

6º CONGRESSO BRASILEIRO DE PESQUISA E DESENVOLVIMENTO EM PETRÓLEO E GÁS



TÍTULO DO TRABALHO:

Desenvolvimento de estratégia para geração de modelos granulares com baixa porosidade

AUTORES:

Lucas Pereira de Gouveia

Diogo Tenório Cintra

William Wagner Matos Lira

INSTITUIÇÃO:

Universidade Federal de Alagoas

Este Trabalho foi preparado para apresentação no 6º Congresso Brasileiro de Pesquisa e Desenvolvimento em Petróleo e Gás- 6º PDPETRO, realizado pela a Associação Brasileira de P&D em Petróleo e Gás-ABPG, no período de 09 a 13 de outubro de 2011, em Florianópolis-SC. Esse Trabalho foi selecionado pelo Comitê Científico do evento para apresentação, seguindo as informações contidas no documento submetido pelo(s) autor(es). O conteúdo do Trabalho, como apresentado, não foi revisado pela ABPG. Os organizadores não irão traduzir ou corrigir os textos recebidos. O material conforme, apresentado, não necessariamente reflete as opiniões da Associação Brasileira de P&D em Petróleo e Gás. O(s) autor(es) tem conhecimento e aprovação de que este Trabalho seja publicado nos Anais do 6ºPDPETRO.

DESENVOLVIMENTO DE ESTRATÉGIA PARA GERAÇÃO DE MODELOS GRANULARES COM BAIXA POROSIDADE

Abstract

This work analyses and tries to improve an existing strategy for particle generation for numerical simulations of dense media. The developed strategy can be used in the investigation of engineering problems involving granular media by the Discrete Element Method, for example. Numerical simulations of fragmentation and compacted soil problems are application examples of the applied strategy. In this work, the Levenberg-Marquardt method of function minimization is employed in order to determine the optimized geometric configuration of a given set of particles. This method can become very computationally expensive due the matrix operations that need to be solved. To reduce the computational time of the minimization algorithm, some high performance computing techniques are applied, including an optimized way to handle sparse matrices more compactly. A case of study is presented aiming to demonstrate the computational performance of the developed strategy.

Introdução

O Método dos Elementos Discretos tem sido utilizado para alcançar respostas cada vez significativas em problemas que envolvem meios granulares, requerendo a discretização do domínio através de um conjunto de discos (2D) ou esferas (3D) (ARAÚJO, LAGES e LIRA, 2009). O processo de geração dessas partículas pode ser feito de várias maneiras e diversas estratégias vêm sendo propostas. Algumas dessas estratégias buscam a geração de meios granulares com baixa porosidade onde podem ser aplicados em problemas de fragmentação ou fraturamento.

Na literatura encontram-se, de maneira geral, técnicas dinâmicas e geométricas de geração de partículas. As técnicas dinâmicas consistem em rearranjar as partículas geradas com auxílio de forças externas, como a força da gravidade. Desta forma, as partículas se acomodariam, podendo até serem vibradas, para alcançar um arranjo final mais denso. O problema desse tipo de técnica é o demasiado tempo gasto para se alcançar esse arranjo final, podendo se tornar inviável. As técnicas geométricas se baseiam apenas em um modelo estático, onde podem ser aproveitadas as propriedades de uma malha inicial ou do domínio trabalhado.

Segundo Labra e Oñate (2008), uma configuração geométrica não densa de partículas pode ser obtida através de uma malha triangular, onde cada partícula está circunscrita em um respectivo triângulo ou nó de uma malha. Em outras metodologias a configuração inicial não densa de partículas pode ser obtida através de números randômicos para a posição e raio da partícula. Observe que no primeiro caso existe um maior controle das sobreposições, podendo elas existirem ou não. No segundo caso, as sobreposições devem existir com maior frequência, entretanto, dependendo da estratégia de geração usada, elas podem ser necessárias para obtenção de melhores resultados, como no caso em que a incidência de uma partícula é dada pelas partículas sobrepostas a ela. A geração através de números randômicos também pode ser feita adicionando restrições que impeçam as sobreposições, evitando regiões que já estejam ocupadas por outras partículas.

