

# *Desempenho de um Algoritmo Multigrid Paralelo Aplicado às Equações de Navier-Stokes*

Réverton Luis Antunes Neundorf  
PPGMNE  
Universidade Federal do Paraná  
Curitiba-PR, Brasil  
revert@ufpr.br

Marcio Augusto Villela Pinto, Luciano Kiyoshi Araki  
Departamento de Engenharia Mecânica  
Universidade Federal do Paraná  
Curitiba-PR, Brasil  
marcio\_villela@ufpr.br, lucaraki@ufpr.br

Leonardo Calvetti  
Faculdade de Meteorologia (FMet)  
Universidade Federal de Pelotas  
Pelotas-RS, Brasil  
lcalvetti@ufpel.edu.br

**Resumo**—O escoamento laminar bidimensional de um fluido incompressível, governado pelas equações de Navier-Stokes em coordenadas cartesianas, é o foco deste trabalho. Para tanto, utilizou-se o método de volumes finitos (MVF) com esquema de aproximação de segunda ordem (CDS), malhas quadrangulares, além de um método de projeção com correção incremental na pressão e convergência temporal de segunda ordem. O *solver* utilizado foi o método de Gauss-Seidel *red-black*. Para a obtenção da solução, foi empregado o método *multigrid* geométrico, com esquema de correção CS, restrição por ponderação completa, prolongação utilizando interpolação bilinear e número máximo de níveis para os casos estudados. A paralelização do *multigrid* foi realizada aplicando-se uma metodologia de particionamento do domínio a cada uma de suas componentes algorítmicas: *solver*, restrição, prolongação e cálculo do resíduo. Foi possível testar a superioridade do *multigrid* em relação à utilização do método de malha única (*singlegrid*). Finalmente, com a paralelização do método *multigrid* foi possível reduzir em até 8 vezes, utilizando 14 processadores, o tempo de CPU necessário para se obter as soluções das equações de Navier-Stokes.

**Palavras-chave**—Navier-Stokes; multigrid; métodos iterativos; métodos de projeção; método de volumes finitos; paralelização.

## I. INTRODUÇÃO

Neste trabalho foi resolvido numericamente o problema do escoamento laminar bidimensional de um fluido incompressível governado pelas equações de Navier-Stokes, em coordenadas cartesianas, com condições de contorno de Dirichlet para as velocidades e malhas quadrangulares.

Utilizou-se um método de projeção com correção incremental na pressão e convergência temporal de segunda ordem. O primeiro, e mais simples, método considerado de projeção foi proposto por [8] e [27]. Utilizando o método de Euler na discretização temporal o algoritmo consiste em criar uma velocidade auxiliar com a qual se estima as velocidades no passo de tempo atual a partir do valor das velocidades no

passo anterior ignorando a pressão. A partir da velocidade auxiliar é calculada a pressão, são aplicadas as condições de incompressibilidade e realizada a correção da velocidade no tempo atual. O aspecto mais atrativo do método é a solução de equações elípticas em cada uma das etapas do algoritmo, no entanto, a acurácia do método é prejudicada devido à imposição de condições de contorno de Neumann artificiais.

Em [11] foi observado que adicionando o valor do gradiente de pressão de uma iteração anterior à primeira etapa do algoritmo de [8], a qual ignora a pressão completamente na primeira etapa, a acurácia do algoritmo aumenta. Esta observação foi utilizada por [31] na formulação de um método de correção na pressão de segunda ordem. O método em questão é conhecido como esquema de correção incremental na pressão na forma padrão.

Em [2] encontra-se um método de projeção bastante conhecido, similar ao de [31], ou seja, consiste em um esquema de correção incremental na pressão na forma padrão.

Na tentativa de superar o problema de se ter a acurácia afetada devido a condições de contorno impostas à pressão, [29] propuseram adicionar o divergente das velocidades auxiliares na segunda etapa do algoritmo de [31]. Esta pequena modificação conduz a melhorias na taxa de convergência da pressão.

Em [18] encontra-se um método de projeção que, como o método de [8], ignora a pressão na primeira etapa do algoritmo e não realiza a correção da pressão na segunda etapa, no entanto apresenta taxas de convergência similares às obtidas pelo esquema de correção incremental da pressão na forma rotacional de [29]. Pelo fato de ignorar a pressão na primeira etapa e não realizar a correção da mesma na segunda etapa [4] referem-se ao esquema de [18] como um método de projeção independente da pressão.

O método numérico utilizado para a discretização do modelo matemático foi o método dos volumes finitos [19].

Foram utilizadas malhas desencontradas e condições de contorno impostas com o auxílio de volumes fictícios. Para o sistema de equações algébricas utilizou-se o método *multigrid*, [3], [30], que pertence à família dos métodos iterativos utilizados para resolver com eficiência sistemas de equações oriundos da discretização de equações diferenciais parciais e têm como principal objetivo acelerar a convergência do esquema iterativo em que são aplicados [26]. O interesse em aplicar o método *multigrid* em problemas de larga escala, mantendo o custo computacional aceitável, conduz à necessidade de paralelização do mesmo. Vários autores [5], [6], [15], [16], [28] e [20] estudaram a paralelização dos métodos *multigrid*. Contudo, segundo [30], o *multigrid* padrão não é completamente paralelizável, pois as malhas são processadas sequencialmente e o grau de paralelismo é diferente para cada malha considerada (menor para malhas mais grossas).

Neste trabalho a paralelização do método *multigrid* foi realizada adotando-se a estratégia da decomposição de dados através do particionamento do domínio em subdomínios, ou seja, considera-se que existe apenas um único sistema de equações para todos os subdomínios, e portanto, cada subdomínio pode ser resolvido com o mesmo algoritmo.

