$ através da LFA e dos experimentos numéricos e não serão expostos aqui.

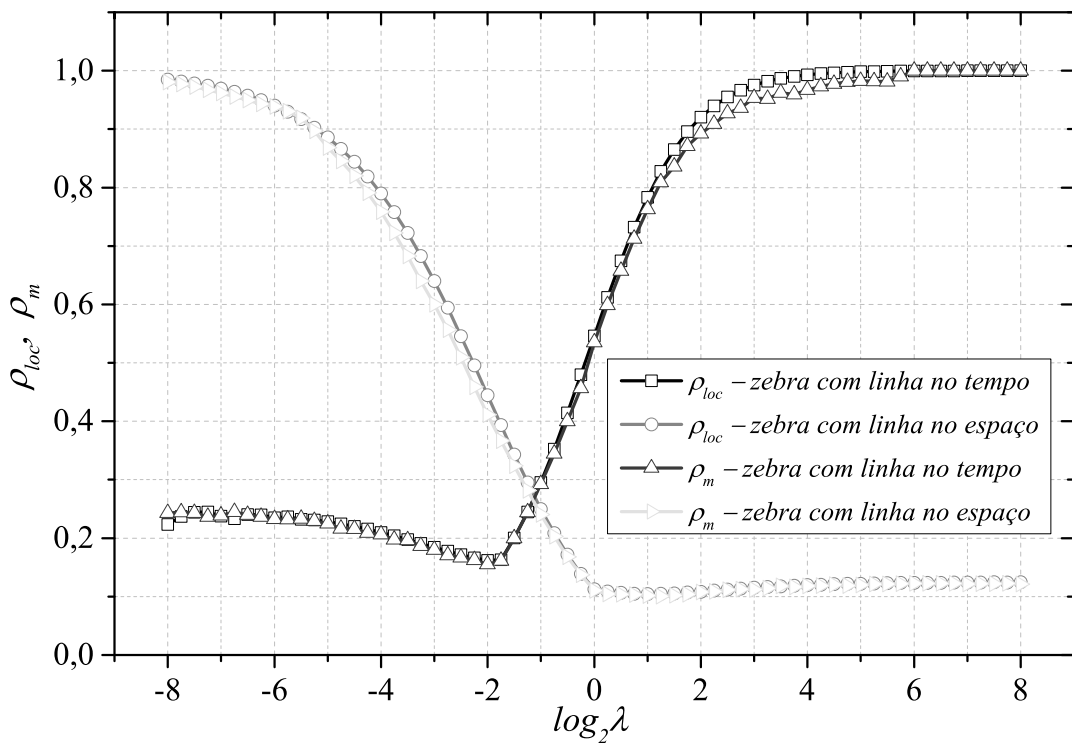


Figura 72 – Fatores de convergência preditos por LFA (ρ_{loc}) correspondentes ao método *multigrid* com suavizadores zebra com linha no tempo e zebra com linha no espaço, e fatores de convergência médio (ρ_m) obtido experimentalmente, para diferentes valores do parâmetro λ aplicados ao problema modelo unidimensional da equação do calor.

10.1.2 Robustez

Para realizar os experimentos numéricos que confirmam a robustez do método *Space-Time - multigrid* com engrossamento padrão, seguiu-se a mesma ideia mostrada na seção 9.1. Nestes testes numéricos, a condição inicial e de contornos são consideradas nulas, a estimativa inicial para os pontos internos é considerada aleatória entre 0 e 1, o ciclo escolhido também é o ciclo $V(0, 1)$, o método para a aproximação temporal usado foi o método de Euler (o método de CN não funciona se usar engrossamento no tempo), o valor usado como interruptor para selecionar o suavizador é $\lambda_{crit} = 0,46$ para todos os casos estudados.

Nas Figs. 73 – 76 são mostrados os fatores de convergência assintótico em função de λ . Nas Fig. 73 e Fig. 74 considerou-se o problema unidimensional com os valores das discretizações espacial e temporal fixas, respectivamente. No caso em que a discretização espacial é fixa, considerou-se $N_x = 2^7 + 1$ e o refinamento entre cada simulação deu-se apenas na discretização temporal. No caso em que a discretização temporal é fixa, considerou-se também $N_t = 2^7 + 1$ e o refinamento entre cada simulação deu-se apenas na discretização espacial.

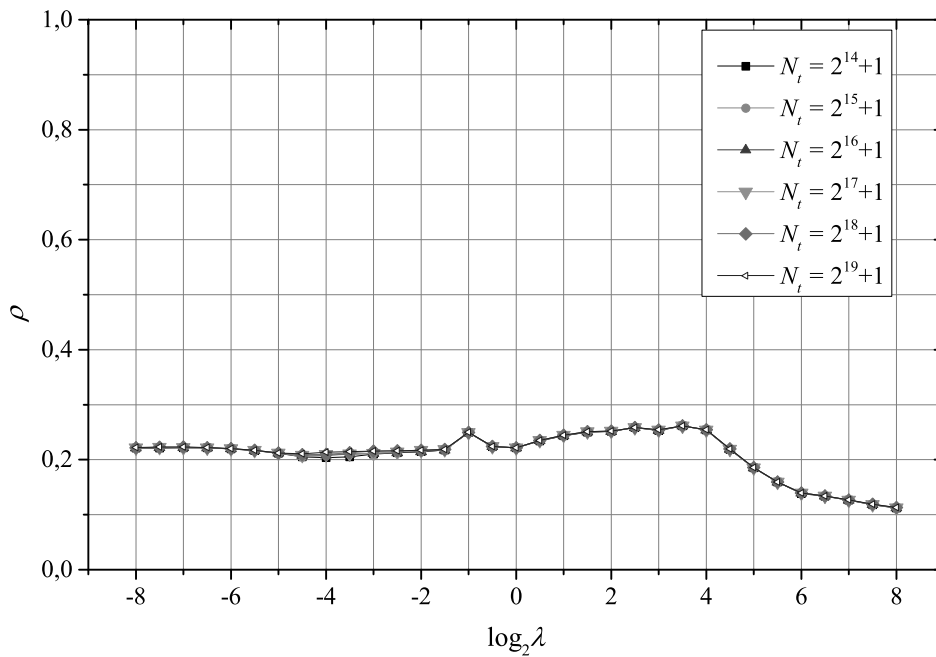


Figura 73 – Robustez do método *Space-Time* com engrossamento padrão para resolver a equação do calor 1D com o método de Euler, ciclo $V(0, 1)$ e $N_x = 2^7 + 1$.

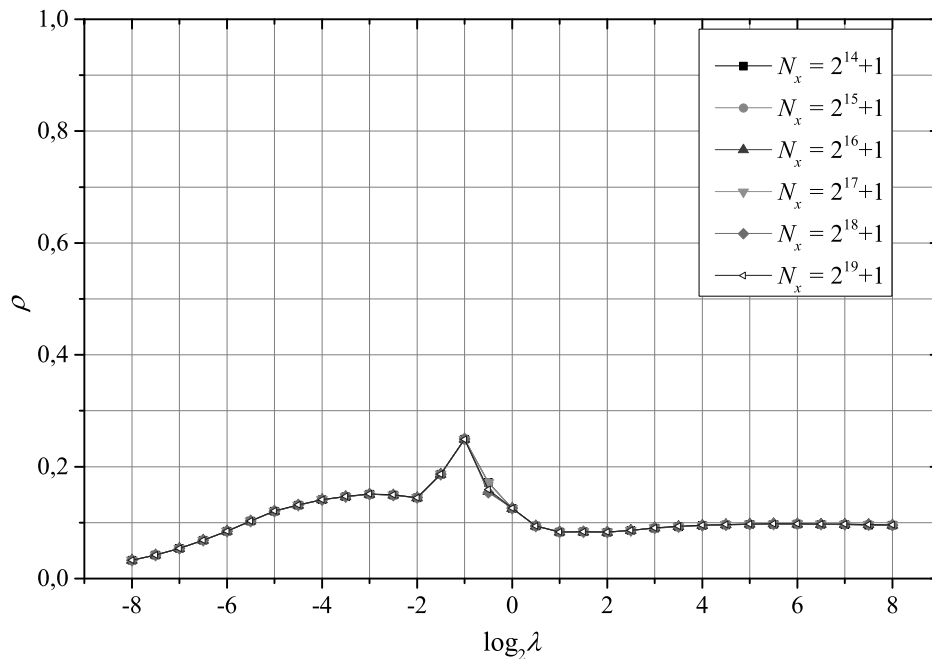


Figura 74 – Robustez do método *Space-Time* com engrossamento padrão para resolver a equação do calor 1D com o método de Euler, ciclo $V(0, 1)$ e $N_t = 2^7 + 1$.

Com esses resultados (Fig. 73 e Fig. 74) pode-se observar que quando ocorre refinamento da malha e para qualquer fator de anisotropia λ , o fator de convergência ρ se estabiliza. Sendo assim, pode-se dizer que o método *Space-Time - multigrid* com engrossamento padrão apresenta robustez em relação às discretizações espacial e temporal

e em relação ao fator de anisotropia λ , para a equação do calor 1D.

Nas Fig. 75 e Fig. 76 considerou-se o problema bidimensional, similar ao caso unidimensional, com os valores das discretizações espacial e temporal fixas, respectivamente. No caso em que a discretização espacial é fixa, considerou-se $N_x = N_y = 2^5 + 1$ e o refinamento entre cada simulação deu-se apenas na discretização temporal. No caso em que a discretização temporal é fixa, considerou-se $N_t = 2^5 + 1$ e o refinamento entre cada simulação deu-se apenas na discretização espacial.

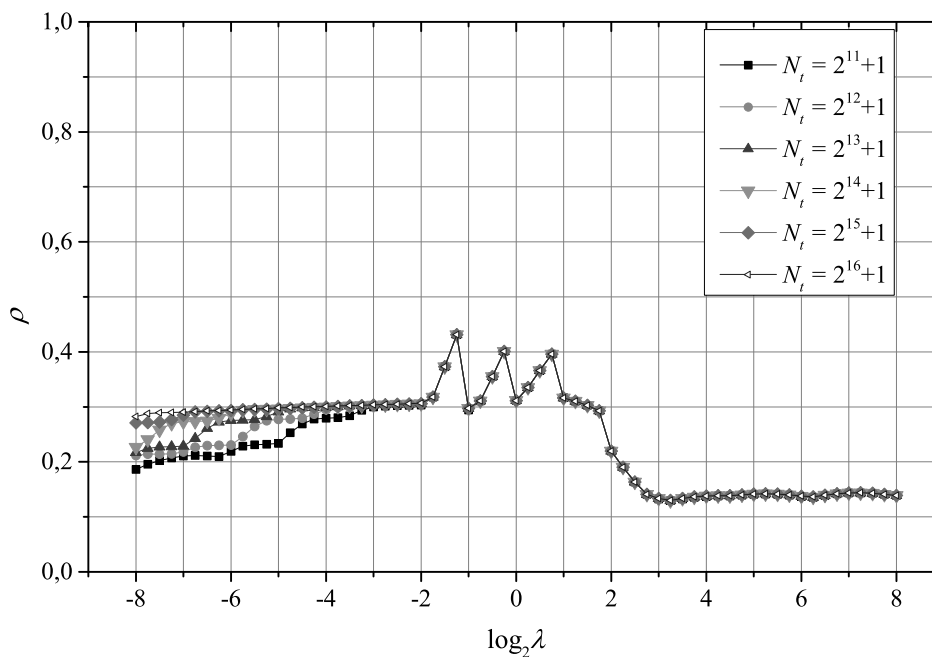


Figura 75 – Robustez do método *Space-Time* com engrossamento padrão para resolver a equação do calor 2D com o método de Euler, ciclo $V(0, 1)$ e $N_x = N_y = 2^5 + 1$.

Semelhante ao caso 1D, com os resultados mostrados nas Fig. 75 e Fig. 76 pode-se dizer que o referido método é robusto em relação às discretizações espacial e temporal e também em relação ao fator de anisotropia λ , para a equação do calor 2D. Com isso, pode-se garantir que esse método pode ser usado para resolver problemas em malhas mais refinadas e para qualquer fator de anisotropia.

10.1.3 Otimização dos parâmetros *multigrid*

Para verificar os parâmetros ótimos do método *Space-Time* com engrossamento padrão juntamente com o método *multigrid* fizeram-se testes considerando-se o t_{CPU} em função do fator de anisotropia $\log_2 \lambda$. Para esses testes, usou-se a discretização de malha com $N_x = N_t = 2^{13} + 1$ para o caso 1D e $N_x = N_y = N_t = 2^9 + 1$ para o caso 2D. A aproximação temporal nesta seção deu-se exclusivamente pelo método de Euler e considerou-se o valor $\lambda_{crit} = 0,46$ para todos os casos estudados. Os ciclos a serem

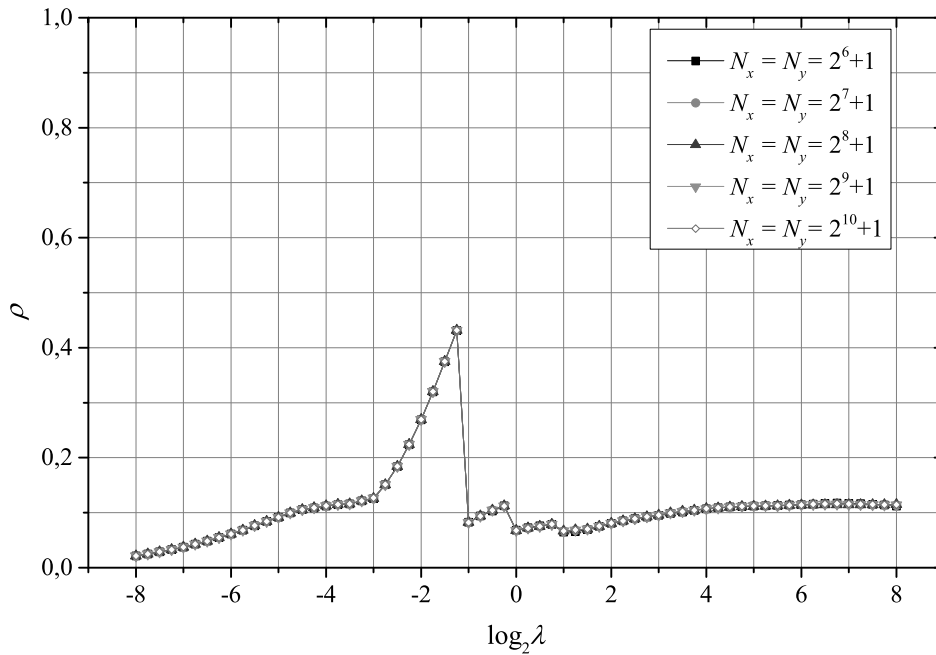


Figura 76 – Robustez do método *Space-Time* com engrossamento padrão para resolver a equação do calor 2D com o método de Euler, ciclo $V(0, 1)$ e $N_t = 2^5 + 1$.

comparados são V , F e W e com um número de suavizações internas que permita verificar seu valor ótimo. O critério de parada usado foi $\frac{\|r(it)\|_\infty}{\|r(0)\|_\infty} \leq 10^{-10}$, em que $r(it)$ é o resíduo na iterada atual e $r(0)$ é o resíduo considerando-se a estimativa inicial.

Na Fig. 77 mostra-se o t_{CPU} em função do fator de anisotropia λ para a equação do calor 1D. Nessa figura, percebe-se que em geral, com 1 suavização interna (qualquer ciclo), ou ainda com o ciclo V e 2 suavizações internas, obtém-se os melhores resultados. Para melhor visualizar os parâmetros ótimos plotou-se um gráfico com apenas os casos citados e que produzem os melhores resultados, ver a Fig. 78.

Com os resultados descritos na Fig. 78 pode-se dizer que o ciclo $V(0, 1)$ apresenta os melhores resultados se $\lambda \leq \lambda_{crit}$ ou $\lambda \geq 2^6$ e o ciclo $V(1, 1)$ apresenta os melhores resultados para os demais casos. Em geral, ao considerar-se apenas um tipo de ciclo como parâmetro ótimo está minimizando-se o tempo computacional de uma forma geral, ou seja, ao se escolher um determinado ciclo, o aumento do t_{CPU} nos casos em que ele não é o melhor, é percentualmente pequeno, se comparado ao aumento do t_{CPU} de outro ciclo em que ele não é o melhor ciclo. Assim, pode-se eleger o ciclo $V(0, 1)$ como parâmetro ótimo.

Na descrição dos parâmetros ótimos para a equação do calor bidimensional acrescentam-se os parâmetros do método *multigrid* usados na relaxação por planos. São eles:

- V_{xy} indica ciclo V por planos (a notação xy indica que o referido ciclo V será usado exclusivamente no plano xy);

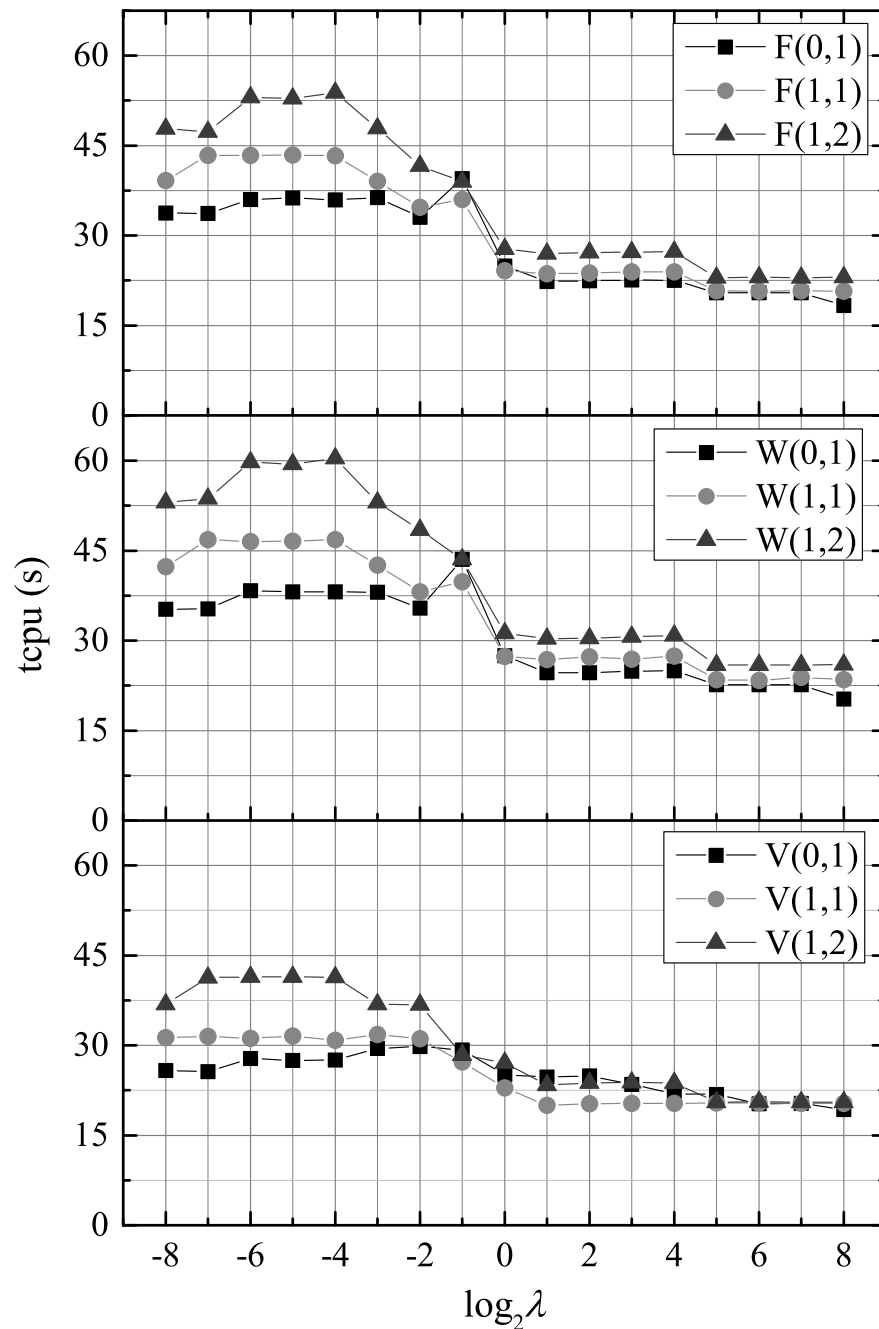


Figura 77 – Otimização dos parâmetros *multigrid* com o método *Space-Time - multigrid* com engrossamento padrão usado na solução da equação do calor 1D.

- $num_{V_{xy}}$ indica o número de ciclos V_{xy} por planos;
- ν_{1xy} e ν_{2xy} representam os números de pré e pós-suavizações internas, respectivamente, aplicadas no ciclo por planos.

Após vários testes numéricos, em que variou-se os tipos de ciclos por planos (V_{xy} , F_{xy} ou W_{xy}), o número de ciclos ($num_{V_{xy}}$, $num_{F_{xy}}$ ou $num_{W_{xy}}$) e o número de pré e pós-suavizações (ν_{1xy} e ν_{2xy}), pôde-se perceber que os parâmetros mais eficientes estão

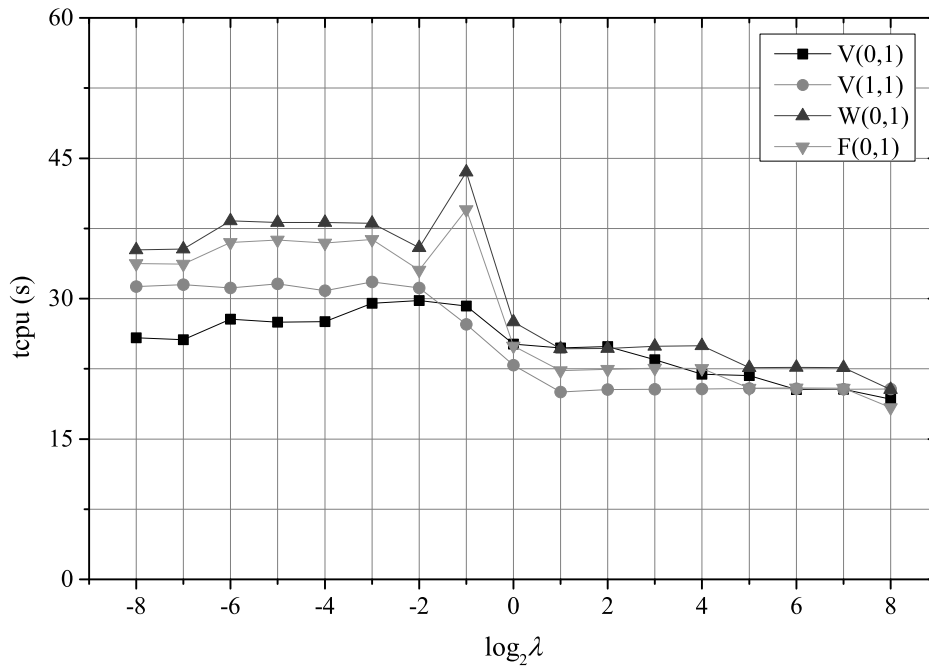


Figura 78 – Comparação entre os parâmetros *multigrid* ótimos para o método *Space-Time - multigrid* com engrossamento padrão usado na solução da equação do calor 1D.

entre $num_V_{xy} = 1$ ou $num_V_{xy} = 2$ e os ciclos $V_{xy}(1, 1)$, $V_{xy}(1, 2)$ ou $V_{xy}(2, 2)$. Nas Figs. 79 a 81 são mostrados alguns testes do t_{CPU} com esses parâmetros.

Observa-se nas Fig. 79 a 81, que se forem feitas 2 ou mais suavizações internas e no *multigrid* aplicado ao plano de suavização forem realizadas apenas 2 suavizações internas, perde-se a eficiência do método (observar os pontos descritos por um quadrado preto cheio na parte superior dos gráficos). Isso possivelmente deve-se ao fato de que com o ciclo V e apenas 2 suavizações internas, os respectivos planos não foram completamente suavizados, por isso houve um considerável aumento do t_{CPU} necessário para resolver o referido problema.

Para melhorar a interpretação dos resultados ótimos, plotou-se um gráfico contendo os ciclos V , F e W com 1, 2 e 3 suavizações internas (ver Fig. 82). No eixo vertical dessa figura é apresentada a soma dos t_{CPU} referente aos experimentos com uso de $1V_{xy}(1, 1)$, $1V_{xy}(1, 2)$, $1V_{xy}(2, 2)$, $2V_{xy}(1, 1)$, $2V_{xy}(1, 2)$ e $2V_{xy}(2, 2)$ para cada valor de $\log_2 \lambda$, ou seja, o t_{CPU} resultante é fruto da soma dos t_{CPU} de 6 testes numéricos contendo os referidos parâmetros $1V_{xy}(1, 1)$, $1V_{xy}(1, 2)$, $1V_{xy}(2, 2)$, $2V_{xy}(1, 1)$, $2V_{xy}(1, 2)$ e $2V_{xy}(2, 2)$.

