

EFEITO DE MALHAS ANISOTRÓPICAS BIDIMENSIONAIS SOBRE O DESEMPENHO DO MÉTODO *MULTIGRID* GEOMÉTRICO

Fabiane de Oliveira, faboliveira@uepg.br¹
Marcio Augusto Villela Pinto, marcio_villela@yahoo.com.br^{1,2}
Carlos Henrique Marchi, marchi@ufpr.br²
Luciano Kiyoshi Araki, lucaraki@ufpr.br²

¹Universidade Estadual de Ponta Grossa, Departamento de Matemática e Estatística. Av. General Carlos Cavalcanti, 4748. Campus de Uvaranas. 84030-900, Ponta Grossa – PR.

²Universidade Federal do Paraná, Departamento de Engenharia Mecânica. Curitiba – PR.

Resumo: *O objetivo deste trabalho é reduzir o tempo de CPU necessário para resolver problemas difusivos bidimensionais, discretizados com malhas anisotrópicas. Os modelos matemáticos considerados referem-se a três problemas bidimensionais lineares de condução de calor, governados pelas equações de Laplace e Poisson, com condições de contorno de Dirichlet. O método de diferenças finitas é usado para discretizar as equações diferenciais com esquema de diferença central (CDS) de segunda ordem de acurácia. Os sistemas de equações algébricas são resolvidos usando-se os métodos Gauss-Seidel lexicográfico e red-black, associados ao método multigrid geométrico com esquema de correção (CS) e ciclo V. Considerou-se problemas anisotrópicos com as razões de aspecto, $Q = 1/64, 1/16, 1/4, 1, 4, 16$ e 64 . São feitas comparações entre diversos algoritmos de engrossamento: engrossamento padrão (EP), semi-engrossamento (SE), semi-engrossamento completo (SEC), semi-engrossamento seguido de engrossamento padrão (SE-EP) e engrossamento padrão seguido de semi-engrossamento (EP-SE). Também são realizadas comparações entre alguns tipos de restrições: injeção, meia ponderação, ponderação completa. São propostos três tipos de restrição para problemas anisotrópicos: meia ponderação geométrica, ponderação geométrica completa e ponderação parcial. O processo de prolongação utilizado é a interpolação bilinear. Também investiga-se o efeito sobre o tempo de CPU causado por: número de pontos na malha (N); número de iterações internas no solver (v); e número de malhas (L). Verificou-se que: o algoritmo SE-EP é o mais rápido entre os cinco algoritmos testados; e confirmou-se que, para problemas isotrópicos e anisotrópicos, o solver Gauss-Seidel red-black com restrição por ponderação parcial resulta em menor tempo de CPU em relação ao Gauss-Seidel lexicográfico.*

Palavras-chave: semi-engrossamento, diferenças finitas, Laplace, Poisson, algoritmo.

1. INTRODUÇÃO

O método *multigrid* (Briggs et al., 2000) é um dos métodos iterativos mais eficientes e gerais conhecidos para se resolver sistemas de equações. Consiste no uso de malhas auxiliares mais grossas (com menor número de nós) do que a malha na qual se quer resolver o problema. São usados processos de restrição e prolongação para transferir informações entre as diversas malhas. Podem ser usados diversos *solvers* no processo de relaxação ou suavização. A seqüência com que as diversas malhas são usadas denomina-se ciclo *multigrid* (ciclo V, W, F entre outros). Podem ser usados dois tipos de esquemas com o método *multigrid*: CS (*Correction Storage*) e FAS (*Full Approximation Storage*), que são mais indicados para problemas lineares e não-lineares, respectivamente (Brandt, 1977). O método *multigrid* classifica-se em geométrico e algébrico, indicados para malhas estruturadas e não-estruturadas, respectivamente.

Wesseling e Oosterlee (2001) fizeram uma revisão dos desenvolvimentos, na década de 90, do método *multigrid* geométrico. Em relação ao tempo computacional necessário para se resolver um problema de CFD (*Computational Fluid Dynamics*), por exemplo, as equações de Navier-Stokes, o tempo de CPU pode ser reduzido ainda de 10 a 100 vezes do atual (Brandt et al., 2002). Um aumento na eficiência do método também permite, no mesmo tempo computacional, resolver um problema em uma malha mais refinada, isto é, com maior número de nós; isto significa obter uma solução numérica com menor erro de discretização (Roache, 1998).

A eficiência do método *multigrid* não tem sido totalmente alcançada em aplicações realísticas da engenharia, na área de CFD. Uma importante razão para isto é que em CFD, em geral, são resolvidos problemas com perturbações singulares, fortes anisotropias, alta razão de aspecto, equações governantes que podem exibir um comportamento em uma parte do domínio e outro em outra parte do domínio (Wesseling e Oosterlee, 2001).

A anisotropia pode ser classificada em física e geométrica. A anisotropia geométrica, ou anisotropia de malha, ocorre quando a discretização da malha possui elementos com tamanho constante, porém distintos nas direções

coordenadas. Ela pode ser caracterizada através das componentes: distância entre dois nós consecutivos (h_x e h_y), número de nós (N_x e N_y) ou comprimento do domínio de cálculo (C_x e C_y) nas direções x e y , respectivamente. Neste trabalho aborda-se o caso particular onde: $h_x \neq h_y$, $N_x \neq N_y$ e $C_x = C_y$. A anisotropia pode ser avaliada através da razão de aspecto definida por $Q = h_x/h_y$, onde $h_x = 1/(N_x - 1)$ e $h_y = 1/(N_y - 1)$ são as distâncias entre dois nós consecutivos nas direções coordenadas x e y , respectivamente. Se $Q = 1$, o problema é denominado isotrópico (Briggs et al., 2000).

Para $0 < Q \ll 1$ ou $Q \gg 1$ (fortes anisotropias) a razão de convergência do método *multigrid* padrão se deteriora (Wesseling e Oosterlee, 2001), podendo até mesmo ocorrer divergência (Larsson et al., 2005). Este tipo de anisotropia é muito comum em problemas de camada limite, onde a razão de aspecto pode ser da ordem de 10^3 , 10^4 ou mais (Wesseling e Oosterlee, 2001). Por isto, a importância de se estudar algoritmos eficientes para razões de aspecto distintas da unidade.