## II. MODELO MATEMÁTICO

O modelo matemático considerado refere-se ao escoamento laminar bidimensional de um fluido incompressível governado pelas equações de Navier-Stokes [18]:

$$\begin{aligned} \nabla \cdot \mathbf{u} &= 0 \\ \frac{\partial u}{\partial t} + \mathbf{u} \cdot \nabla u &= -\frac{\partial p}{\partial x} + \frac{1}{\text{Re}} \nabla^2 u \\ \frac{\partial v}{\partial t} + \mathbf{u} \cdot \nabla v &= -\frac{\partial p}{\partial y} + \frac{1}{\text{Re}} \nabla^2 v \end{aligned} \quad (1)$$

Em (1)  $x$  e  $y$  são as coordenadas espaciais (variáveis independentes),  $t$  é a coordenada temporal (variável independente),  $u$  e  $v$  são as componentes do vetor velocidade  $\mathbf{u}$  nas direções  $x$  e  $y$ , respectivamente,  $p$  é a pressão do fluido e  $\text{Re}$  é o número de Reynolds. Cada termo de (1) foi discretizado utilizando o método dos volumes finitos empregando malhas “desencontradas” (*staggered grid*).

Para testar a metodologia numérica foram resolvidos dois problemas:

**Problema 1:** Vórtices de Taylor-Green [23]

Domínio dado por  $\{(x, y) \in R^2 : 0 \leq x \leq \pi \text{ e } 0 \leq y \leq \pi\}$  e solução analítica

$$\begin{aligned} u &= -(\cos x \cdot \sin y) e^{-2(t+Ti)} \\ v &= (\sin x \cdot \cos y) e^{-2(t+Ti)} \\ p &= -\frac{1}{4}(\cos 2x + \sin 2y) e^{-4(t+Ti)} \end{aligned} \quad (2)$$

Em (2)  $Ti$  é o tempo inicial; neste trabalho foi utilizado  $Ti = 0$ , e  $t = kh_t$  onde  $k$  é o passo de tempo que se está resolvendo e  $h_t$  é o refinamento temporal. As condições iniciais e de contorno são obtidas diretamente da solução analítica.

**Problema 2:** Cavidade com tampa deslizante [10]

Domínio dado por  $\{(x, y) \in R^2 : 0 \leq x \leq 1 \text{ e } 0 \leq y \leq 1\}$  cujas condições de contorno são:

$$\begin{aligned} u(0,1) &= 1 \\ u(0,y) = u(1,y) = u(x,0) &= 0 \\ v(0,y) = v(1,y) = v(x,0) &= v(x,1) = 0 \end{aligned} \quad (3)$$

O Problema 2 não possui solução analítica, no entanto as soluções numéricas apresentadas por [10] são amplamente utilizadas na literatura como referência na comparação dos resultados numéricos.

## III. METODOLOGIA

### A. Métodos de Projeção

Referência [13] fazem uma ampla revisão dos métodos de projeção disponíveis na literatura. Discutem os principais métodos de projeção, as taxas de convergência e a relação destes com outros métodos de projeção. Apresentam uma generalização dos métodos de projeção com correção na pressão que para certa combinação de parâmetros ( $q = 2$ ,  $r = 2$  e  $\chi = 1$ ), resulta no esquema descrito a seguir:

1. Primeiro passo:

$$\frac{(3\mathbf{u}^t - 4\mathbf{u}^n + \mathbf{u}^{n-1})}{2h_t} = \beta_1 g_1(\mathbf{u}^n) + \beta_0 g_0(\mathbf{u}^{n-1}) + \nu \nabla^2 \mathbf{u}^t + 2\nabla P^n - \nabla P^{n-1}, \quad (4)$$

$$\mathbf{u}^t|_{\partial\Omega} = \mathbf{b}^n$$

onde  $\mathbf{u}^t$  é o campo de velocidades auxiliar,  $\mathbf{u}^n$  o campo de velocidades no passo de tempo  $n$ ,  $h_t$  é o refinamento temporal,  $P^n$  é pressão no passo de tempo  $n$  e  $\mathbf{b}^n$  as condições de contorno no passo de tempo  $n$ ,  $\beta \mathbf{g}$  com  $\beta$  constante representa os termos convectivos  $\mathbf{u} \cdot \nabla \mathbf{u}$ . A escolha das constantes  $\beta_1$  e  $\beta_0$  vai depender de como se quer tratar (explicitamente ou implicitamente) os termos convectivos. Para este trabalho, foi utilizado  $\beta_1 = -3/2$  e  $\beta_0 = 1/2$ .

2. Segundo passo:

$$\begin{aligned} \nabla^2 \phi^{n+1} &= \frac{3}{2h_t} \nabla \cdot \mathbf{u}^t \\ \mathbf{u}^{n+1} &= \mathbf{u}^t - \frac{2h_t}{3} \nabla \phi^{n+1} \\ P^{n+1} &= P^{n-1} - P^n + \phi^{n+1} - \nu \nabla \cdot \mathbf{u}^t \end{aligned} \quad (5)$$

Na seção de resultados serão apresentadas evidências numéricas que o esquema resultante da combinação  $q=2$ ,  $r=2$  e  $\chi=1$  é de segunda ordem em relação às normas  $L_2$ ,  $L_1$  e  $L_\infty$ .

### B. Tipo de Malha Utilizada

Neste trabalho foram utilizadas malhas “desencontradas” (*staggered grid*) do tipo [14], também utilizadas por [19], onde as velocidades são posicionadas nas faces e a pressão no centro dos volumes. A razão disso é para evitar instabilidades numéricas na solução da pressão [24].

A Fig. 1 ilustra uma malha de 6 por 6 volumes, nela é apresentada a ordenação adotada ao longo do trabalho. Na Fig. 1 os volumes com linha tracejada são chamados de volumes fictícios e não pertencem ao domínio físico do problema. Além disso, pode-se observar que as condições de contorno para a variável  $u$  são automaticamente prescritas nos contornos oeste e leste. Nos contornos norte e sul utilizou-se extrapolação linear para que os volumes fictícios levem em consideração a informação dos contornos. De forma similar ao caso da variável  $u$ , nota-se, na Fig. 1, que as condições de contorno na variável  $v$  são automaticamente prescritas nos contornos sul e norte mas nos contornos oeste e leste foi utilizada extrapolação linear.