Observa-se na Fig. 82 que, em geral, os ciclos com 1 suavização interna apresentam os melhores resultados. Dessa forma, com o objetivo de verificar os melhores parâmetros do método *multigrid* aplicados ao plano de suavização, na Fig. 83 são mostrados os valores máximo dos t_{CPU} considerando-se os ciclos $V(0, 1)$, $F(0, 1)$ e $W(0, 1)$ fixos. Dessa forma, para cada valor de $\log_2 \lambda$ cada t_{CPU} indicado na Fig. 83 representa o valor máximo

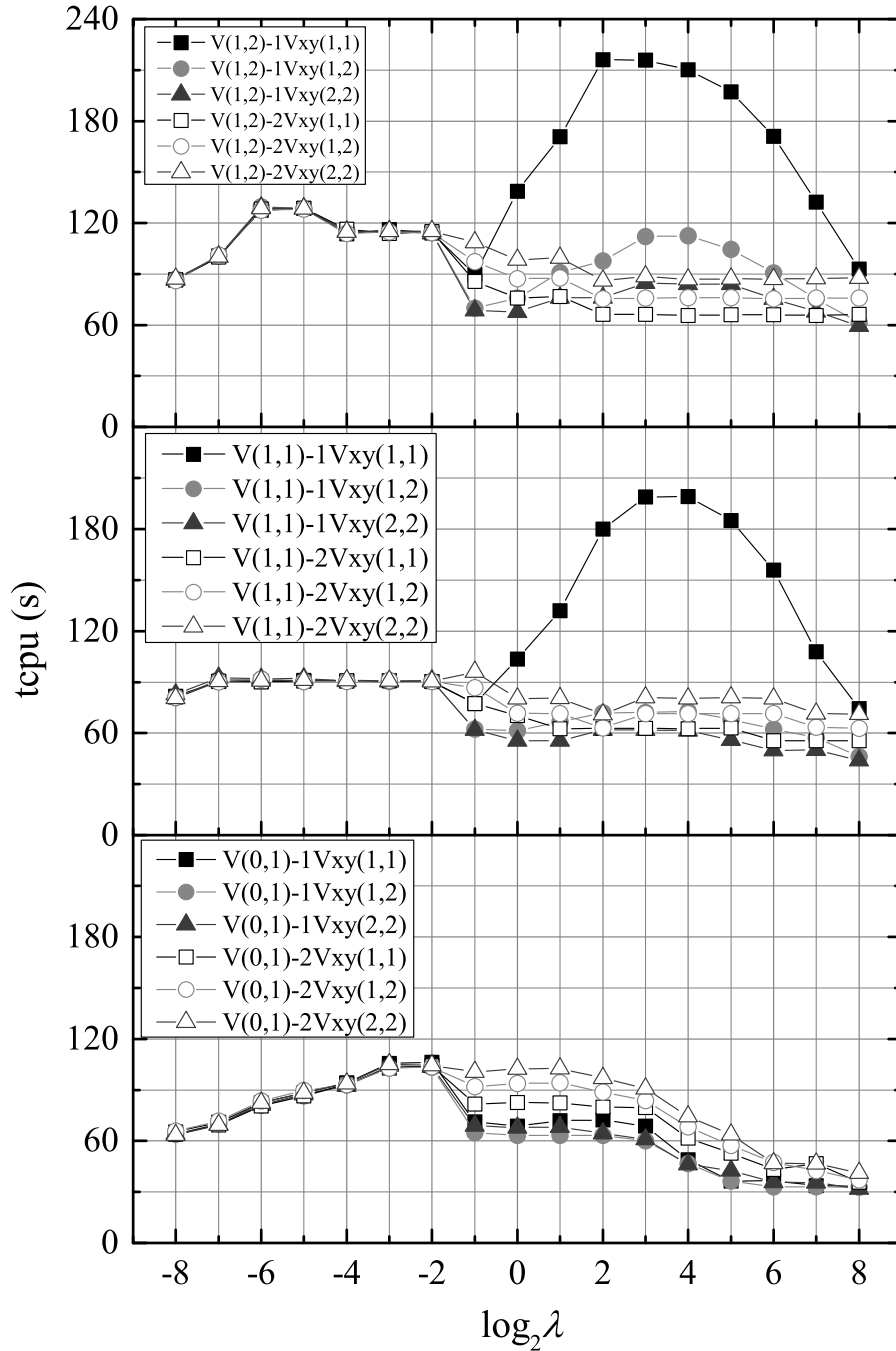


Figura 79 – Otimização dos parâmetros *multigrid* através do método *Space-Time* com engrossamento padrão, método de Euler e ciclo *V* para resolver a equação do calor 2D.

encontrado, considerando-se os ciclos $\{V(0, 1), F(0, 1), W(0, 1)\}$.

Com os resultados descritos na Fig. 83 pode-se concluir que, com 1 suavização interna (ciclos V , F ou W) e ao usar-se a suavização por planos, os parâmetros $num_V_{xy} = 1$, $\nu_{1xy} = 1$ e $\nu_{2xy} = 2$ (ou $1V_{xy}(1, 2)$) fornecem os melhores t_{CPU} .

Observando-se atentamente os gráficos inferiores nas Figs. 79 a 81 percebe-se que para valores de $\lambda > \lambda_{crit}$ e $\lambda < 2^5$, os t_{CPU} relacionados ao ciclo $V(0, 1)$ são um pouco

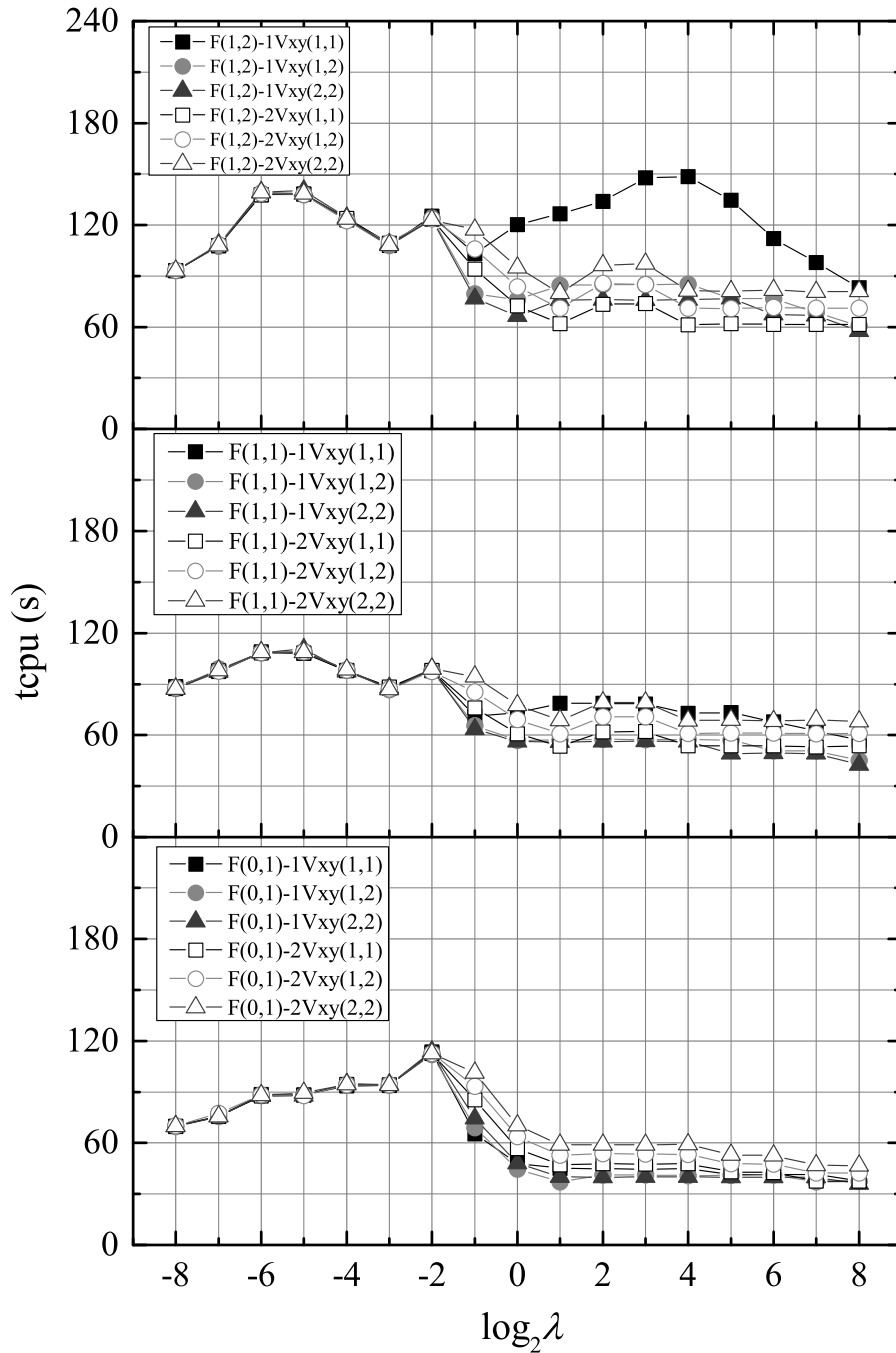


Figura 80 – Otimização dos parâmetros *multigrid* através do método *Space-Time* com engrossamento padrão, método de Euler e ciclo F para resolver a equação do calor 2D.

maiores para que os t_{CPU} relacionados ao ciclo $F(0, 1)$ ou $W(0, 1)$. Porém, como o ciclo $V(0, 1)$ apresentou robustez e maior eficiência na maioria dos casos analisados, optou-se por considerá-lo como o parâmetro ótimo para este caso, ou seja, o ciclos $V(0, 1)$ para o método *multigrid* geral e no caso de se usar suavização por planos, realizar 1 ciclo $V_{xy}(1, 2)$ ($1V_{xy}(1, 2)$) são os melhores parâmetros do método *multigrid* para esse caso.

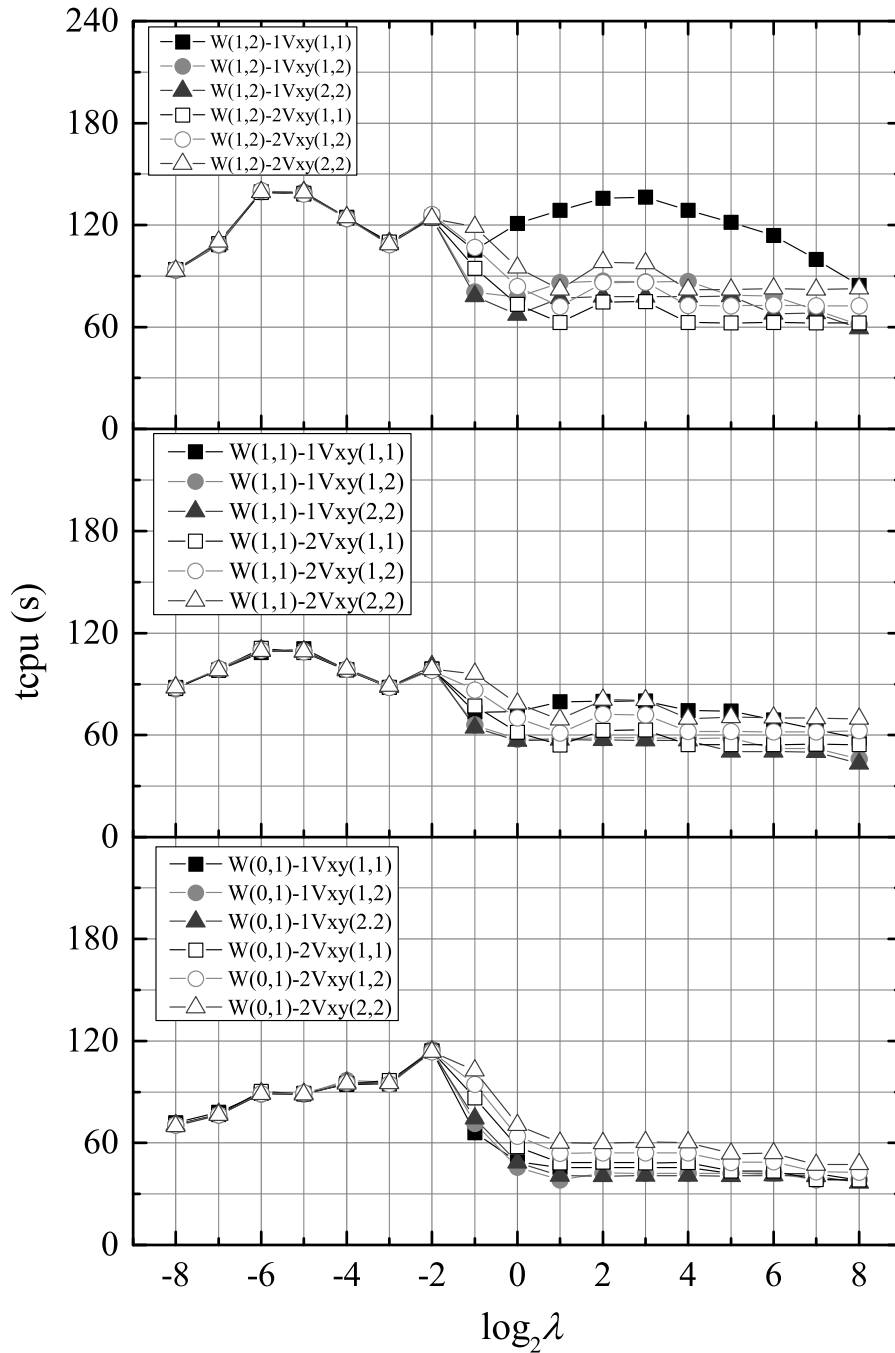


Figura 81 – Otimização dos parâmetros *multigrid* através do método *Space-Time* com engrossamento padrão, método de Euler e ciclo W para resolver a equação do calor 2D.

10.1.4 Fatores de convergência

Na Fig. 84 (FRANCO et al., 2018a) é apresentado o fator de convergência médio, considerando os parâmetros ótimos descritos anteriormente (ciclo $V(0,1)$ para os casos 1D e 2D, e na suavização por planos no caso bidimensional, $1V_{xy}(1,2)$). Nessas simulações consideraram-se $tol = 10^{-10}$, $N_x = 2^{10} + 1$ e $N_t = 2^{16} + 1$ para o caso unidimensional (Fig. 84(a)) e $N_x = N_y = 2^7 + 1$ e $N_t = 2^{12} + 1$ para o caso bidimensional (Fig. 84(b)).

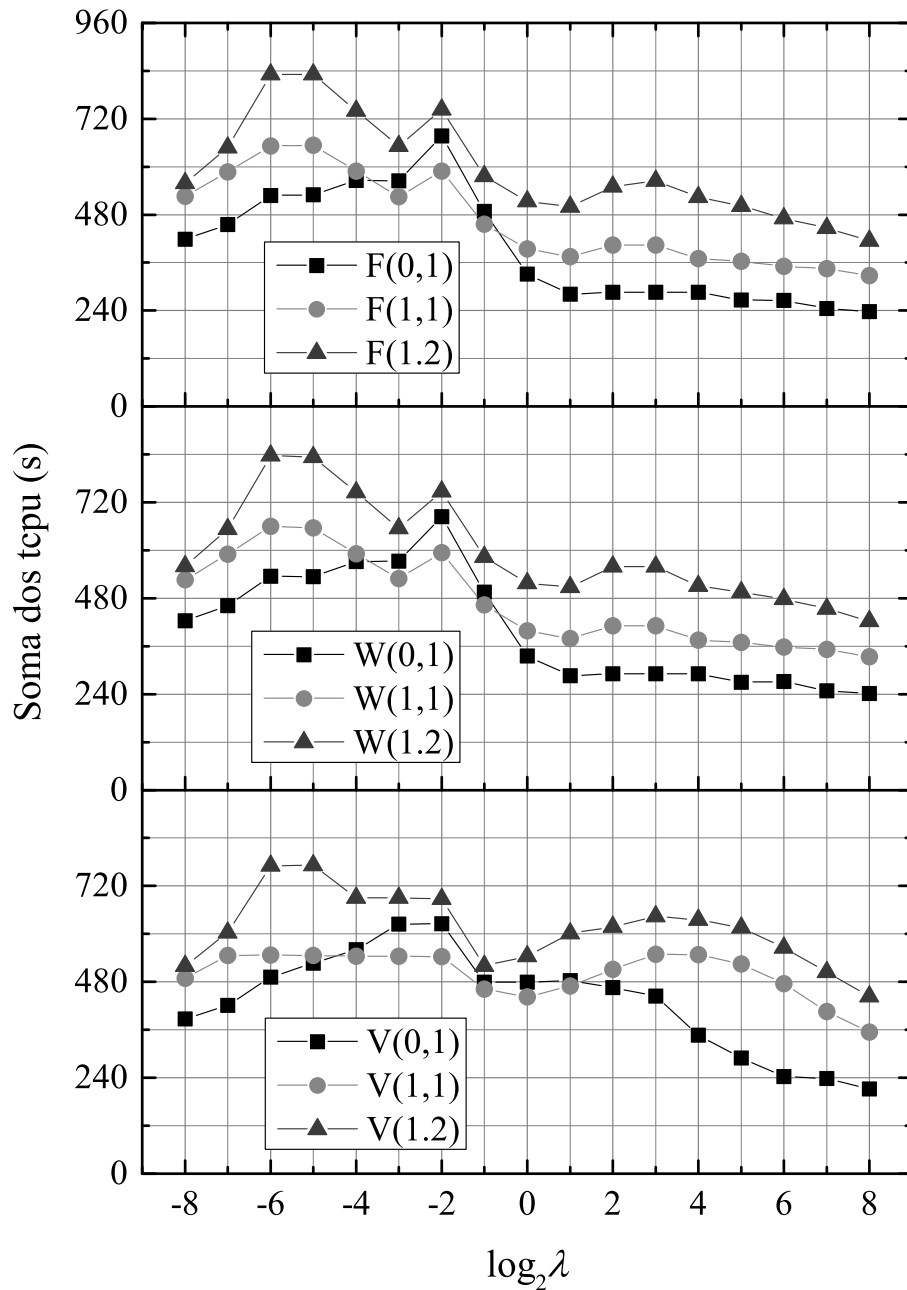


Figura 82 – Comparação dos resultados descritos nas Fig. 79, Fig. 80 e Fig. 81. No eixo vertical são descritas as somas de todos os t_{CPU} relacionados aos casos usados para a suavização por planos para cada λ .

Com os resultados apresentados na Fig. 84, observa-se que para ambos os problemas 1D e 2D, mesmo com uma suavização interna, os fatores de convergência foram menores que 0,3, indicando eficiência para resolver esse tipo de problema.

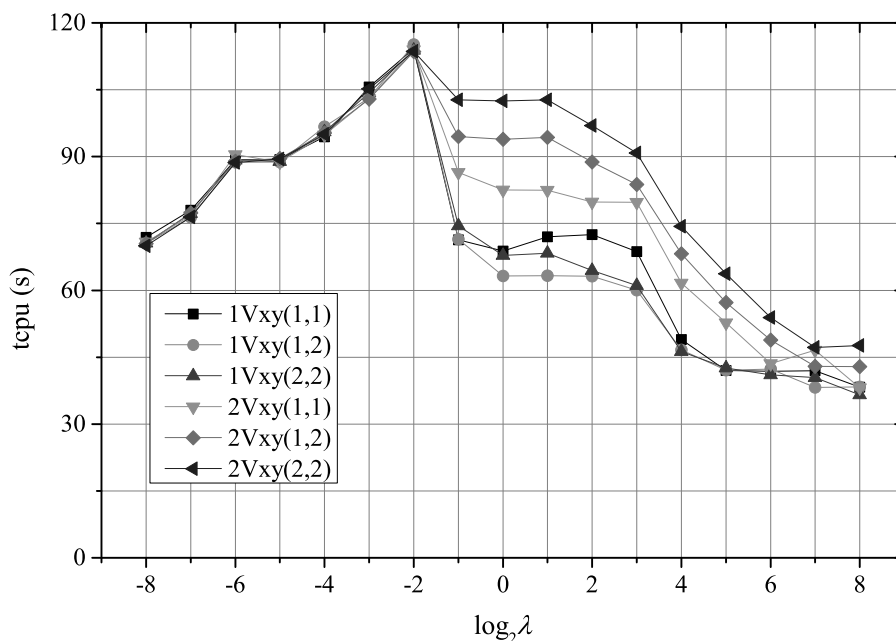


Figura 83 – Comparação dos resultados descritos nas Figs. 79 a 81. No eixo vertical são mostrados os valores máximo dos t_{CPU} considerando-se os ciclos $V(0, 1)$, $F(0, 1)$ e $W(0, 1)$.

10.1.5 Método da dupla discretização

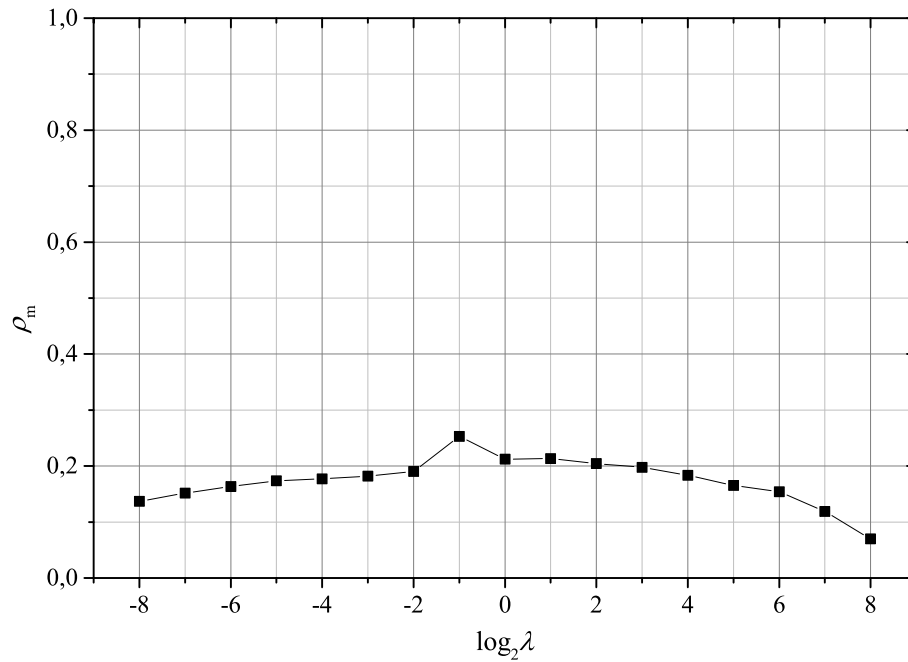
O método *multigrid* oferece possibilidades para melhorar não só a convergência algébrica, mas também a precisão da solução discreta em relação à solução analítica. Essa é a ideia básica da dupla discretização (BRANDT; LIVNE, 2011), que pode ser vista como uma técnica especial de correção de resíduos aplicada junto ao método *multigrid*.

Devido ao fato do método de Crank-Nicolson (CN) não ser eficiente quando ocorre engrossamento no tempo (ver a Fig. 68), esse método é uma alternativa para melhorar a ordem de acurácia nesses casos.

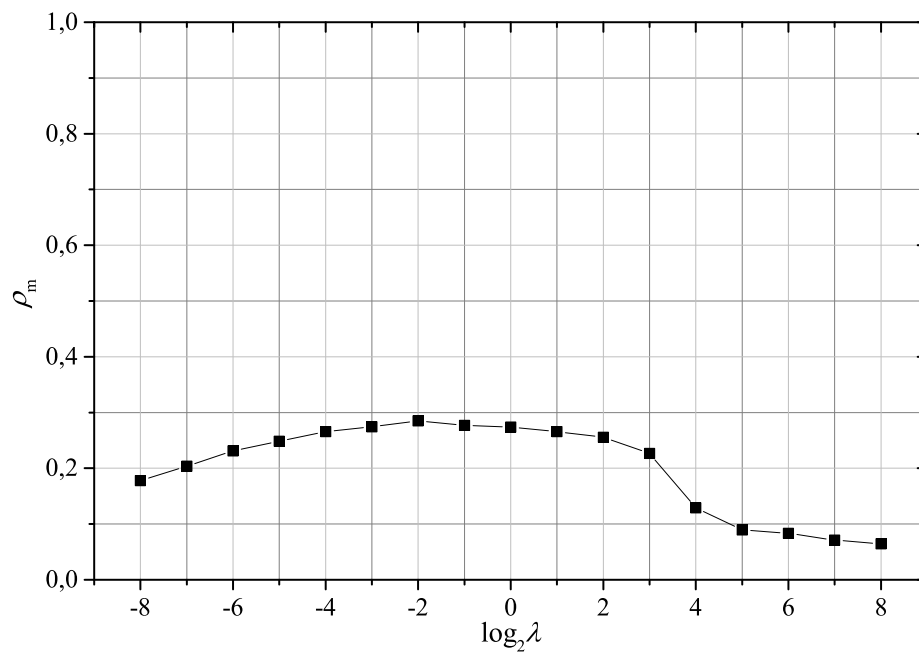
De acordo com Brandt e Livne (2011) a dupla discretização consiste em empregar um esquema de discretização estável para as varreduras de suavização, que pode ser de ordem inferior ao desejado, combinado com o uso de uma discretização da ordem solicitada para calcular os resíduos transferidos para a malha grossa, ou seja, discretiza-se com um método para efetuar as suavizações e com outro método para calcular os resíduos.