Aplicar o método *multigrid* com engrossamento padrão (EP) em problemas anisotrópicos, faz com que as taxas de convergência se degenerem, surgindo a necessidade de métodos mais eficazes para este tipo de problema. Diversos estudos têm dado ênfase para algoritmos de semi-engrossamento (SE). Mulder (1989) e Naik e Van Rosendale (1993) utilizaram o SE múltiplo no lugar do SE padrão, e concluíram ser mais vantajoso. Radespiel e Swanson (1995) apresentam uma variante do SE para as equações de Navier-Stokes. Larsson et al. (2005) resolveram problemas anisotrópicos com SE condicional, mantendo altas taxas de convergência com alta razão de aspecto.

Em Zhang (2002), são abordados problemas anisotrópicos acoplando-se a técnica de semi-engrossamento seguida de engrossamento padrão (SE-EP) intitulado como “*partial semicoarsening*”. Foram feitas comparações para a equação de Poisson bidimensional com a técnica denominada de EP (*full coarsening*). Zhang (2002) comparou estas técnicas para as razões de aspecto 1/2, 1/4, 1/8 e 1/16. Constatou-se que o algoritmo *partial semicoarsening* com os suavizadores Gauss-Seidel *red-black* e *four-color* são eficientes para as anisotropias estudadas.

Fischer e Huckle (2006) fazem uma análise de problemas no contexto de sistemas anisotrópicos. São geradas funções e suas curvas de nível, permitindo o desenvolvimento de métodos *multigrid* para sistemas onde a anisotropia ocorre em direções arbitrárias e não ao longo dos eixos coordenados. Fischer e Huckle (2008) dão continuidade a este estudo incluindo o uso de técnicas de suavização sofisticadas que podem ser combinadas com EP e SE. Gee et. al, (2009) propõem um novo tipo de prolongação para problemas anisotrópicos e o método *multigrid* algébrico. Isto motiva o estudo das propriedades do método *multigrid* geométrico, tanto em sua fundamentação como em aplicações, para que se possam elaborar algoritmos mais eficientes para classes de problemas mais abrangentes.

No presente trabalho compara-se cinco algoritmos de engrossamento para problemas anisotrópicos: engrossamento padrão (EP) proposto por Brandt (1977); semi-engrossamento (SE) proposto por Mulder (1989); semi-engrossamento completo (SEC) proposto neste artigo; engrossamento padrão seguido de semi-engrossamento (EP-SE) proposto por Pinto e Marchi (2006); e semi-engrossamento seguido de engrossamento padrão (SE-EP) proposto por Zhang (2002). Também são realizadas comparações entre alguns tipos de restrições: injeção, meia ponderação e ponderação completa. São propostos três tipos de restrição para problemas anisotrópicos: meia ponderação geométrica, ponderação geométrica completa e ponderação parcial.

São considerados os seguintes parâmetros neste trabalho: número de iterações internas no *solver* (v); número de níveis de malha (L); número de elementos (E) 4.096, 65.536, 262.144 e 1.048.576; razão de aspecto (Q) 1/64, 1/16, 1/4, 1, 4, 16 e 64; *solver* Gauss-Seidel lexicográfico e *red-black*; e restrição por injeção, meia ponderação, ponderação completa e ponderação parcial. O objetivo deste trabalho é verificar o efeito desses parâmetros sobre o tempo de CPU para o método *multigrid*. O algoritmo *multigrid* adotado é o esquema de correção (*Correction Scheme*) com ciclo V e prolongação por interpolação bilinear para a razão de engrossamento padrão, $r = 2$. Os modelos matemáticos considerados referem-se a três problemas bidimensionais lineares de condução de calor, governados pelas equações de Laplace e Poisson, com condições de contorno de Dirichlet.

Este artigo está organizado da seguinte forma: na seção 2, são apresentados os modelos matemáticos e numéricos, incluindo detalhes do método *multigrid*. Nas seções 3 e 4 são apresentados os resultados e a conclusão do trabalho.

2. MODELOS MATEMÁTICOS E NUMÉRICOS

2.1. Modelos Matemáticos e Suas Discretizações

Os modelos matemáticos considerados referem-se a três problemas bidimensionais lineares de condução de calor, governados pelas equações de Laplace e Poisson, com condições de contorno de Dirichlet (Incropera e DeWitt, 1998):

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = S \quad (1)$$

onde x e y são as direções coordenadas e T representa a temperatura. O termo fonte (S), as condições de contorno e as soluções analíticas estão na Tab. 1.

Tabela 1. Problemas resolvidos.

Termo Fonte	Solução Analítica	Condições de Contorno
$S = 0$	$T(x, y) = xy$	$T(0, y) = T(x, 0) = 0, T(x, 1) = x$ $T(1, y) = y$
$S = 0$	$T(x, y) = \text{sen}(\pi x) \frac{\text{senh}(\pi y)}{\text{senh}(\pi)}$	$T(0, y) = T(x, 0) = T(1, y) = 0$ $T(x, 1) = \text{sen}(\pi x)$
$S = -2[(1 - 6x^2)y^2(1 - y^2) + (1 - 6y^2)x^2(1 - x^2)]$	$T(x, y) = (x^2 - x^4)(y^4 - y^2)$	$T(0, y) = T(x, 0) = 0$ $T(1, y) = T(x, 1) = 0$

A discretização do domínio é feita com o método das diferenças finitas (Tannehill et al., 1997) com aproximação de segunda ordem de acurácia e malhas uniformes. O número de incógnitas (ou número de pontos) é dado por $N = N_x N_y$, onde N_x e N_y são os números de pontos nas direções coordenadas x e y respectivamente, incluindo os contornos. Para cada um dos $(N_x - 2)$ por $(N_y - 2)$ pontos interiores da malha, a Eq. (1) é discretizada com o esquema de diferença central (CDS) (Tannehill et al., 1997) resultando em um sistema de equações algébricas do tipo

$$AT = b \tag{2}$$

onde a matriz dos coeficientes A é pentadiagonal N por N , simétrica e definida positiva (Briggs et al., 2000), T é um vetor de incógnitas e b o termo fonte. Em problemas unidimensionais, dois nós consecutivos da malha determinam um subintervalo, neste trabalho denominado por “elemento” (E). A distância entre estes dois nós é definida por h_x . Para problemas bidimensionais, h_x e h_y são os comprimentos de cada elemento nas direções x e y , respectivamente, denotados por E_x e E_y . O número total de elementos é dado por $E = E_x E_y$.

