

UMA ABORDAGEM ALTERNATIVA AO USO DE MPI PARA PARALELIZAÇÃO DE CÓDIGOS FORTRAM EM COMPUTAÇÃO DISTRIBUÍDA

João Ricardo Masuero (masuero@cpgec.ufrgs.br)

Armando Miguel Awruch (amawruch@vortex.ufrgs.br)

Centro de Mecânica Aplicada e Computacional - CEMACOM

Programa de Pós-Graduação em Engenharia Civil -PPGEC

Universidade Federal do Rio Grande do Sul -UFRGS

Av. Osvaldo Aranha, 99, 3º andar, Porto Alegre, 90035-190, Brasil

A enorme necessidade de capacidade de armazenamento e poder de processamento necessários à simulação numérica de problemas complexos de Mecânica dos Sólidos e Mecânica dos Flúidos utilizando o Método dos Elementos Finitos têm levado os pesquisadores a empregar soluções computacionais de grande desempenho, como supercomputadores e clusters de estações de trabalho, as quais apresentam, em geral, grandes custos de implantação e operação. Nos últimos anos, o emprego de clusters de microcomputadores comuns tem se tornado cada vez mais freqüente como solução de alto desempenho para problemas de porte médio, em função dos baixíssimos custos de implantação e manutenção apresentados. Em tais configurações, o emprego da biblioteca *MPI* (*Message Passing Interface*) é bastante comum, visto ser eficiente tanto em máquinas de memória compartilhada quanto distribuída e ser uma biblioteca padrão cujas implementações obedecem a uma norma de sintaxe, garantindo a portabilidade dos códigos desenvolvidos entre diversas plataformas.

Este trabalho visa apresentar uma abordagem alternativa ao uso de *MPI* voltada para clusters temporários de microcomputadores que é baseada inteiramente nos recursos dos sistemas operacionais atuais e nos comandos padrões das linguagens de programação.

Entende-se como clusters temporários um conjunto de microcomputadores comuns que momentaneamente são conectados em rede para o processamento paralelo/distribuído de um determinado problema, voltando ao uso normal como estações de trabalho independentes após o processamento. Esta abordagem permite a otimização de recursos computacionais de um laboratório de pesquisa, pois as máquinas de uso geral podem ser utilizadas para processamento intensivo quando necessário e vice-versa, minimizando a ociosidade do conjunto e permitindo a implantação gradativa de uma cultura de processamento paralelo / distribuído.

As características chaves dos sistemas operacionais atuais para implementação de códigos paralelos na presente abordagem são a capacidade de compartilhar pastas na rede para operações de leitura e escrita e a capacidade de mapear pastas compartilhadas como unidades de armazenamento remoto, permitindo que o acesso a arquivos nestas pastas seja feito de forma idêntica ao de arquivos contidos no disco rígido local do computador. Desta forma, o envio de dados de uma máquina para outra através da rede se resume a uma operação de gravação de dados em um arquivo, e o recebimento a uma operação de leitura. Essas características são inerentes ao sistema operacional e totalmente transparentes às linguagens de programação. Como qualquer linguagem de programação permite a manipulação de arquivos, qualquer uma delas poderia ser utilizada para a implementação de algoritmos paralelos. Da mesma forma, qualquer máquina que tenha um sistema operacional que apresente essas características poderia ser utilizada no cluster temporário, independentemente do sistema operacional utilizado pelas demais. Desta forma, computadores com Windows 95, 98, Me, NT, 2000 e XP, ou Linux poderiam ser agrupados.

No presente trabalho, foram utilizados clusters temporários formados por até 8 computadores utilizando Windows 98, 2000 e XP. Foram implementados códigos em Fortran 77/90 utilizando o Método dos Elementos Finitos para a análise elástica linear estática de sólidos

tridimensionais com solução por Gradientes Conjugados e para análise de escoamentos compressíveis tridimensionais utilizando o esquema explícito de Taylor-Galerkin, ambos utilizando elementos hexaédricos de 8 nós com um ponto de integração.

São apresentados resultados da escalabilidade do tempo de solução com o tamanho do problema e com o número de computadores empregados no cluster, bem como de eficiência de paralelização, utilizando tanto a abordagem proposta quanto *MPI*.

Resultados iniciais mostram que *MPI* é vantajoso para problemas pequenos, mas à medida que o tamanho de problema cresce as abordagens se tornam equivalentes em termos de eficiência.