Essa estratégia já foi aplicada com sucesso em vários problemas: com aproximações de quarta e sexta ordens para a equação de Poisson, ver por exemplo Schaeffer (1982); em aproximações de segunda ordem para problemas de perturbação singulares, ver Brandt (1977); e em dinâmica de fluidos, ver Brandt e Livne (2011).

Para comprovar os fatores de convergência do método *Space-Time - multigrid* com engrossamento padrão, suavizadores do tipo zebra e o método de CN, fizeram-se alguns



(a) Problema 1D, $N_x = 2^{10} + 1$ e $N_t = 2^{16} + 1$



(b) Problema 2D, $N_x = N_y = 2^7 + 1$ e $N_t = 2^{12} + 1$ e $1V_{xy}(1, 2)$

Figura 84 – Fator de convergência médio do método *Space-Time* com engrossamento padrão e ciclo $V(0, 1)$.

experimentos que estão expressos na Fig. 85. Esses testes foram realizados com a equação do calor 1D, número de pontos nas discretizações espacial e temporal iguais $N = 2^{13} + 1$, engrossamento padrão em ambas as direções com razão de engrossamento $q = 2$, ciclo $V(0, 1)$, aproximação temporal com o método de CN e condição inicial dada pela Eq. (6.2).

Na linha vertical do gráfico, apresentou-se o fator de convergência médio após 10 ciclos *multigrid*. Os pontos que não estão expressos nesse gráfico indicam que o referido método divergiu ($\rho_m > 1$) para o respectivo valor de λ .

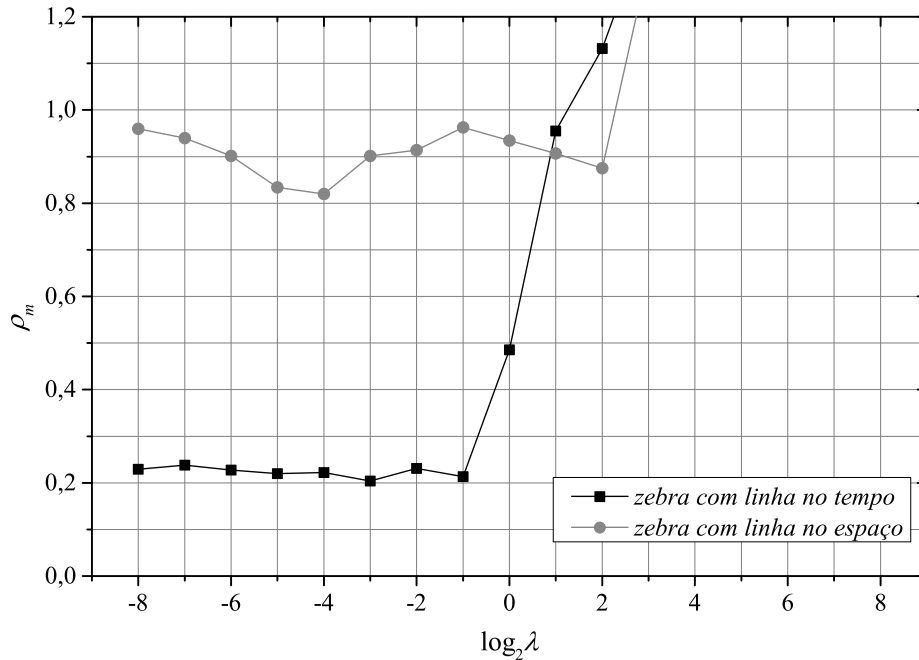


Figura 85 – Experimentos com o método *multigrid*, aproximação temporal com o método de CN, engrossamento padrão, ciclo $V(0, 1)$ e suavizadores *zebra com linha no tempo* e *zebra com linha no espaço* para resolver a equação do calor 1D.

Com os resultados descritos na Fig. 85 observa-se que com $\lambda \leq \lambda_{crit} = 0,46$ o suavizador *zebra com linha no tempo* apresenta bons fatores de convergência mesmo com a aproximação temporal sendo feita pelo método de CN, ou seja, nesse intervalo ($\lambda \leq \lambda_{crit}$) o método *multigrid* com engrossamento padrão associado ao método de CN apresenta bons fatores de convergência. Porém, se $\lambda > \lambda_{crit}$ o método diverge para ambos os suavizadores (com o método de CN), por isso a necessidade do método de dupla discretização.

O referido método de dupla discretização será usado (nesta tese) se $\lambda > \lambda_{crit}$ e com a aproximação temporal feita com o método de Euler (primeira ordem e estável) para fazer as suavizações em todos os níveis de malha, enquanto que os resíduos são calculados com o método de Crank-Nicolson. Esse último método (CN) determina a precisão da solução numérica. Pode-se ver o processo detalhado para os casos 1D no algoritmo 13. Nesse algoritmo observa-se que os termos "**método de Euler**" e "**método de Crank-Nicolson**" em negrito representam a ideia principal do método da dupla discretização.

Para o caso 2D, basta usar os referidos suavizadores e operadores de restrição/prolongação descritos para o caso bidimensional no algoritmo 13.

O método de dupla discretização não tem o controle do resíduo, pois se utiliza um suavizador em que a aproximação da variável temporal é feita através de um método de

Algoritmo 13: Ciclo $V(\nu_1, \nu_2)$ com o método de dupla discretização, nível de malha (l)

```

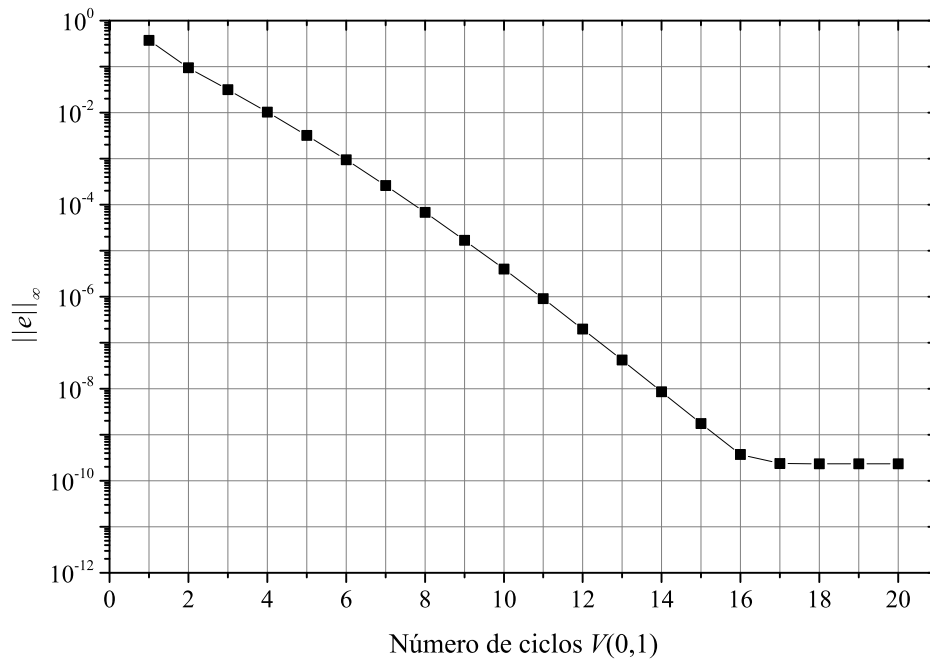
if  $l$  é o nível de malha mais grossa,  $L_{max}$  then
  Resolver exatamente o sistema de equações na malha grossa.
else
  if  $\lambda_l > \lambda_{crit}$  then
    Suavizar  $\nu_1$  vezes  $A_l v^{(l)} = f^{(l)}$  usando o suavizador zebra com linha no espaço e
    aproximação temporal dada pelo método de Euler.
  else
    Suavizar  $\nu_1$  vezes  $A_l v^{(l)} = f^{(l)}$  usando o suavizador zebra com linha no tempo e
    aproximação temporal dada pelo método de Crank-Nicolson.
  end if
  Calcular o resíduo usando o método de Crank-Nicolson.
  Restringir o resíduo.
  Resolver a equação residual no nível de malha  $l + 1$  usando o Algoritmo 13.
  Interpolar a correção.
  Calcular uma nova aproximação.
  if  $\lambda_l > \lambda_{crit}$  then
    Suavizar  $\nu_2$  vezes  $A_l v^{(l)} = f^{(l)}$  usando o suavizador zebra com linha no espaço e
    aproximação temporal dada pelo método de Euler.
  else
    Suavizar  $\nu_2$  vezes  $A_l v^{(l)} = f^{(l)}$  usando o suavizador zebra com linha no tempo e
    aproximação temporal dada pelo método de Crank-Nicolson.
  end if
end if

```

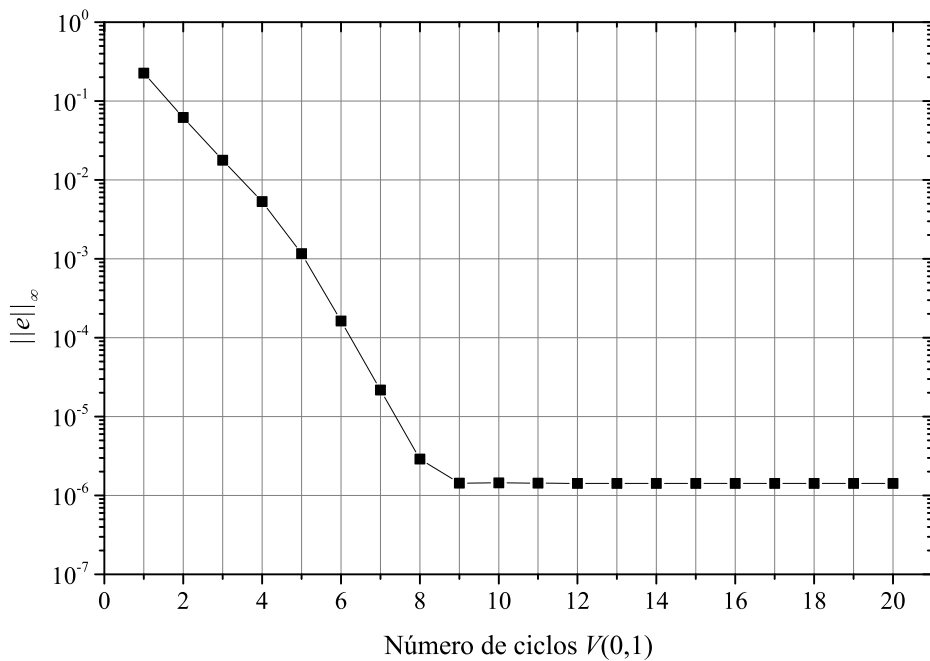
primeira ordem (Euler) e o resíduo é calculado usando uma discretização para a variável temporal com um método de segunda ordem (CN). Dessa forma, considerando-se por exemplo, $\lambda = 2^4$, uma discretização com $N = 2^{13} + 1$ ($N = N_x = N_t$) para o problema unidimensional e com $N = N_x = N_y = N_t = 2^9 + 1$ ($N = N_x = N_y = N_t$) para o problema bidimensional, mostra-se na Fig. 86 que com 16 ciclos $V(0, 1)$ para o caso 1D e com apenas 10 ciclos $V(0, 1)$ para o caso 2D, o erro de máquina já foi alcançado, o que torna esse método muito atraente.

Para garantir que a técnica de dupla discretização forneça a precisão de segunda ordem de acurácia, nas Tab. 11 e Tab. 12 mostram-se as normas máximas do erro numérico ($\|e\|_\infty$) para diferentes tamanhos de discretização de malha variando de $h = \tau = 2^{-6}$ a $h = \tau = 2^{-13}$ ($h = h_x$) para o caso unidimensional e de $h = \tau = 2^{-4}$ a $h = \tau = 2^{-9}$ ($h = h_x = h_y$) para o caso bidimensional. O tamanho do passo temporal é dado por $\tau = h$ em ambos os casos 1D e 2D. Pode-se verificar que quando os tamanhos das malhas são divididos pela metade, a $\|e\|_\infty$ são reduzidos em um quarto, o que implica a segunda ordem de acurácia relacionada ao método de Crank-Nicolson.

Na Fig. 87 mostrada em Franco et al. (2018a) é apresentado o fator de convergência médio do método *Space-Time* com engrossamento padrão e com a técnica de dupla



(a) Problema 1D com $N = 2^{13} + 1$



(b) Problema 2D com $N = 2^9 + 1$

Figura 86 – Número de ciclos $V(0, 1)$ necessários para alcançar o erro de máquina usando o método de dupla discretização.

discretização após se efetuar 20 ciclos *multigrid* com os parâmetros ótimos (ciclo $V(0, 1)$ e no caso 2D, $1V_{xy}(1, 2)$ para suavizar cada plano espacial) descritos anteriormente. Nessas simulações considerou-se solução analítica nula, $N_x = 2^{10} + 1$ e $N_t = 2^{16} + 1$ para o caso unidimensional (Fig. 87(a)) e $N_x = N_y = 2^7 + 1$ e $N_t = 2^{12} + 1$ para o caso bidimensional

Tabela 11 – Norma máxima do erro numérico, $\|e\|_\infty$, para o problema 1D e diferentes tamanhos da malha com $h = \tau = \frac{1}{N}$

N	$2^6 + 1$	$2^7 + 1$	$2^8 + 1$	$2^9 + 1$	$2^{10} + 1$	$2^{11} + 1$	$2^{12} + 1$	$2^{13} + 1$
$\ e\ _\infty$	3,4E-04	9,1E-05	2,4E-05	6,0E-06	1,5E-06	3,8E-07	9,6E-08	2,4E-08

Tabela 12 – Norma máxima do erro numérico, $\|e\|_\infty$, para o problema 2D e diferentes tamanhos da malha com $h = \tau = \frac{1}{N}$

N	$2^3 + 1$	$2^4 + 1$	$2^5 + 1$	$2^6 + 1$	$2^7 + 1$	$2^8 + 1$	$2^9 + 1$
$\ e\ _\infty$	1,2E-02	3,7E-03	1,1E-03	3,4E-04	9,6E-05	2,6E-05	6,9E-06

(Fig. 87(b)).

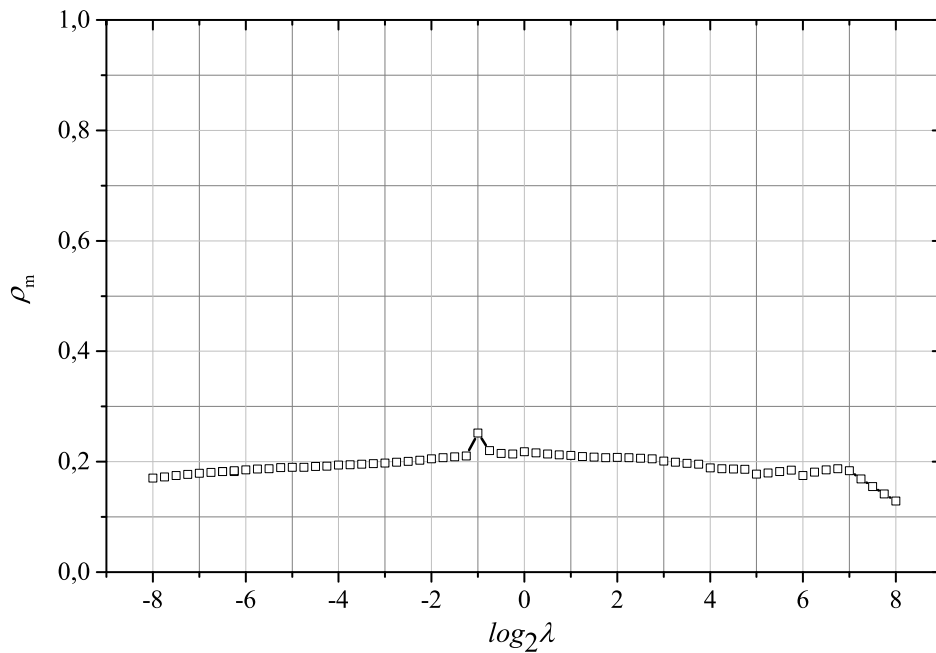
Com os resultados descritos na Fig. 87 se observa que para ambos os problemas 1D e 2D, mesmo com apenas uma suavização interna, os fatores de convergência foram menores que 0,3, fornecendo resultados muito satisfatórios, e que são similares à convergência mostrada na Fig. 84 a partir do método *Space-Time* com engrossamento padrão para o esquema de Euler de primeira ordem.

Portanto, tem-se um método que usa engrossamento padrão (simplicidade na programação), permite o uso do método de CN (aproximação de 2^a ordem para o erro), apresenta robustez com o ciclo V , possui fatores de convergência comparáveis aos da dupla discretização e possibilita o uso de suavizadores coloridos (altamente paralelizáveis no espaço e no tempo). Essas características, aliadas ao fato de que no caso do uso de programação paralela, o ciclo V é mais eficiente que os demais ciclos, tornam esse método muito atraente, pelo grande potencial de eficiência e robustez. Esse é um dos temas de pesquisa para trabalho futuro.

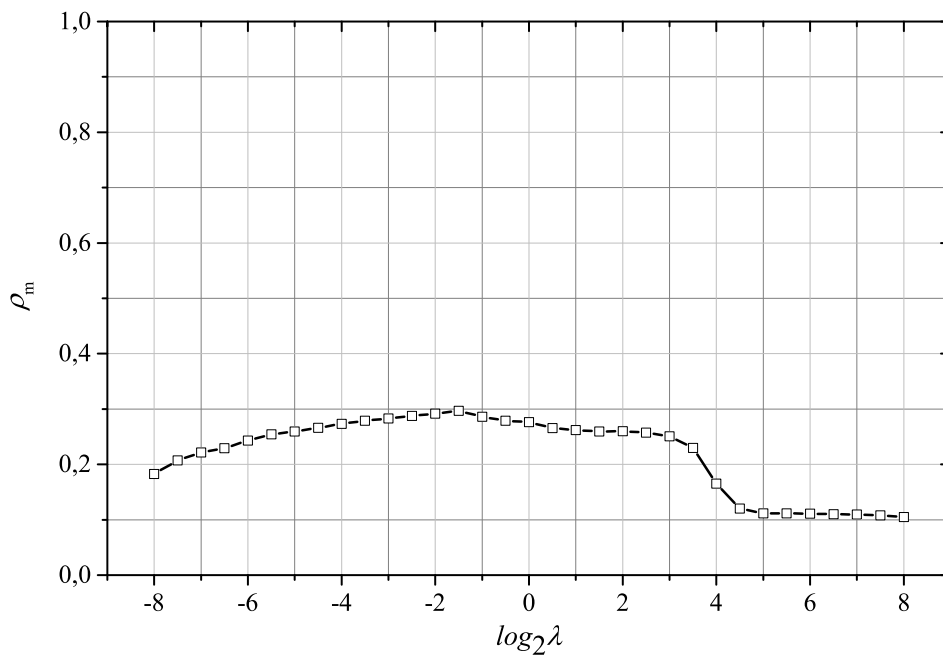
10.2 Equações da poroelasticidade

Até o momento, os algoritmos de solução mais usados para o problema da poroelasticidade baseiam-se em uma abordagem *Time-Stepping* (seção 7.1), em que cada passo de tempo é resolvido após o outro de forma sequencial. Esse método tradicional tem a grande vantagem de se resolver uma equação elíptica a cada passo de tempo, mas tem o problema de não permitir paralelização da variável temporal. Como consequência, deve-se procurar métodos que possam ser paralelizados no tempo e também no espaço-tempo.

Nesta tese será proposto um algoritmo de relaxamento na forma *Waveform Relaxation - multigrid* (LUBICH; OSTERMANN, 1987; VANDEWALLE, 1993), em que a novidade é o método *Waveform Relaxation - multigrid* para os problemas da poroelasticidade 1D e 2D. A chave para o bom funcionamento desse método é o uso do



(a) Problema 1D, $N_x = 2^{10} + 1$ e $N_t = 2^{16} + 1$



(b) Problema 2D, $N_x = N_y = 2^7 + 1$ e $N_t = 2^{12} + 1$ e $1V_{xy}(1,2)$

Figura 87 – Fator de convergência médio do método *Space-Time* com engrossamento padrão associado a técnica da dupla discretização e ciclo $V(0,1)$.

suavizador Vanka (VANKA, 1986) para aproximar a solução numérica. As formulações matemáticas e numéricas desse problemas encontram-se nas seções 6.2.1 e 6.2.2.

Para as simulações numéricas usaram-se as discretizações do modelo matemático para as equações da poroelasticidade apresentadas na seção 6.2.2. O suavizador usado

foi o Vanka de 3 pontos para o caso unidimensional e o Vanka de 5 pontos para o caso bidimensional, como descrito na Seção A. Com o objetivo de desenvolver algoritmos paralelizáveis, propuseram-se formas coloridas para efetuar as suavizações com o suavizador Vanka, ver a seção 6.3.

Os exemplos numéricos simulados nesta tese (problema 1D e 2D) referem-se ao problema da uma concha cilíndrica de material poroso deformável, como descrito em Rodrigo (2010). Os coeficientes de Lamé λ e μ dependem do valor do módulo de Young, que neste caso é $E = 10^4 N/m^2$ e da razão de Poisson que é dado por $\nu = 0,2$ (adimensional). Os valores de λ e μ podem ser calculados pelas seguintes relações

$$\lambda = \frac{\nu E}{(1 + \nu)(1 - 2\nu)}, \quad \mu = \frac{E}{2(1 + \nu)}. \quad (10.8)$$

10.2.1 Robustez

Os primeiros testes para verificar a eficiência de um novo método são os testes de robustez. Para isso verificam-se os fatores de convergência do método *Waveform Relaxation - multigrid* com o suavizador Vanka em relação aos parâmetros físicos e de discretização para os casos 1D e 2D à medida em que a malha é refinada.

Nos primeiros testes consideraram-se $K = 10^{-6}$, solução analítica nula, método de Euler e o suavizador Vanka de 3 pontos e 4 cores para o problema da poroelasticidade 1D, como mostra a Fig. 88.

Observa-se na Fig. 88 que quando ocorre refinamento de malha, ou seja $N = N_x = N_t$ aumenta, os fatores de convergência relacionado aos ciclos V e também ao ciclo F com 1 ou 2 suavizações internas (independente de ser pré ou pós-suavização) aumentam com o refino da malha. Portanto, para esses casos pode-se dizer que o referido método não é robusto em relação à discretização de malhas. Nesses casos testados, os resultados com o método de Crank-Nicolson (CN) apresentaram resultados similares.

Para confirmar se o ciclo F com mais de 2 suavizações internas e o ciclo W com 1 suavização interna apresentam ou não robustez, fizeram-se testes envolvendo as equações da poroelasticidade 2D. Nesses testes usaram-se também $K = 10^{-6}$, solução analítica nula, método de Euler e o suavizador Vanka de 5 pontos e 12 cores. Os resultados são expostos na Fig. 89.

Com os resultados descritos na Fig. 89 observa-se que quando se usa o ciclo $F(2, 2)$ o fator de convergência aumenta com o refino da malha, chegando a divergir com $N = 2^9 + 1$ pontos ($N = N_x = N_y = N_t$). Assim, pode-se dizer que apenas o ciclo W é robusto em relação à discretização de malhas. Para verificar também a robustez com relação ao método de CN e às propriedades físicas, como a condutividade hidráulica (K), fizeram-se alguns testes adicionais para o problema unidimensional com $N = N_x = N_t$ pontos, que são

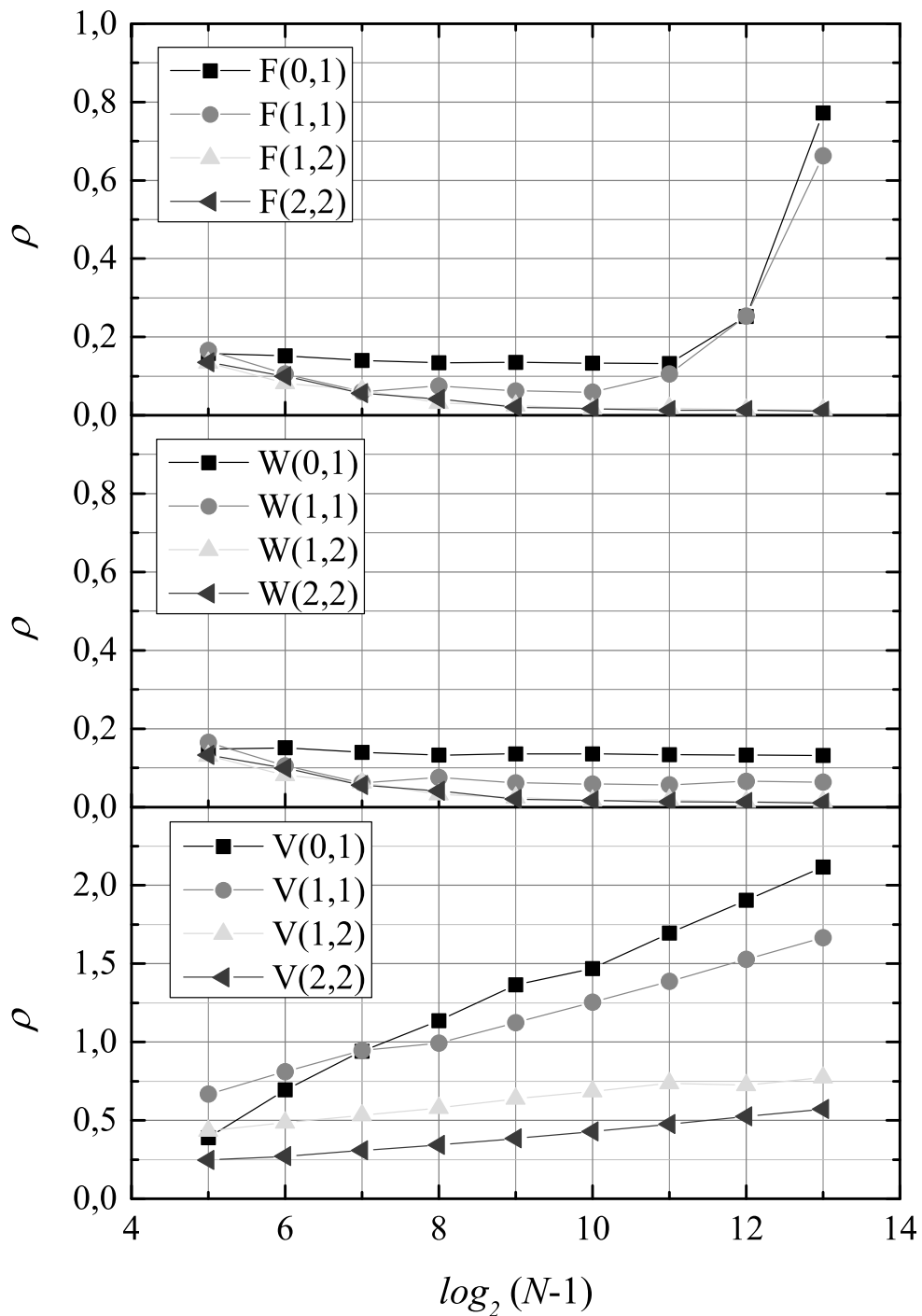


Figura 88 – Fator de convergência após 20 ciclos *multigrid*, considerando $E = 10^4 N/m^2$, $K = 10^{-6}$, solução analítica nula, método de Euler e o suavizador Vanka de 3 pontos e 4 cores para o problema da poroelasticidade 1D.

mostrados na Fig. 90, e para o problema bidimensional com $N = N_x = N_y = N_t$ pontos, exposto na Fig. 91.