2.2. Método Multigrid e Detalhes Numéricos

O sistema de equações algébricas representado pela Eq. (2) é resolvido com o método *multigrid* geométrico, como descrito por Wesseling (1992), usando-se o esquema de correção CS. Conforme Fletcher (1991), o esquema de aproximação completa FAS é de 5% a 10% mais caro que o CS por restringir o resíduo e a solução para as malhas mais grossas. O tipo de ciclo utilizado é o V com *multigrid* padrão. O ciclo V foi escolhido porque o W é cerca de 50% mais caro (Hirsch, 1988). O processo de restrição é feito por injeção e posteriormente comparado com meia ponderação, ponderação completa, ponderação parcial, meia ponderação geométrica e ponderação geométrica completa. A prolongação é feita através de interpolação bilinear (Trottenberg et al., 2001). Neste trabalho utiliza-se a razão de engrossamento de malhas padrão, ou seja, $r = 2$.

São comparados os seguintes suavizadores ou *solvers*: Gauss-Seidel lexicográfico (GS) (Burden e Faires, 2008) e Gauss-Seidel *red-black* (Trottenberg et al., 2001). No presente trabalho, o número de iterações internas ou suavizações feitas em cada malha é representado por ν .

O número de vezes que o ciclo V é repetido é denominado de iterações externas. O critério de convergência usado para interromper as iterações externas é a norma l_2 do resíduo adimensionalizada pela norma l_2 do resíduo da estimativa inicial (norma frequentemente utilizada na literatura). Entre os trabalhos que usam esta norma pode-se citar: Trottenberg et al. (2001) e Zhang (2002). O processo iterativo é interrompido quando a norma utilizada é menor ou igual à tolerância (ϵ) de 10^{-10} . Utilizou-se o valor nulo como estimativa inicial da solução de cada problema.

Em todas as simulações, considerou-se um número de níveis de malha L tal que $1 \leq l \leq L \leq L_{\text{máximo}}$, onde l é o número do nível de uma malha em particular e $L_{\text{máximo}}$ representa o número máximo possível de malhas que se pode usar para uma dada malha mais fina, com a malha mais grossa tendo apenas um nó interno. Por exemplo, se $N = 513 \times 513$ nós com $r = 2$, as malhas são 513×513 , 257×257 , 129×129 , 65×65 , 33×33 , 17×17 , 9×9 , 5×5 e 3×3 nós; neste exemplo, portanto, $L_{\text{máximo}} = 9$.

O foco deste trabalho é a minimização do tempo de CPU (t_{CPU}). Entende-se por t_{CPU} o tempo gasto para realizar a geração de malhas, atribuir a estimativa inicial, calcular os coeficientes e resolver o sistema linear representado pela Eq. (2) até ser atingida a tolerância estabelecida com base no critério de convergência. Este tempo é medido em segundos (s) usando-se a sub-rotina CPU_TIME do Fortran 2003.

2.3. Algoritmos de Engrossamento

Neste trabalho são testados cinco algoritmos de engrossamento em problemas anisotrópicos (anisotropia geométrica).

- Engrossamento padrão (EP): apresentado por Brandt (1977). Realiza o engrossamento simultaneamente em ambas as direções. Este algoritmo está apresentado na Fig. 1a.
- Semi-engrossamento (SE): apresentado em Mulder (1989). Realiza o engrossamento em apenas uma das direções coordenadas, onde isto seja possível ou desejado. O semi-engrossamento é realizado até que a malha torne-se isotrópica. Pode ser aplicado em problemas anisotrópicos nos quais seja conhecida a direção de forte acoplamento. Este algoritmo para o semi-engrossamento em x está apresentado na Fig. 1b.
- Semi-engrossamento completo (SEC): proposto neste trabalho. Realiza o engrossamento em apenas uma das direções coordenadas como o algoritmo SE, porém o engrossamento é realizado até a malha mais grossa possível. Este algoritmo para o semi-engrossamento em x está apresentado na Fig. 1c.
- Engrossamento padrão seguido de semi-engrossamento (EP-SE): proposto por Pinto e Marchi (2006). Aplica-se primeiro o engrossamento padrão e a seguir o semi-engrossamento. Este algoritmo para o semi-engrossamento em x está apresentado na Fig. 1d.
- Semi-engrossamento seguido de engrossamento padrão (SE-EP Zhang): apresentado em Zhang (2002). Aplica-se o semi-engrossamento até a que a malha torne-se isotrópica, e a seguir aplica-se o engrossamento padrão. Este algoritmo para o semi-engrossamento em x está apresentado na Fig. 1e.
- Semi-engrossamento seguido de engrossamento padrão ótimo (SE-EP ótimo): a idéia do algoritmo SE-EP é aplicar o semi-engrossamento até a malha tornar-se isotrópica e em seguida aplicar o engrossamento padrão. Como são utilizados dois algoritmos diferentes os parâmetros ótimos podem diferir para cada um deles. O algoritmo SE-EP ótimo faz um estudo dos seguintes parâmetros: *solvers* (Gauss-Seidel lexicográfico e *red-black*) e tipo de restrição (injeção, meia ponderação, ponderação completa, meia ponderação geométrica, ponderação geométrica completa e ponderação parcial), para cada parte do algoritmo separadamente.

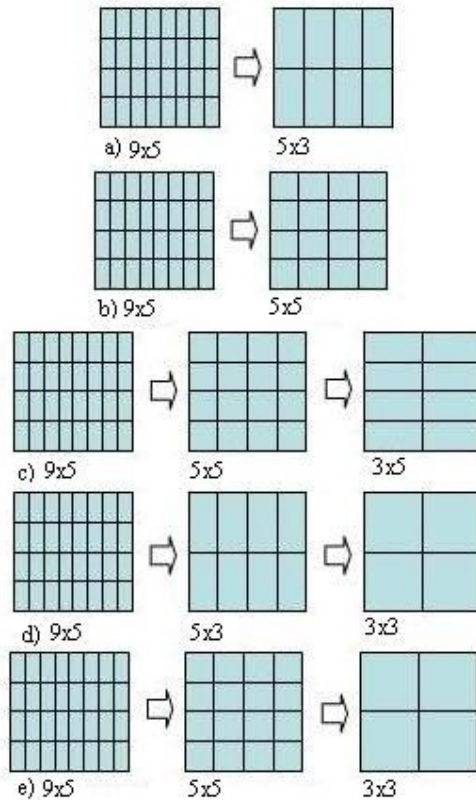


Figura 1. Algoritmos de engrossamento: a) EP, b) SE, c) SEC, d) EP-SE, e) SE-EP.