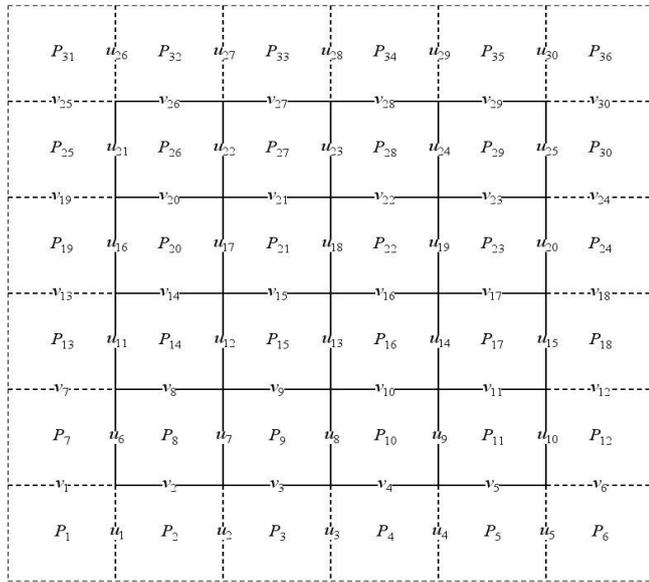


Fig. 1. Malha e ordenação lexicográfica das variáveis pressão e velocidade.

### C. Restrição no Passo de Tempo

Se o passo de tempo,  $h_t$ , for muito grande em relação ao refinamento espacial ( $h_x$  e  $h_y$ ) as simulações podem se tornar instáveis. O critério utilizado neste trabalho para evitar instabilidades é dado por [19]:

$$h_t < \frac{1}{2\text{Re}} \frac{h_x^2 h_y^2}{h_x^2 + h_y^2}. \quad (6)$$

### D. Condição de Integrabilidade

Uma característica comum dos métodos de projeção é que estes impõem condições de contorno de Neumann homogêneas sobre a pressão nos contornos. Devido a estas condições de contorno, a seguinte condição de integrabilidade precisa ser satisfeita para garantir a existência e unicidade da solução [3]:

$$\int_{\Omega} \nabla^2 \phi^{n+1} = \int_{\Omega} \frac{3}{2h_t} \nabla \cdot \mathbf{u}^n = 0. \quad (7)$$

Para detalhes de como (7) é satisfeita a cada passo de tempo, ver o procedimento proposto por [33].

### E. Método Multigrid

O método *multigrid* consiste, basicamente, em aplicar algum método iterativo básico, por exemplo os métodos de Jacobi e Gauss-Seidel [3], [30], em uma dada malha até que as componentes do erro tornem-se suaves. Na sequência, deve-se transferir o “problema” para uma malha mais grossa (onde os erros suavizados se tornam mais oscilatórios) e transferir o “problema” da malha mais grossa para a malha mais fina melhorando a solução da malha fina com o erro suavizado da malha grossa. Este procedimento para duas malhas pode ser facilmente estendido para diversas malhas e ele pode ser repetido até que a malha mais grossa (ou conveniente) seja atingida.

Considerando malhas uniformes, a razão de engrossamento é definida como  $r = H/h$ , onde  $h$  representa o tamanho dos elementos da malha fina  $\Omega^h$  e  $H$  o tamanho dos elementos da malha grossa  $\Omega^H$ . Neste trabalho será utilizado  $r=2$ . O número de malhas que podem ser visitadas durante o processo de engrossamento depende do problema considerado e da razão de engrossamento utilizada.

A Fig. 2 mostra o processo de engrossamento de uma malha de  $32 \times 32$  volumes de tamanho  $h$  (em cada direção coordenada) e razão de engrossamento  $r=2$ , ou seja,  $H = 2h$ . Neste caso,  $L = L_{\max} = 5$  níveis de malha

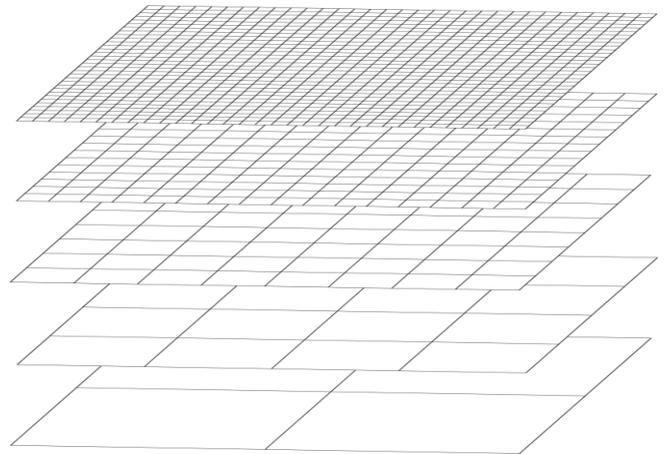


Fig. 2. Processo de engrossamento de uma malha uniforme.

A ordem na qual as malhas são visitadas é chamada de ciclo *multigrid*. Existem vários tipos de ciclos *multigrid* tais como o ciclo V, o ciclo W, o ciclo dente-de-serra, entre outros. Neste trabalho foi utilizado o ciclo V.

#### F. Processo de Restrição

Os operadores que transferem informações da malha fina  $\Omega^h$  para a malha grossa  $\Omega^{2h}$  são chamados de operadores de restrição. Estes operadores “convertem” vetores da malha fina  $\mathbf{v}^h$  em vetores da malha grossa  $\mathbf{v}^{2h}$ .

Um operador muito conhecido e utilizado neste trabalho é o operador de restrição por ponderação completa [33], que para o caso da pressão é dado por

$$\phi_{i,j}^{2h} = \frac{(\phi_{i,j}^h + \phi_{i+1,j}^h + \phi_{i,j+1}^h + \phi_{i+1,j+1}^h)}{4}, \quad (8)$$

onde  $\phi_{i,j}^{2h}$  é o valor da pressão na posição  $i, j$  da malha grossa  $\Omega^{2h}$ . De forma similar  $\phi_{i,j}^h$  é o valor da pressão na posição  $i, j$  da malha fina  $\Omega^h$ .

