



20° POSMEC

SIMPÓSIO DO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Implementação de uma estrutura octonária para refinamento adaptativo de malhas cartesianas

Autor:

Marcos Antonio de Souza LOURENÇO

Orientadores:

Aristeu da SILVEIRA NETO
Elie Luis Martínez PADILLA

Grande parte dos fenômenos físicos presentes nas simulações de mecânica dos fluidos é caracterizada por múltiplas escalas no espaço e no tempo. Muitas vezes as pequenas escalas se localizam apenas em regiões específicas e de exígua extensão se comparadas ao domínio total de solução. Esse é o caso, por exemplo, de regiões próximas à paredes, onde os efeitos viscosos muitas vezes são mais importantes do que aqueles que ocorrem no restante do domínio. A maneira mais inteligente e econômica de se conseguir bons resultados em tais problemas é efetuar um *refinamento local* da malha nos pontos de interesse. Quanto mais otimizada a distribuição dos nós/volumes na discretização, menor será o tempo computacional de solução.

O presente trabalho apresenta o desenvolvimento de uma malha semi-estruturada na forma de uma árvore octonária. Além do refinamento adaptativo, esta também permite a solução paralela por decomposição de domínios com balanceamento de carga, por meio de uma curva de preenchimento (SFC). O propósito é utilizar essa estrutura para simulação do escoamento incompressível através da solução das equações de Navier-Stokes, com o código descrito em (Padilla & Neto, 2004), mas de forma paralela.

Com a crescente utilização de computadores de múltiplos processadores e de clusters de computadores, o balanceamento de carga dinâmico se tornou algo de elevada importância na simulação de problemas de grande porte. Existem softwares livres e comerciais apenas com o objetivo de realizar a

distribuição dos nós/volumes de um determinado problema e realizar o balanceamento de carga, como é o caso de Zoltan (Catalyurek *et al.*, 2007). Uma maneira de realizar esse particionamento é através da aplicação de uma curva de preenchimento como as de Sierpiński, de Lebesgue e de Hilbert, como discutido em Hendrickson & Devine (2000). Dentre as três curvas, as duas últimas foram testadas no trabalho e são ilustradas na Fig. 1.

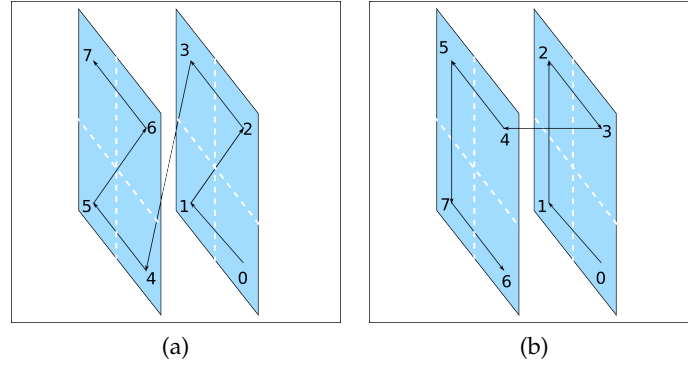


Figura 1: Exemplos de curvas de preenchimento de (a) Morton e de (b) Hilbert.

A estrutura da malha utilizada neste trabalho se aproxima daquela empregada nos trabalhos de Popinet & Rickard (2007), na forma de uma árvore octonária, diferindo daquelas baseadas em blocos superpostos, ou *patch based*, como em (Villar, 2007) e no aplicativo *FLASH* (Fryxell *et al.*, 2000).

A Fig. 2 ilustra duas malhas de estrutura octonária. Na figura da esquerda, construída utilizando uma curva do tipo Morton, pode-se observar que as partições não têm boa qualidade, ou seja, em um mesmo processo (em vermelho) existem faces que não se tocam. Isto não é bom para o desempenho, uma vez que diminui a relação processamento/(transferência de informações). A Fig. 2b utiliza a curva do tipo Hilbert e em suas partições, além de mais compactas, não existem faces que não se tocam no mesmo processo.