Com esses testes mostrados nas Fig. 90 e Fig. 91, pode-se verificar que o método *Waveform Relaxation - multigrid* usando o suavizador Vanka é robusto em relação às

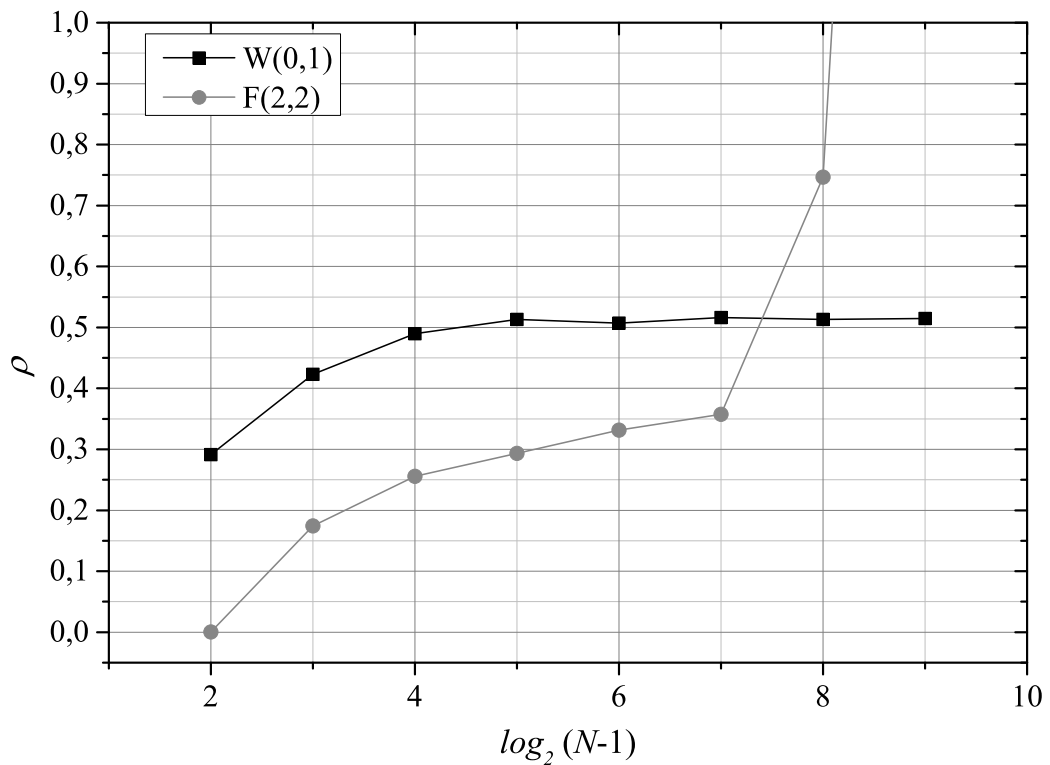


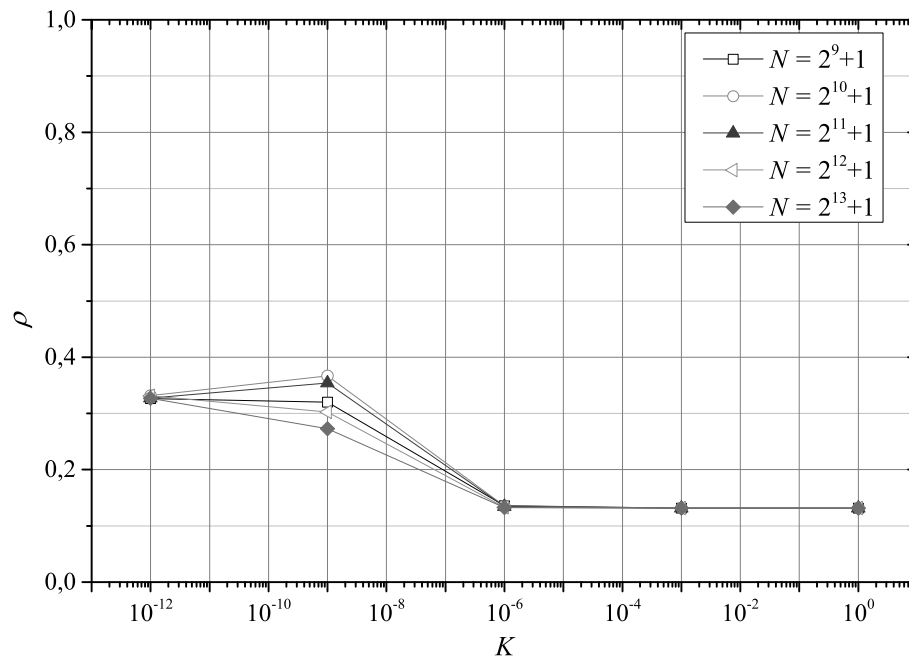
Figura 89 – Fator de convergência após 20 ciclos *multigrid*, considerando $E = 10^4$, $\nu = 0,2$, $K = 10^{-6}$, solução analítica nula, método de Euler e o suavizador Vanka de 5 pontos e 12 cores para o problema da poroelasticidade 2D.

propriedades físicas e em relação à discretização de malhas quando se usa o ciclo W .

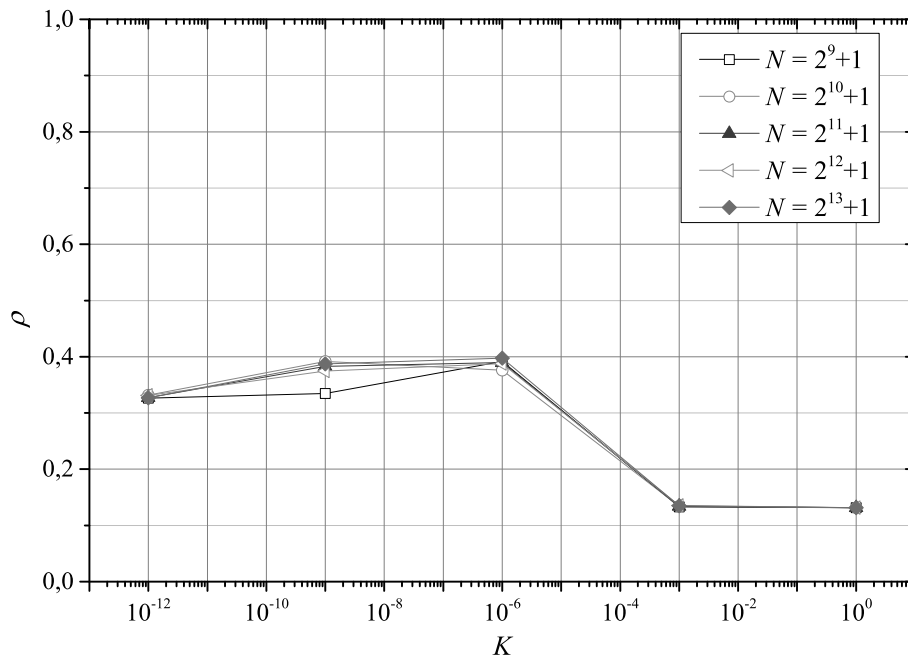
10.2.2 Otimização dos parâmetros *multigrid*

A fim de encontrar os parâmetros que melhoram o desempenho do método *multigrid* para os problemas da poroelasticidade 1D e 2D resolvido com o método *Waveform Relaxation - multigrid* e o suavizador Vanka com ordem lexicográfica e colorida, fizeram-se alguns testes numéricos considerando o ciclo W , que apresentou robustez. O critério de parada adotado é dado por $\frac{\|r(it)\|_\infty}{\|r(0)\|_\infty} \leq 10^{-10}$, em que $\|r(it)\|_\infty$ representa o máximo resíduo entre o(s) deslocamento(s) e a pressão na iterada atual e $\|r(0)\|_\infty$ o correspondente resíduo considerando-se a estimativa inicial dada por números aleatórios entre 0 e 1. Nesta seção serão expostos os resultados considerando-se o método de Euler para a aproximação da variável temporal, pois o método de CN apresentou resultados similares e não serão expostos aqui.

Para o caso unidimensional consideraram-se o módulo de Young $E = 10^4$, a discretização da malha com $N = 2^{13} + 1$ ($N = N_x = N_t$) pontos e as soluções analíticas para u e p dadas nas Eq. (6.26) e Eq. (6.27), respectivamente. Nas Fig. 92 e Fig. 93 são mostrados os tempos computacionais t_{CPU} e os fatores de convergência médio ρ_m resultantes dos testes numéricos envolvendo a equação da poroelasticidade 1D para



(a) Método de Euler

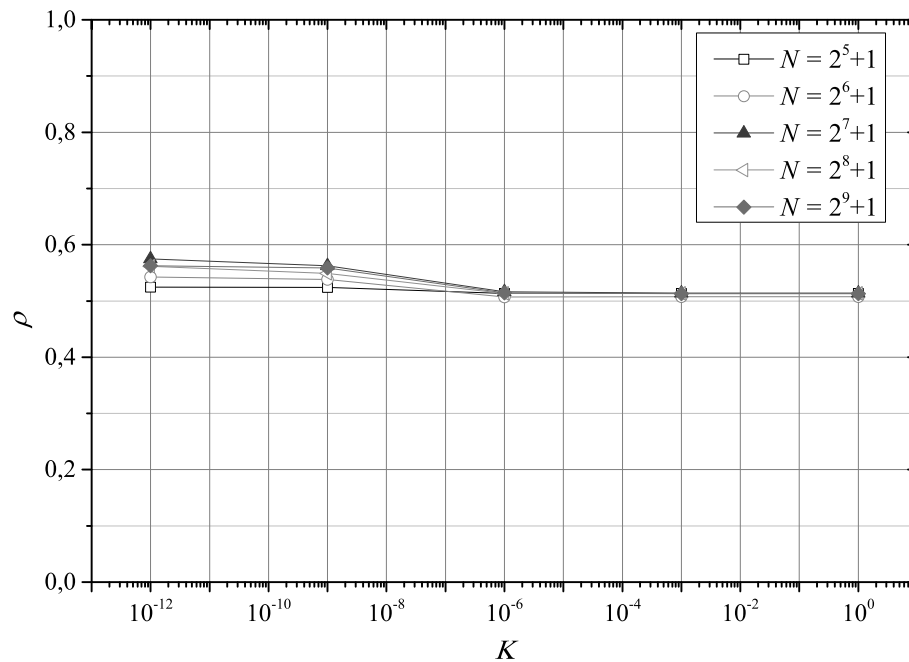


(b) Método de CN

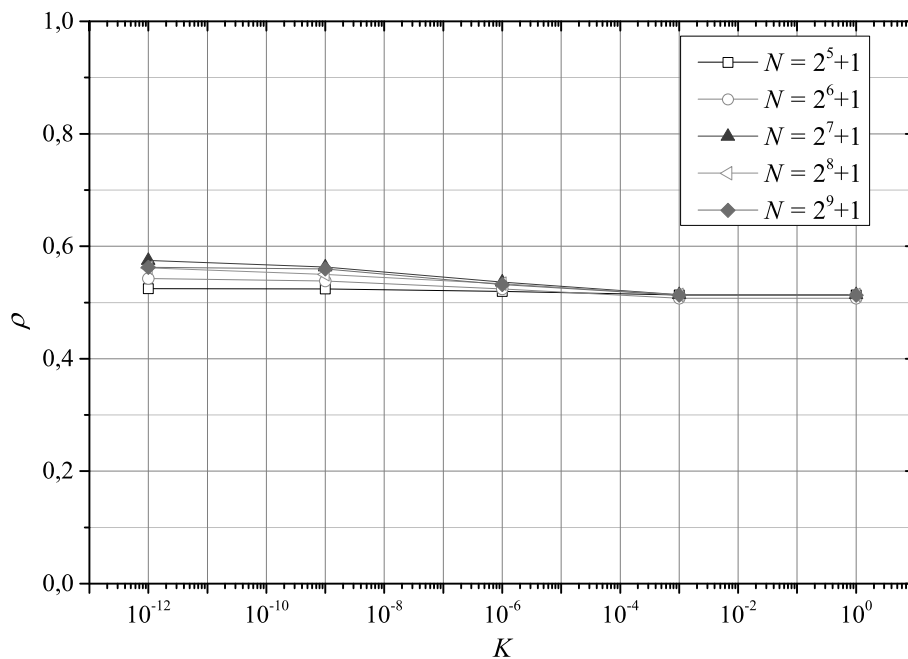
Figura 90 – Fator de convergência para mostrar a robustez do método *Waveform Relaxation - multigrid* para o problema da poroelasticidade 1D, usando o ciclo $W(0, 1)$.

diferentes valores da condutividade hidráulica K , considerando o suavizador Vanka de 3 pontos com ordenação lexicográfica e de 4 cores, respectivamente.

Para o caso bidimensional consideraram-se o módulo de Young $E = 10^4$, a razão de Poisson $\nu = 0,2$, e os valores das constantes de Lamé λ e μ dadas pela Eq. (10.8),



(a) Método de Euler



(b) Método de CN

Figura 91 – Fator de convergência para mostrar a robustez do método *Waveform Relaxation - multigrid* para o problema da poroelasticidade 2D, usando o ciclo $W(0, 1)$.

a discretização da malha com $N = 2^9 + 1$ ($N = N_x = N_y = N_t$) pontos e as soluções analíticas para u , v e p dada na Eq. (6.32). Nas Fig. 94 e Fig. 95, assim como para o caso 1D, são mostrados os t_{CPU} e os ρ_m resultantes dos testes numéricos envolvendo a equação da poroelasticidade 2D para diferentes valores da condutividade hidráulica K ,

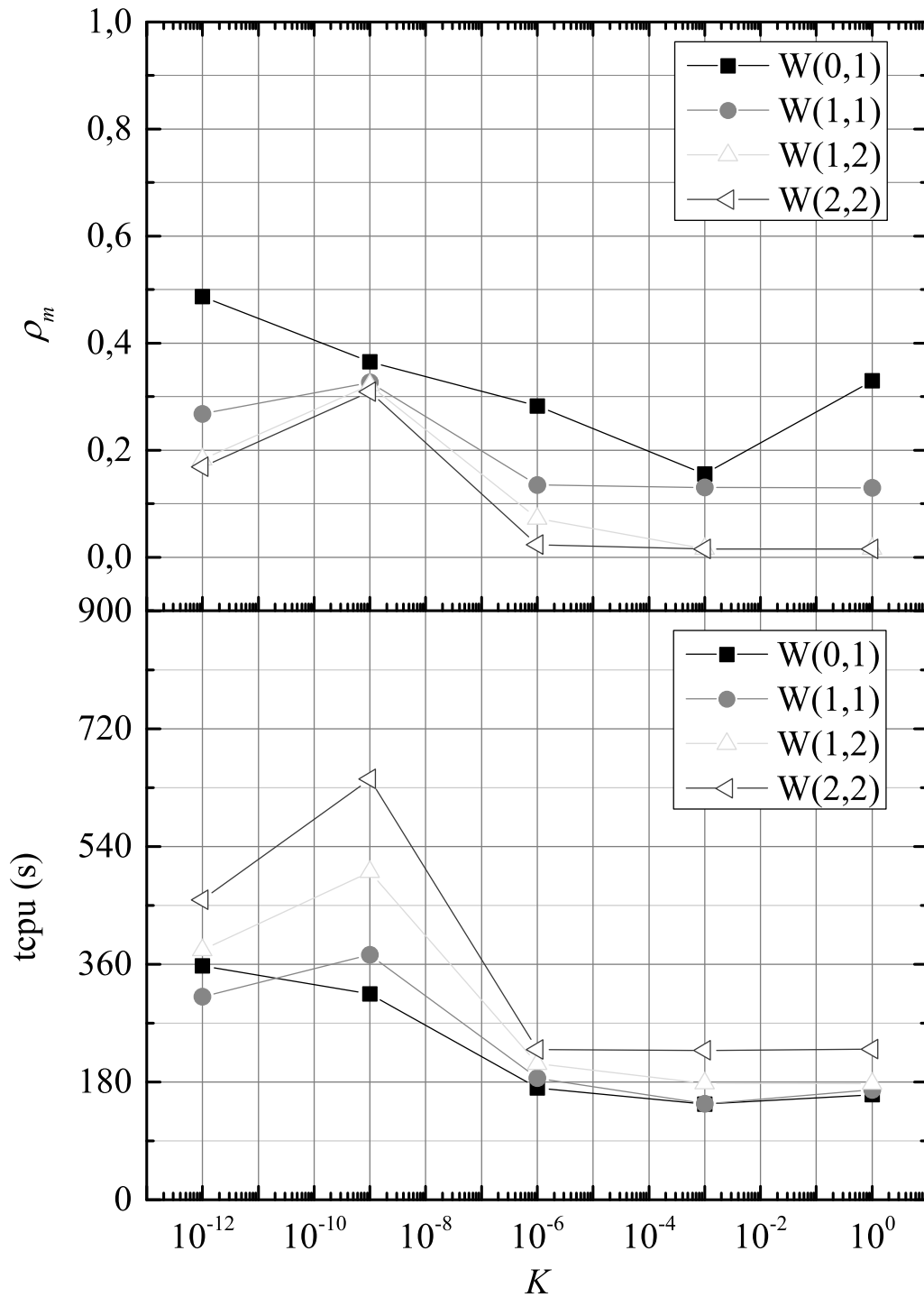


Figura 92 – Fator de convergência médio ρ_m e t_{CPU} considerando o método *Waveform Relaxation - multigrid*, Euler e o suavizador Vanka de 3 pontos com ordenação lexicográfica para o problema da poroelasticidade 1D.

considerando o suavizador Vanka de 5 pontos com ordenação lexicográfica e de 12 cores, respectivamente.

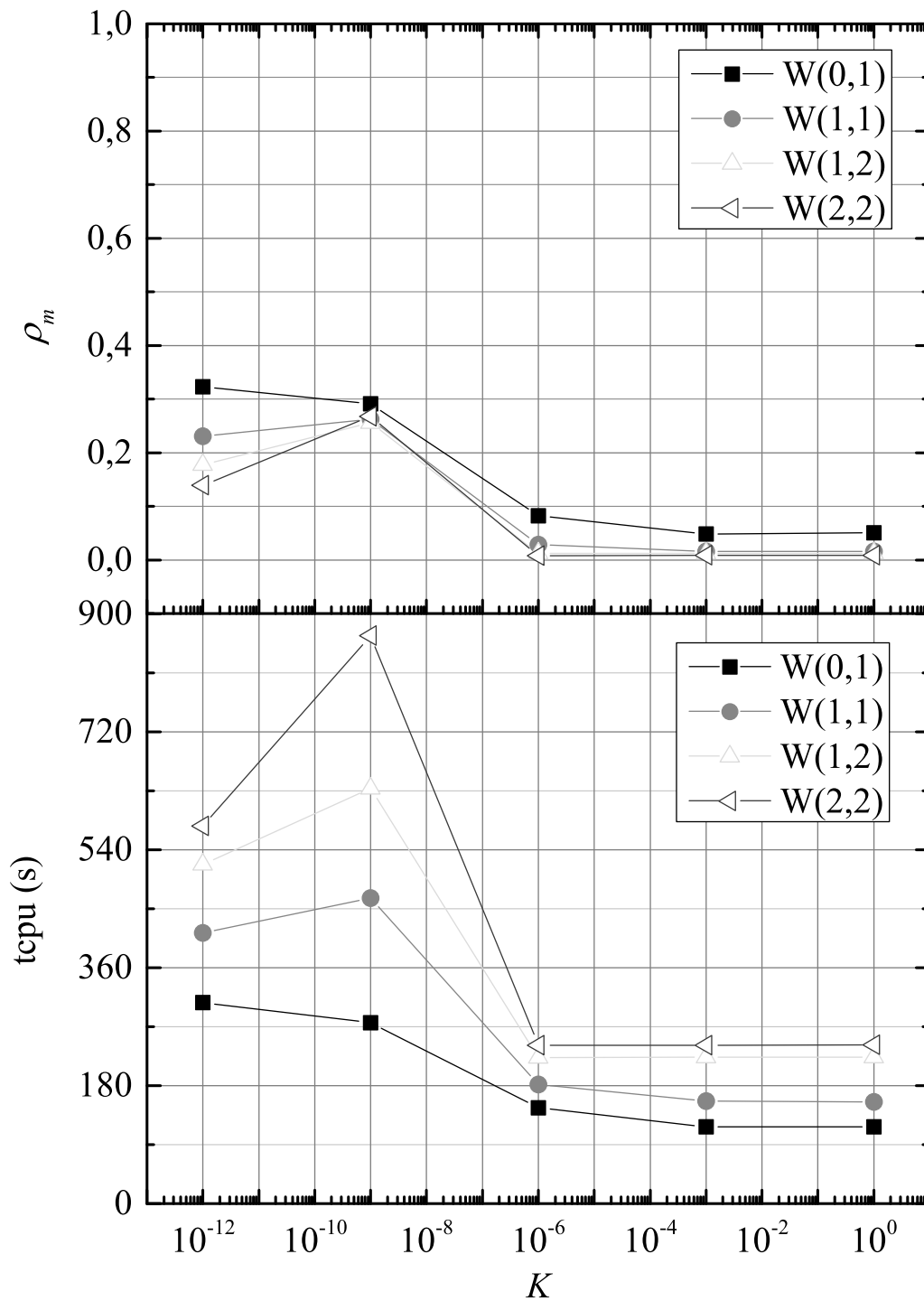


Figura 93 – Fator de convergência médio ρ_m e t_{CPU} considerando o método *Waveform Relaxation - multigrid*, Euler e o suavizador Vanka de 3 pontos com ordenação de 4 cores para o problema da poroelasticidade 1D.