2.4 Operadores de Restrição

O operador de restrição transfere informações entre uma malha fina (Ω^h) e outra grossa (Ω^H). Ele é denominado por I_h^H onde $H = 2h$. Neste trabalho utiliza-se os operadores de restrição: injeção (I), meia ponderação (HW), ponderação completa (FW), ponderação parcial (PW), meia ponderação geométrica (GHW) e ponderação geométrica completa (GFW). As restrições HW e FW podem ser encontradas em (Trottenberg et al., 2001). As restrições GHW e GFW podem ser encontradas em (Oliveira, 2010). A restrição PW utiliza pesos somente na direção em que é realizado o engrossamento; apresenta-se a seguir os estênceis para o engrossamento na direção x e na direção y , respectivamente.

$$I_h^H = \frac{1}{4} \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix}_h^H \quad \text{e} \quad I_h^H = \frac{1}{4} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 1 & 0 \end{bmatrix}_h^H \quad (3)$$

3. RESULTADOS NUMÉRICOS

Os algoritmos foram implementados na linguagem Fortran 2003, versão 9.1 da Intel com precisão dupla. As simulações foram realizadas num microcomputador com processador Intel Core 2 Duo de 2.66 GHz, 8 GB de RAM e sistema operacional Windows xp 64 bits.

Cerca de 2.800 simulações foram realizadas com as seguintes variantes: equações (Laplace senoidal, Laplace linear e Poisson); algoritmos (EP, SE, SEC, EP-SE e SE-EP); número de elementos (65.536 e 262.144); razões de aspecto (1/64, 1/16, 1/4, 1, 4, 16, 64); número de iterações internas (1 a 1.000); número de níveis ($L_{\max}, L_{\max-1}, L_{\max-2}, L_{\max-3}, L_{\max-4}$); *solvers* (Gauss-Seidel lexicográfico e *red-black*); e restrição (injeção, meia ponderação, ponderação completa, meia ponderação geométrica, ponderação geométrica completa e ponderação parcial). Para todos os algoritmos e razões de aspecto em estudo considera-se o mesmo número de elementos. Utilizar o mesmo número de nós nem sempre é viável, pois quando este número é ímpar não é possível obter o mesmo número de nós para todas as razões de aspecto. São apresentados a seguir os resultados mais representativos.

3.1 Iterações Internas (ν)

Para o estudo do $\nu_{\text{ótimo}}$ foram fixados os seguintes parâmetros: número máximo de níveis, *solver* Gauss-Seidel lexicográfico e restrição por injeção. O parâmetro $\nu_{\text{ótimo}}$ é definido como o número de iterações internas no *solver* que resulta no menor tempo de CPU para um conjunto de dados constituído por: algoritmo, razão de aspecto e número de incógnitas. Para a equação de Laplace senoidal foram utilizadas malhas com 65.536 e 262.144 elementos e razões de aspecto 1/64, 1/16, 1/4, 1, 4, 16 e 64. Para as equações de Laplace linear e Poisson foram utilizadas malhas com 262.144 elementos e razões de aspecto 1/16, 1 e 16. Determinou-se $\nu_{\text{ótimo}}$ para os cinco algoritmos de engrossamento (EP, SE, SEC, EP-SE, SE-EP) e as equações de Laplace senoidal (LS), Laplace linear (LL) e Poisson (P). Para a razão de aspecto $Q = 1$ (problema isotrópico) utilizou-se somente o EP, pois todos os outros algoritmos recaem no EP quando a razão de aspecto é igual a 1.

Através dos resultados obtidos verificou-se que para problemas anisotrópicos $\nu_{\text{ótimo}}$ é muito alto para todos os algoritmos ($\nu_{\text{ótimo}}$ entre 10 e 250), exceto para o SE-EP. Verificou-se também que o tamanho do problema e a razão de aspecto influenciam no $\nu_{\text{ótimo}}$ para todos os algoritmos, com exceção do SE-EP.

A Fig. 2 faz uma comparação do $\nu_{\text{ótimo}}$ para as equações de Laplace senoidal, Laplace linear e Poisson com o uso do algoritmo SE-EP. Foram consideradas duas razões de aspecto: $Q = 1/64$ (Fig. 2.a) e $Q = 64$ (Fig. 2.b). Verifica-se na Fig. 2.a que para a equação de Laplace senoidal $\nu_{\text{ótimo}} = 2$. Para a equação de Laplace linear $\nu_{\text{ótimo}} = 4$ e para a equação de Poisson o $\nu_{\text{ótimo}}$ ocorre em dois pontos $\nu = 2$ e $\nu = 4$. Na Fig. 2.b verifica-se que para as três equações em estudo $\nu_{\text{ótimo}} = 4$. Fazendo uma comparação entre os resultados obtidos para as duas razões de aspecto apresentadas verifica-se que para as equações em estudo e razão de aspecto $Q = 64$, tem-se $\nu_{\text{ótimo}} = 4$. Para a razão de aspecto $Q = 1/64$ o $\nu_{\text{ótimo}}$ varia de acordo com a equação. Para a equação de Laplace linear o $\nu_{\text{ótimo}}$ para as razões de aspecto 1/64 e 64 são iguais. Isto acontece devido à equação possuir a mesma oscilação em ambas as direções x e y . Portanto, não importa se a razão de aspecto é maior ou menor que a unidade que ela não influenciará no $\nu_{\text{ótimo}}$. As equações de Laplace senoidal e Poisson não possuem a mesma oscilação nas duas direções coordenadas, logo o $\nu_{\text{ótimo}}$ varia em função da razão de aspecto maior ou menor que a unidade.