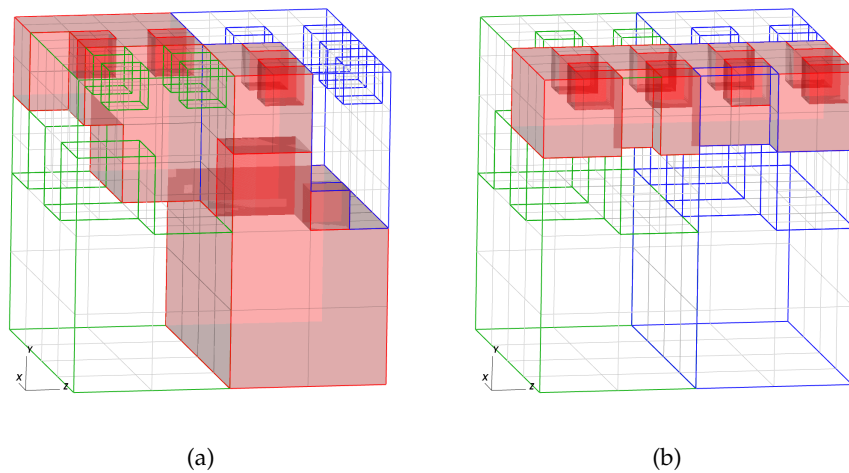


Figura 2: Duas decomposições de domínio: (a) utilizando a curva de preenchimento de Morton e (b) com a curva de Hilbert.

Embora a utilização da curva de Hilbert resulte em partições melhor definidas, sua aplicação é mais trabalhosa, requerendo mais processamento computacional que com o método de Morton. Nesse caso, é feito primeiramente um particionamento por Morton e então convertido para Hilbert através de uma tabela de coordenadas previamente definida.

Para a solução paralela do sistema linear de equações foi utilizado o framework Trilinos (Heroux *et al.*, 2003). Ele possui código aberto e diversos métodos para solução de sistemas lineares e não-lineares e oferece recursos como pré-condicionadores pré-definidos e/ou personalizados pelo usuário. Como exemplo de aplicação, foi simulado o escoamento dentro de uma cavidade quadrada com tampa deslizante, Fig. 3. A malha é uniforme e trata-se apenas de um teste simples de balanceamento de carga do solver adotado.

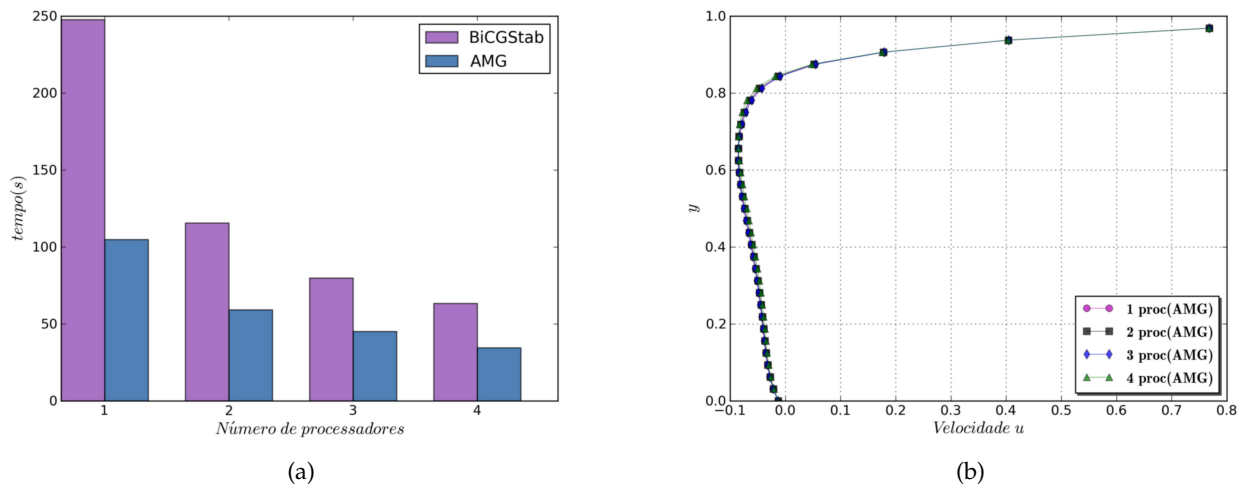


Figura 3: Simulação do escoamento em uma cavidade quadrada com tampa deslizante e malha uniforme $32 \times 32 \times 32$. (a) Balanceamento de carga e (b) Perfis da solução com número de processadores de 1 a 4. O perfil se encontra na direção y no centro da cavidade.

