

PARALELIZAÇÃO DE UM CÓDIGO DE CFD BASEADO NO MÉTODO DE VOLUMES FINITOS EM MALHA NÃO ESTRUTURADA

Diego Alves de Moro Martins, Faculdade de Engenharia Mecânica da Universidade Federal de Uberlândia, moromartins@gmail.com

Francisco José de Souza, Faculdade de Engenharia Mecânica da Universidade Federal de Uberlândia, fjsouza@mecanica.ufu.br

Resumo. Neste trabalho foi realizada a adaptação de um código computacional de CFD (Computational Fluid Dynamics), escrito em Fortran 90, para ambiente de computação paralela. Este código utiliza o método dos volumes finitos com malhas não estruturadas para solucionar escoamentos. A paralelização do código computacional foi realizada utilizando os recursos da interface MPI (Message Passing Interface), e a validação do código foi realizada com base nos resultados gerados pelo código na versão serial.

Palavras chave: Computação paralela, CFD, Malha não estruturada, Gradiente bi-conjugado

1. INTRODUÇÃO

O detalhamento necessário nos casos de dinâmica dos fluidos computacional está cada vez maior, exigindo computadores de alto desempenho e códigos computacionais capazes de solucionar problemas utilizando computação paralela, ou seja, usando mais de um processador para solucionar um caso. Acompanhado as necessidades da engenharia, o código computacional UNSCYFL3D (*Unsteady Cyclone Flow 3D*) foi paralelizado com o intuito de capturar detalhes do escoamento somente observados com malhas muito finas, e com vários processadores trabalhando em conjunto. Neste código, as equações de Navier-Stokes na formulação incompressível são resolvidas numericamente através do método de volumes finitos (Ferziger e Peric, 2002) em malhas não estruturadas, que podem ser compostas por hexaedros, tetraedros, prismas, pirâmides e cunhas. Para o acoplamento pressão-velocidade, o algoritmo SIMPLE (Ferziger e Peric, 2002) é utilizado. Escoamentos em regime permanente e transiente podem ser simulados, e 5 tipos de condição de contorno podem ser prescritas: velocidade imposta, simetria, escoamento desenvolvido, não-deslizamento e pressão imposta. Os termos difusivos são discretizados por diferenças centradas de 2ª ordem, para os termos advectivos o esquema upwind de 1ª ordem e o esquema centrado de 2ª ordem podem ser combinados. O esquema upwind de 2ª ordem também pode ser utilizado. Para o avanço no tempo, os esquema de Euler implícito (1ª ordem) e de três níveis no tempo (2ª ordem) podem ser empregados.

A paralelização do código foi realizada utilizando os recursos da interface MPI (*Message Passing Interface*), na linguagem de programação Fortran 90.

2. MÉTODO

A arquitetura utilizada na paralelização do código UNSCYFL3D foi a de memória distribuída, ou seja, múltiplos processadores operam independentemente, sendo que, cada um possui sua própria memória. Os dados são compartilhados através de uma interface de comunicação (rede ou switch), utilizando o sistema de “Message-Passing”.

A decomposição do domínio computacional é realizada pela ferramenta METIS 5.0.2 (Karypis et al., 1998), um pacote de fonte aberta para o particionamento de grafos, de malhas, e computação de preenchimento reduzido e ordenação de matrizes esparsas. O METIS é uma ferramenta de fonte aberta, amplamente utilizada pelos códigos comerciais, depurada e eficiente através do uso corriqueiro em problemas de CFD.

Os dados de entrada para o particionamento de malha utilizando a ferramenta METIS 5.0.2 são dois vetores com informações dos vértices de cada elemento, e a quantidade de partições desejadas. Um dos vetores de entrada tem que conter a sequência de identificação dos vértices que compõem cada elemento, e o outro vetor tem que conter o número de vértices de cada elemento. Como saída o METIS gera um vetor que indica em qual partição está cada elemento, e um vetor que indica em qual partição está cada vértice. A Tabela (1) mostra o número de elementos por partição da malha de um duto com 1.428.094 elementos tetraédricos e prismáticos, dividida em quatro partições pela ferramenta METIS 5.0.2.

