O Método de Elementos de Contorno e a Visualização Científica aliados à resolução de problemas da Mecânica Computacional

Marlucio Barbosa
Universidade Federal do Rio de Janeiro, Programa de Engenharia Civil, COPPE, UFRJ
E-mail: marlucio@coc.ufrj.br

Edivaldo Figueiredo Fontes Junior
Universidade Federal do Rio de Janeiro, Programa de Engenharia Civil, COPPE, UFRJ
E-mail: fontesjunior@coc.ufrj.br

Carlos Andrés Reyna Vera-Tudela
Universidade Federal Rural do Rio de Janeiro - Departamento de Matemática
Caixa Postal 74517, CEP 23890-971, Seropédica, RJ
E-mail: candres@ufrrj.br

José Cláudio de Faria Telles
Universidade Federal do Rio de Janeiro, Programa de Engenharia Civil, COPPE, UFRJ
Caixa Postal 68506, CEP 21945-970, Rio de Janeiro, RJ
E-mail: telles@coc.ufrj.br

Resumo: *Métodos Numéricos são utilizados para dar soluções aproximadas a problemas cuja solução analítica é complexa ou impossível de ser encontrada. Com a evolução dos computadores, métodos numéricos ditos impraticáveis ganharam força e tornaram populares, como os Métodos dos Elementos Finitos e de Contorno, solucionando problemas em diversas áreas. Todavia, tais soluções numéricas, na maioria das vezes densas, possuem uma análise custosa. Este trabalho apresenta uma aplicação do Método dos Elementos de Contorno (MEC), aliado à Computação Gráfica, na Engenharia Civil. A aplicação fora feita na área de Mecânica dos Sólidos, com a elaboração do software MEMEC. O MEMEC realiza a Análise Numérica e a Visualização Científica de problemas da elasticidade linear, típicos de análise de tensões em estruturas, modelados com o Método dos Elementos de Contorno. O software foi desenvolvido pela necessidade de se aliar a precisão numérica do MEC com uma análise qualitativa e eficaz do conjunto de dados gerados pelo método.*

Introdução

O campo da Visualização Científica vem se desenvolvendo de uma forma paralela ao dos computadores; a oferta maior de recursos permite que o usuário final possa exigir mais do seu trabalho, assim como esperar programas mais poderosos, rápidos e que manipulem uma maior quantidade de dados.

Como no capitalismo, a computação é movida pela lei da demanda. A crescente necessidade de sistemas de Visualização Científica leva ao desenvolvimento da Computação Gráfica como um todo, e suas aplicações vêm se tornando evidentes em várias áreas do conhecimento.

Neste texto, são apresentados resultados do uso da Computação Gráfica, particularmente da Visualização Científica, em problemas de elasticidade linear, típicos de análise de tensões em estruturas, comuns para Engenharia Civil.

Para isso, uniu-se um poderoso método numérico, Elementos de Contorno, a técnicas sofisticadas de Computação Gráfica.

O software Mecânica Elastostática - Método de Elementos de Contorno (MEMEC) foi desenvolvido a partir de um programa em linguagem Fortran 77 que efetuava somente análises numéricas de problemas específicos da Mecânica dos Sólidos via Método dos Elementos de Contorno. O programa MEMEC foi criado pela necessidade de poder manipular uma grande quantidade de dados para num pós-processamento, apresentar os mesmos de uma forma moderna e ágil, incluindo numa visualização 2D ou 3D. Esta visualização é fundamental para eficientemente analisar o assunto em estudo.

O Método dos Elementos de Contorno

As condições de equilíbrio para um problema elastostático com carga de domínio nula é representado pela conhecida Equação de Navier:

$$\mu\, u_{j,ii} + (\lambda + \mu) u_{i,ij} = 0 \tag{1}$$

onde u representa o campo de deslocamentos; μ e λ são as constantes de Lamê. A relação entre o tensor de deslocamentos e o tensor de tensões é dado por:

$$p_i = \sigma_{ij}\, n_j \tag{2}$$

onde n_j é a normal externa à superfície.