Com os resultados apresentados nas Fig. 92 a 95 e considerando-se apenas o t_{CPU} como parâmetro para avaliar os parâmetros ótimos do referido método, pode-se considerar o ciclo $W(0,1)$ como a melhor opção. Em Franco et al. (2018b) considerou-se também o

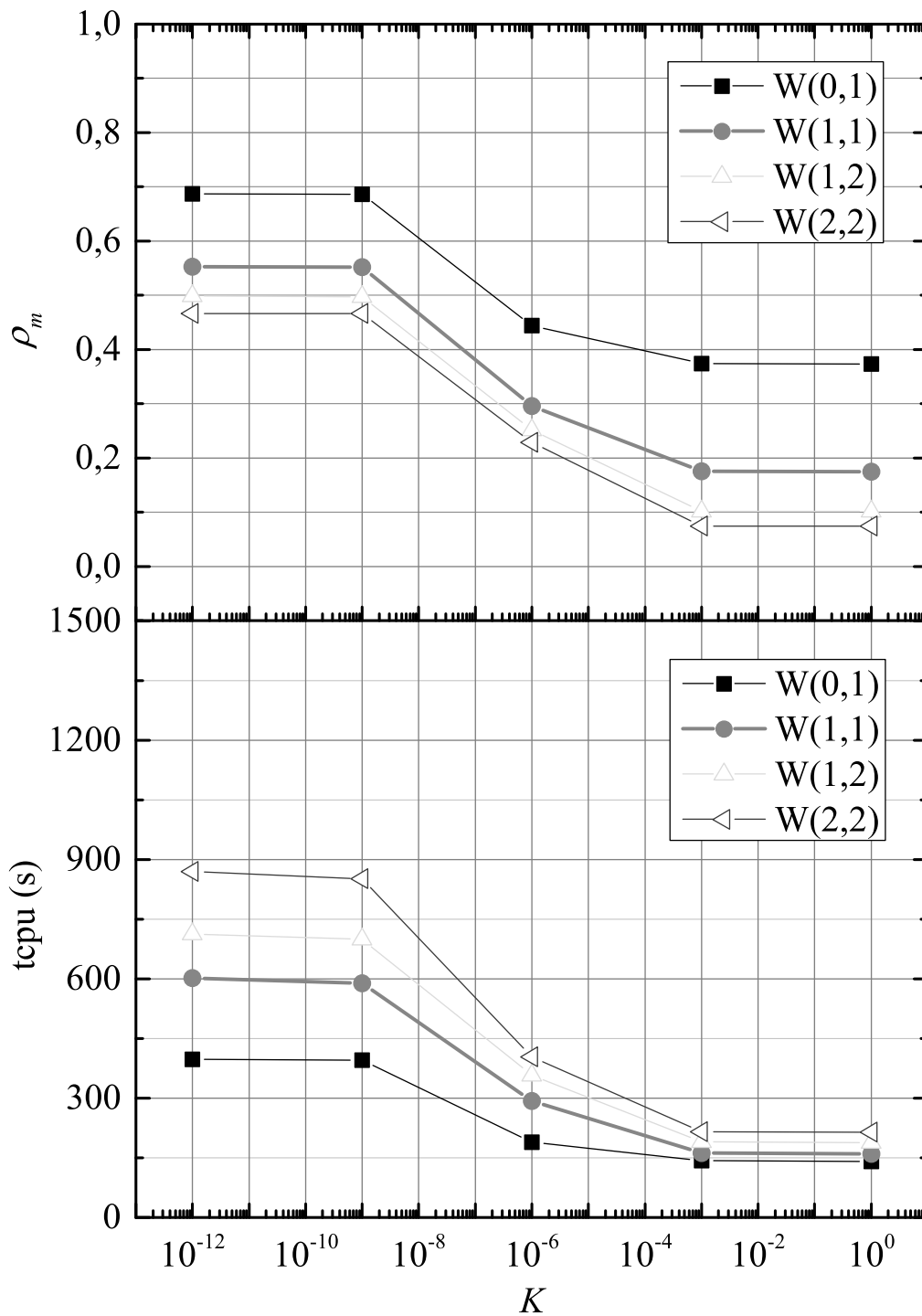


Figura 94 – Fator de convergência médio ρ_m e t_{CPU} considerando o método *Waveform Relaxation - multigrid*, Euler e o suavizador Vanka de 5 pontos com ordenação lexicográfica para o problema da poroelasticidade 2D.

fator de convergência médio para eleger tais parâmetros.

Portando, o método *Waveform Relaxation - multigrid* com o uso do suavizador

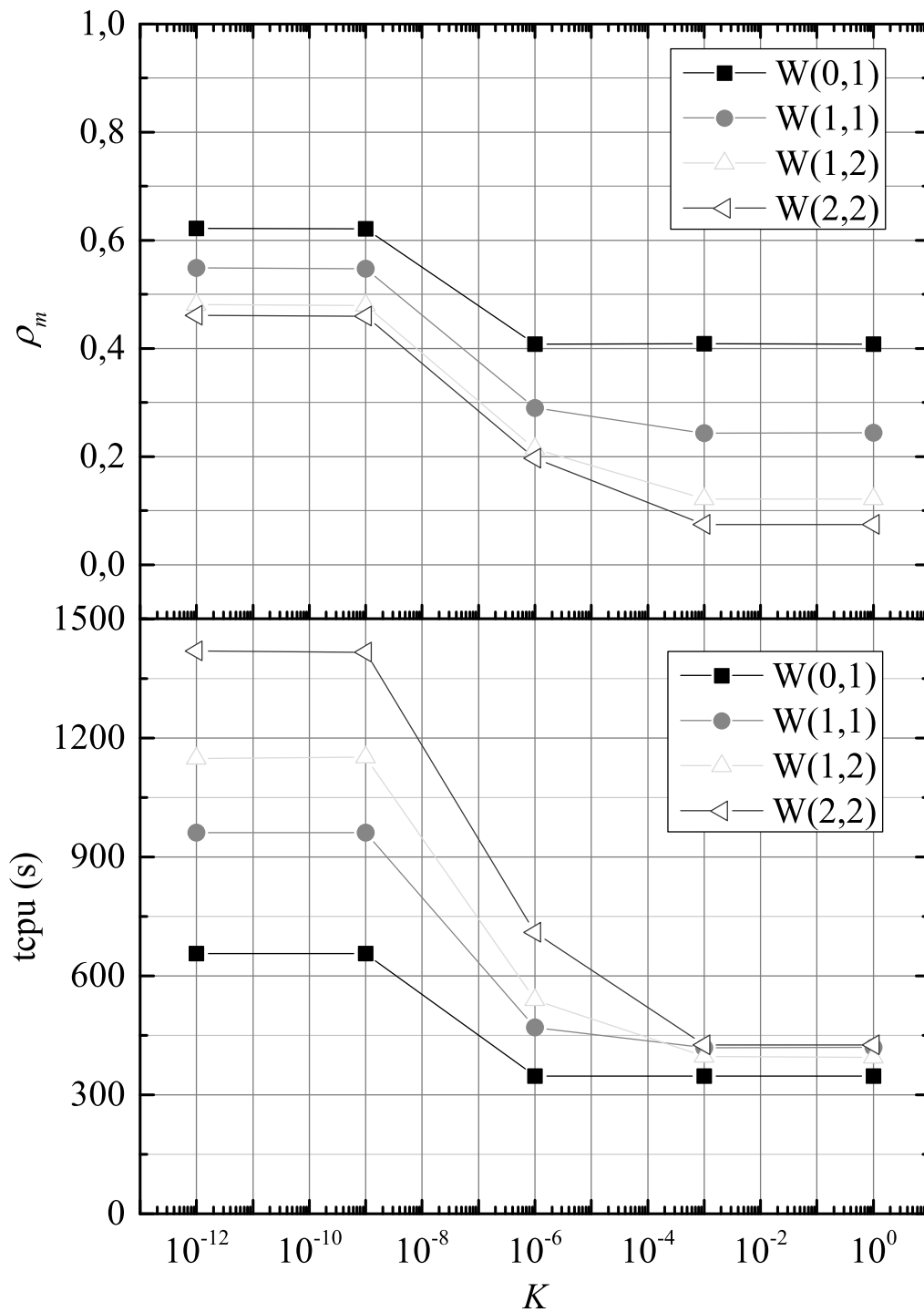


Figura 95 – Fator de convergência médio ρ_m e t_{CPU} considerando o método *Waveform Relaxation - multigrid*, Euler e o suavizador Vanka de 5 pontos com ordenação de 12 cores para o problema da poroelasticidade 2D.

Vanka, proposto nesta tese, mostrou-se um excelente método para aproximar a solução de problemas da poroelasticidade uni e bidimensional usando os métodos de Euler e de CN para a aproximação temporal. Observa-se que com apenas 2 suavizações internas, em todos

os casos dos problemas uni e bidimensional realísticos (para $K \leq 10^{-6}$, por exemplo), os fatores de convergência médio sempre foram menores que 0,35. A formulação colorida do suavizador Vanka (altamente paralelizável), permite ao método *Waveform Relaxation - multigrid* uma melhor eficiência tornando esse método muito mais atrativo se comparado com o método *Time-Stepping* existente para resolver o mesmo problema.

11 Conclusão

Este trabalho teve como principal objetivo o estudo dos métodos de solução para problemas transientes, dados pelas equações do calor e da poroelasticidade uni e bidimensionais com o uso do método *multigrid*.

Os primeiros métodos estudados nesta tese para resolver a equação do calor foram os métodos *Time-Stepping - multigrid*, *Waveform Relaxation - multigrid* e *Space-Time - multigrid*. Os métodos *Time-Stepping* e *Waveform Relaxation* apresentam robustez mesmo com o uso do ciclo V para os métodos de Euler e de Crank-Nicolson (CN), enquanto que o método *Space-Time* necessita de ciclo F ou W para ser robusto e além disso, não converge com o método de CN. Verificou-se também através de experimentos numéricos que o método *Time-Stepping* é mais eficiente que os demais métodos citados para resolver a equação do calor usando apenas um núcleo de processamento. Porém, se pensar em paralelismo de computadores, verificou-se na literatura (veja por exemplo, Falgout et al. (2017) e Gander e Neumüller (2016)) que o método *Space-Time* é muito bom, mas os métodos *Waveform Relaxation* com redução no tempo e o novo *space-time* com suavização por blocos (não usados nesta tese) apresentam melhor eficiência em relação ao paralelismo e assintoticamente poderão superar o método *Space-Time*. Porém, o método *Space-Time* com engrossamento padrão, proposto nesta tese, apresentou fatores de convergência comparáveis aos demais métodos estudados, possibilidade de alcançar 2ª ordem de acurácia usando o método da dupla discretização (com CN), além de robustez com o ciclo V e suavizador colorido, tornando o método altamente paralelizável no espaço e no tempo. Essas características fazem com que esse método seja atrativo para resolver esse tipo de problema.

A teoria da poroelasticidade, inicialmente descrita por Biot e que descreve o comportamento acoplado entre os fluidos e sólidos em meios porosos era, até então, resolvida numericamente com o uso do método *multigrid*, basicamente através do método *Time-Stepping*. Devido a esse método não ser paralelizável no tempo, propôs-se nesta tese o método *Waveform Relaxation - multigrid*, que é um método que atualiza os valores no espaço e tempo e é paralelizável. O uso desse método aliado ao suavizador Vanka (com ordenação lexicográfica ou colorida) apresentou robustez e bons fatores de convergência, mostrando a bom desempenho do método *Waveform Relaxation - multigrid*. Essas características fazem com que esse método seja atrativo para resolver esse tipo de problema.

11.1 Contribuições

Método *Space-Time* com engrossamento padrão para a equação do calor

Nesta tese foi desenvolvido um método robusto *Space-Time - multigrid* que combina engrossamento padrão com a seleção de suavizadores adequados, dependendo da anisotropia do problema, para resolver a equação de calor. A escolha do suavizador utilizado é feita usando um valor crítico do parâmetro λ , que representa o grau de anisotropia no operador discreto, resultando em uma estratégia de suavização adaptativa. Este parâmetro foi selecionado com a ajuda de uma análise local de Fourier (LFA) e comprovado através de experimentos numéricos.

Foram considerados problemas com uma e duas dimensões espaciais e a aproximação temporal foi realizada usando os esquemas bem conhecidos de primeira e segunda ordem correspondentes aos métodos de Euler e Crank-Nicolson (CN). Este último é resolvido aplicando a técnica da discretização dupla no algoritmo de *Space-Time - multigrid* com engrossamento padrão. Este é um dos pontos fortes do método proposto, uma vez que a literatura não fornece um método *Space-Time - multigrid* com engrossamento no tempo para o esquema de CN.

Os suavizadores utilizados são: zebra com linha no tempo ou zebra com linha no espaço para o problema unidimensional; e linha no tempo *red-black* ou zebra com planos no espaço para o caso bidimensional. Para o suavizador zebra com planos no espaço, ao invés de se resolver de forma direta (caro computacionalmente), fez-se suavizações com um ciclo *multigrid* $V(1, 2)$ em cada plano espacial. O suavizador utilizado neste plano espacial foi o Gauss-Seidel *red-black*.

O fato desse método apresentar robustez com o ciclo $V(0, 1)$ poderá representar uma grande vantagem se for utilizado paralelismo de computadores (lembrando que o método *Space-Time - multigrid* necessita de ciclo F ou W para ser robusto), pois o ciclo V é mais eficiente que os demais ciclos em relação a paralelismo de computadores.

O método da dupla discretização se mostrou muito atraente por apresentar fatores de convergência comparáveis aos fatores de apresentados pelo método de Euler, esforço computacional similar e 2^a ordem de acurácia para o decaimento dos erros.

Método *Waveform Relaxation* para o problema da poroelasticidade

Como apenas o método *Time-Stepping - multigrid* era usado para encontrar a solução de problemas da poroelasticidade, pensou-se em usar as metodologias adotadas na solução da equação do calor, como o método *Waveform Relaxation - multigrid*, para esse tipo de problema.

Assim, um eficiente e robusto método *multigrid* no espaço e no tempo foi proposto

para resolver o modelo de consolidação linear de Biot usando a discretização através do Método das Diferenças Finitas. Este método *multigrid* é construído em uma iteração de relaxamento de forma *Waveform Relaxation* com base em um suavizador Vanka de 3 pontos para o caso 1D e Vanka de 5 pontos para o caso 2D, que é a chave para o bom desempenho do método. Os resultados mostraram um bom desempenho do novo método, que também permite uma melhor eficiência com o uso de paralelização com o suavizador Vanka de 3 pontos e 4 cores para o caso 1D, e Vanka 5 pontos e 12 cores para o caso 2D.

11.2 Proposta de trabalhos futuros

Com a finalidade de complementar e expandir os estudos deste trabalho, são sugeridos os seguintes temas:

- Estudo do método *Space-Time* com engrossamento padrão para os problemas da poroelasticidade uni e bidimensionais (já em fase avançada de estudo);
- Desenvolvimento de um código em paralelo (paralelização em GPU) para o método *Space-Time* com engrossamento padrão aplicado à equação do calor;
- Programação do método *multigrid* com redução no tempo (MGRIT), proposto em Falgout et al. (2017) e o novo método *Space-Time* paralelo (STMG-BR), proposto em Gander e Neumüller (2016), usando paralelização em GPU;
- Comparação entre os métodos descritos nos dois últimos itens aplicados na solução do problema de condução de calor uni e bidimensional. Acredita-se que o método *Space-Time* com engrossamento padrão apresentará melhores resultados que os demais métodos, devido ao fato de apresentar robustez com o ciclo V e usar suavizadores coloridos (altamente paralelizáveis) no espaço e tempo;
- Comparação entre os métodos descritos no item anterior aplicados na solução do problema de condução de calor tridimensional;
- Aplicação de paralelização aos problemas da poroelasticidade com os métodos estudados;
- Testar os métodos descritos nesta tese em malhas malhas semiestruturadas triangulares;
- Testar os métodos descritos nesta tese para o problema da cavidade quadrada descrito pelas equações de Navier-Stokes, nas variáveis primitivas.

Referências

- BAKHVALOV, N. S. On the convergence of relaxation method with natural constraints on the elliptic operator. *URSS Comp. Math. Phys.*, v. 1, n. 5, 1966.
- BIOT, M. A. General theory of three dimensional consolidation. *Journal of Applied Physics*, v. 12, n. 2, p. 155–164, 1941.
- BIOT, M. A. Theory of elasticity and consolidation for a porous anisotropic solid. *Journal of Applied Physics*, v. 26, n. 2, p. 182–185, 1955.
- BRANDT, A. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, v. 31, n. 138, p. 333–390, 1977.
- BRANDT, A.; LIVNE, O. Multigrid techniques: 1984 guide with applications to fluid dynamics. *Society for Industrial and Applied Mathematics*, 2011.
- BRIGGS, W. L.; HENSON, V. E.; MCCORMICK, S. F. *A Multigrid Tutorial*. 2nd. ed. Philadelphia: SIAM, 2000.
- BURDEN, R. L.; FAIRES, J. D. *Numerical Analysis*. 10nd. ed. Boston: Cengage Learning, 2016.
- CHOPTUIK, M. W. *Using Multigrid to Solve Time Dependent PDEs*. 2008. 4 Feb. 2008 <<http://wwwsfb.tpi.uni-jena.de/VideoSeminar/Files/20080204-choptuik.pdf>>.
- EHLERS, W.; BLUHM, J. *Porous Media: Theory, Experiments and Numerical Applications*. Berlin, Germany: Springer, 2002.
- EMMETT, M.; MINION, M. L. Toward an efficient parallel in time method for partial differential equations. *Communications in Applied Mathematics and Computational Science*, v. 22, n. 1, p. 105–132, 2012.
- FALGOUT, R. D.; FRIEDHOFF, S.; KOLEV, T. V.; MACLACHLAN, S. P.; SCHRODER, J. B.; VANDEWALLE, S. Parallel time integration with multigrid. *SIAM J. Sci. Comput.*, v. 36, p. C625–C661, 2014.
- FALGOUT, R. D.; FRIEDHOFF, S.; KOLEV, T. V.; MACLACHLAN, S. P.; SCHRODER, J. B.; VANDEWALLE, S. Multigrid methods with space-time concurrency. *Computing and Visualization in Science*, v. 18, p. 123–143, 2017.
- FEDORENKO, R. P. On the speed of convergence of an iteration process. *USSR Comput. Math. and Math. Phys.*, n. 4, p. 227, 1964.
- FERZIGER, J. H.; PERIC, M. *Computational Methods for Fluid Dynamics*. 3nd. ed. New York: Springer, 2002.
- FORTUNA, A. O. *Técnicas Computacionais para Dinâmica dos Fluidos: conceitos básicos e aplicações*. 2nd. ed. São Paulo: Edusp, 2000.

- FRANCO, S. R.; GASPAR, F. J.; PINTO, M. A. V.; RODRIGO, C. Multigrid method based on a space-time approach with standard coarsening for parabolic problems. *Applied Mathematics and Computation*, v. 317, p. 25–34, 2018a.
- FRANCO, S. R.; PINTO, M. A. V.; VARGAS, A. P. d. S. Comparison of efficiency between the time stepping and waveform relaxation methods for two-dimensional fourier equation. In: *23rd ABCM International Congress of Mechanical Engineering*. Rio de Janeiro: 23rd ABCM International Congress of Mechanical Engineering, 2015.
- FRANCO, S. R.; RODRIGO, C.; GASPAR, F. J.; PINTO, M. A. V. A multigrid waveform relaxation method for solving the poroelasticity equations. *Submetido para Computational and Applied Mathematics*, 2018b.
- GANDER, M. 50 years of time parallel time integration. *Springer International Publishing*, p. 69–113, 2015.
- GANDER, M. J.; NEUMÜLLER, M. Analysis of a new space-time parallel multigrid algorithm for parabolic problems. *SIAM J. SCI. COMPUT*, v. 38, n. 4, p. A2173–A2208, 2016.
- GANDER, M. J.; VANDEWALLE, S. Analysis of the parareal time-parallel time-integration method. *Society for Industrial and Applied Mathematics*, v. 29, n. 2, p. 556–578, 2007.
- GASPAR, F.; LISBONA, F.; OOSTERLEE, C. W.; WIENANDS, R. A systematic comparison of coupled and distributive smoothing in multigrid for the poroelasticity system. *Numerical Linear Algebra with Applications*, v. 11, n. 28, p. 93–113, 2004.
- GASPAR, F. J.; LISBONA, F. J.; OOSTERLEE, C. W.; VABISHCHEVICH, P. N. An efficient multigrid solver for a reformulated version of the poroelasticity system. *Computer Methods Applied Mechanics Engineering*, v. 196, 2007.
- GASPAR, F. J.; LISBONA, F. J.; VABISHCHEVICH, P. N. A finite difference analysis of biot's consolidation model. *Applied Numerical Mathematics*, v. 44, p. 487–506, 2003.
- GASPAR, F. J.; LISBONA, F. J.; VABISHCHEVICH, P. N. Staggered grid discretizations for the quasi-static biot's consolidation problem. *Applied Numerical Mathematics*, v. 56, p. 888–898, 2006.
- GASPAR, F. J.; RODRIGO, C. Local fourier analysis of vanka smoother based multigrid for staggered discretization of biot's consolidation problem. *Australian Journal of Basic and Applied Sciences*, v. 9, n. 28, p. 18–25, 2015.
- GHIA, U.; GHIA, K. N.; GHIN, C. T. High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of Computational Physics*, v. 48, p. 387–411, 1982.
- GOLUB, G. H.; LOAN, C. V. *Matrix Computations*. 2nd. ed. Baltimore: Johns Hopkins Press, 1989.
- GOLUB, G. H.; ORTEGA, J. M. *Scientific Computing and Differential Equations: an Introduction to Numerical Methods*. [S.l.]: Academic Press, 1992.
- HACKBUSCH, W. *Multi-grid Methods and Applications*. Berlin: Springer-Verlag, 1985.

- HIRSCH, C. *Numerical Computation of Internal and External Flows: The Fundamentals of Computational Fluid Dynamics*. 2nd. ed. New York, USA: John Wiley & Sons, 2007.
- HORTON, G.; VANDEWALLE, S. A space time multigrid method for parabolic partial differential equations. *SIAM J. Sci. Comput.*, v. 16, n. 4, p. 848–864, 1995.
- HORTON, G.; VANDEWALLE, S.; WORLEY, P. An algorithm with polylog parallel complexity for solving parabolic partial differential equations. *SIAM Journal on Scientific Computing*, v. 16, n. 3, p. 531–541, 1995.
- INCROPERA, F. P.; DEWITT, D. P.; BERGMAN, T. L.; LAVINE, T. L. *Fundamentos de Transferência de Calor e Massa*. 6nd. ed. Rio de Janeiro: LTC, 2008.
- JANSEN, J. K. M.; MATTHEIJ, R. M. M.; PENDERS, M. T. M. Stability and efficiency of waveform relaxation methods. *Computers Math. Applic.*, v. 28, n. 1-3, p. 153–166, 1994.
- JANSSEN, J. *Acceleration of Waveform Relaxation Methods for Linear Ordinary and Partial Differential Equations*. Tese (Doutorado) — Katholieke Universiteit Leuven, Heverlee, Belgium, 1997.
- JOHN, V. A comparison of parallel solvers for the incompressible navier-stokes equations. *Computing and Visualization Science*, v. 1, p. 193–200, 1999.
- KREYSZIG, E. *Advanced Engineering Mathematics*. 8nd. ed. Wiley: Springer, 1999.
- LELARASMEE, E.; RUEHLI, A. E.; VINCENNELLI, S. A. L. The waveform relaxation method for time domain analysis of large scale integrated circuits theory and applications. *IEEE Trans. Comput. Aided Design Integr. Circ. Systems*, n. 1, p. 131–145, 1982.
- LENT, J. V. *Multigrid Method for Time-Dependent Partial Differential Equations*. Tese (Doutorado) — Katholieke Universiteit Leuven, 2006.
- LENT, J. V.; VANDEWALLE, S. Multigrid waveform relaxation for anisotropic partial differential equations. *Numerical Algorithms*, v. 31, p. 361–380, 2002.
- LENT, J. V.; VANDEWALLE, S. Multigrid methods for implicit runge–kutta and boundary value method discretizations of parabolic pdes. *Siam J. Sci. Comput.*, v. 27, n. 1, p. 67–92, 2005.
- LLORENTE, I. M.; MELSON, N. D. Robust multigrid smoothers for three dimensional elliptic equations with strong anisotropies. *ICASE*, v. 98-37, 1998.
- LLORENTE, I. M.; MELSON, N. D. Behavior of plane relaxation methods as multigrid smoothers. *Electronic Transactions on Numerical Analysis*, v. 10, p. 92–114, 2000.
- LUBICH, C.; OSTERMANN, A. Multigrid dynamic iteration for parabolic equations. *BIT*, v. 27, n. 2, p. 216–234, 1987.
- MALISKA, C. R. *Transferência de Calor e Mecânica dos Fluidos Computacionais*. 2nd edição. ed. Rio de Janeiro: LTC, 2004.
- MARCHI, C. H. *Verificação de soluções numéricas unidimensionais em dinâmica dos fluidos*. Tese (Doutorado) — Universidade Federal de Santa Catarina, Florianópolis, 2001.