A estratégia geométrica proposta por Labra e Oñate (2008) consiste em modificar centro e raio de uma partícula em um arranjo inicial, que pode ser qualquer uma das citadas acima. Essa modificação se dá com base no algoritmo de minimização de funções de Levenberg-Marquardt (MAFFRA e GATTAS, 2008), que busca minimizar as distâncias entre as partículas. Esse algoritmo pode se tornar bastante dispendioso computacionalmente para grandes gerações, então, o presente trabalho consiste em aplicar ferramentas da computação de alto desempenho a essa etapa de minimização de funções.

Metodologia

A ideia básica da estratégia usada é minimizar as distâncias entre as partículas até que o método atinja uma configuração densa. Desta forma, inicialmente partimos de uma configuração inicial não densa, gerada a partir de uma malha triangular, como mostra a Figura 1a.

Então, é aplicado o método numérico e iterativo de minimização de funções de Levenberg-Marquardt (MAFFRA e GATTAS, 2008). Em resumo, sua rotina consiste em incrementar ao vetor incógnita X (centro e raios das partículas) um ganho em direção ao valor mínimo das funções:

$$X_{k+1} = X_k + h_k \quad (1)$$

Nesta equação, h_k representa o ganho em direção ao valor mínimo das funções. As funções são as distâncias entre as partículas. O ganho é calculado através de um sistema de equações linear de equações dado por:

$$H \cdot h_k = B \quad (2)$$

onde a matriz de coeficientes do sistema é H , dada pela equação:

$$H = J^T J + \mu I \quad (3)$$

Nesta equação, J é o jacobiano das funções a serem minimizadas, μ um parâmetro de amortecimento e I a matriz identidade. O produto μI serve para garantir que H será positiva definida. O vetor do sistema de equações, B , seria dado por:

$$B = J^T E \quad (4)$$

onde E é um vetor com as funções a serem minimizadas. Uma configuração final da estratégia utilizada pode ser vista na Figura 1b.

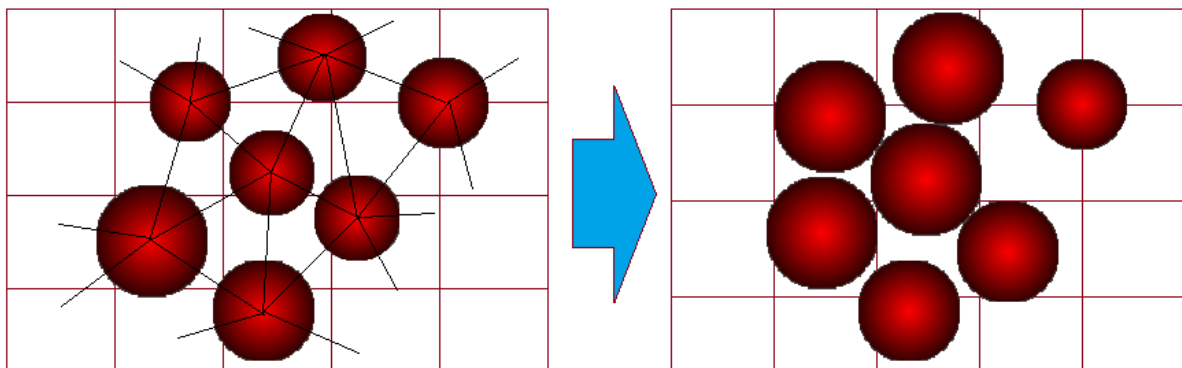


Figura 1 - a) Configuração inicial randômica; b) Configuração final otimizada.

A Figura 1b mostra que a estratégia utilizada permite a variação do centro (coordenadas cartesianas x e y) e do raio da partícula, possibilitando a obtenção de uma configuração em que o módulo da distância entre as partículas é mínimo. Algumas partículas não sofreram alteração, pois a estratégia só irá minimizar as distâncias entre partículas consideradas vizinhas, e, no caso, as partículas que não sofreram alterações não eram consideradas vizinhas da partícula central em análise.