A formulação tradicional do Método dos Elementos de Contorno [1] consiste em ponderar a equação (1) por uma função u*, com características especiais e depois integrá-la no domínio. Por meio de um tratamento matemático adequado, que envolve integração por partes e tomando em consideração o princípio de reciprocidade, transforma-se esta equação integral de domínio em uma equação integral de contorno.

$$C_{ij}(\xi) u_j(\xi) + \int_\Gamma p^*_{ij}(\xi, x)\, u_j(x) d\Gamma(x)$$

$$= \int_\Gamma u^*_{ij}(\xi, x) p_j(x) d\Gamma(x) \tag{3}$$

A solução fundamental para o problema de estado plano num meio elástico infinito é dado pela expressão seguinte:

$$u^*_{ij}(\xi, x) = \frac{-1}{8\,\pi\,(1-\nu)} \{$$

$$[(3-4\nu)\ln(r)]\delta_{ij}\} - r_{,i}\, r_{,j} \tag{4}$$

e

$$p^*_{ij}(\xi, x) = \frac{-1}{4\,\pi\,(1-\nu)\, r}$$

$$\left\{ \left[(1-2\nu)\delta_{ij} + 2\, r_{,i}\, r_{,j} \right] \frac{\partial r}{\partial n} - \right.$$

$$\left. (1-2\,\nu)(r_{,i}\, n_j - r_{,j}\, n_i) \right\} \tag{5}$$

onde r é a distância do ponto de aplicação da carga ao ponto em consideração.

A implementação

Em termos de Análise de Sistemas, a implementação de um método numérico, não é nada trivial. Várias questões têm que ser levantadas e sanadas de forma adequada. O MEMEC foi desenvolvido buscando sempre otimizar processos e operações de forma a ter ganho computacional aliado a precisão numérica. Para isso, a escolha da Linguagem Java [2] e da biblioteca VTK [4] foi a mais adequada, pois dessa forma o MEMEC também ganha portabilidade.

Em um problema de Elasticidade Linear devemos analisar com critério a sua geometria, e nesse sentido a elaboração de um software que se adapte a diversas geometrias é custoso. O MEMEC, em sua versão atual, apresenta soluções para problemas específicos da Mecânica dos Sólidos, como em problemas de barras (Figura 1).

Figura 1: Representação física da barra engastada e tracionada.

A discretização da geometria considera o contorno Γ composto por elementos distintos, sobre os quais são definidas variações para o deslocamento e a tensão em função de valores em determinados pontos (denominados nós ou pontos nodais).

Para ilustrar considere a discretização mostrada na Figura 2. Conforme é feito no método de Elementos Finitos, normalmente, os elementos de conectividade são orientados no sentido anti-horário.

Isso garante que a normal esteja sempre apontada para fora do domínio, além da convergência do gerador de malhas.

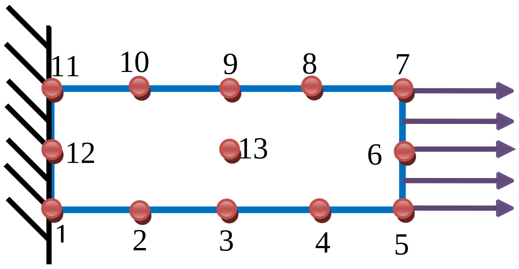


Figura 2: Representação da discretização para a barra engastada.

O Método dos Elementos de Contorno atua no contorno da Geometria do problema, todavia, em termos de Computação Gráfica, os pontos interiores são importantes para a geração de malhas (veja considerações finais) as quais serão utilizadas para a construção da visualização científica. Os escalares que são obtidos para esses pontos interiores após a execução do método são obtidos por interpolação e não via o método diretamente.