- MIEKKALA, U.; NEVANLINNA, O. Convergence of dynamic iteration methods for initial value problems. *SIAM J. Sci. Statist. Comput.*, v. 8, p. 459–482, 1997.
- MOSSI, A. *Modelagem da radiação térmica em chamas laminares da combustão de metano em ar*. Tese (Doutorado) — Universidade Federal do Rio Grande do Sul, Porto Alegre, 2011.
- MOW, V. C.; LAI, W. M. Recent developments in synovial joint biomechanics. *SIAM Review*, v. 7, p. 275–317, 1980.
- NGUYEN, V. T.; GRAF, T.; MOREL, C. R. Free thermal convection in heterogeneous porous media. *Geothermics*, 2016.
- OLIVEIRA, F.; PINTO, M. A. V.; MARCHI, C. H.; K., A. L. Optimized partial semicoarsening multigrid algorithm for heat diffusion problems and anisotropic grids. *Applied Mathematical Modelling*, v. 36, p. 4665–4676, 2012.
- OOSTERLEE, C. W.; GASPAR, F. J. Multigrid relaxation methods for systems of saddle point type. *Appl. Numer. Math.*, v. 58, p. 1933–1950, 2008.
- PINTO, M. A. V.; MARCHI, C. H. Optimum parameters of a geometric multigrid for the two-dimensional laplace's equation. *Proceedings of COBEM*, 2007.
- PRIETO, M.; SANTIAGO, R.; ESPADAS, D.; LLORENTE, I. M.; TIRADO, F. Parallel multigrid for anisotropic elliptic equations. *Journal of Parallel and Distributed Computing*, v. 61, p. 96–114, 2001.
- ROACHE, P. J. *Fundamentals of Computational Fluid Dynamics*. Albuquerque, USA: Hermosa Publishers, 1998.
- RODRIGO, C. *Geometric Multigrid Methods on Semi-Structured Triangular Grids*. Tese (Doutorado) — Universidad de Zaragoza, Zaragoza, España, 2010.
- RODRIGO, C.; GASPAR, F. J.; LISBONA, F. J. On a local fourier analysis for overlapping block smoothers on triangular grids. *Applied Numerical Mathematics*, v. 105, p. 96–111, 2016.
- RUGE, J. W.; STUBEN, K. Algebraic multigrid. In: MCCORMICK, S. (Ed.). *Frontiers in Applied Mathematics*. Philadelphia: SIAM, 1986. v. 5.
- RUGGIERO, M. A. G.; LOPES, V. L. d. R. *Cálculo Numérico - Aspectos Teóricos e Computacionais*. 2nd edição. ed. São Paulo: Pearson, 1996.
- SAAD, Y. *Iterative Methods for Sparse Linear Systems*. 2nd. ed. Philadelphia: PWS, 2003.
- SANTIAGO, C. D.; MARCHI, C. H.; SOUZA, L. F. Performance of geometric multigrid method for coupled two-dimensional systems in cfd. *Applied Mathematical Modelling*, v. 39, p. 2602–2616, 2015.
- SCHAEFFER, S. *High-order multigrid methods*. Tese (Doutorado) — Colorado State University, Fort Collins, Colorado, 1982.
- STÜBEN, K. A review of algebraic multigrid. *Journal of Computational and Applied Mathematics*, n. 128, p. 281–309, 2001.

- STÜBEN, K.; TROTTEBERG, U. Multigrid methods: Fundamental algorithms, model problem analysis and applications. In: HACKBUSCH, W.; TROTTEBERG, U. (Ed.). *Proceedings of the Conference Held at Köln-Porz*. New York: Springer-Verlag, 1981. p. 1–176.
- STEINBRUCH, A.; WINTERLE, P. *Introdução à álgebra linear*. São Paulo: Pearson Education do Brasil, 1997.
- STRIKWERDA, J. C. *Finite Difference Schemes and Partial Differential Equations*. 2nd. ed. Pacific Grove, CA: SIAM, 1989.
- TANNEHILL, J. C.; ANDERSON, D. A.; PLETCHER, R. H. *Computational Fluid Mechanics and Heat Transfer*. Philadelphia: Taylor & Francis, 1997.
- TERZAGHI, K. *Theoretical Soil Mechanics*. New York: John Wiley & Sons, 1943.
- THOLE, C.; TROTTEBERG, U. Basic smoothing procedures for the multigrid treatment of elliptic 3d operators. *Appl. Math. Comput.*, v. 333, n. 19, 1986.
- TROTTEBERG, U.; OOSTERLEE, C.; SCHÜLLER, A. *Multigrid*. San Diego: Academic Press, 2001.
- TUREK, S. *Efficient solvers for incompressible flow problems: an algorithmic and computational approach*. Berlin: Springer, 1999.
- VANDEWALLE, S. *Parallel Multigrid Waveform Relaxation for Parabolic Problems*. Stuttgart: Teubner, 1993.
- VANDEWALLE, S.; HORTON, G. Multicomputer-multigrid solution of parabolic partial differential equations. In: HEMKE, P. W.; WESSELIN, P. (Ed.). *Multigrid Methods IV: Proceedings of the Fourth European Multigrid Conference*. Amsterdam: Springer Basel AG, 1993. p. 97–109.
- VANDEWALLE, S.; HORTON, G. Fourier mode analysis of the multigrid waveform relaxation and time-parallel multigrid methods. *Computing*, v. 54, p. 317–330, 1995.
- VANKA, S. P. Block-implicit multigrid solution of navier-stokes equations in primitive variables. *Journal of Computational Physics*, v. 65, n. 1, p. 138–158, 1986.
- VARGAS, A. P. S. *Multiextrapolação de Richardson e esquemas de 1ª e 2ª ordens, mistos e Crank-Nicolson sobre as equações 2D de advecção-difusão Fourier*. Tese (Doutorado) — Universidade Federal do Paraná, Curitiba, 2013.
- WESSELING, P. *An Introduction to Multigrid Methods*. Chichester: John Wiley & Sons, 1992.
- WESSELING, P.; OOSTERLEE, C. Multigrid with applications to computational fluid dynamics. *Comput. Appl. Math.*, v. 128, p. 311–334, 2001.
- WIENANDS, R.; GASPAR, F. J.; LISBONA, F. J.; OOSTERLEE, C. W. Analysis of a pointwise smoother in multigrid for poroelasticity. *Journal Computational Fluid and Solid Mechanics*, v. 2, 2003.

WIENANDS, R.; GASPAR, F. J.; LISBONA, F. J.; OOSTERLEE, C. W. An efficient multigrid solver based on distributive smoothing for poroelasticity equations. *Computing*, v. 73, p. 99–119, 2004.

WIENANDS, R.; JOPPICH, W. *Practical Fourier Analysis for Multigrid Methods*. Washington, USA: Chapman & Hall/CRC Press, 2005. v. 4.

Apêndices

APÊNDICE A – Programação do suavizador Vanka

Nesta seção será descrita a construção dos sistemas de equações para o suavizador Vanka de 3 pontos para o caso unidimensional e Vanka de 5 pontos para o caso bidimensional.

Caso unidimensional

Método de Euler

Considerando-se o sistema de Eq. (6.40) descrito usando o método de Euler e as incógnitas u_{i-1}^{m+1} , p_i^{m+1} e u_{i+1}^{m+1} , para $i = 3, 4, \dots, N - 2$ tem-se

$$\left\{ \begin{array}{l} \frac{2E}{h^2} u_{i-1}^{m+1} = \frac{E}{h^2} (u_{i-2}^{m+1} + u_i^{m+1}) + \frac{1}{2h} (p_{i-2}^{m+1} - p_i^{m+1}) + \mathcal{U}_{i-1}^{m+1} \\ \left(\frac{h^2 + 4E\tau k}{2E\tau h^2} \right) p_i^{m+1} = \frac{1}{2\tau h} (u_{i-1}^{m+1} - u_{i+1}^{m+1}) + \left(\frac{h^2 + 4E\tau k}{4E\tau h^2} \right) \cdot \\ \cdot (p_{i-1}^{m+1} + p_{i+1}^{m+1}) - \frac{1}{2\tau h} (u_{i-1}^m - u_{i+1}^m) - \frac{1}{4E\tau} (p_{i-1}^m - 2p_i^m + p_{i+1}^m) + \mathcal{P}_i^{m+1} \\ \frac{2E}{h^2} u_{i+1}^{m+1} = \frac{E}{h^2} (u_i^{m+1} + u_{i+2}^{m+1}) + \frac{1}{2h} (p_i^{m+1} - p_{i+2}^{m+1}) + \mathcal{U}_{i+1}^{m+1} \end{array} \right. , \quad (\text{A.1})$$

reorganizando os termos obtém-se

$$\left\{ \begin{array}{l} \frac{2E}{h^2} u_{i-1}^{m+1} + \frac{1}{2h} p_i^{m+1} = \frac{E}{h^2} (u_{i-2}^{m+1} + u_i^{m+1}) + \frac{1}{2h} p_{i-2}^{m+1} + \mathcal{U}_{i-1}^{m+1} \\ -\frac{1}{2\tau h} u_{i-1}^{m+1} + \left(\frac{h^2 + 4E\tau k}{2E\tau h^2} \right) p_i^{m+1} + \frac{1}{2\tau h} u_{i+1}^{m+1} = \left(\frac{h^2 + 4E\tau k}{4E\tau h^2} \right) \cdot \\ \cdot (p_{i-1}^{m+1} + p_{i+1}^{m+1}) - \frac{1}{2\tau h} (u_{i-1}^m - u_{i+1}^m) - \frac{1}{4E\tau} (p_{i-1}^m - 2p_i^m + p_{i+1}^m) + \mathcal{P}_i^{m+1} \\ -\frac{1}{2h} p_i^{m+1} + \frac{2E}{h^2} u_{i+1}^{m+1} = \frac{E}{h^2} (u_i^{m+1} + u_{i+2}^{m+1}) - \frac{1}{2h} p_{i+2}^{m+1} + \mathcal{U}_{i+1}^{m+1} \end{array} \right. . \quad (\text{A.2})$$

Em forma matricial, para $i = 3, 4, \dots, N - 2$, tem-se

$$\begin{aligned}
 & \begin{pmatrix} \frac{2E}{h^2} & \frac{1}{2h} & 0 \\ -\frac{1}{2\tau h} & \frac{h^2+4E\tau k}{2E\tau h^2} & \frac{1}{2\tau h} \\ 0 & -\frac{1}{2h} & \frac{2E}{h^2} \end{pmatrix} \begin{pmatrix} u_{i-1}^{m+1} \\ p_i^{m+1} \\ u_{i+1}^{m+1} \end{pmatrix} = \\
 & = \begin{pmatrix} \frac{E}{h^2} & \frac{1}{2h} & 0 & 0 & 0 \\ 0 & 0 & \frac{h^2+4E\tau k}{4E\tau h^2} & 0 & 0 \\ 0 & 0 & 0 & \frac{E}{h^2} & -\frac{1}{2h} \end{pmatrix} \begin{pmatrix} u_{i-2}^{m+1} + u_i^{m+1} \\ p_{i-2}^{m+1} \\ p_{i-1}^{m+1} + p_{i+1}^{m+1} \\ u_i^{m+1} + u_{i+2}^{m+1} \\ p_{i+2}^{m+1} \end{pmatrix} + \\
 & + \begin{pmatrix} 0 & 0 \\ -\frac{1}{2\tau h} & -\frac{1}{4E\tau} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} u_{i-1}^m - u_{i+1}^m \\ p_{i-1}^m - 2p_i^m + p_{i+1}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{i-1}^{m+1} \\ \mathcal{P}_i^{m+1} \\ \mathcal{U}_{i+1}^{m+1} \end{pmatrix}. \tag{A.3}
 \end{aligned}$$

Em $i = 2$, considerando-se a Eq. (6.45) e as equações da pressão e deslocamento em Eq. (6.40), tem-se

$$\left\{ \begin{aligned}
 & \frac{2E}{h^2} u_1^{m+1} + \frac{1}{h} p_2^{m+1} = \frac{2E}{h^2} u_2^{m+1} + \mathcal{U}_1^{m+1} \\
 & -\frac{1}{2\tau h} u_1^{m+1} + \left(\frac{h^2 + 4E\tau k}{2E\tau h^2} \right) p_2^{m+1} + \frac{1}{2\tau h} u_3^{m+1} = \left(\frac{h^2 + 4E\tau k}{4E\tau h^2} \right) p_3^{m+1} \\
 & \quad -\frac{1}{2\tau h} (u_1^m - u_3^m) - \frac{1}{4E\tau} (p_1^m - 2p_2^m + p_3^m) + \mathcal{P}_2^{m+1} \\
 & -\frac{1}{2h} p_2^{m+1} + \frac{2E}{h^2} u_3^{m+1} = \frac{E}{h^2} (u_2^{m+1} + u_4^{m+1}) - \frac{1}{2h} p_4^{m+1} + \mathcal{U}_3^{m+1}
 \end{aligned} \right. \tag{A.4}$$

Em forma matricial, para $i = 2$ obtém-se

$$\begin{aligned}
 & \begin{pmatrix} \frac{2E}{h^2} & \frac{1}{h} & 0 \\ -\frac{1}{2\tau h} & \frac{h^2+4E\tau k}{2E\tau h^2} & \frac{1}{2\tau h} \\ 0 & -\frac{1}{2h} & \frac{2E}{h^2} \end{pmatrix} \begin{pmatrix} u_1^{m+1} \\ p_2^{m+1} \\ u_3^{m+1} \end{pmatrix} = \begin{pmatrix} \frac{2E}{h^2} & 0 & 0 & 0 \\ 0 & 0 & \frac{h^2+4E\tau k}{4E\tau h^2} & 0 \\ \frac{E}{h^2} & \frac{E}{h^2} & 0 & -\frac{1}{2h} \end{pmatrix} \begin{pmatrix} u_2^{m+1} \\ u_4^{m+1} \\ p_3^{m+1} \\ p_4^{m+1} \end{pmatrix} \\
 & + \begin{pmatrix} 0 & 0 \\ -\frac{1}{2\tau h} & -\frac{1}{4E\tau} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} u_1^m - u_3^m \\ p_3^m - 2p_2^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_1^{m+1} \\ \mathcal{P}_2^{m+1} \\ \mathcal{U}_3^{m+1} \end{pmatrix}. \tag{A.5}
 \end{aligned}$$

Em $i = N - 1$, considerando-se a condição de contorno $u_N = 0$ e a Eq. (6.40)

tem-se

$$\begin{cases} \frac{2E}{h^2}u_{N-2}^{m+1} + \frac{1}{2h}p_{N-1}^{m+1} = \frac{E}{h^2}(u_{N-3}^{m+1} + u_{N-1}^{m+1}) + \frac{1}{2h}p_{N-3}^{m+1} + \mathcal{U}_{N-2}^{m+1} \\ -\frac{1}{2\tau h}u_{N-2}^{m+1} + \left(\frac{h^2 + 4E\tau k}{2E\tau h^2}\right)p_{N-1}^{m+1} = \left(\frac{h^2 + 4E\tau k}{4E\tau h^2}\right)(p_{N-2}^{m+1} + p_N^{m+1}) \\ -\frac{1}{2\tau h}u_{N-2}^m - \frac{1}{4E\tau}(p_{N-2}^m - 2p_{N-1}^m + p_N^m) + \mathcal{P}_{N-1}^{m+1} \end{cases} \quad (\text{A.6})$$

A forma matricial para este caso é

$$\begin{pmatrix} \frac{E}{h^2} & \frac{1}{2h} & 0 & 0 & 0 \\ 0 & 0 & \frac{h^2+4E\tau k}{4E\tau h^2} & -\frac{1}{2\tau h} & -\frac{1}{4E\tau} \end{pmatrix} \begin{pmatrix} u_{N-3}^{m+1} + u_{N-1}^{m+1} \\ p_{N-3}^{m+1} \\ p_{N-2}^{m+1} + p_N^{m+1} \\ u_{N-2}^m \\ p_{N-2}^m - 2p_{N-1}^m + p_N^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{N-2}^{m+1} \\ \mathcal{P}_{N-1}^{m+1} \end{pmatrix} = \begin{pmatrix} \frac{2E}{h^2} & \frac{1}{2h} \\ -\frac{1}{2\tau h} & \frac{h^2+4E\tau k}{2E\tau h^2} \end{pmatrix} \begin{pmatrix} u_{N-2}^{m+1} \\ p_{N-1}^{m+1} \end{pmatrix} \quad (\text{A.7})$$

Em $i = N$ ($x = L$), considerando-se a condição de contorno $u_N = 0$, a equação do deslocamento em Eq. (6.40) e a Eq. (6.50), tem-se

$$\begin{cases} \frac{2E}{h^2}u_{N-1}^{m+1} + \frac{1}{2h}p_N^{m+1} = \frac{E}{h^2}u_{N-2}^{m+1} + \frac{1}{2h}p_{N-2}^{m+1} + \mathcal{U}_{N-1}^{m+1} \\ -\frac{1}{\tau h}u_{N-1}^{m+1} + \left(\frac{h^2 + 4E\tau k}{2E\tau h^2}\right)p_N^{m+1} = \left(\frac{h^2 + 4E\tau k}{2E\tau h^2}\right)p_{N-1}^{m+1} \\ -\frac{1}{\tau h}u_{N-1}^m - \frac{1}{2E\tau}(p_{N-1}^m - p_N^m) + \mathcal{P}_N^{m+1} \end{cases} \quad (\text{A.8})$$

Na forma matricial esse sistema pode ser escrito

$$\begin{pmatrix} \frac{E}{h^2} & \frac{1}{2h} & 0 & 0 & 0 \\ 0 & 0 & \frac{h^2+4E\tau k}{2E\tau h^2} & -\frac{1}{\tau h} & -\frac{1}{2E\tau} \end{pmatrix} \begin{pmatrix} u_{N-2}^{m+1} \\ p_{N-2}^{m+1} \\ p_{N-1}^{m+1} \\ u_{N-1}^m \\ p_{N-1}^m - p_N^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{N-1}^{m+1} \\ \mathcal{P}_N^{m+1} \end{pmatrix} = \begin{pmatrix} \frac{2E}{h^2} & \frac{1}{2h} \\ -\frac{1}{\tau h} & \frac{h^2+4E\tau k}{2E\tau h^2} \end{pmatrix} \begin{pmatrix} u_{N-1}^{m+1} \\ p_N^{m+1} \end{pmatrix} \quad (\text{A.9})$$

Portanto, o suavizador Vanka de 3 pontos é formado pelos sistemas descritos nas Eqs. (A.5), (A.3), (A.7) e Eq. (A.9).

Para a malha mais grossa possível, com somente 3 pontos, de acordo com a teoria do método *multigrid*, o sistema deve ser resolvido de forma direta. Considerando-se as

condições de contorno $p_1 = 0$ e $u_3 = 0$, têm-se as incógnitas u_1, u_2, p_2, p_3 . A partir das Eq. (6.45), Eq. (6.40) e Eq. (6.50), tem-se

$$\left\{ \begin{array}{l} \frac{2E}{h^2} u_1^{m+1} = \frac{2E}{h^2} u_2^{m+1} - \frac{1}{h} p_2^{m+1} + \mathcal{U}_1^{m+1} \\ \frac{2E}{h^2} u_2^{m+1} = \frac{E}{h^2} u_1^{m+1} - \frac{1}{2h} p_3^{m+1} + \mathcal{U}_2^{m+1} \\ \left(\frac{h^2 + 4E\tau k}{2E\tau h^2} \right) p_2^{m+1} = \frac{1}{2\tau h} u_1^{m+1} + \left(\frac{h^2 + 4E\tau k}{4E\tau h^2} \right) p_3^{m+1} + \\ \quad - \frac{1}{2\tau h} u_1^m - \frac{1}{4E\tau} (-2p_2^m + p_3^m) + \mathcal{P}_2^{m+1} \\ \left(\frac{h^2 + 4E\tau k}{2E\tau h^2} \right) p_3^{m+1} = \frac{1}{\tau h} u_2^{m+1} + \left(\frac{h^2 + 4E\tau k}{2E\tau h^2} \right) p_2^{m+1} + \\ \quad - \frac{1}{\tau h} u_2^m - \frac{1}{2E\tau} (p_2^m - p_3^m) + \mathcal{P}_3^{m+1} \end{array} \right. , \quad (\text{A.10})$$

ou então

$$\left\{ \begin{array}{l} \frac{2E}{h^2} u_1^{m+1} - \frac{2E}{h^2} u_2^{m+1} + \frac{1}{h} p_2^{m+1} = \mathcal{U}_1^{m+1} \\ -\frac{E}{(h)^2} u_1^{m+1} + \frac{2E}{(h)^2} u_2^{m+1} + \frac{1}{2h} p_3^{m+1} = \mathcal{U}_2^{m+1} \\ -\frac{1}{2\tau h} u_1^{m+1} + \left(\frac{h^2 + 4E\tau k}{2E\tau h^2} \right) p_2^{m+1} - \left(\frac{h^2 + 4E\tau k}{4E\tau h^2} \right) p_3^{m+1} = \\ \quad - \frac{1}{2\tau h} u_1^m - \frac{1}{4E\tau} (p_3^m - 2p_2^m) + \mathcal{P}_2^{m+1} \\ -\frac{1}{\tau h} u_2^{m+1} - \left(\frac{h^2 + 4E\tau k}{2E\tau h^2} \right) p_2^{m+1} + \left(\frac{h^2 + 4E\tau k}{2E\tau h^2} \right) p_3^{m+1} = \\ \quad - \frac{1}{\tau h} u_2^m - \frac{1}{2E\tau} (p_2^m - p_3^m) + \mathcal{P}_3^{m+1} \end{array} \right. . \quad (\text{A.11})$$

Em forma matricial, pode-se exprimir como

$$\begin{pmatrix} \frac{2E}{h^2} & -\frac{2E}{h^2} & \frac{1}{h} & 0 \\ -\frac{E}{h^2} & \frac{2E}{h^2} & 0 & \frac{1}{2h} \\ -\frac{1}{2\tau h} & 0 & \frac{h^2 + 4E\tau k}{2E\tau h^2} & -\frac{h^2 + 4E\tau k}{4E\tau h^2} \\ 0 & -\frac{1}{\tau h} & -\frac{h^2 + 4E\tau k}{2E\tau h^2} & \frac{h^2 + 4E\tau k}{2E\tau h^2} \end{pmatrix} \begin{pmatrix} u_1^{m+1} \\ u_2^{m+1} \\ p_2^{m+1} \\ p_3^{m+1} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{1}{2\tau h} & 0 & \frac{1}{2E\tau} & -\frac{1}{4E\tau} \\ 0 & -\frac{1}{\tau h} & -\frac{1}{2E\tau} & \frac{1}{2E\tau} \end{pmatrix} \begin{pmatrix} u_1^m \\ u_2^m \\ p_2^m \\ p_3^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_1^{m+1} \\ \mathcal{U}_2^{m+1} \\ \mathcal{P}_2^{m+1} \\ \mathcal{P}_3^{m+1} \end{pmatrix} . \quad (\text{A.12})$$

Método de Crank-Nicolson

Para o uso do método de CN na aproximação temporal, verificou-se na seção 6.1.2 que basta substituir a equação referente à pressão nos respectivos sistemas de equações descritos usando o método de Euler. Para a implementação do suavizador Vanka de 3 pontos, consideram-se as equações da pressão no sistema descrito na Eq. (6.40) para o uso do método de Euler e a Eq. (6.55) descrita para o uso do método CN. As respectivas equações são

$$\begin{aligned} \left(\frac{h^2 + 4E\tau K}{2E\tau h^2}\right) p_i^{m+1} = & \frac{1}{2\tau h} (u_{i-1}^{m+1} - u_{i+1}^{m+1}) + \left(\frac{h^2 + 4E\tau K}{4E\tau h^2}\right) (p_{i-1}^{m+1} + p_{i+1}^{m+1}) + \\ & -\frac{1}{2\tau h} (u_{i-1}^m - u_{i+1}^m) - \frac{1}{4E\tau} (p_{i-1}^m - 2p_i^m + p_{i+1}^m) + \mathcal{P}_i^{m+1} \end{aligned}$$

e

$$\begin{aligned} \left(\frac{h^2 + 2E\tau k}{2E\tau h^2}\right) p_i^{m+1} = & \frac{1}{2\tau h} (u_{i-1}^{m+1} - u_{i+1}^{m+1}) + \left(\frac{h^2 + 2E\tau k}{4E\tau h^2}\right) (p_{i+1}^{m+1} + p_{i-1}^{m+1}) - \\ & -\frac{1}{2\tau h} (u_{i-1}^m - u_{i+1}^m) + \left(\frac{2E\tau k - h^2}{4E\tau h^2}\right) (p_{i-1}^m - 2p_i^m + p_{i+1}^m) + \frac{\mathcal{P}_i^{m+1} + \mathcal{P}_i^m}{2}. \end{aligned}$$

Com estas equações pode-se observar que para transformar as aproximações feitas pelo método de Euler em aproximações com o método de CN, basta fazer a seguinte substituição dos coeficientes

$$\frac{h^2 + 4E\tau K}{2E\tau h^2} \leftarrow \frac{h^2 + 2E\tau k}{2E\tau h^2} \quad (\text{A.13})$$

$$\frac{h^2 + 4E\tau K}{4E\tau h^2} \leftarrow \frac{h^2 + 2E\tau k}{4E\tau h^2} \quad (\text{A.14})$$

$$\frac{1}{4E\tau} \leftarrow \frac{2E\tau k - h^2}{4E\tau h^2} \quad (\text{A.15})$$

$$\mathcal{P}_i^{m+1} \leftarrow \frac{\mathcal{P}_i^{m+1} + \mathcal{P}_i^m}{2} \quad (\text{A.16})$$

A solução direta na malha mais grossa possível (apenas 3 pontos espaciais) com o método de CN é efetuada resolvendo o sistema de Eq. (A.12) e fazendo as mesmas substituições descritas nas Eq. (A.13) – (A.16) em cada um dos pontos $i = 1, 2, 3$.