Nessa estratégia são observadas três etapas em que o custo computacional se torna dispendioso:

- A montagem do jacobiano: J ;

- A multiplicação de matrizes: $J^T J$;
- A resolução do sistema linear $H \cdot h_k = B$.

As seções seguintes descrevem algumas observações encontradas após o estudo e implementação computacional do procedimento adotado neste trabalho para minimização das distâncias entre partículas.

- **Montagem do jacobiano**

O jacobiano desse problema de minimização é uma matriz esparsa. Isso ocorre porque o termo i do vetor E é composto pelo somatório das distâncias entre a partícula i e as suas vizinhas. A vizinhança de uma partícula pode ser determinada por um método de incidência por visibilidade com auxílio de um *grid*, onde se as partículas estão localizadas em células adjacentes do *grid*, elas são partículas vizinhas entre si (Figura 1a). Outros métodos de detecção de incidência de uma partícula podem ser encontrados, como o método de incidências por contatos, ou seja, se uma partícula está em contato com outra elas são vizinhas.

Desta forma, para a montagem do jacobiano foram adotadas as seguintes estratégias:

- ✓ Usar *containers* da biblioteca STL do C++;
- ✓ Usar *arrays* do formato compactado de matrizes esparsas CSR;

A biblioteca STL do C++ foi testada inicialmente, pois ela facilitaria muito a montagem do jacobiano, guardando de maneira estratégica os termos não-nulos do jacobiano para que as seguintes operações de multiplicação e resolução do sistema linear fossem feitas de forma mais eficiente. Os *containers* usados foram o *vector* e o *map*.

Os *arrays* do CSR (Compressed Sparse Row) são três: *values* e *columns*, de tamanho igual ao número de elementos não-nulos, e, *rowIndex*, de tamanho igual ao número de linhas da matriz esparsa. Esses *arrays* são capazes de armazenar os elementos não-nulos da matriz, resumindo a matriz esparsa com menos memória.

- **Multiplicação de matrizes**

O produto $J^T J$ pode ser bastante dispendioso para uma grande quantidade de partículas a serem geradas. Entretanto, como essa multiplicação envolve matrizes esparsas, são propostas duas estratégias de multiplicação mais eficientes para esse tipo de produto:

- ✓ Empregando *containers* da STL;
- ✓ Empregando *arrays* CSR, com o uso da Intel® MKL.

Vale ressaltar que a matriz resultante da multiplicação com o CSR seja dada como *arrays* do CSR, e, quando forem usados os *containers* da STL a matriz resultante seja dada como *containers* também.

- **Resolução do sistema linear**

A resolução desse sistema linear é proposta através de apenas uma estratégia.

- ✓ Usando uma função de resolução direta do sistema de equações linear da biblioteca Intel® MKL.

No caso, a função da MKL que resolve o sistema linear recebe como parâmetros *arrays* no formato CSR, portanto, se estivermos trabalhando com os *containers* da STL será necessário fazer uma conversão desses *containers* para os *arrays* CSR.

Resultados e Discussão

Os problemas analisados partiram de configurações iniciais não-densas geradas a partir de uma malha triangular, e, em seguida, foi aplicada a estratégia de Labra e Oñate (2008). Os testes foram feitos em uma máquina com processador Intel® Core™ i3 de 2,93GHz. Os resultados mostrados a seguir referem-se a um estudo de *profile* dos problemas analisados, ou seja, eles demonstram quanto tempo levou as etapas críticas apresentadas na metodologia do trabalho para serem resolvidas. Além disso, essas tomadas de tempo referem-se apenas a etapa de otimização do modelo, onde o arranjo inicial de partículas já foi criado.