O método de interpolação utilizado é o descrito em [1] e garante uma solução aproximada de qualidade para os pontos interiores, pois decorre diretamente da construção do MEC.

Os dados da discretização do objeto como das forças que atuam sobre ele e das propriedades do material que o constitui são guardados em arquivo (Tabela 1) ASCII. Utilizamos, convencionalmente, o formato .txt.

Entrada	**Descrição**
Parâmetros	Tipo de domínio, número de elementos de contorno e de nós de contorno, número de pontos interiores, indicador do tipo de estado e de simetria, módulo de elasticidade e coeficiente de Poisson.
Coordenadas dos Nós	Coordenadas dos nós indicando se é nó duplo ou não e condições de simetria se houver.
Coordenadas dos Pontos Interiores	Coordenadas dos pontos interiores e condições de simetria se houver.
Conectividade	Conectividade dos elementos.
Número de nós com condições Prescritas	Número de nós com condição de deslocamento prescrito e de nós somente com condições de tensão prescrita.
Nós com condições de deslocamentos Prescritos	Indica o nó e o valor prescrito em cada direção do nó.
Nós com condições de tensões prescritas	Indica o nó e o valor da tensão prescrita em cada direção do nó.

Tabela 1: Descrição do arquivo de entrada.

A Tabela 1 representa a geometria do problema, isto é, descreve o objeto com as suas condições de contorno naturais e essenciais, como também as forças que atuam sobre o objeto.

O fluxograma de execução do MEMEC é apresentado na

Figura 3. Em (1) o arquivo é lido e armazenado na memória de forma a aperfeiçoar a execução do programa, evitando, ao máximo, buscas descontinuadas na memória principal e de recursos presentes na memória secundária. Em (2) os dados são verificados, e essa etapa é importante para verificar falha humana na construção do arquivo de dados antes da aplicação do método.

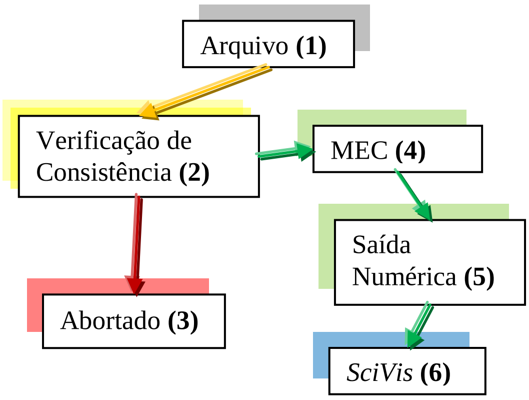


Figura 3: Fluxograma de execução do MEMEC.

Se a verificação em (2) falhar então o programa segue para (3) e é abortado apontando em que parte do arquivo, possivelmente, está a inconsistência. Se a verificação em (2) é feita com sucesso o programa segue para (4), onde o método dos elementos de contorno é executado já interpolando os pontos interiores.

A etapa (5) imprime os resultados numéricos na tela do programa em uma estrutura organizacional de abas. Em (6), executa-se a Visualização Científica (*SciVis*). Essa etapa precisa de interação com o usuário, uma vez que, ele vai escolher a visualização científica mais adequada ao que quer analisar.

O MEMEC suporta diferentes estratégias de visualização que ficam organizadas em uma estrutura de *desktop*, e isso facilita a comparação e análise de resultados. Para isso, várias janelas podem ser dispostas dentro do *desktop* interno do software (ver Figura 4).

O objetivo que norteou a construção do MEMEC foi de aliar o Método dos Elementos de Contorno a *SciVis*.

As técnicas de Visualização Científica buscam tornar o processo de análise de resultados numéricos mais dinâmico e qualitativo. Essa área da Computação Gráfica é foco de estudo de diversos pesquisadores no mundo e está em desenvolvimento constante.

O MEMEC utiliza diversas técnicas dessa área, dentre as quais podemos citar o algoritmo de triangulação de Delaunay [3].