Para o algoritmo SE-EP ótimo foi feito um estudo dos seguintes parâmetros: *solvers* (Gauss-Seidel lexicográfico e *red-black*) e tipo de restrição (I, HW, FW, GHW, GFW e PW) e número de iterações internas para cada parte do algoritmo separadamente. Primeiramente fixa-se os parâmetros referentes ao EP: *solver* Gauss-Seidel lexicográfico e restrição por injeção. Realiza-se uma seqüência de simulações variando-se o *solver* e o tipo de restrição para o SE. Em seguida faz a análise do algoritmo EP fixando-se para o SE os parâmetros ótimos obtidos no passo anterior. Posteriormente é feita uma análise do número de iterações internas para cada algoritmo.

Através dos resultados obtidos nas simulações conclui-se que para problemas isotrópicos ($Q = 1$), o menor tempo de CPU é obtido com o uso do algoritmo EP, *solver* Gauss-Seidel *red-black*. O tipo de restrição não afeta significativamente o tempo de CPU. Para problemas anisotrópicos ($Q \neq 1$), o menor tempo de CPU é obtido com o uso do algoritmo SE-EP, *solver* Gauss-Seidel *red-black*. Para o SE restrição por ponderação parcial e para o EP restrição por ponderação completa tanto para o SE como para o EP. Posteriormente fez-se o estudo do ν para este algoritmo.

A Fig. 3 apresenta o número de iterações internas para o algoritmo SE-EP ótimo para a equação de Laplace senoidal. Este número foi determinado através de uma combinação do número de iterações internas para o EP (v_{EP}) e do SE (v_{SE}). Verifica-se que $v_{ótimo} = 1$ tanto para o EP como para o SE. Observa-se também que o $v_{ótimo}$ é o mesmo para as razões de aspecto maiores e menores que a unidade. Este processo foi repetido para as equações de Laplace linear e Poisson e os resultados obtidos foram similares. Para o problema isotrópico e o algoritmo EP obteve-se $v_{ótimo} = 2$. Para a razão de aspecto $Q = 1$ (problema isotrópico) utilizou-se somente o EP, pois todos os outros algoritmos recaem no EP quando a razão de aspecto é igual a 1; neste caso, $v_{ótimo} = 2$ para todas as equações.

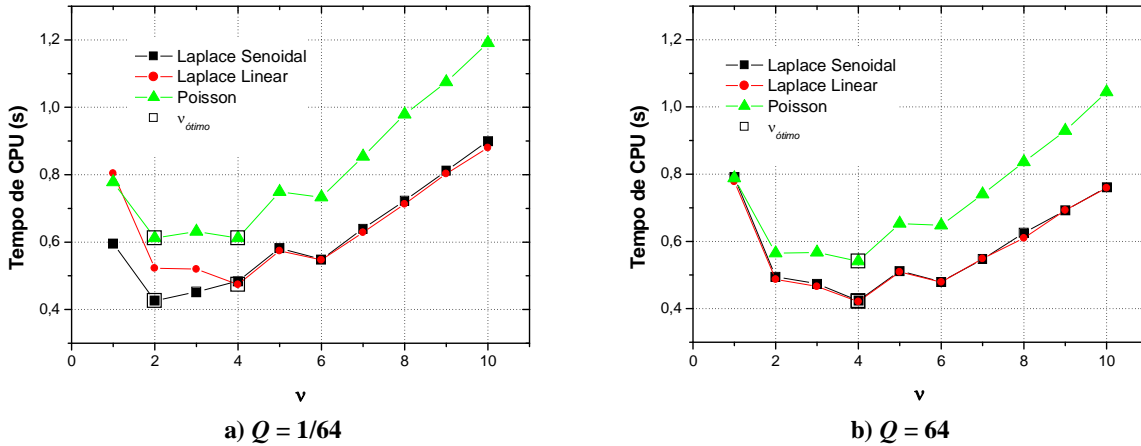


Figura 2. Tempo de CPU versus v para $Q = 1/64$ (Fig. a), e $Q = 64$ (Fig. b), $E = 262.144$ e SE-EP (Zhang).

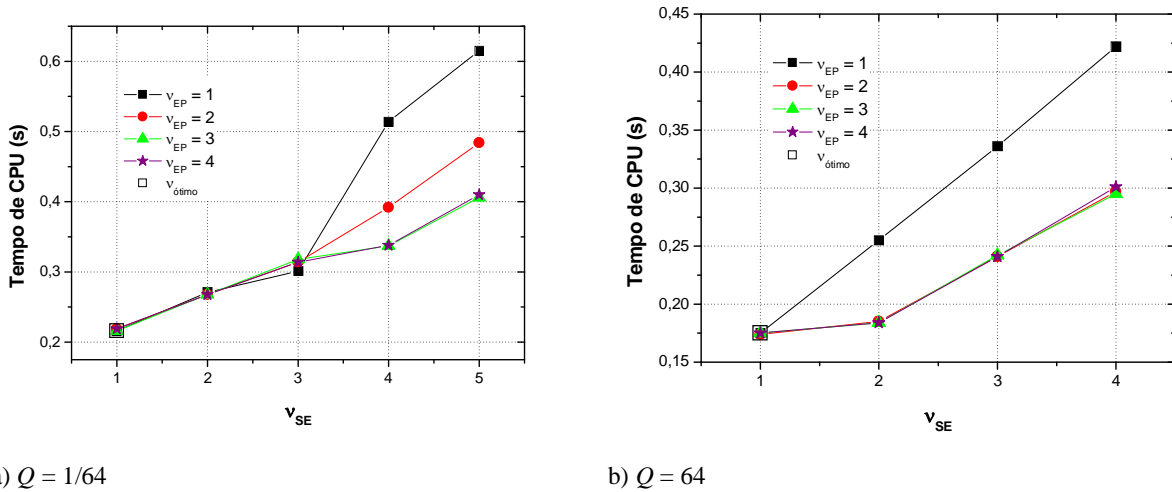


Figura 3. Tempo de CPU versus v para $Q = 1/64$ (Fig. a), e $Q = 64$ (Fig. b), $E = 262.144$ e SE-EP (ótimo).