Dois métodos de solução foram utilizados: o método de gradiente bi-conjugado estabilizado (BiCGStab) e o método residual mínimo generalizado (GMRES) preconditionado com Multigrid Algébrico (AMG). É possível observar que o método BiCGStab escala um pouco melhor que o método AMG, embora o tempo de solução no segundo caso seja consideravelmente menor. Entretanto, o fato da máquina utilizada possuir apenas 2 núcleos de processamento físico e 4 virtuais (*Hyper-Threading*) e de algumas aplicações do sistema operacional estarem executando outras tarefas no momento da simulação pode influenciar muito no balanceamento de carga e no desempenho da solução.

Os resultados determinados até o presente momento atestam o caminho adotado para criação de um aplicativo computacional e paralelo de alto desempenho para solução das equações de Navier-Stokes. O que se encontra em andamento é apenas a criação de ferramentas que possibilitem a simulação externa e se possível interna, de risers de alta razão de aspecto. No exemplo mostrado, o solver foi aplicado para solução apenas no nível base, composto pela malha $32 \times 32 \times 32$. Futuramente, a solução deverá ser via multigrid geométrico, na qual cada nível será resolvido de maneira independente entre os processadores

que o contém. Mesmo assim, o solver continuará atuando como uma caixa preta, que recebe um sistema linear esparsa e retorna o vetor de solução.

Deve ser enfatizado que o trabalho desenvolvido até agora deverá ser acoplado à outras partes do projeto ainda não concluídas, como o modelo estrutural por exemplo, para a simulação fluido-estrutural. Seu desenvolvimento se encontra bem inserido dentro do campo de pesquisa do Laboratório de Mecânica dos Fluidos (MFLab) e acredita-se que as ferramentas necessárias para a determinação dos resultados possam contribuir para outros projetos em andamento no laboratório.

Referências

CATALYUREK, U. *et al.* Hypergraph-based dynamic load balancing for adaptive scientific computations. In: *Proc. of 21st International Parallel and Distributed Processing Symposium (IPDPS'07)*. [S.l.]: IEEE, 2007. Best Algorithms Paper Award.

FRYXELL, B. *et al.* Flash: An adaptive mesh hydrodynamics code for modeling astrophysical thermonuclear flashes. *Astrophysical Journal*, v. 131, n. (Suppl.), p. 273–334, 2000.

HENDRICKSON, B.; DEVINE, K. Dynamic load balancing in computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, v. 184, n. 2-4, p. 485 – 500, 2000. ISSN 0045-7825. Disponível em: <http://www.sciencedirect.com/science/article/B6V29-3YW270P-J/2/bbdf31e2be710e61790593b96a2da1b>.

HEROUX, M. *et al.* *An Overview of Trilinos*. [S.l.], 2003.

PADILLA, E. L. M.; NETO, A. da S. Validação de um código numérico 3d para análise de problemas de dinâmica dos fluidos. *POSMEC - Simpósio do Programa de Pós-Graduação em Engenharia Mecânica*, v. 14, 2004.

POPINET, S.; RICKARD, G. A tree-based solver for adaptive ocean modelling. *Ocean Modelling*, v. 16, n. 3-4, p. 224 – 249, 2007. ISSN 1463-5003. Disponível em: <http://www.sciencedirect.com/science/article/B6VPS-4MC14YJ-1/2/7a845021b7e7874aa500e899a00a0b5b>.

VILLAR, M. M. *Análise numérica detalhada de escoamentos multifásicos bidimensionais*. Tese (Doutorado) — Universidade Federal de Uberlândia, Uberlândia, Brasil, 2007.