Para o caso das velocidades  $u$  e  $v$  foi utilizado o procedimento apresentado por [32]

$$u_{i,j}^{2h} = \frac{(u_{i,j}^h + u_{i+1,j}^h)}{2}, \quad v_{i,j}^{2h} = \frac{(v_{i,j}^h + v_{i+1,j}^h)}{2}. \quad (9)$$

#### G. Processo de Prolongação

Os operadores que transferem informações da malha grossa  $\Omega^{2h}$  para a malha fina  $\Omega^h$  são chamados de operadores de prolongação, também conhecidos como interpolação. Estes operadores “convertem” vetores da malha grossa  $\mathbf{v}^{2h}$  em vetores da malha fina  $\mathbf{v}^h$ .

Um dos operadores de prolongação mais utilizados é o operador de interpolação bilinear. Para o caso da pressão foi utilizado o procedimento apresentado em [33]

$$\left\{ \begin{array}{l} \phi_{i,j}^h = \frac{9\phi_{i,j}^{2h} + 3\phi_{i-1,j}^{2h} + 3\phi_{i,j-1}^{2h} + \phi_{i-1,j-1}^{2h}}{16} \\ \phi_{i+1,j}^h = \frac{9\phi_{i,j}^{2h} + 3\phi_{i+1,j}^{2h} + 3\phi_{i,j-1}^{2h} + \phi_{i+1,j-1}^{2h}}{16} \\ \phi_{i,j+1}^h = \frac{9\phi_{i,j}^{2h} + 3\phi_{i-1,j}^{2h} + 3\phi_{i,j+1}^{2h} + \phi_{i-1,j+1}^{2h}}{16} \\ \phi_{i+1,j+1}^h = \frac{9\phi_{i,j}^{2h} + 3\phi_{i+1,j}^{2h} + 3\phi_{i,j+1}^{2h} + \phi_{i+1,j+1}^{2h}}{16} \end{array} \right. \quad (10)$$

Para o caso da variável  $u$  as equações de interpolação utilizadas seguem os procedimentos utilizados por [33]

$$\left\{ \begin{array}{l} u_{i,j}^h = \frac{3u_{i,j}^{2h} + u_{i,j-1}^{2h}}{4} \\ u_{i-1,j}^h = \frac{3u_{i,j}^{2h} + 3u_{i-1,j}^{2h} + u_{i,j-1}^{2h} + u_{i-1,j-1}^{2h}}{8} \\ u_{i+1,j}^h = \frac{3u_{i,j}^{2h} + 3u_{i+1,j}^{2h} + u_{i,j-1}^{2h} + u_{i+1,j-1}^{2h}}{8} \\ u_{i,j+1}^h = \frac{3u_{i,j}^{2h} + u_{i,j+1}^{2h}}{4} \\ u_{i-1,j+1}^h = \frac{3u_{i,j}^{2h} + 3u_{i-1,j}^{2h} + u_{i,j+1}^{2h} + u_{i-1,j+1}^{2h}}{8} \\ u_{i+1,j+1}^h = \frac{3u_{i,j}^{2h} + 3u_{i+1,j}^{2h} + u_{i,j+1}^{2h} + u_{i+1,j+1}^{2h}}{8} \end{array} \right. , \quad (11)$$

para o caso da variável  $v$  procedimentos similares podem ser encontrados em [33].

#### H. Paralelização do Método Multigrid

A paralelização do método *multigrid* foi realizada adotando-se a estratégia da decomposição de dados através do particionamento do domínio em subdomínios, ou seja, considera-se que existe apenas um único sistema de equações para todos os subdomínios, e portanto, cada subdomínio pode ser resolvido com o mesmo algoritmo. O modelo de paralelização que aplica o mesmo programa em cada um dos subdomínios é conhecido como SPMD (*Single Program Multiple Data*)[7].

O particionamento do domínio consiste em dividir o domínio em subdomínios de forma que cada *thread* (processador físico ou lógico) resolva uma parcela do sistema de equações que define o problema em um menor intervalo de tempo.

O método de particionamento do domínio deste trabalho está diretamente relacionado com a ordenação lexicográfica dos volumes do domínio. Como são considerados apenas modelos com condições de contorno de Dirichlet, ou seja, modelos com solução conhecida nos contornos, o número de volumes de cada subdomínio pode ser calculado como

$$N_{sub} = \left\lfloor \frac{(N_x - 2)(N_y - 2)}{\bar{T}} \right\rfloor, \quad (12)$$

onde  $\bar{T}$  é o número total de *threads* disponíveis na máquina. O símbolo  $\lfloor \cdot \rfloor$  refere-se à função piso.

Equação (12) distribui o número de volumes no interior do domínio entre os *threads* disponíveis de forma que o número máximo de volumes que podem “sobrar” na divisão é  $\bar{T} - 1$ , e estes, podem ser redistribuídos entre os *threads* de forma que a diferença entre o número de volumes entre os subdomínios não seja maior que um. A escolha de quais *threads* receberão os

volumes que “sobram” é arbitrária, desde que cada *thread* receba apenas 1 volume extra (balanceamento da carga). Neste trabalho optou-se por redistribuir os nós entre os *threads*  $t \geq t_c$ , onde

$$t_c = \bar{T} + 1 - \left[ (N_x - 2)(N_y - 2) - \bar{T}N_{sub} \right]. \quad (13)$$

Os índices que particionam o domínio em subdomínios, em relação à ordenação lexicográfica, são dados por

$$\begin{aligned} i_t &= (t-1)N_{sub} + 1 + C(t-t_c) \\ f_t &= tN_{sub} + C(t-t_c + 1) \end{aligned}, \quad (14)$$

onde  $i_t$  representa o início e  $f_t$  representa o fim de cada subdomínio. O termo  $C$  é calculado com

$$C = \begin{cases} 1 & \text{se } t \geq t_c \\ 0 & \text{se } t < t_c \end{cases}. \quad (15)$$

Desta forma cada *thread* a partir de  $t_c$  (este *thread* incluso) receberá 1 volume dos que sobraram na divisão do domínio com (12).

A Fig. 3 ilustra um domínio com 49 volumes distribuídos, utilizando (14), entre: (a) 4 *threads* e (b) 8 *threads*. As formas geométricas representam os subdomínios. Pode-se observar nas duas situações, (a) e (b), que a diferença no número de volumes entre os subdomínios não passa de um.