3.2 Número de Níveis (L)

Nesta seção determina-se o número de níveis que resulta no menor tempo de CPU ($L_{ótimo}$), para um conjunto de dados constituído por: algoritmo, razão de aspecto, número de incógnitas e iterações internas. Foi analisado o efeito do número de níveis sobre a equação de Laplace senoidal e para todos os algoritmos de engrossamento em estudo. Foram utilizadas as mesmas malhas e razões de aspecto do estudo do $v_{ótimo}$ e seus valores da seção 3.1.

A Fig. 4 apresenta o tempo de CPU versus número de níveis para a equação de Laplace senoidal, algoritmo SE-EP ótimo e um problema com 262.144 elementos. Observa-se que o tempo de CPU aumenta à medida que se diminui o número de níveis, e que os menores tempos de CPU ocorrem ao se utilizar um número maior de níveis: L_{max} , L_{max-1} ou L_{max-2} . Utilizando-se as mesmas razões de aspecto, verificou-se que os resultados foram similares para outros tamanhos de malha e outros algoritmos. Portanto recomenda-se utilizar o número máximo de níveis, ou seja, $L = L_{max}$. Esta análise também já foi realizada para outras equações nos trabalhos de Pinto e Marchi (2006) e Oliveira et al. (2006). Portanto, para as equações de Laplace linear e Poisson optou-se também por utilizar $L = L_{max}$.

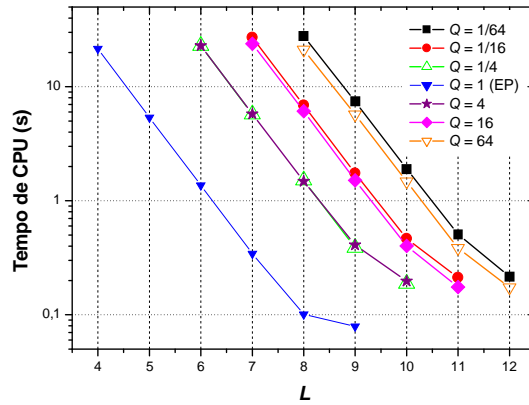


Figura 4. Tempo de CPU versus número de níveis (L) para $E = 262.144$, algoritmo SE-EP ótimo e equação de Laplace senoidal.

3.3 Comparação Entre os Algoritmos

A Fig. 5 apresenta uma comparação entre os diversos algoritmos para as razões de aspecto $Q = 1/64, 1/16, 1/4, 1, 4, 16$ e 64 para a equação de Laplace senoidal. Para problemas anisotrópicos o algoritmo SE-EP ótimo apresenta o menor tempo de CPU entre os algoritmos e as razões de aspecto testadas, exceto para $Q = 1$ onde o EP é mais rápido. Pode-se notar que os algoritmos EP e EP-SE ficam mais lentos à medida que se aumenta a razão de aspecto ou o seu inverso. Os algoritmos SE e SEC são mais rápidos à medida que aumenta a razão de aspecto, mas nunca melhores que o SE-EP ótimo. Em resumo, para os mesmos parâmetros (número de iterações internas, razão de aspecto, número de níveis e tamanho do problema), dentre os cinco algoritmos estudados tem-se que o tempo de CPU do algoritmo SE-EP ótimo é menor que o tempo de CPU dos demais algoritmos (SE, SEC, EP-SE), exceto para problemas isotrópicos onde o algoritmo EP apresenta bons resultados. Esta melhora no tempo de CPU foi entre 2 a 2.786 vezes, o que comprova a vantagem do uso do algoritmo SE-EP para problemas anisotrópicos. Para as equações de Laplace linear e Poisson os resultados foram similares.

Zhang (2002) já havia constatado que o algoritmo *partial semicoarsening* (aqui denominado de SE-EP) é mais rápido, se comparado ao EP para a equação de Poisson bidimensional envolvendo razões de aspecto menores que a unidade ($1/2, 1/4, 1/8$ e $1/16$).

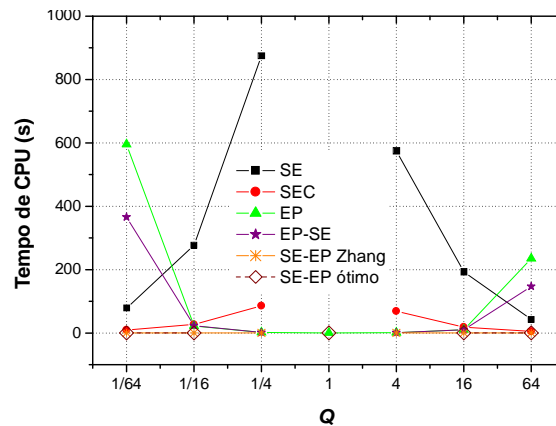


Figura 5. Tempo de CPU versus razão de aspecto (Q) para diversas razões de aspecto, $E = 262.144$ e equação de Laplace Senoidal.

A Fig. 6 faz uma comparação entre os algoritmos SE-EP Zhang e SE-EP ótimo para a equação de Laplace Senoidal. A Fig. 6.a apresenta a razão de aspecto $Q = 1/64$ e a Fig. 6.b a razão $Q = 64$. Verifica-se que o algoritmo SE-EP (ótimo) obteve um menor tempo de CPU em relação ao SE-EP Zhang. Esta redução no tempo de CPU é de em média 2,3 vezes. Por exemplo, para $Q = 16$ e a equação de Poisson esta melhora é de aproximadamente 65%. Os resultados foram similares para outras razões de aspecto e para as três equações em estudo.

Considerando-se a equação

$$t_{CPU}(E) = cE^p \quad (4)$$

sabe-se que para o método *multigrid* ideal, $p = 1$, significando que o tempo de CPU cresce linearmente com o tamanho da malha (Brandt, 1977; Trottenberg et al., 2001; e Hirsch, 1988), isto é, com E . Assim, quanto menor forem c e p , mais eficientes são o algoritmo, *hardware* e compilador utilizados. Através dos resultados obtidos, verificou-se que todos os valores de p são próximos de 1, para todas as razões de aspecto, o que indica que o algoritmo SE-EP ótimo está bem próximo do método *multigrid* ideal. O algoritmo SE-EP ótimo possuiu desempenho melhor em relação ao SE-EP Zhang. Por exemplo, para a equação de Laplace senoidal e $Q = 64$, $p = 1,11$ para o algoritmo SE-EP Zhang, e para o SE-EP ótimo $p = 1,07$. Os resultados foram similares para outras razões de aspecto e também para as equações de Laplace linear e Poisson.