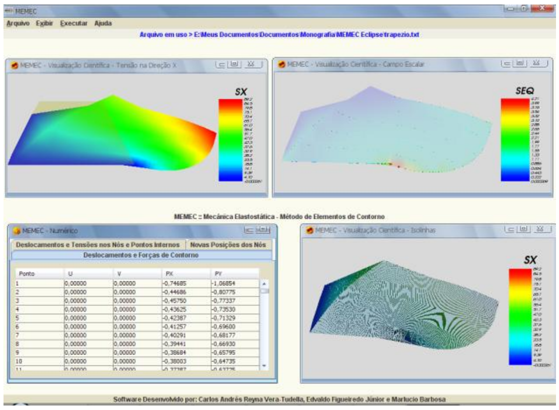


Figura 4: Tela do MEMEC com quatro janelas posicionadas no *desktop*.

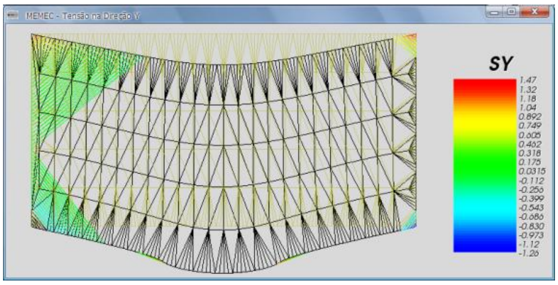
A triangulação de Delaunay D de V, introduzida, em 1934, pelo matemático russo Boris Nikolaevich Delone — depois chamado Boris Delaunay —, é um grafo definido como segue. Qualquer círculo no plano é tido como vazio se não cerca nenhum vértice de V. Vértices são permitidos sobre o círculo. Sejam u e v dois vértices de V. Um círculo-circundante da aresta vw é qualquer círculo que passa através de u e v. A aresta uv está em D se e somente se existe um círculo-circundante de uv. Uma aresta que satisfaz essa propriedade é dita ser Delaunay.

A triangulação de Delaunay contribui para a qualidade da malha final, visto que, dado um conjunto de vértices, maximiza o ângulo mínimo entre todas as maneiras possíveis de triangular aquele conjunto [3].

A triangulação de Delaunay foi escolhida para o processo de geração de malhas no MEMEC.

Uma vez definidos os deslocamentos sofridos nos pontos de contorno e nos pontos interiores, um algoritmo interno no MEMEC gera uma malha para o objeto com seus pontos já deslocados utilizando triangulação de Delaunay.

Na Figura 5, podemos ver a malha gerada para uma viga com as extremidades fixas e tracionada ao centro no sentido negativo de Y.



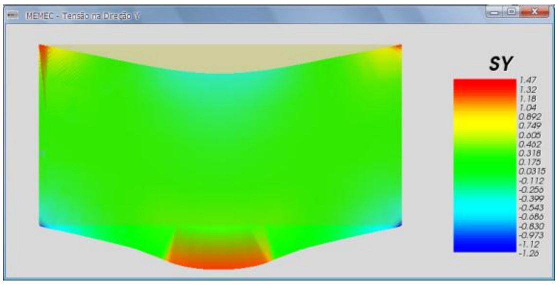


Figura 5: *SciVis*, em *wireframe*, de uma barra com duas extremidades fixas gerada pelo MEMEC com tensão na direção Y.

Figura 7: *SciVis*, em *surface*, de uma barra com duas extremidades fixas gerada pelo MEMEC com tensão na direção Y.

Resultados Prévios

Como já dito, o MEMEC propõe soluções para problemas de Elasticidade em objetos sólidos com geometria bem geral, bastando a geometria estar bem definida dentro das hipóteses do Método dos Elementos de Contorno. Considere por exemplo o problema descrito na Figura 6.