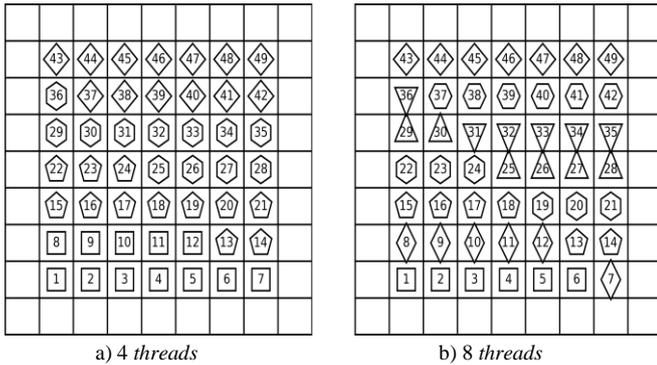


Fig. 3. Domínio com 49 volumes particionado em (a) 4 subdomínios e (b) 8 subdomínios.

Na avaliação da paralelização do método *multigrid* foram utilizados os parâmetros tempo de CPU ( $t_{CPU}$ ), medido com a função `OMP_GET_WTIME()`, e o fator *speed-up* ( $S_P$ ), interpretado como uma medida do aumento na velocidade, ou “aceleração”, do programa, definido como [7]

$$S_P = \frac{t_1}{t_P}, \quad (16)$$

onde  $t_1$  é o tempo de CPU utilizando apenas 1 processador e  $t_P$  é o tempo de CPU utilizando  $P$  processadores (tempo de execução em paralelo do programa).

Na literatura [25] é comum encontrar o *speed-up* teórico máximo, conhecido como lei de Amdahl [1], em sua forma mais genérica como

$$S_{Amdahl} = \frac{1}{(1-f) + \frac{f}{m}}, \quad (17)$$

onde  $f$  representa a parcela do processamento que pode ser melhorada por um fator  $m$  e o restante  $(1-f)$ , a parcela que não pode ser melhorada. No contexto do paralelismo,  $f$  representa a parcela do trabalho que pode ser paralelizada e  $m$ , o número e processadores utilizados.

## IV. RESULTADOS

### A. Dados de Implementação

Nesta seção são apresentados os parâmetros empregados no método *multigrid* padrão utilizado neste trabalho.

O tipo de ciclo utilizado é o V. O processo de restrição é feito empregando-se (8) e (9). O processo de prolongação é realizado empregando-se (10) e (11). A razão de engrossamento das malhas ( $r$ ) é a razão de engrossamento padrão, ou seja,  $r = 2$  [3].

O suavizador (*solver*) utilizado foi o método de Gauss-Seidel *red-black*. Neste trabalho, o número de iterações internas empregadas na pré-suavização,  $\nu_1$ , é igual ao número de iterações internas utilizadas na pós-suavização,  $\nu_2$ . Neste trabalho utilizou-se  $\nu_1 = \nu_2 = 3$  [21].

A cada passo de tempo são resolvidas numericamente, com *multigrid* ou *singlegrid* (solução obtida pelo suavizador utilizando-se apenas uma malha), as velocidades e a pressão. O critério de parada usado para interromper o processo iterativo de cada uma das variáveis (velocidades e pressão) é a norma euclidiana do resíduo adimensionalizada pela norma euclidiana do resíduo na estimativa inicial [3], [30], ou seja,  $\frac{\|\mathbf{r}^{(i)}\|_2}{\|\mathbf{r}^{(0)}\|_2}$ , onde o termo  $\mathbf{r}^{(i)}$  é o resíduo na iteração  $i$  e  $\mathbf{r}^{(0)}$ , o resíduo na estimativa inicial. Entre os trabalhos que usam essa norma pode-se citar: [3]; [21] e [30]. O processo iterativo é interrompido quando a norma é menor ou igual à tolerância  $\varepsilon$ , isto é  $\frac{\|\mathbf{r}^{(i)}\|_2}{\|\mathbf{r}^{(0)}\|_2} \leq \varepsilon$ . Neste trabalho foi utilizado  $\varepsilon = 10^{-9}$ .

Em todas as simulações, o método *multigrid* partiu da malha mais fina, nível 1, e foi até a malha mais grossa possível, nível  $L_{\max}$ .

Para avaliar o fator *speed-up* devido à paralelização é necessário medir o tempo de CPU. Este tempo é medido em segundos (s) com o uso da função `OMP_GET_WTIME()`. Entende-se por tempo de CPU o tempo gasto para realizar a geração de malhas, atribuir a estimativa inicial, calcular os coeficientes e resolver o sistema linear até atingir a tolerância estabelecida para o primeiro passo de tempo na comparação *multigrid versus singlegrid*. Como os tempos de CPU entre dois passos de tempo são muito próximos e assumindo o princípio da proporcionalidade, pode-se avaliar o tempo de CPU considerando-se apenas o tempo medido no primeiro passo de tempo. Na avaliação do paralelismo, até atingir a tolerância estabelecida para 10 passos de tempo. Não se considera o tempo de CPU no último passo de tempo porque o número de passos de tempo é variável e depende do refinamento temporal utilizado ( $h_t$ ).

Os algoritmos foram implementados na linguagem Fortran 2003 utilizando o compilador GNU Fortran (GFortran), versão 4.6.3, com a opção OpenMP e precisão dupla. Os testes foram realizados em um servidor disponibilizados pelo Sistema Meteorológico do Paraná-SIMEPAR. O servidor utilizado está equipado com processadores Intel Xeon E5-2690 de 2,9 GHz e totalizando 32 processadores, dos quais 14 estiveram disponíveis para testes; 32 GB de memória RAM e sistema operacional Linux 64 bits.

### B. Avaliação das Ordens de Convergência

Para avaliação do erro numérico foi utilizado o Problema 1 que possui solução analítica conhecida e portanto o erro numérico pode ser determinado.

Da Tabela I pode-se observar que o método estudado é de segunda ordem, para todas as variáveis estudadas, em relação a todas as normas empregadas. Da Tabela I conclui-se que os erros introduzidos pela imposição de condições de Neumann artificiais (não necessariamente obedecidas pelo fluido real) pelo método de projecção sobre a pressão é menor, ou de mesma ordem, que os erros de discretização obtidos.