Caso bidimensional

Método de Euler

Considerando-se o método de Euler e para facilitar a escrita dos sistemas de equações da poroelasticidade no caso 2D descrito nas Eqs. (6.59), são realizadas as seguintes substituições de variáveis

$$\begin{aligned}
c_1 &= \frac{2(\lambda + 3\mu)}{h^2} \\
c_2 &= \frac{\lambda + 2\mu}{h^2} \\
c_3 &= \frac{\mu}{h^2} \\
c_4 &= \frac{\lambda + \mu}{4h^2} \\
c_5 &= \frac{1}{2h} \\
c_6 &= \frac{h^2 + 4k(\lambda + 2\mu)\tau}{h^2(\lambda + 2\mu)\tau} \\
c_7 &= \frac{1}{2h\tau} \\
c_8 &= \frac{h^2 + 4k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \\
c_9 &= \frac{1}{4(\lambda + 2\mu)\tau}
\end{aligned} \tag{A.17}$$

Considerando-se então o sistema de Eqs. (6.59), as substituições de variáveis descritas nas Eqs. (A.17), o suavizador Vanka de 5 pontos (GASPAR et al., 2007; RODRIGO, 2010) e calculando-se simultaneamente os valores dos deslocamentos nas posições $u_{i-1,j}^{m+1}$, $u_{i+1,j}^{m+1}$, $v_{i,j-1}^{m+1}$, $v_{i,j+1}^{m+1}$ e da pressão na posição $p_{i,j}^{m+1}$, têm-se para os pontos internos $i, j = 3, 4, \dots, N - 2$

$$\left\{ \begin{array}{l}
 c_1 u_{i-1,j}^{m+1} = c_2 \left(u_{i-2,j}^{m+1} + u_{i,j}^{m+1} \right) + c_3 \left(u_{i-1,j-1}^{m+1} + u_{i-1,j+1}^{m+1} \right) + c_4 \left(v_{i,j+1}^{m+1} \right. \\
 \quad \left. - v_{i,j-1}^{m+1} - v_{i-2,j+1}^{m+1} + v_{i-2,j-1}^{m+1} \right) + c_5 \left(p_{i-2,j}^{m+1} - p_{i,j}^{m+1} \right) + \mathcal{U}_{i-1,j}^{m+1} \\
 c_1 u_{i+1,j}^{m+1} = c_2 \left(u_{i,j}^{m+1} + u_{i+2,j}^{m+1} \right) + c_3 \left(u_{i+1,j-1}^{m+1} + u_{i+1,j+1}^{m+1} \right) + c_4 \left(v_{i+2,j+1}^{m+1} \right. \\
 \quad \left. - v_{i+2,j-1}^{m+1} - v_{i,j+1}^{m+1} + v_{i,j-1}^{m+1} \right) + c_5 \left(p_{i,j}^{m+1} - p_{i+2,j}^{m+1} \right) + \mathcal{U}_{i+1,j}^{m+1} \\
 c_1 v_{i,j-1}^{m+1} = c_4 \left(u_{i+1,j}^{m+1} - u_{i+1,j-2}^{m+1} - u_{i-1,j}^{m+1} + u_{i-1,j-2}^{m+1} \right) + c_3 \left(v_{i-1,j-1}^{m+1} \right. \\
 \quad \left. + v_{i+1,j-1}^{m+1} \right) + c_2 \left(v_{i,j-2}^{m+1} + v_{i,j}^{m+1} \right) + c_5 \left(p_{i,j-2}^{m+1} - p_{i,j}^{m+1} \right) + \mathcal{V}_{i,j-1}^{m+1} \\
 c_1 v_{i,j+1}^{m+1} = c_4 \left(u_{i+1,j+2}^{m+1} - u_{i+1,j}^{m+1} - u_{i-1,j+2}^{m+1} + u_{i-1,j}^{m+1} \right) + c_3 \left(v_{i-1,j+1}^{m+1} \right. \\
 \quad \left. + v_{i+1,j+1}^{m+1} \right) + c_2 \left(v_{i,j}^{m+1} + v_{i,j+2}^{m+1} \right) + c_5 \left(p_{i,j}^{m+1} - p_{i,j+2}^{m+1} \right) + \mathcal{V}_{i,j+1}^{m+1} \\
 c_6 p_{i,j}^{m+1} = c_7 \left(u_{i-1,j}^{m+1} - u_{i+1,j}^{m+1} + v_{i,j-1}^{m+1} - v_{i,j+1}^{m+1} \right) + c_8 \left(p_{i-1,j}^{m+1} + p_{i+1,j}^{m+1} \right. \\
 \quad \left. + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1} \right) + c_7 \left(u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m \right) \\
 \quad \left. + c_9 \left(4p_{i,j}^m - p_{i-1,j}^m - p_{i+1,j}^m - p_{i,j-1}^m - p_{i,j+1}^m \right) + \mathcal{P}_{i,j}^{m+1} \right.
 \end{array} \right. \quad . \quad (\text{A.18})$$

Colocando-se todas as incógnitas $u_{i-1,j}^{m+1}$, $u_{i+1,j}^{m+1}$, $v_{i,j-1}^{m+1}$, $v_{i,j+1}^{m+1}$ e $p_{i,j}^{m+1}$ nos primeiros

membros das equações, obtém-se

$$\left\{ \begin{array}{l}
 c_1 u_{i-1,j}^{m+1} + c_4 v_{i,j-1}^{m+1} - c_4 v_{i,j+1}^{m+1} + c_5 p_{i,j}^{m+1} = c_2 \left(u_{i-2,j}^{m+1} + u_{i,j}^{m+1} \right) \\
 + c_3 \left(u_{i-1,j-1}^{m+1} + u_{i-1,j+1}^{m+1} \right) + c_4 \left(v_{i-2,j-1}^{m+1} - v_{i-2,j+1}^{m+1} \right) + c_5 p_{i-2,j}^{m+1} + \mathcal{U}_{i-1,j}^{m+1} \\
 c_1 u_{i+1,j}^{m+1} - c_4 v_{i,j-1}^{m+1} + c_4 v_{i,j+1}^{m+1} - c_5 p_{i,j}^{m+1} = c_2 \left(u_{i,j}^{m+1} + u_{i+2,j}^{m+1} \right) \\
 + c_3 \left(u_{i+1,j-1}^{m+1} + u_{i+1,j+1}^{m+1} \right) + c_4 \left(v_{i+2,j+1}^{m+1} - v_{i+2,j-1}^{m+1} \right) - c_5 p_{i+2,j}^{m+1} + \mathcal{U}_{i+1,j}^{m+1} \\
 c_4 u_{i-1,j}^{m+1} - c_4 u_{i+1,j}^{m+1} + c_1 v_{i,j-1}^{m+1} + c_5 p_{i,j}^{m+1} = c_4 \left(u_{i-1,j-2}^{m+1} - u_{i+1,j-2}^{m+1} \right) \\
 + c_3 \left(v_{i-1,j-1}^{m+1} + v_{i+1,j-1}^{m+1} \right) + c_2 \left(v_{i,j-2}^{m+1} + v_{i,j}^{m+1} \right) + c_5 p_{i,j-2}^{m+1} + \mathcal{V}_{i,j-1}^{m+1} \quad \cdot \quad (\text{A.19}) \\
 -c_4 u_{i-1,j}^{m+1} + c_4 u_{i+1,j}^{m+1} + c_1 v_{i,j+1}^{m+1} - c_5 p_{i,j}^{m+1} = c_4 \left(u_{i+1,j+2}^{m+1} - u_{i-1,j+2}^{m+1} \right) \\
 + c_3 \left(v_{i-1,j+1}^{m+1} + v_{i+1,j+1}^{m+1} \right) + c_2 \left(v_{i,j}^{m+1} + v_{i,j+2}^{m+1} \right) - c_5 p_{i,j+2}^{m+1} + \mathcal{V}_{i,j+1}^{m+1} \\
 -c_7 u_{i-1,j}^{m+1} + c_7 u_{i+1,j}^{m+1} - c_7 v_{i,j-1}^{m+1} + c_7 v_{i,j+1}^{m+1} + c_6 p_{i,j}^{m+1} = \\
 c_8 \left(p_{i-1,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1} \right) + c_7 \left(u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m \right) \\
 + c_9 \left(4p_{i,j}^m - p_{i-1,j}^m - p_{i+1,j}^m - p_{i,j-1}^m - p_{i,j+1}^m \right) + \mathcal{P}_{i,j}^{m+1}
 \end{array} \right.$$

Em forma matricial, para os pontos internos com $i, j = 3, 4, \dots, N-2$, a Eq. (A.19)

pode ser escrita da seguinte maneira

$$\begin{pmatrix} c_1 & 0 & c_4 & -c_4 & c_5 \\ 0 & c_1 & -c_4 & c_4 & -c_5 \\ c_4 & -c_4 & c_1 & 0 & c_5 \\ -c_4 & c_4 & 0 & c_1 & -c_5 \\ -c_7 & c_7 & -c_7 & c_7 & c_6 \end{pmatrix} \begin{pmatrix} u_{i-1,j}^{m+1} \\ u_{i+1,j}^{m+1} \\ v_{i,j-1}^{m+1} \\ v_{i,j+1}^{m+1} \\ p_{i,j}^{m+1} \end{pmatrix} = \begin{pmatrix} c_2 & c_3 & c_4 & c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_2 & c_3 & c_4 & -c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & c_5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & -c_5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_8 & c_7 & c_9 \end{pmatrix} \begin{pmatrix} u_{i-2,j}^{m+1} + u_{i,j}^{m+1} \\ u_{i-1,j-1}^{m+1} + u_{i-1,j+1}^{m+1} \\ v_{i-2,j-1}^{m+1} - v_{i-2,j+1}^{m+1} \\ p_{i-2,j}^{m+1} \\ u_{i,j}^{m+1} + u_{i+2,j}^{m+1} \\ u_{i+1,j-1}^{m+1} + u_{i+1,j+1}^{m+1} \\ v_{i+2,j+1}^{m+1} - v_{i+2,j-1}^{m+1} \\ p_{i+2,j}^{m+1} \\ u_{i-1,j-2}^{m+1} - u_{i+1,j-2}^{m+1} \\ v_{i-1,j-1}^{m+1} + v_{i+1,j-1}^{m+1} \\ v_{i,j-2}^{m+1} + v_{i,j}^{m+1} \\ p_{i,j-2}^{m+1} \\ u_{i+1,j+2}^{m+1} - u_{i-1,j+2}^{m+1} \\ v_{i-1,j+1}^{m+1} + v_{i+1,j+1}^{m+1} \\ v_{i,j}^{m+1} + v_{i,j+2}^{m+1} \\ p_{i,j+2}^{m+1} \\ p_{i-1,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1} \\ u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m \\ 4p_{i,j}^m - p_{i-1,j}^m - p_{i+1,j}^m - p_{i,j-1}^m - p_{i,j+1}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{i-1,j}^{m+1} \\ \mathcal{U}_{i+1,j}^{m+1} \\ \mathcal{V}_{i,j-1}^{m+1} \\ \mathcal{V}_{i,j+1}^{m+1} \\ \mathcal{P}_{i,j}^{m+1} \end{pmatrix}. \quad (\text{A.20})$$

Para os contornos, considerando as condições de contorno do tipo Dirichlet, têm-se adicionalmente 8 sistemas de equações diferentes que são calculados nos pontos localizados nas posições Sudoeste, Sul, Sudeste, Oeste, Leste, Noroeste, Norte e Nordeste, como mostra a Fig. 96. Para isso, consideram-se o sistema descrito na Eq. (A.19) e os respectivos valores nulos das incógnitas localizadas nos contornos.

Para as posições Sudoeste, com $i = j = 2$, as incógnitas são: $u_{3,2}^{m+1}$, $v_{2,3}^{m+1}$ e $p_{2,2}^{m+1}$.

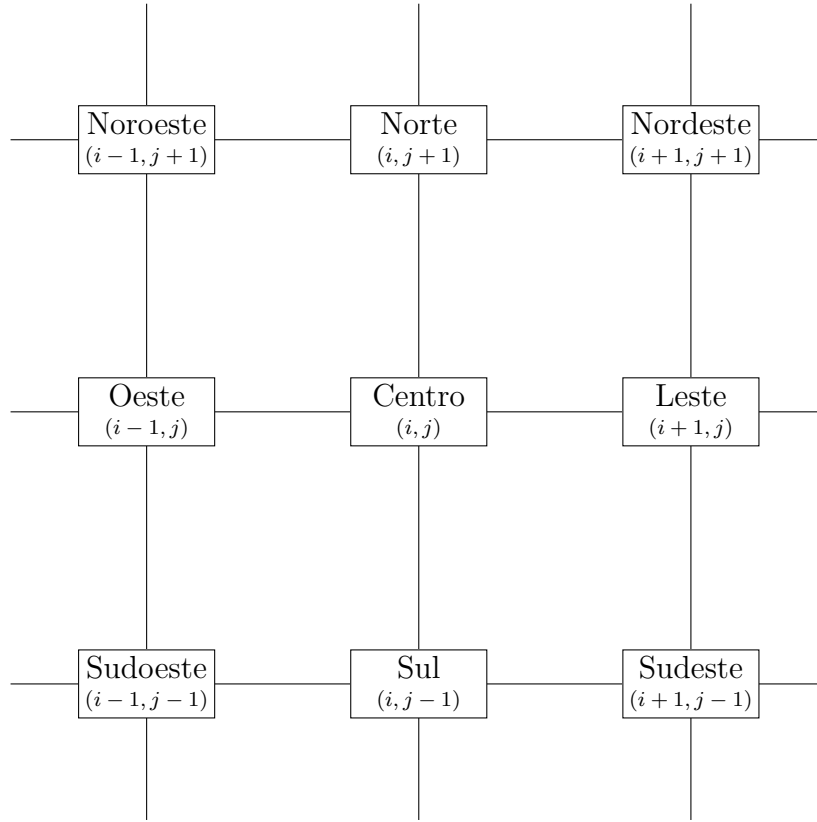


Figura 96 – Posições dos pontos em um sistema de coordenadas cartesianas e em relação a um ponto central.

Assim, tem-se

$$\left\{ \begin{array}{l}
 c_1 u_{3,2}^{m+1} + c_4 v_{2,3}^{m+1} - c_5 p_{2,2}^{m+1} = c_2 (u_{2,2}^{m+1} + u_{4,2}^{m+1}) \\
 \quad + c_3 u_{3,3}^{m+1} + c_4 v_{4,3}^{m+1} - c_5 p_{4,2}^{m+1} + \mathcal{U}_{3,2}^{m+1} \\
 c_4 u_{3,2}^{m+1} + c_1 v_{2,3}^{m+1} - c_5 p_{2,2}^{m+1} = c_4 (u_{3,4}^{m+1}) \\
 \quad + c_3 (v_{3,3}^{m+1}) + c_2 (v_{2,2}^{m+1} + v_{2,4}^{m+1}) - c_5 p_{2,4}^{m+1} + \mathcal{V}_{2,3}^{m+1} \\
 c_7 u_{3,2}^{m+1} + c_7 v_{2,3}^{m+1} + c_6 p_{2,2}^{m+1} = c_8 (p_{3,2}^{m+1} + p_{2,3}^{m+1}) \\
 \quad + c_7 (u_{3,2}^m + v_{2,3}^m) + c_9 (4p_{2,2}^m - p_{3,2}^m - p_{2,3}^m) + \mathcal{P}_{2,2}^{m+1}
 \end{array} \right. \quad . \quad (A.21)$$

Em forma matricial a Eq. (A.21) pode ser escrita como

$$\begin{pmatrix} c_1 & c_4 & -c_5 \\ c_4 & c_1 & -c_5 \\ c_7 & c_7 & c_6 \end{pmatrix} \begin{pmatrix} u_{3,2}^{m+1} \\ v_{2,3}^{m+1} \\ p_{2,2}^{m+1} \end{pmatrix} = \begin{pmatrix} c_2 & c_3 & c_4 & -c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & -c_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_8 & c_7 & c_9 \end{pmatrix} \begin{pmatrix} u_{2,2}^{m+1} + u_{4,2}^{m+1} \\ u_{3,3}^{m+1} \\ v_{4,3}^{m+1} \\ p_{4,2}^{m+1} \\ u_{3,4}^{m+1} \\ v_{3,3}^{m+1} \\ v_{2,2}^{m+1} + v_{2,4}^{m+1} \\ p_{2,4}^{m+1} \\ p_{3,2}^{m+1} + p_{2,3}^{m+1} \\ u_{3,2}^m + v_{2,3}^m \\ 4p_{2,2}^m - p_{3,2}^m - p_{2,3}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{3,2}^{m+1} \\ \mathcal{V}_{2,3}^{m+1} \\ \mathcal{P}_{2,2}^{m+1} \end{pmatrix}. \tag{A.22}$$

Considerando-se as posições Sul, em que $j = 2$ e $i = 3, 4, \dots, N - 2$, têm-se as incógnitas: $u_{i-1,2}^{m+1}$, $u_{i+1,2}^{m+1}$, $v_{i,3}^{m+1}$ e $p_{i,2}^{m+1}$. Assim, a partir da Eq. (A.19) tem-se para

$i = 3, 4, \dots, N - 2$

$$\left\{ \begin{array}{l}
 c_1 u_{i-1,2}^{m+1} - c_4 v_{i,3}^{m+1} + c_5 p_{i,2}^{m+1} = c_2 \left(u_{i-2,2}^{m+1} + u_{i,2}^{m+1} \right) \\
 \quad + c_3 u_{i-1,3}^{m+1} + c_4 \left(-v_{i-2,3}^{m+1} \right) + c_5 p_{i-2,2}^{m+1} + \mathcal{U}_{i-1,2}^{m+1} \\
 c_1 u_{i+1,2}^{m+1} + c_4 v_{i,3}^{m+1} - c_5 p_{i,2}^{m+1} = c_2 \left(u_{i,2}^{m+1} + u_{i+2,2}^{m+1} \right) \\
 \quad + c_3 u_{i+1,3}^{m+1} + c_4 v_{i+2,3}^{m+1} - c_5 p_{i+2,2}^{m+1} + \mathcal{U}_{i+1,2}^{m+1} \\
 -c_4 u_{i-1,2}^{m+1} + c_4 u_{i+1,2}^{m+1} + c_1 v_{i,3}^{m+1} - c_5 p_{i,2}^{m+1} = c_4 \left(u_{i+1,4}^{m+1} - u_{i-1,4}^{m+1} \right) \\
 \quad + c_3 \left(v_{i-1,3}^{m+1} + v_{i+1,3}^{m+1} \right) + c_2 \left(v_{i,2}^{m+1} + v_{i,4}^{m+1} \right) - c_5 p_{i,4}^{m+1} + \mathcal{V}_{i,3}^{m+1} \\
 -c_7 u_{i-1,2}^{m+1} + c_7 u_{i+1,2}^{m+1} + c_7 v_{i,3}^{m+1} + c_6 p_{i,2}^{m+1} = \\
 \quad c_8 \left(p_{i-1,2}^{m+1} + p_{i+1,2}^{m+1} + p_{i,3}^{m+1} \right) + c_7 \left(u_{i+1,2}^m - u_{i-1,2}^m + v_{i,3}^m \right) \\
 \quad + c_9 \left(4p_{i,2}^m - p_{i-1,2}^m - p_{i+1,2}^m - p_{i,3}^m \right) + \mathcal{P}_{i,2}^{m+1}
 \end{array} \right. \quad (\text{A.23})$$

Em forma matricial, a Eq. (A.23) ($i = 3, 4, \dots, N - 2$) pode ser escrita como

$$\begin{pmatrix} c_2 & c_3 & c_4 & c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_2 & c_3 & c_4 & -c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & -c_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_8 & c_7 & c_9 \end{pmatrix} \begin{pmatrix} u_{i-2,2}^{m+1} + u_{i,2}^{m+1} \\ u_{i-1,3}^{m+1} \\ -v_{i-2,3}^{m+1} \\ p_{i-2,2}^{m+1} \\ u_{i,2}^{m+1} + u_{i+2,2}^{m+1} \\ u_{i+1,3}^{m+1} \\ v_{i+2,3}^{m+1} \\ p_{i+2,2}^{m+1} \\ u_{i+1,4}^{m+1} - u_{i-1,4}^{m+1} \\ v_{i-1,3}^{m+1} + v_{i+1,3}^{m+1} \\ v_{i,2}^{m+1} + v_{i,4}^{m+1} \\ p_{i,4}^{m+1} \\ p_{i-1,2}^{m+1} + p_{i+1,2}^{m+1} + p_{i,3}^{m+1} \\ u_{i+1,2}^m - u_{i-1,2}^m + v_{i,3}^m \\ 4p_{i,2}^m - p_{i-1,2}^m - p_{i+1,2}^m - p_{i,3}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{i-1,2}^{m+1} \\ \mathcal{U}_{i+1,2}^{m+1} \\ \mathcal{V}_{i,3}^{m+1} \\ \mathcal{P}_{i,2}^{m+1} \end{pmatrix}. \quad (\text{A.24})$$

Para o contorno localizado nas posições Sudeste, com $j = 2$ e $i = N - 1$, as incógnitas são: $u_{N-2,2}^{m+1}$, $v_{N-1,3}^{m+1}$ e $p_{N-1,2}^{m+1}$. Assim, pode-se escrever o sistema de equações

como

$$\left\{ \begin{array}{l}
 c_1 u_{N-2,2}^{m+1} - c_4 v_{N-1,3}^{m+1} + c_5 p_{N-1,2}^{m+1} = c_2 \left(u_{N-3,2}^{m+1} + u_{N-1,2}^{m+1} \right) \\
 \quad + c_3 u_{N-2,3}^{m+1} + c_4 \left(-v_{N-3,3}^{m+1} \right) + c_5 p_{N-3,2}^{m+1} + \mathcal{U}_{N-2,2}^{m+1} \\
 -c_4 u_{N-2,2}^{m+1} + c_1 v_{N-1,3}^{m+1} - c_5 p_{N-1,2}^{m+1} = c_4 \left(-u_{N-2,4}^{m+1} \right) \\
 \quad + c_3 v_{N-2,3}^{m+1} + c_2 \left(v_{N-1,2}^{m+1} + v_{N-1,4}^{m+1} \right) - c_5 p_{N-1,4}^{m+1} + \mathcal{V}_{N-1,3}^{m+1} \\
 -c_7 u_{N-2,2}^{m+1} + c_7 v_{N-1,3}^{m+1} + c_6 p_{N-1,2}^{m+1} = \\
 \quad c_8 \left(p_{N-2,2}^{m+1} + p_{N-1,3}^{m+1} \right) + c_7 \left(-u_{N-2,2}^m + v_{N-1,3}^m \right) \\
 \quad + c_9 \left(4p_{N-1,2}^m - p_{N-2,2}^m - p_{N-1,3}^m \right) + \mathcal{P}_{N-1,2}^{m+1}
 \end{array} \right. \quad (A.25)$$

Em forma matricial a Eq. (A.25) torna-se

$$\begin{pmatrix} c_1 & -c_4 & c_5 \\ -c_4 & c_1 & -c_5 \\ -c_7 & c_7 & c_6 \end{pmatrix} \begin{pmatrix} u_{N-2,2}^{m+1} \\ v_{N-1,3}^{m+1} \\ p_{N-1,2}^{m+1} \end{pmatrix} = \\
 \begin{pmatrix} c_2 & c_3 & c_4 & c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & -c_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_8 & c_7 & c_9 \end{pmatrix} \\
 \begin{pmatrix} u_{N-3,2}^{m+1} + u_{N-1,2}^{m+1} \\ u_{N-2,3}^{m+1} \\ -v_{N-3,3}^{m+1} \\ p_{N-3,2}^{m+1} \\ -u_{N-2,4}^{m+1} \\ v_{N-2,3}^{m+1} \\ v_{N-1,2}^{m+1} + v_{N-1,4}^{m+1} \\ p_{N-1,4}^{m+1} \\ p_{N-2,2}^{m+1} + p_{N-1,3}^{m+1} \\ -u_{N-2,2}^m + v_{N-1,3}^m \\ 4p_{N-1,2}^m - p_{N-2,2}^m - p_{N-1,3}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{N-2,2}^{m+1} \\ \mathcal{V}_{N-1,3}^{m+1} \\ \mathcal{P}_{N-1,2}^{m+1} \end{pmatrix}. \quad (A.26)$$

Nas posições do contorno Oeste, em que $i = 2$ e $j = 3, 4, \dots, N - 2$, têm-se as incógnitas: $u_{3,j}^{m+1}$, $v_{2,j-1}^{m+1}$, $v_{2,j+1}^{m+1}$ e $p_{2,j}^{m+1}$. O respectivo sistema de equações (com $j =$