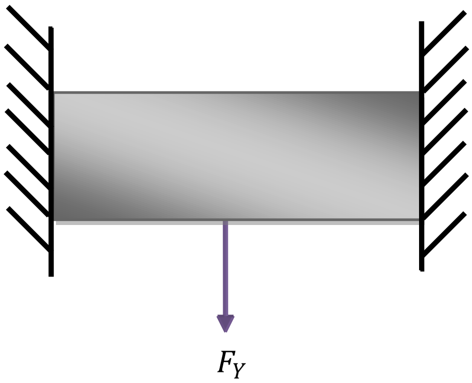


Figura 6: Barra com extremidades fixas e tracionada ao centro.

O estudo da distribuição das forças ao longo do objeto descrito e da deformação causada pelas mesmas constitui o problema cujo MEMEC propõe solucionar. Uma *SciVis* para o problema descrito na Figura 6 pode ser observado na Figura 7.

Dentre as visualizações científicas suportadas pelo MEMEC podemos citar: deslocamentos e tensões nas direções X e Y, deslocamento e tensão equivalente, campo escalar (Figura 8) e curvas de níveis.

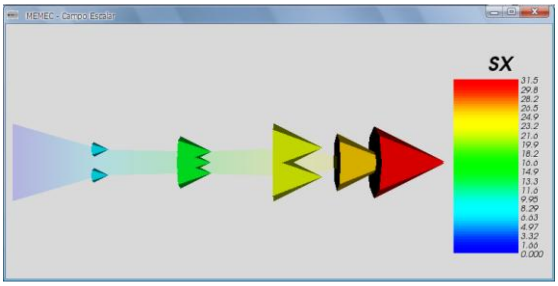


Figura 8: *SciVis*, em campo escalar, da barra discretizada na Figura 2 com tensão na direção X.

Considerações Finais

A visualização por meio do computador apoiada em técnicas de Computação Gráfica tem proporcionado inúmeros benefícios para as mais diversas áreas. Benefícios como o aumento de produtividade e maior eficiência na tomada de decisões baseadas na análise de grandes conjuntos de dados.

O software MEMEC mostra a versatilidade da Computação Gráfica e sua importância como objeto de estudo nos dias atuais. A busca por algoritmos eficientes é constante e sua demanda se torna cada vez maior. Uma das causas é o crescente número de áreas que buscam auxilio na Computação Gráfica para resolução de problemas reais.

A união da Computação Gráfica ao Método dos Elementos de Contorno potencializa o método e sua aplicabilidade.

O MEMEC está em desenvolvimento contínuo e está em sua primeira versão. Pretende-se para o futuro incorporar ao MEMEC rotinas que aperfeiçoem e automatizem o problema de inserção de pontos internos. Tais rotinas retirariam do usuário a tarefa de gerar pontos interiores, a menos que, seja estritamente necessária como no caso de problemas com furos. A construção de pontos interiores mais adequados otimizaria o processo de geração de malhas.

Pretende-se também inserir a variável tempo, isto é, o comportamento do objeto na variação do tempo. Essa tarefa aumenta significativamente a quantidade de dados a serem processados e para isso pretende-se desenvolver rotinas para computação paralela.

Agradecimentos

Este trabalho tem o apoio da FAPERJ e do Fundo Setorial de Infra-Estrutura (CT-INFRA) por intermédio do MCT/CNPq.

Referências

[1] C. A.Brebbia, J. C. F. Telles & L. C. Wrobel, Boundary Elements Techniques: Theory and Application, Berlin, Springer-Verlag, 1984.

[2] H. M. Deitel, P. J. Deitel, Java How to Program (6th Edition), Prentice Hall, 2004.

[3] A. L. Moura, “Uma Proposta para a Triangulação de Delaunay 2D e Localização Planar de Pontos em OCaml", Tese de Doutorado, Universidade Federal de Uberlândia, 2006.

[4] W. Schroeder, M. Ken and B. Lorensen, “The Visualization Toolkit (3rd Edition)” Kitware, Inc. Publishers, 2004.