TABELA I. TAXAS DE CONVERGÊNCIA OBSERVADAS

	$L_1$	$L_2$	$L_\infty$
$P$	1,995	1,993	1,944
$u$	1,978	1,980	1,977
$v$	1,979	1,983	1,984

### C. Avaliação dos Resultados para o Problema 2

Apesar deste problema não ter solução analítica para comparações diretas, em [10] podem ser encontrados resultados numéricos para comparação amplamente utilizados na literatura. Como não é conhecido o momento a partir do qual o problema da cavidade com tampa deslizante (problema 2) atinge o regime permanente, não é possível definir um tempo final ( $T_f$ ) para este problema, logo, para interromper o processo iterativo foi considerado que a maior diferença em módulo entre uma variável no passo de tempo atual e anterior

deve ser menor que  $10^{-6}$ , ou seja,  $\max(|\varphi^{n+1} - \varphi^n|) \leq 10^{-6}$ , onde  $\varphi$  representa a pressão ou velocidades  $u$  e  $v$ . O processo só foi interrompido quando todas as variáveis atingiram a tolerância estabelecida.

Na Fig. 4 são apresentadas as simulações para o Problema 2, [10]. O refinamento de malha considerado foi de  $128 \times 128$  e Reynolds variando de 100 até 3200. Pode-se observar boa concordância entre os resultados. Pode-se observar também que há uma forte dependência dos perfis das velocidades em relação ao número de Reynolds. Além disso, foi observado que o tempo necessário para atingir a tolerância aumenta com o número de Reynolds, sendo necessários muitos passos de tempo para Reynolds maiores que 1000. Visto que o tempo de processamento cresce significativamente, para números de Reynolds maiores que 1000, foi utilizada a versão paralelizada do *multigrid*.

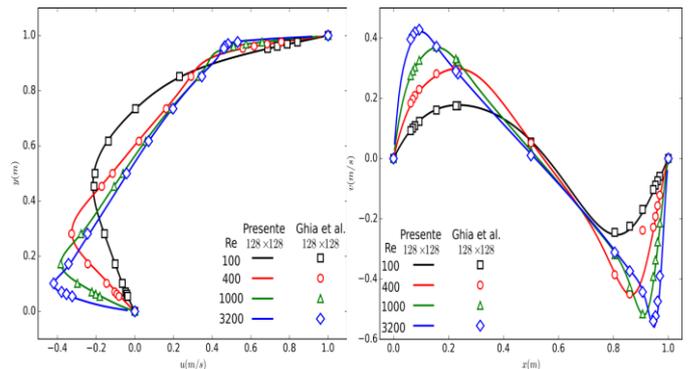


Fig. 4. Perfis das velocidades  $u$  e  $v$  em  $x=0,5$  e  $y=0,5$  para malhas de  $128 \times 128$  volumes.

### D. Multigrid versus Singlegrid

Nesta seção são apresentados os resultados do *multigrid versus singlegrid*. Basicamente foi utilizado o método iterativo Gauss-Seidel *red-black* para o *singlegrid* e como suavizador para o método *multigrid*.

Para gerar o tempo de CPU foi resolvido o Problema 1 para Reynolds igual a 1 com ambos os métodos e considerado apenas o primeiro passo de tempo (lembrando que  $h_t$  depende do refinamento espacial utilizado de acordo com as restrições ao passo de tempo). A solução numérica depende do tempo e diferentes refinamentos de tempo podem gerar diferentes números de passos de tempo, logo, para que a comparação tenha sentido foi realizado apenas um passo de tempo para vários tamanhos de malha. Foram realizadas pelo menos 5 simulações (repetições) de cada experimento numérico com as respectivas coletas de tempo de CPU para cada tamanho de malha. O tempo apresentado na Fig. 5 é uma média destes tempos (esta figura está na escala log log). Além disso, foi utilizado o mesmo critério de parada para ambos os métodos: norma  $L_2$  do resíduo da iteração atual dividida pela norma  $L_2$  do resíduo na primeira iteração deve ser menor ou igual a uma tolerância, isto é  $\|\mathbf{r}^{(i)}\|_2 / \|\mathbf{r}^{(0)}\|_2 \leq \varepsilon$ . Neste trabalho foi utilizado  $\varepsilon = 10^{-9}$ .

Na Tabela II são apresentados os tempos de CPU do *multigrid* (serial), do *singlegrid* e o fator *speed-up* (aceleração do *multigrid* em relação ao *singlegrid*). Pode-se observar um fator *speed-up* significativo do *multigrid* em relação ao *singlegrid*. Pode-se observar ainda que o fator *speed-up* aumenta com o refinamento da malha, uma propriedade desejável.

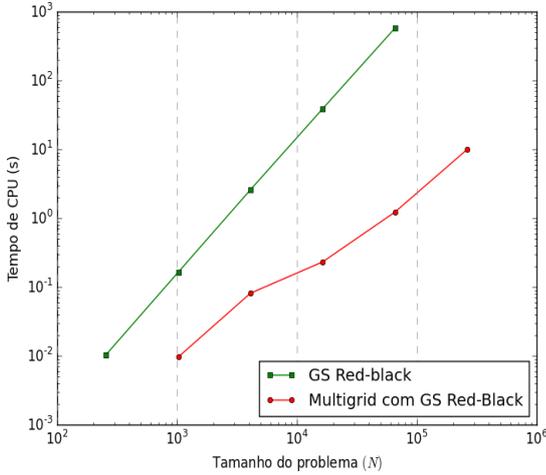


Fig. 5. Tempo de CPU para *multigrid* versus *singlegrid* para Reynolds igual a 1.