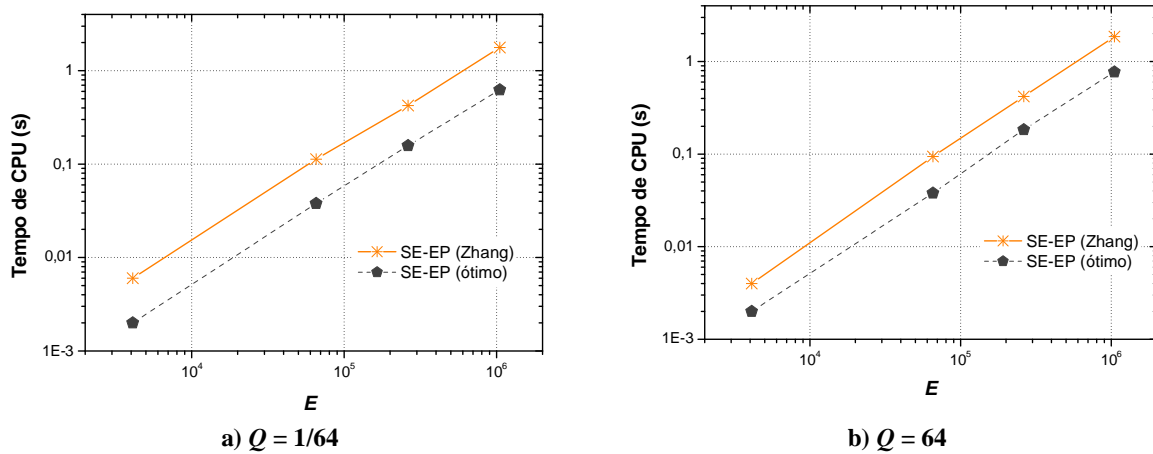


Figura 6. SE-EP ótimo versus SE-EP para as razões de aspecto (a) $Q = 1/64$ e (b) $Q = 64$ e equação de Laplace senoidal.

4. CONCLUSÃO

Neste trabalho foram resolvidos numericamente três problemas bidimensionais lineares de condução de calor, governados pelas equações de Laplace e Poisson, com condições de contorno de Dirichlet. Utilizou-se o esquema de aproximação CDS e o esquema de correção CS do método *multigrid* geométrico. Foram usados cinco algoritmos utilizados em problemas anisotrópicos: engrossamento padrão (EP), semi-engrossamento (SE), semi-engrossamento completo (SEC), engrossamento padrão seguido de semi-engrossamento (EP-SE) e semi-engrossamento seguido de engrossamento padrão (SE-EP).

Com base nos resultados obtidos neste trabalho, verificou-se que:

1. Para os mesmos parâmetros (ν , Q , L e E), o algoritmo SE-EP apresentou o menor tempo de CPU entre os cinco algoritmos analisados para os problemas anisotrópicos. A melhora no tempo de CPU ao utilizar-se SE-EP está entre 2 a 2.786 vezes.
2. A variação da razão de aspecto resulta em grande variação do $\nu_{ótimo}$ para os algoritmos EP, SE, SEC e EP-SE. Para o algoritmo SE-EP, o $\nu_{ótimo}$ permanece constante em todas as razões de aspecto. Para o algoritmo SE-EP ótimo os parâmetros que apresentam o menor tempo de CPU são: $\nu_{SE} = \nu_{EP} = 1$ para $Q \neq 1$. Para $Q = 1$ (problema isotrópico) o algoritmo EP com $\nu = 2$ apresenta o menor tempo de CPU.
3. Para todos os algoritmos de engrossamento analisados o $L_{máximo}$ resulta praticamente no menor tempo de CPU para qualquer razão de aspecto e qualquer tamanho de problema.
4. Para problemas isotrópicos ($Q = 1$) o tipo de restrição não afeta significativamente o tempo de CPU. A restrição por ponderação parcial apresenta o menor tempo de CPU para os problemas anisotrópicos.
5. Para problemas isotrópicos e anisotrópicos o *solver* Gauss-Seidel *red-black* resultou no menor tempo de CPU em relação ao Gauss-Seidel lexicográfico.
6. O algoritmo SE-EP ótimo obteve uma redução no tempo de CPU de em média 2,3 vezes em relação ao algoritmo SE-EP (Zhang).
7. O comportamento qualitativo dos parâmetros estudados é o mesmo para as equações de Laplace senoidal, Laplace linear e Poisson.

5. AGRADECIMENTOS

O primeiro e o segundo autores agradecem o apoio do Laboratório de Experimentação Numérica (LENA), do Departamento de Engenharia Mecânica da UFPR por disponibilizar sua infra-estrutura. O primeiro autor agradece à Universidade Estadual de Ponta Grossa pelo suporte financeiro e aos amigos do LENA. O terceiro autor é bolsista do CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico). Os autores agradecem o apoio financeiro do CNPq, da Fundação Araucária (Paraná) e da Agência Espacial Brasileira (AEB) através do Programa Uniespaço.