Tabela 1. Número de elementos por processo no particionamento da malha de um duto.

Processo	Nº de Elementos
1	357.527
2	355.848
3	357.192
4	357.527

A Figura (1) mostra a geometria desta malha particionada, e a Fig. (2) mostra o detalhamento dos elementos utilizados.

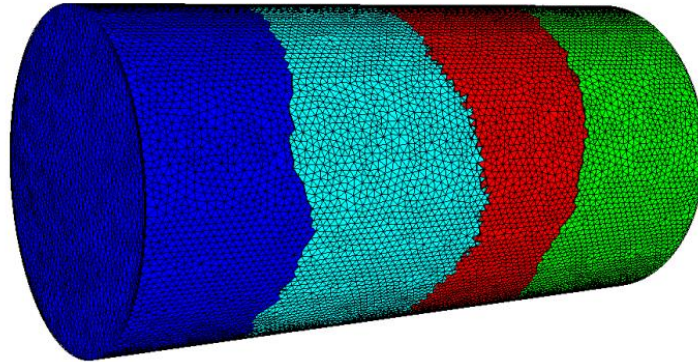


Figura 1. Malha de um duto particionada pela ferramenta METIS.

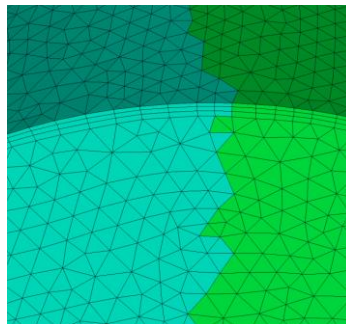


Figura 2. Detalhe dos elementos tetraédricos e prismáticos da malha particionada.

A paralelização do código foi realizada utilizando os recursos da interface MPI, com funções de comunicação ponto a ponto e coletiva. A comunicação entre os domínios é feita por elementos denominados de “halo”. Estes elementos se localizam na interface da partição, armazenando os valores das variáveis dos elementos vizinhos que estão em outras partições. Cada elemento “halo” recebe a informação do seu respectivo elemento da outra partição, como mostra a Fig. (3). Assim, os valores armazenados nos elementos “halo” devem ser atualizados a cada iteração, para que o processo iterativo paralelo gere os mesmos resultados que o processo serial.

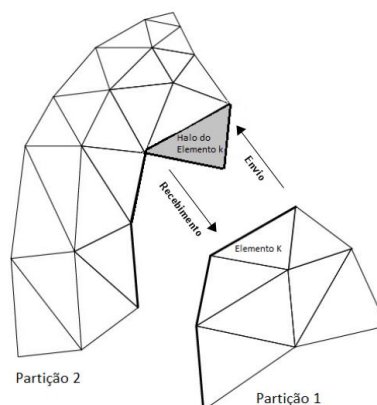


Figura 3. Esquema ilustrativo de troca de mensagens entre as partições (adaptado do Manual FEM).

O método SIMPLE (Semi-Implicit Method for Pressure-Linked Equation) foi utilizado no acoplamento pressão-velocidade do código UNSCYFL3D. A Fig. (4a) mostra o esquema do processo iterativo do método SIMPLE com as trocas de informação entre os processos, para um caso transiente. O método do gradiente bi-conjugado (Bi-CG) foi utilizado como solucionador de todos os sistemas lineares do código UNSCYFL3D na versão paralela. Este foi

adaptado da versão serial proposta por Ferziger e Peric (2002). A Fig. (4b) mostra o esquema ilustrativo do algoritmo do gradiente bi-conjugado paralelo utilizado do código.