TABELA II. TEMPO DE CPU MULTIGRID (MG) VERSUS SINGLEGRID (SG) PARA REYNOLDS IGUAL 1

Malha	Multigrid	Singlegrid	Speed-up
32×32	0,01	0,16	16
64×64	0,08	2,58	32
128×128	0,23	38,8	169
256×256	1,22	572,94	470
512×512	9,91	---	---

Para calcular o esforço computacional (ordem de complexidade) foi realizado o ajuste geométrico dos dados da Tabela II considerando a função

$$T_{CPU}(N) = cN^p. \quad (18)$$

onde  $p$  representa a ordem do *solver* utilizado (*multigrid* ou *singlegrid*) ou a inclinação das curvas apresentadas na Fig. 5. No caso do *multigrid*, quanto mais próximo  $p$  estiver de 1 melhor o desempenho do algoritmo testado.

Para os dados da Tabela II os valores de  $p$  calculados utilizando (18) foram  $p = 1,196$  para o algoritmo *multigrid* e  $p = 1,978$  para o algoritmo *singlegrid*.

### E. Avaliação do Desempenho Paralelo

Nesta seção são apresentados os resultados da paralelização do método *multigrid*. Para a geração dos tempos computacionais foram simulados 10 passos de tempo de forma que os tempos totais de CPU medidos ficaram acima de 1s e,

portanto, acima da precisão da função `OMP_GET_WTIME()` do módulo OMP da linguagem FORTRAN. Além disso, foram feitas pelo menos 5 simulações (repetições) de cada experimento numérico com as respectivas coletas de tempo de CPU para cada tamanho de problema e *thread* utilizada. O tempo de CPU foi considerado como a média destes tempos.

A Fig. 6 mostra o fator *speed-up* para várias malhas. Nela são apresentados o número de processadores no eixo  $x$  e o fator *speed-up* (aceleração da versão paralela em relação à versão serial do *multigrid*) no eixo  $y$ . Além disso, são apresentadas as curvas de *speed-up* teórico máximo previsto por (17), para programas que têm, hipoteticamente, frações paralelas de 90%, 95% e 99%.

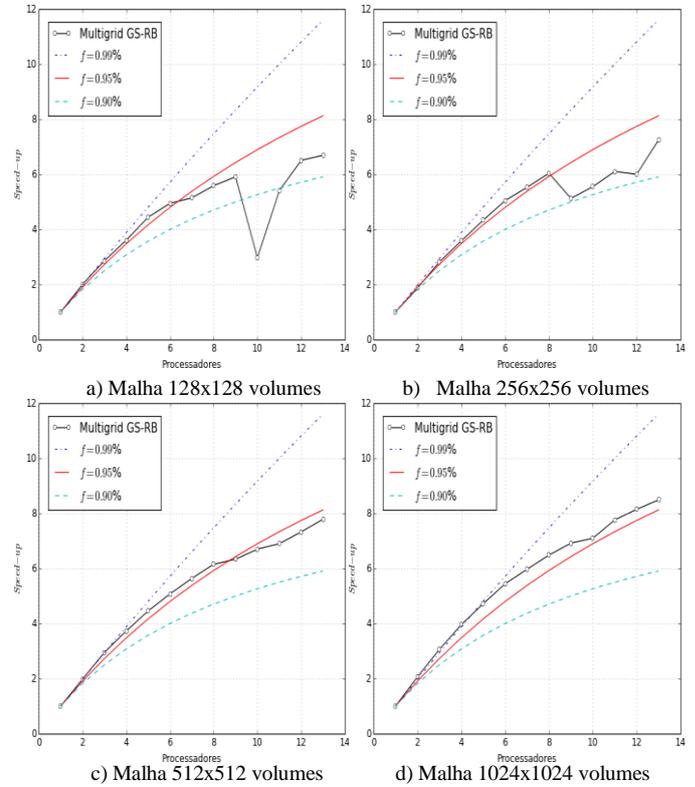


Fig. 6. Fator *speed-up* para várias malhas.

Uma análise das Figs. 6 a), b), c) e d) mostra que o fator *speed-up* encontra-se abaixo do pico máximo teórico de 14. As principais justificativas para estes resultados podem ser resumidas nas seguintes constatações:

- 1) Na versão do método *multigrid* utilizada, muitos processadores ficam ociosos durante o processamento da malha mais grossa. Para os experimentos realizados, na malha mais grossa, 12 (de 13) processadores ficam ociosos enquanto esperam que 1 processador termine suas tarefas.
- 2) No método Gauss-Seidel *red-black* apenas metade do domínio pode ser processado em paralelo.
- 3) O método de particionamento do domínio, apesar do ótimo balanceamento de carga não é ideal, pois produz subdomínios com perímetros consideráveis elevando os custos de comunicação entre os processadores.

## V. CONCLUSÃO

Neste trabalho foi apresentado um método de projeção com correção incremental na pressão e convergência temporal de segunda ordem para a resolução das equações de Navier-Stokes na variáveis primárias. Foi utilizado o método dos volumes finitos em malhas cartesianas e com variáveis descontraídas para a discretização das variáveis. Os sistemas lineares gerados foram resolvidos com o método *multigrid* geométrico com esquema CS e ciclo V. O método *multigrid* foi usado em conjunto com técnicas de paralelização. Foram realizados testes numéricos com resultados satisfatórios para um problema com solução analítica conhecida e o problema da cavidade com tampa móvel [10]. Foi possível testar a superioridade do *multigrid* em relação à utilização do método de malha única (*singlegrid*). Finalmente, com a paralelização do método *multigrid* foi possível reduzir em até 8 vezes, utilizando 14 processadores, o tempo de CPU necessário para se obter as soluções das equações de Navier-Stokes. Os resultados podem ser considerados positivos, pois verificou-se que, além do tempo de processamento ter sido reduzido significativamente empregando o *multigrid* paralelizado, este diminuiu à medida que aumentou-se o número processadores utilizados.

## AGRADECIMENTOS

Os autores agradecem o apoio do Sistema Meteorológico do Paraná – SIMEPAR, do PPGMNE (Programa de Pós-Graduação em Métodos Numéricos em Engenharia) e do Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq. O primeiro autor é bolsista do Sistema Meteorológico do Paraná – SIMEPAR.