6. REFERÊNCIAS

- Brandt, A., 1977, "Multi-Level Adaptive Solutions to Boundary-Value Problems", *Mathematics of Computation*, Vol. 31, pp. 333-390.
- Brandt, A., Diskin B., Thomas J.L., 2002. "Recent advances in achieving textbook multigrid efficiency for computational fluid dynamics simulations". Hampton, VA, USA: ICASE/NASA;NASA/CR-211656.
- Briggs, W. L., Henson, V.E., McCormick, S.F., 2000, "A Multigrid Tutorial", 2ª ed., SIAM.
- Burden, R. L.; Faires, J. D., 2008. *Análise Numérica*, Tradutor: Ricardo Lenzi Tombi. São Paulo: Pioneira Thomson Learning.
- Fischer, R., Huckle T., 2006, "Multigrid methods for anisotropic BTTB systems". *Linear Algebra and its applications* Vol. 417. pp. 314-334.
- Fischer, R., Huckle T., 2008, "Multigrid solution techniques for anisotropic structured linear systems", *Applied Numerical mathematics*, Vol. 58. pp. 407-421.
- Fletcher, C.A.J, 1991. "Computational Methods for Fluid Dynamics". Vol.1, 2 d. Berlin: Springer 2002. Vol.. II.
- Gee W.M, Hu, J.J., Tuminaro S.R., 2009, "A new smoothed aggregation multigrid method for anisotropic problems", *Numerical Linear Algebra with applications*; Vo. 16, pp.19-37.
- Hirsch, C., 1988, "Numerical Computational of Internal and External Flows", Vol..I, Wiley.
- Incropera, F. P., & DeWitt, D. P., 1998. "Fundamentos de Transferência de Calor e Massa". 4 ed. Rio de Janeiro:LTC Editora.
- Larsson, J., Lien, F. S., Yee, E., 2005. "Conditional Semicoarsening Multigrid Algorithm for the Poisson Equation on Anisotropic Grids", *Journal of Computational Physics*, Vol. 208, pp. 368-383.
- Mulder, W.A., 1989. "A New Multigrid Approach to Convection Problems", *Journal of Computational Physics*, Vol. 83, pp. 303-323.
- Naik, N.H.; Van Rosendale, J., 1993. "The Improved Robustness of Multigrid Elliptic Solvers Based on Multiple Semicoarsened Grids", *SIAM Journal in Numerical Analysis*, Vol. 30, pp. 215-229.
- Oliveira, F., 2010. "Otimização de Ciclos, Roteiros e Parâmetros do Multigrid Geométrico com e sem Anisotropia Geométrica", Curitiba, PR. Universidade Federal do Paraná. Tese de doutorado em Engenharia Mecânica.
- Oliveira, F., Pinto, M. A. V., Santiago, C. D., Marchi, C. H., 2006, "Efeito de Parâmetros do Método Multigrid CS e FAS sobre o Tempo de CPU em Problemas 1D Lineares e Não-lineares", *Proceedings of the XXVII Iberian Latin American Congress on the Computational Methods in Engineering (CILAMCE)*, Belém, Brasil.
- Pinto, M. A. V., Marchi, C. H., 2006. "Efeito dos Parâmetros do Método Multigrid CS e FAS sobre o tempo de CPU para a Equação de Laplace Bidimensional", *Proceedings of the 11th Brazilian Congresso of Thermal Sciences and Engineering (ENCIT)*.
- Radespiel, R. E Swanson, R.C., 1995. "Progress with Multigrid Schemes for Hyperbolic Flow Problems", *Journal of Computational Physics*, Vol. 116, pp. 103-122.
- Roache, P. J., 1998. "Fundamentals of Computational Fluid Dynamics", Hermosa Publishers.
- Tannehill, J. C., Anderson, D. A., Fletcher, R. H., 1997. "Computational Fluid Mechanics and Heat Transfer", 2 ed., Washington: Taylor & Francis.
- Trottenberg, U., Oosterlee, C., Schüller, A., 2001, "Multigrid", Academic Press.
- Wesseling, P., 1992, "An Introduction to Multigrid Methods", John Wiley & Sons.
- Wesseling, P.; Oosterlee, C. W., 2001. "Multigrid with Applications to Computational Fluid Dynamics", *Journal of Computation and Applied Mathematics*, Vol. 128, pp. 311-334.
- Zhang, J., 2002, "Multigrid Method and Fourth-Order Compact Scheme for 2D Poisson Equation with Unequal Mesh-Size Discretization", *Journal of Computational Physics*, Vol. 179, pp. 170-179.



VI CONGRESSO NACIONAL DE ENGENHARIA MECÂNICA
VI NATIONAL CONGRESS OF MECHANICAL ENGINEERING
18 a 21 de agosto de 2010 – Campina Grande – Paraíba - Brasil
August 18 – 21, 2010 – Campina Grande – Paraíba – Brazil

EFFECT OF TWO-DIMENSIONAL ANISOTROPIC GRIDS ON THE PERFORMANCE OF THE GEOMETRIC MULTIGRID METHOD

Fabiane de Oliveira, faboliveira@uepg.br¹
Marcio Augusto Villela Pinto, marcio_villela@yahoo.com.br^{1,2}
Carlos Henrique Marchi, marchi@ufpr.br²
Luciano Kiyoshi Araki, lucaraki@ufpr.br²

¹ State University of Ponta Grossa, Department of Matematic and Statistics. 4748, General Carlos Cavalcanti Avenue. Campus of Uvaranas. 84030-900, Ponta Grossa – PR.

² Federal University of Paraná, Department of Mechanical Engineering. Curitiba – PR.

Abstract. *The purpose of this work is to reduce the CPU time necessary to solve two-dimensional diffusive problems, discretized with anisotropic grids. The mathematical models considered are refer to three two-dimensional linear problems of heat diffusion, governed by Laplace and Poisson equations, with Dirichlet boundary conditions. The finite difference method is used to discretize the differential equations with central difference scheme (CDS) with second-order of accuracy. The systems of equations are solved with the lexicographical Gauss-Seidel and red-black methods, associated to geometric multigrid method with correction scheme (CS) and V cycle. Anisotropic problems considered have aspect ratios, $Q = 1/64, 1/16, 1/4, 1, 4, 16$ e 64 . Comparisons among several semi-coarsening algorithms are made: standard coarsening (EP), semi-coarsening (SE), full semi-coarsening (SEC), semi-coarsening followed by standard coarsening (SE-EP) and standard coarsening followed by semi-coarsening (EP-SE). Comparisons are made among some restrictions schemes: injection, half weighting, full weighting. Three restrictions schemes for anisotropic problems are proposed: geometric half, weighting, geometric full weighting and partial weighting. The prolongation process used is the linear interpolation. It is also investigated the CPU time effects caused by: number of nodes of a grid (N); number of inner iterations of the solver (v); and number of grids used (L). It was verified that: the algorithm SE-EP is the fastest of the five algorithms tested; for isotropic and anisotropic problems it was verified that Gauss-Seidel red-black solver with partial restriction results in smaller CPU time compared to lexicographical Gauss-Seidel.*

Keywords: *semi-coarsening, finite difference, Laplace, Poisson, algorithm.*