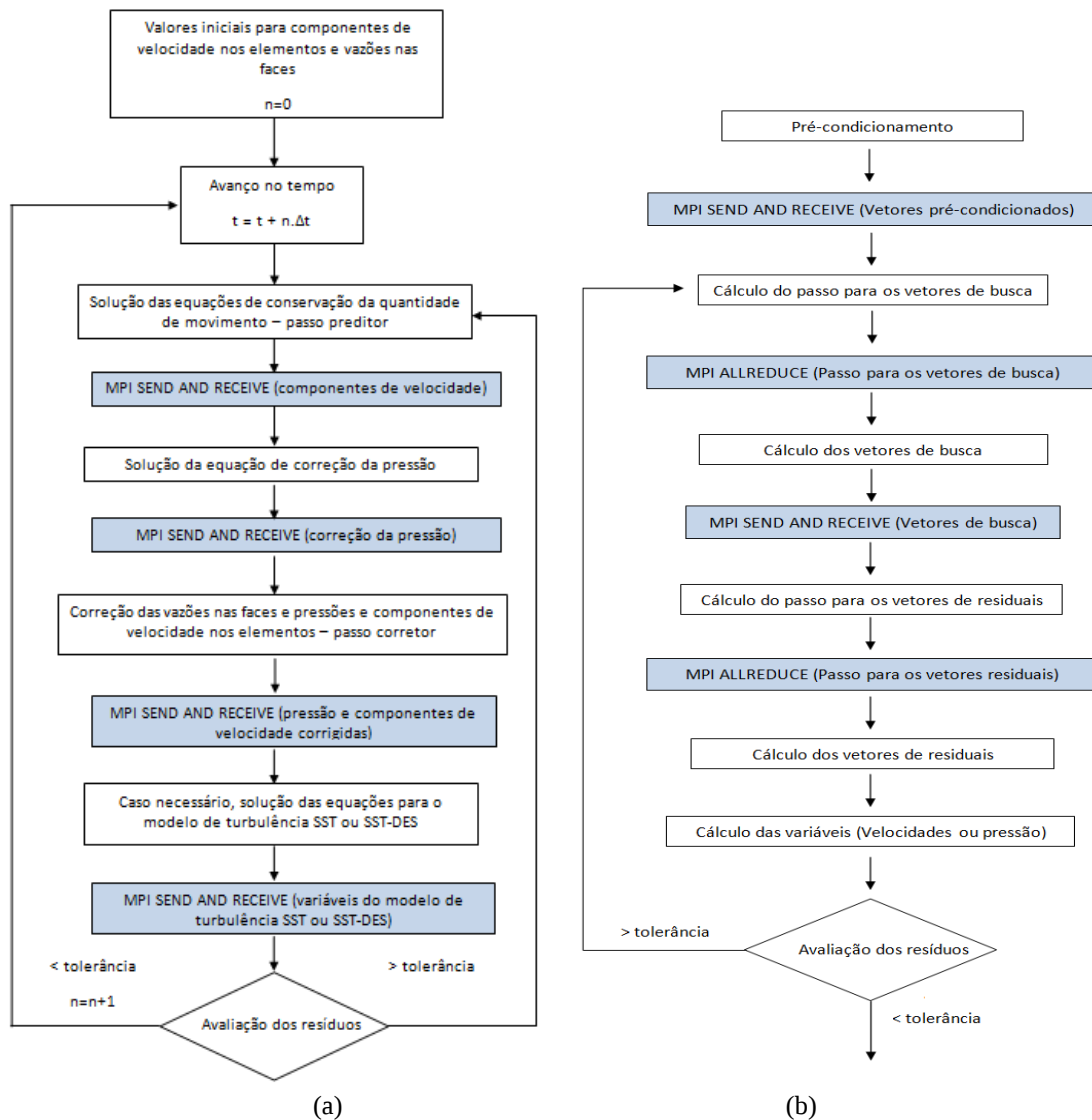


Figura 4. Esquema do processo iterativo no SIMPLE (a) e no gradiente bi-conjugado (b).

3. RESULTADOS

A validação do código UNSCYFL3D na versão paralela foi realizada com base nos resultados obtidos com o código na versão serial, validado por Salvo (2009). Os resultados para comparação foram obtidos com a simulação do escoamento em um duto não periódico com número de Reynolds de 1000. A malha do duto é mostrada na Fig. (1).

Foram posicionadas cinco sondas ao longo do domínio computacional nas versões serial e paralela do código, e foram comparados os valores de velocidade e pressão obtidos em cada sonda, após a convergência das simulações. A Tab. (2) mostra que o erro relativo entre as versões serial e paralela do código é da ordem do erro de máquina para a componente axial de velocidade e para a pressão nas cinco sondas analisadas.

Tabela 2. Erro relativo entre as versões serial e paralela do código UNSCYFL3D.