## REFERÊNCIAS

- [1] G. M. Amdahl, "Validity of the single processor approach to achieving large scale". AFIPS spring joint computer conference. IBM Sunnyvale, California: [s.n.], pp. 483-485, 1967.
- [2] J. Bell, P. Colella and H. Glaz, "A second-order projection method for the incompressible navier-stokes equations", *Journal of Computational Physics*, vol. 85, no. 2, pp. 257-283, 1989.
- [3] W. Briggs, V. Henson and S. McCormick, *A multigrid tutorial*. Second Edition, SIAM, Philadelphia, 2000.
- [4] D. Brown, R. Cortez and M. Minion, "Accurate Projection Methods for the Incompressible Navier-Stokes Equations", *Journal of Computational Physics*, vol. 168, no. 2, pp. 464-499, 2001.
- [5] T. Chan and R. Schreiber, "Parallel Networks for Multi-Grid Algorithms: Architecture and Complexity", *SIAM Journal on Scientific and Statistical Computing*, vol. 6, no. 3, pp. 698-711, 1985.
- [6] Chan, T. F., & Schreiber, R., 1985. Parallel Networks for Multigrid Algorithms: Architecture and Complexity, *SIAM J. Sci. Comput*, v. 6, pp. 698-711.
- [7] B. Chapman, G. Jost and R. Pas, *Using OpenMP*. Cambridge, Mass.: MIT Press, 2008.
- [8] A. Chorin, "Numerical solution of the Navier-Stokes equations", *Mathematics of Computation*, vol. 22, no. 104, pp. 745-745, 1968.
- [9] J. Ferziger and M. Perić, *Computational methods for fluid dynamics*. Berlin: Springer, 2002.
- [10] U. Ghia, K. Ghia and C. Shin, "High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method", *Journal of Computational Physics*, vol. 48, no. 3, pp. 387-411, 1982.
- [11] K. Goda, "A multistep technique with implicit difference schemes for calculating two- or three-dimensional cavity flows", *Journal of Computational Physics*, vol. 30, no. 1, pp. 76-95, 1979.
- [12] J. Guermond and J. Shen, "Velocity-Correction Projection Methods for Incompressible Flows", *SIAM Journal on Numerical Analysis*, vol. 41, no. 1, pp. 112-134, 2003.
- [13] J. Guermond, P. Mineev and J. Shen, "An overview of projection methods for incompressible flows", *Computer Methods in Applied Mechanics and Engineering*, vol. 195, no. 44-47, pp. 6011-6045, 2006.
- [14] F. Harlow and J. Welch, "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface", *Physics of Fluids*, vol. 8, no. 12, p. 2182, 1965.
- [15] Hempel, R., & Schüller, A., 1988. Experiments with Parallel Multigrid using the SUPRENUM Communications Library, GMD-Studie, n. 141.
- [16] R. Herbin, S. Gerbi and V. Sonnad, "Parallel implementation of a multigrid method on the experimental ICAP supercomputer", *Applied Mathematics and Computation*, vol. 27, no. 4, pp. 281-312, 1988.
- [17] G. Karniadakis, M. Israeli and S. Orszag, "High-order splitting methods for the incompressible Navier-Stokes equations", *Journal of Computational Physics*, vol. 97, no. 2, pp. 414-443, 1991.
- [18] J. Kim and P. Moin, "Application of a fractional-step method to incompressible Navier-Stokes equations", *Journal of Computational Physics*, vol. 59, no. 2, pp. 308-323, 1985.
- [19] C. Maliska, *Transferência de calor e mecânica dos fluidos computacional*. Rio de Janeiro: Livros Tecnicos e Científicos, 2004.
- [20] O. McBryan, P. Frederickson, J. Lindenand, A. Schüller, K. Solchenbach, K. Stüben, C. Thole and U. Trottenberg, "Multigrid methods on parallel computers—A survey of recent developments", *IMPACT of Computing in Science and Engineering*, vol. 3, no. 1, pp. 1-75, 1991.
- [21] F. Oliveira, Efeitos de malhas anisotrópicas bidimensionais sobre o desempenho do método Multigrid geométrico, Doutor, Universidade Federal do Paraná, Curitiba, PR, 2010.
- [22] S. Orszag, M. Israeli and M. Deville, "Boundary conditions for incompressible flows", *J Sci Comput*, vol. 1, no. 1, pp. 75-111, 1986.
- [23] C. E. Pearson, "A computational method for time dependent two dimensional incompressible viscous flow problems", Sperry-Rand Research Center, Sudbury, Mass., Report No. SRRR-RR-64-17, 1964.
- [24] T. Shih, C. Tan and B. Hwang, "Effects of grid staggering on numerical schemes", *International Journal for Numerical Methods in Fluids*, vol. 9, no. 2, pp. 193-212, 1989.
- [25] X. Sun and Y. Chen, "Reevaluating Amdahl's law in the multicore era", *Journal of Parallel and Distributed Computing*, vol. 70, no. 2, pp. 183-188, 2010.
- [26] J. C. Tannehill, D. A. Anderson, R. H. Pletcher, *Computational Fluid Mechanics and Heat Transfer*, 2 ed. Washington: Taylor & Francis, 1997.
- [27] R. Temam, "Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des fractionnaires", *Archive for Rational Mechanics and Analysis*, v. 33, n. 5, pp. 377-385, 1969.
- [28] C. A. Thole, "Experiments with Multigrid Methods on the CalTech-hypercube", GMD-Studie, n. 103, 1985.
- [29] L. Timmermans, P. Mineev and F. Van de Vosse, "An Approximate Projection Scheme for Incompressible Flow using Spectral Elements", *International Journal for Numerical Methods in Fluids*, vol. 22, no. 7, pp. 673-688, 1996.
- [30] U. Trottenberg, C. Oosterlee and A. Schüller, *Multigrid*. San Diego: Academic Press, 2001.
- [31] J. van Kan, "A Second-Order Accurate Pressure-Correction Scheme for Viscous Incompressible Flow", *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 3, pp. 870-891, 1986.
- [32] S. Vanka, "Block-implicit multigrid solution of Navier-Stokes equations in primitive variables", *Journal of Computational Physics*, vol. 65, no. 1, pp. 138-158, 1986.
- [33] M. M. Villar, "Análise Numérica Detalhada de Escoamentos Multifásicos Bidimensionais", Doutor, Universidade Federal de Uberlândia, 2007.