$3, 4, \dots, N - 2$) é descrito da seguinte maneira

$$\left\{ \begin{array}{l}
 c_1 u_{3,j}^{m+1} - c_4 v_{2,j-1}^{m+1} + c_4 v_{2,j+1}^{m+1} - c_5 p_{2,j}^{m+1} = c_2 (u_{2,j}^{m+1} + u_{4,j}^{m+1}) \\
 + c_3 (u_{3,j-1}^{m+1} + u_{3,j+1}^{m+1}) + c_4 (v_{4,j+1}^{m+1} - v_{4,j-1}^{m+1}) - c_5 p_{4,j}^{m+1} + \mathcal{U}_{3,j}^{m+1} \\
 - c_4 u_{3,j}^{m+1} + c_1 v_{2,j-1}^{m+1} + c_5 p_{2,j}^{m+1} = c_4 (-u_{3,j-2}^{m+1}) \\
 + c_3 (v_{3,j-1}^{m+1}) + c_2 (v_{2,j-2}^{m+1} + v_{2,j}^{m+1}) + c_5 p_{2,j-2}^{m+1} + \mathcal{V}_{2,j-1}^{m+1} \\
 c_4 u_{3,j}^{m+1} + c_1 v_{2,j+1}^{m+1} - c_5 p_{2,j}^{m+1} = c_4 (u_{3,j+2}^{m+1}) \\
 + c_3 v_{3,j+1}^{m+1} + c_2 (v_{2,j}^{m+1} + v_{2,j+2}^{m+1}) - c_5 p_{2,j+2}^{m+1} + \mathcal{V}_{2,j+1}^{m+1} \\
 c_7 u_{3,j}^{m+1} - c_7 v_{2,j-1}^{m+1} + c_7 v_{2,j+1}^{m+1} + c_6 p_{2,j}^{m+1} = \\
 c_8 (p_{3,j}^{m+1} + p_{2,j-1}^{m+1} + p_{2,j+1}^{m+1}) + c_7 (u_{3,j}^m + v_{2,j+1}^m - v_{2,j-1}^m) \\
 + c_9 (4p_{2,j}^m - p_{3,j}^m - p_{2,j-1}^m - p_{2,j+1}^m) + \mathcal{P}_{2,j}^{m+1}
 \end{array} \right. \quad . \quad (\text{A.27})$$

Em forma matricial, a Eq. (A.27) com $j = 3, 4, \dots, N - 2$ é escrita da seguinte

maneira

$$\begin{pmatrix} c_1 & -c_4 & c_4 & -c_5 \\ -c_4 & c_1 & 0 & c_5 \\ c_4 & 0 & c_1 & -c_5 \\ c_7 & -c_7 & c_7 & c_6 \end{pmatrix} \begin{pmatrix} u_{3,j}^{m+1} \\ v_{2,j-1}^{m+1} \\ v_{2,j+1}^{m+1} \\ p_{2,j}^{m+1} \end{pmatrix} = \begin{pmatrix} c_2 & c_3 & c_4 & -c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & -c_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_8 & c_7 & c_9 \end{pmatrix} \begin{pmatrix} u_{2,j}^{m+1} + u_{4,j}^{m+1} \\ u_{3,j-1}^{m+1} + u_{3,j+1}^{m+1} \\ v_{4,j+1}^{m+1} - v_{4,j-1}^{m+1} \\ p_{4,j}^{m+1} \\ -u_{3,j-2}^{m+1} \\ v_{3,j-1}^{m+1} \\ v_{2,j-2}^{m+1} + v_{2,j}^{m+1} \\ p_{2,j-2}^{m+1} \\ u_{3,j+2}^{m+1} \\ v_{3,j+1}^{m+1} \\ v_{2,j}^{m+1} + v_{2,j+2}^{m+1} \\ p_{2,j+2}^{m+1} \\ p_{3,j}^{m+1} + p_{2,j-1}^{m+1} + p_{2,j+1}^{m+1} \\ u_{3,j}^m + v_{2,j+1}^m - v_{2,j-1}^m \\ 4p_{2,j}^m - p_{3,j}^m - p_{2,j-1}^m - p_{2,j+1}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{3,j}^{m+1} \\ \mathcal{V}_{2,j-1}^{m+1} \\ \mathcal{V}_{2,j+1}^{m+1} \\ \mathcal{P}_{2,j}^{m+1} \end{pmatrix}. \quad (\text{A.28})$$

Nas posições do contorno Leste, em que $i = N - 1$ e $j = 3, 4, \dots, N - 2$, têm-se as incógnitas: $u_{N-2,j}^{m+1}$, $v_{N-1,j-1}^{m+1}$, $v_{N-1,j+1}^{m+1}$ e $p_{N-1,j}^{m+1}$. O respectivo sistema de equações (com

$j = 3, 4, \dots, N - 2$) é descrito da seguinte maneira

$$\left\{ \begin{array}{l}
 c_1 u_{N-1,j}^{m+1} + c_4 v_{N-1,j-1}^{m+1} - c_4 v_{N-1,j+1}^{m+1} + c_5 p_{N-1,j}^{m+1} = c_2 \left(u_{N-3,j}^{m+1} + u_{N-1,j}^{m+1} \right) \\
 + c_3 \left(u_{N-2,j-1}^{m+1} + u_{N-2,j+1}^{m+1} \right) + c_4 \left(v_{N-3,j-1}^{m+1} - v_{N-3,j+1}^{m+1} \right) + c_5 p_{N-3,j}^{m+1} + \mathcal{U}_{N-2,j}^{m+1} \\
 \\
 c_4 u_{N-2,j}^{m+1} + c_1 v_{N-1,j-1}^{m+1} + c_5 p_{N-1,j}^{m+1} = c_4 \left(u_{N-2,j-2}^{m+1} \right) \\
 + c_3 \left(v_{N-2,j-1}^{m+1} \right) + c_2 \left(v_{N-1,j-2}^{m+1} + v_{N-1,j}^{m+1} \right) + c_5 p_{N-1,j-2}^{m+1} + \mathcal{V}_{N-1,j-1}^{m+1} \\
 \\
 -c_4 u_{N-2,j}^{m+1} + c_1 v_{N-1,j+1}^{m+1} - c_5 p_{N-1,j}^{m+1} = c_4 \left(-u_{N-2,j+2}^{m+1} \right) \\
 + c_3 \left(v_{N-2,j+1}^{m+1} \right) + c_2 \left(v_{N-1,j}^{m+1} + v_{N-1,j+2}^{m+1} \right) - c_5 p_{N-1,j+2}^{m+1} + \mathcal{V}_{N-1,j+1}^{m+1} \\
 \\
 -c_7 u_{N-2,j}^{m+1} - c_7 v_{N-1,j-1}^{m+1} + c_7 v_{N-1,j+1}^{m+1} + c_6 p_{N-1,j}^{m+1} = \\
 c_8 \left(p_{N-2,j}^{m+1} + p_{N-1,j-1}^{m+1} + p_{N-1,j+1}^{m+1} \right) + c_7 \left(-u_{N-2,j}^m + v_{N-1,j+1}^m - v_{N-1,j-1}^m \right) \\
 + c_9 \left(4p_{N-1,j}^m - p_{N-2,j}^m - p_{N-1,j-1}^m - p_{N-1,j+1}^m \right) + \mathcal{P}_{N-1,j}^{m+1}
 \end{array} \right. \quad . \quad (\text{A.29})$$

Em forma matricial a Eq. (A.29) com $j = 3, 4, \dots, N - 2$ é descrita como

$$\begin{pmatrix} c_1 & c_4 & -c_4 & c_5 \\ c_4 & c_1 & 0 & c_5 \\ -c_4 & 0 & c_1 & -c_5 \\ -c_7 & -c_7 & c_7 & c_6 \end{pmatrix} \begin{pmatrix} u_{i-1,j}^{m+1} \\ v_{i,j-1}^{m+1} \\ v_{i,j+1}^{m+1} \\ p_{i,j}^{m+1} \end{pmatrix} = \begin{pmatrix} c_2 & c_3 & c_4 & c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & -c_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_8 & c_7 & c_9 \end{pmatrix} \begin{pmatrix} u_{N-3,j}^{m+1} + u_{N-1,j}^{m+1} \\ u_{N-2,j-1}^{m+1} + u_{N-2,j+1}^{m+1} \\ v_{N-3,j-1}^{m+1} - v_{N-3,j+1}^{m+1} \\ p_{N-3,j}^{m+1} \\ u_{N-2,j-2}^{m+1} \\ v_{N-2,j-1}^{m+1} \\ v_{N-1,j-2}^{m+1} + v_{N-1,j}^{m+1} \\ p_{N-1,j-2}^{m+1} \\ -u_{N-2,j+2}^{m+1} \\ v_{N-2,j+1}^{m+1} \\ v_{N-1,j}^{m+1} + v_{N-1,j+2}^{m+1} \\ p_{N-1,j+2}^{m+1} \\ p_{N-2,j}^{m+1} + p_{N-1,j-1}^{m+1} + p_{N-1,j+1}^{m+1} \\ -u_{N-2,j}^m + v_{N-1,j+1}^m - v_{N-1,j-1}^m \\ 4p_{N-1,j}^m - p_{N-2,j}^m - p_{N-1,j-1}^m - p_{N-1,j+1}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{N-2,j}^{m+1} \\ \mathcal{V}_{N-1,j-1}^{m+1} \\ \mathcal{V}_{N-1,j+1}^{m+1} \\ \mathcal{P}_{N-1,j}^{m+1} \end{pmatrix}. \quad (\text{A.30})$$

Considerando-se o ponto localizado nos contornos Noroeste, ou seja, $i = 2$ e $j = N - 1$, as incógnitas são: $u_{3,N-1}^{m+1}$, $v_{2,N-1}^{m+1}$ e $p_{2,N-1}^{m+1}$. Nestas posições, o sistema de

equações fica

$$\left\{ \begin{array}{l}
 c_1 u_{3,N-1}^{m+1} - c_4 v_{2,N-2}^{m+1} - c_5 p_{2,N-1}^{m+1} = c_2 \left(u_{2,N-1}^{m+1} + u_{4,N-1}^{m+1} \right) \\
 \quad + c_3 \left(u_{3,N-2}^{m+1} \right) + c_4 \left(-v_{4,N-2}^{m+1} \right) - c_5 p_{4,N-1}^{m+1} + \mathcal{U}_{3,N-1}^{m+1} \\
 -c_4 u_{3,N-1}^{m+1} + c_1 v_{2,N-2}^{m+1} + c_5 p_{2,N-1}^{m+1} = c_4 \left(-u_{3,N-3}^{m+1} \right) \\
 \quad + c_3 \left(v_{3,N-2}^{m+1} \right) + c_2 \left(v_{2,N-3}^{m+1} + v_{2,N-1}^{m+1} \right) + c_5 p_{2,N-3}^{m+1} + \mathcal{V}_{2,N-2}^{m+1} \\
 c_7 u_{3,N-1}^{m+1} - c_7 v_{2,N-2}^{m+1} + c_6 p_{2,N-1}^{m+1} = c_8 \left(p_{3,N-1}^{m+1} + p_{2,N-2}^{m+1} \right) \\
 \quad + c_7 \left(u_{3,N-1}^m - v_{2,N-2}^m \right) + c_9 \left(4p_{2,N-1}^m - p_{3,N-1}^m - p_{2,N-2}^m \right) + \mathcal{P}_{2,N-1}^{m+1}
 \end{array} \right. \quad . \quad (\text{A.31})$$

Em forma matricial, a Eq. (A.31) torna-se

$$\begin{pmatrix} c_1 & -c_4 & -c_5 \\ -c_4 & c_1 & c_5 \\ c_7 & -c_7 & c_6 \end{pmatrix} \begin{pmatrix} u_{i+1,j}^{m+1} \\ v_{i,j-1}^{m+1} \\ p_{i,j}^{m+1} \end{pmatrix} = \\
 \begin{pmatrix} c_2 & c_3 & c_4 & -c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & c_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_8 & c_7 & c_9 \end{pmatrix} \\
 \begin{pmatrix} u_{2,N-1}^{m+1} + u_{4,N-1}^{m+1} \\ u_{3,N-2}^{m+1} \\ -v_{4,N-2}^{m+1} \\ p_{4,N-1}^{m+1} \\ -u_{3,N-3}^{m+1} \\ v_{3,N-2}^{m+1} \\ v_{2,N-3}^{m+1} + v_{2,N-1}^{m+1} \\ p_{2,N-3}^{m+1} \\ p_{3,N-1}^{m+1} + p_{2,N-2}^{m+1} \\ u_{3,N-1}^m - v_{2,N-2}^m \\ 4p_{2,N-1}^m - p_{3,N-1}^m - p_{2,N-2}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{3,N-1}^{m+1} \\ \mathcal{V}_{2,N-2}^{m+1} \\ \mathcal{P}_{2,N-1}^{m+1} \end{pmatrix}. \quad (\text{A.32})$$

Nos pontos localizados nas posições Norte, em que $i = 3, 4, \dots, N-2$ e $j = N-1$ as incógnitas são: $u_{i-1,N-1}^{m+1}$, $u_{i+1,N-1}^{m+1}$, $v_{i,N-2}^{m+1}$ e $p_{i,N-1}^{m+1}$. Nesses pontos, para $i = 3, 4, \dots, N-2$

o sistema pode ser escrito como

$$\left\{ \begin{array}{l}
 c_1 u_{i-1, N-1}^{m+1} + c_4 v_{i, N-2}^{m+1} + c_5 p_{i, N-1}^{m+1} = c_2 \left(u_{i-2, N-1}^{m+1} + u_{i, N-1}^{m+1} \right) \\
 \quad + c_3 \left(u_{i-1, N-2}^{m+1} \right) + c_4 \left(v_{i-2, N-2}^{m+1} \right) + c_5 p_{i-2, N-1}^{m+1} + \mathcal{U}_{i-1, N-1}^{m+1} \\
 c_1 u_{i+1, N-1}^{m+1} - c_4 v_{i, N-2}^{m+1} - c_5 p_{i, N-1}^{m+1} = c_2 \left(u_{i, N-1}^{m+1} + u_{i+2, N-1}^{m+1} \right) \\
 \quad + c_3 \left(u_{i+1, N-2}^{m+1} \right) + c_4 \left(-v_{i+2, N-2}^{m+1} \right) - c_5 p_{i+2, N-1}^{m+1} + \mathcal{U}_{i+1, N-1}^{m+1} \\
 c_4 u_{i-1, N-1}^{m+1} - c_4 u_{i+1, N-1}^{m+1} + c_1 v_{i, N-2}^{m+1} + c_5 p_{i, N-1}^{m+1} = c_4 \left(u_{i-1, N-3}^{m+1} - u_{i+1, N-3}^{m+1} \right) \\
 \quad + c_3 \left(v_{i-1, N-2}^{m+1} + v_{i+1, N-2}^{m+1} \right) + c_2 \left(v_{i, N-3}^{m+1} + v_{i, N-1}^{m+1} \right) + c_5 p_{i, N-3}^{m+1} + \mathcal{V}_{i, N-2}^{m+1} \\
 -c_7 u_{i-1, N-1}^{m+1} + c_7 u_{i+1, N-1}^{m+1} - c_7 v_{i, N-2}^{m+1} + c_6 p_{i, N-1}^{m+1} = \\
 c_8 \left(p_{i-1, N-1}^{m+1} + p_{i+1, N-1}^{m+1} + p_{i, N-2}^{m+1} \right) + c_7 \left(u_{i+1, N-1}^m - u_{i-1, N-1}^m - v_{i, N-2}^m \right) \\
 \quad + c_9 \left(4p_{i, N-1}^m - p_{i-1, N-1}^m - p_{i+1, N-1}^m - p_{i, N-2}^m \right) + \mathcal{P}_{i, N-1}^{m+1}
 \end{array} \right. \cdot \quad (\text{A.33})$$

Em forma matricial, a Eq. (A.33) pode ser escrita em $i = 3, 4, \dots, N - 2$ como

$$\begin{pmatrix} c_1 & 0 & c_4 & c_5 \\ 0 & c_1 & -c_4 & -c_5 \\ c_4 & -c_4 & c_1 & c_5 \\ -c_7 & c_7 & -c_7 & c_6 \end{pmatrix} \begin{pmatrix} u_{i-1,N-1}^{m+1} \\ u_{i+1,N-1}^{m+1} \\ v_{i,N-2}^{m+1} \\ p_{i,N-1}^{m+1} \end{pmatrix} = \begin{pmatrix} c_2 & c_3 & c_4 & c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_2 & c_3 & c_4 & -c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & c_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_8 & c_7 & c_9 \end{pmatrix} \begin{pmatrix} u_{i-2,N-1}^{m+1} + u_{i,N-1}^{m+1} \\ u_{i-1,N-2}^{m+1} \\ v_{i-2,N-2}^{m+1} \\ p_{i-2,N-1}^{m+1} \\ u_{i,N-1}^{m+1} + u_{i+2,N-1}^{m+1} \\ u_{i+1,N-2}^{m+1} \\ -v_{i+2,N-2}^{m+1} \\ p_{i+2,N-1}^{m+1} \\ u_{i-1,N-3}^{m+1} - u_{i+1,N-3}^{m+1} \\ v_{i-1,N-2}^{m+1} + v_{i+1,N-2}^{m+1} \\ v_{i,N-3}^{m+1} + v_{i,N-1}^{m+1} \\ p_{i,N-3}^{m+1} \\ p_{i-1,N-1}^{m+1} + p_{i+1,N-1}^{m+1} + p_{i,N-2}^{m+1} \\ u_{i+1,N-1}^m - u_{i-1,N-1}^m - v_{i,N-2}^m \\ 4p_{i,N-1}^m - p_{i-1,N-1}^m - p_{i+1,N-1}^m - p_{i,N-2}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{i-1,N-1}^{m+1} \\ \mathcal{U}_{i+1,N-1}^{m+1} \\ \mathcal{V}_{i,N-2}^{m+1} \\ \mathcal{P}_{i,N-1}^{m+1} \end{pmatrix}. \quad (\text{A.34})$$

Para o último contorno a ser calculado, nas posições Nordeste, com $i = N - 1$ e $j = N - 1$ têm-se as incógnitas: $u_{N-2,N-1}^{m+1}$, $v_{N-1,N-2}^{m+1}$ e $p_{N-1,N-1}^{m+1}$. Nesses pontos, o sistema

assume a forma

$$\left\{ \begin{array}{l}
 c_1 u_{N-2,N-1}^{m+1} + c_4 v_{N-1,N-2}^{m+1} + c_5 p_{N-1,N-1}^{m+1} = c_2 \left(u_{N-3,N-1}^{m+1} + u_{N-1,N-1}^{m+1} \right) \\
 \quad + c_3 \left(u_{N-2,N-2}^{m+1} \right) + c_4 \left(v_{N-3,N-2}^{m+1} \right) + c_5 p_{N-3,N-1}^{m+1} + \mathcal{U}_{N-2,N-1}^{m+1} \\
 c_4 u_{N-2,N-1}^{m+1} + c_1 v_{N-1,N-2}^{m+1} + c_5 p_{N-1,N-1}^{m+1} = c_4 u_{N-2,N-3}^{m+1} + c_3 v_{N-2,N-2}^{m+1} \\
 \quad + c_2 \left(v_{N-1,N-3}^{m+1} + v_{N-1,N-1}^{m+1} \right) + c_5 p_{N-1,N-3}^{m+1} + \mathcal{V}_{N-1,N-2}^{m+1} \\
 -c_7 u_{N-2,N-1}^{m+1} - c_7 v_{N-1,N-2}^{m+1} + c_6 p_{N-1,N-1}^{m+1} = c_8 \left(p_{N-2,N-1}^{m+1} + p_{N-1,N-2}^{m+1} \right) \\
 \quad + c_7 \left(-u_{N-2,N-1}^m - v_{N-1,N-2}^m \right) + c_9 \left(4p_{N-1,N-1}^m - p_{N-2,N-1}^m - p_{N-1,N-2}^m \right) + \mathcal{P}_{N-1,N-1}^{m+1}
 \end{array} \right. \quad . \quad (\text{A.35})$$

A descrição matricial do sistema dado na Eq. (A.35) é

$$\begin{pmatrix} c_1 & c_4 & c_5 \\ c_4 & c_1 & c_5 \\ -c_7 & -c_7 & c_6 \end{pmatrix} \begin{pmatrix} u_{N-2,N-1}^{m+1} \\ v_{N-1,N-2}^{m+1} \\ p_{N-1,N-1}^{m+1} \end{pmatrix} = \begin{pmatrix} c_2 & c_3 & c_4 & c_5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_4 & c_3 & c_2 & c_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_8 & c_7 & c_9 \end{pmatrix} \begin{pmatrix} u_{N-3,N-1}^{m+1} + u_{N-1,N-1}^{m+1} \\ u_{N-2,N-2}^{m+1} \\ v_{N-3,N-2}^{m+1} \\ p_{N-3,N-1}^{m+1} \\ u_{N-2,N-3}^{m+1} \\ v_{N-2,N-2}^{m+1} \\ v_{N-1,N-3}^{m+1} + v_{N-1,N-1}^{m+1} \\ p_{N-1,N-3}^{m+1} \\ p_{N-2,N-1}^{m+1} + p_{N-1,N-2}^{m+1} \\ -u_{N-2,N-1}^m - v_{N-1,N-2}^m \\ 4p_{N-1,N-1}^m - p_{N-2,N-1}^m - p_{N-1,N-2}^m \end{pmatrix} + \begin{pmatrix} \mathcal{U}_{N-2,N-1}^{m+1} \\ \mathcal{V}_{N-1,N-2}^{m+1} \\ \mathcal{P}_{N-1,N-1}^{m+1} \end{pmatrix}. \quad (\text{A.36})$$

A solução direta na malha mais grossa possível, com apenas 3 pontos espaciais em cada direção (sendo que 2 destes pontos são contornos nulos) usando a aproximação temporal com o método de Euler, é efetuada resolvendo o sistema de Eq. (6.59) no ponto $i = j = 2$.

Método de Crank-Nicolson

Para o uso do método de CN na aproximação temporal, assim como para o caso unidimensional, basta substituir a equação referente à pressão no sistemas de Eq. (6.59) pela Eq. (6.60). Ou seja, substituir a equação

$$\begin{aligned} \left(\frac{h^2 + 4k(\lambda + 2\mu)\tau}{h^2(\lambda + 2\mu)\tau} \right) p_{i,j}^{m+1} &= \frac{1}{2h\tau} \left(u_{i-1,j}^{m+1} - u_{i+1,j}^{m+1} + v_{i,j-1}^{m+1} - v_{i,j+1}^{m+1} \right) + \\ &+ \left(\frac{h^2 + 4k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \right) \left(p_{i-1,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1} \right) + \\ &+ \frac{1}{2h\tau} \left(u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m \right) + \\ &+ \frac{1}{4(\lambda + 2\mu)\tau} \left(4p_{i,j}^m - p_{i-1,j}^m - p_{i+1,j}^m - p_{i,j-1}^m - p_{i,j+1}^m \right) + \mathcal{P}_{i,j}^{m+1} \end{aligned}$$

pela equação

$$\begin{aligned} \left(\frac{h^2 + 2k(\lambda + 2\mu)\tau}{h^2(\lambda + 2\mu)\tau} \right) p_{i,j}^{m+1} &= \frac{1}{2h\tau} \left(u_{i-1,j}^{m+1} - u_{i+1,j}^{m+1} + v_{i,j-1}^{m+1} - v_{i,j+1}^{m+1} \right) + \\ \left(\frac{h^2 + 2k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \right) \left(p_{i-1,j}^{m+1} + p_{i+1,j}^{m+1} + p_{i,j-1}^{m+1} + p_{i,j+1}^{m+1} \right) &+ \frac{1}{2h\tau} \left(u_{i+1,j}^m - u_{i-1,j}^m + v_{i,j+1}^m - v_{i,j-1}^m \right) \\ + \left(\frac{h^2 - 2k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \right) \left(4p_{i,j}^m - p_{i-1,j}^m - p_{i+1,j}^m - p_{i,j-1}^m - p_{i,j+1}^m \right) &+ \frac{\mathcal{P}_{i,j}^{m+1} + \mathcal{P}_{i,j}^m}{2}. \end{aligned}$$

Observando-se os coeficientes das respectivas equações acima (ou Eq. (6.59) e Eq. (6.60)), verifica-se que para transformar as aproximações obtidas pelo método de Euler em aproximações com o método de CN, basta fazer a seguinte substituição dos coeficientes

$$\frac{h^2 + 4k(\lambda + 2\mu)\tau}{h^2(\lambda + 2\mu)\tau} \longleftarrow \frac{h^2 + 2k(\lambda + 2\mu)\tau}{h^2(\lambda + 2\mu)\tau} \quad (\text{A.37})$$

$$\frac{h^2 + 4k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \longleftarrow \frac{h^2 + 2k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \quad (\text{A.38})$$

$$\frac{1}{4(\lambda + 2\mu)\tau} \longleftarrow \frac{h^2 - 2k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau} \quad (\text{A.39})$$

$$\mathcal{P}_{i,j}^{m+1} \longleftarrow \frac{\mathcal{P}_{i,j}^{m+1} + \mathcal{P}_{i,j}^m}{2} \quad (\text{A.40})$$

ou então substituir apenas as constantes c_6 , c_8 e c_9 descritas nas Eq. (A.17) por

$$c_6 = \frac{h^2 + 2k(\lambda + 2\mu)\tau}{h^2(\lambda + 2\mu)\tau}, \quad (\text{A.41})$$

$$c_8 = \frac{h^2 + 2k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau}, \quad (\text{A.42})$$

$$c_9 = \frac{h^2 - 2k(\lambda + 2\mu)\tau}{4h^2(\lambda + 2\mu)\tau}, \quad (\text{A.43})$$

e fazer a média dos valores dos termos fonte considerando o passo de tempo atual e anterior, como descrito na Eq. (A.40).

Para a solução direta na malha mais grossa possível, com apenas 3 pontos espaciais em cada direção (sendo que 2 destes pontos são contornos nulos) usando a aproximação temporal com o método de CN, resolve-se um sistema com as equações do deslocamento descritas no sistema de Eq. (6.59) e a equação da pressão descrita em Eq. (6.60), no ponto $i = j = 2$.