Sonda	Erro U	Erro P
1	<1E-15	4,9738E-13
2	<1E-15	1,19904E-14
3	<1E-15	4,9738E-13
4	<1E-15	6,00409E-13
5	<1E-15	3,97904E-13

O duto simulado na validação também foi utilizado na análise de *speed-up* e eficiência. Foram rodadas 15 iterações no SIMPLE em até 32 processos.

As análises de *speed-up* (Sp) e eficiência (Ef) foram feitas seguindo as seguintes equações:

$$Sp_n = \frac{T_1}{T_n} \quad (1)$$

$$Ef_n = \frac{Sp_n}{n} \quad (2)$$

onde T_1 é o tempo computacional gasto com um processo, n é o numero de processos, e T_n é o tempo computacional gasto com n processos.

As Figuras 5a e 5b mostram respectivamente os gráficos de *speed-up* e eficiência do código UNSCYFL3D na versão paralela para até 32 processos, comparando com os resultados ideais.

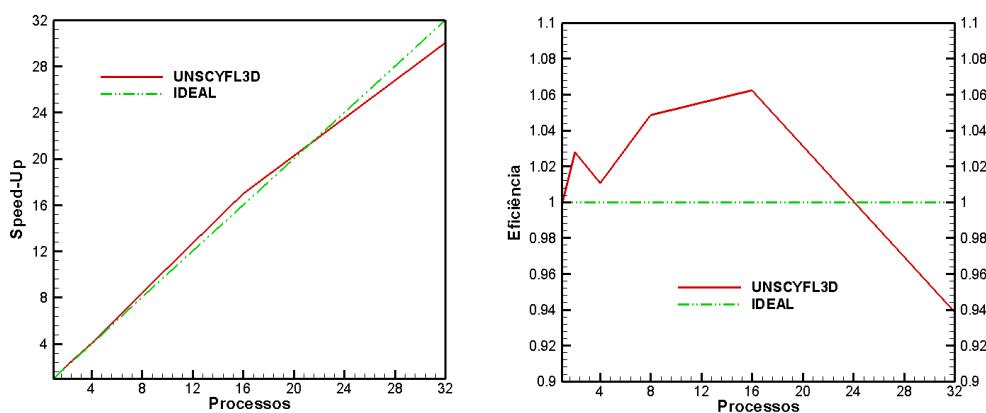


Figura 5. Gráfico do *speed-up* (a) e da eficiência (b) do código UNSCYFL3D comparado com o resultado ideal.

Observando o *speed-up* e a eficiência obtida pelo código UNSCYFL3D na versão paralela, nota-se o efeito de *super speed-up* (eficiência maior que 100%) em relação ao ideal em até 16 processos, posteriormente há um decréscimo na eficiência chegando a 94% com 32 processos. Após vários testes, foi observado que o aumento do número de processos em um mesmo nó reduz a eficiência do caso. Isto explica a apreciável eficiência em até 16 processos, onde foram utilizados menos processos por nó, e a redução da eficiência nos casos com mais processos, onde o número destes é maior por nó.

4. CONCLUSÕES

A validação do código UNSCYFL3D na versão paralela foi realizada com êxito através da comparação com a versão serial, que foi validada por Salvo (2009). A eficiência e *speed-up* com o código na versão paralela foram satisfatórios, principalmente considerando que o mesmo é baseado em malhas não estruturadas, o que dificulta o balanceamento de carga.

5. AGRADECIMENTOS

Os autores gostariam de agradecer ao CNPq, CAPES/PROEX e FAPEMIG pelo suporte financeiro, ao MFLAB pela estrutura física e equipamentos, e ao programa de pós-graduação da faculdade de engenharia mecânica pela oportunidade.

5. REFERÊNCIAS

- Ferziger, J. H. e Peric, M., 2002, "Computational Methods for Fluid Dynamics", Springer.
- Karypis, G. e Kumar, V., 1998, "METIS, A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices Version 4.0" University of Minnesota.
- Salvo, R. V., 2009, "Efeitos de Modelos Submalha em escoamentos em ciclones", Dissertação de Mestrado, Universidade Federal de Uberlândia.