• Resultados com *containers* da STL

Ao utilizar essas ferramentas para aplicação da estratégia observou-se uma maior facilidade de trabalhar com os elementos das matrizes esparsas. O acesso a cada elemento era facilitado, pois os elementos não-nulos eram mapeados de forma estratégica. A seguir, na Tabela 1, são mostrados os tempos de cada operação apresentadas na metodologia, que utilizam os *containers* da STL para serem realizadas.

Tabela 1 - Tempos de processamento das principais operações da estratégia usando *containers* da STL.

Particles	Mount J and J'	Multiply	Linear System	Total
11500	79,902s	17,409s	5,850s	103,161s
14208	131,855s	23,414s	8,257s	163,526s
18000	194,672s	29,609s	11,029s	235,310s

A operação que demanda maior tempo nesse caso seria a montagem do jacobiano. Isso ocorre por que a utilização dos *containers* da STL não é muito eficiente quanto à montagem de grandes matrizes, pois muitas vezes eles se realocam internamente demorando mais para concluir a montagem total da matriz esparsa. Por outro lado, o acesso aos elementos da matriz esparsa acaba sendo mais rápido, pois esses *containers* são bastante eficientes para esse tipo de operação. No caso, o tempo do sistema linear mostrado na Tabela 1 é referente a resolução direta do sistema linear usando a Intel® MKL (ver próxima seção) mais o tempo de conversão dos *containers* da STL para os *arrays* CSR.

• Resultados com uso da Intel® MKL

A Intel® Math Kernel Library (MKL) possui um conjunto de rotinas específicas para resolver problemas que envolvem matrizes esparsas. Dentre as opções existentes foram escolhidas uma rotina que resolve diretamente sistemas lineares de equações, onde a matriz de coeficientes é esparsa, e, uma rotina que multiplica duas matrizes esparsas. As duas rotinas têm as matrizes parâmetros de entrada e saída no formato CSR (*Compressed Sparse Row*). Desta forma, a Tabela 2 mostra os resultados obtidos com o uso dessa biblioteca.

Tabela 2 - Tempos de processamento das principais operações da estratégia usando *arrays* CSR.

Particles	Mount J and J'	Multiply	Linear System	Solve
11500	0,082s	0,524s	3,856s	4,462s
14208	0,105s	0,707s	5,261s	6,073s
18000	0,135s	0,888s	6,848s	7,871s
57183	0,431s	2,799s	26,067s	29,297s

A Tabela 2 comparada a Tabela 1 mostra uma grande diferença de tempo de processamento quando se muda a forma de resolver o problema. Inclusive, é possível extrapolar a otimização para um maior número de partículas, como mostra a última linha da Tabela 2.

Conclusões

Neste trabalho ficou apresentada uma estratégia para geração de meios particulados mais densos, que possuem grande importância em simulações numéricas que envolvem fragmentação de rochas ou solos compactados, por exemplo. A estratégia proposta se torna mais viável computacionalmente utilizando o formato de armazenamento de matrizes esparsas através de *arrays* do CSR.

É importante a continuação desse estudo para que se extrapolem as otimizações mostrados neste trabalho. Para tanto, pode ser proposta uma alternativa à operação de resolver sistema linear, que foi a que mais demandou tempo, de acordo com a Tabela 2. Além disso, outras operações implícitas na estratégia e não mencionadas nesse trabalho também podem ser mais trabalhadas a fim de reduzir o tempo computacional da estratégia como um todo, tais como método de atualização de incidências, montagem do vetor B (ver seção Metodologia), verificação dos critérios de convergência, entre outros. O uso da programação paralela também pode ser explorado nessa estratégia, onde já foi observado que alguns trechos possuem a possibilidade de serem paralelizados.

Agradecimentos

Ao Laboratório de Computação Científica e Visualização e ao Centro de Tecnologia da UFAL por auxiliarem no desenvolvimento da pesquisa fornecendo a infraestrutura necessária.

Ao apoio financeiro da Agência Nacional de Petróleo, Gás Natural e Biocombustíveis (ANP); da Financiadora de Estudos e Projetos (FINEP) e do Ministério da Ciência e Tecnologia (MCT) por meio do Programa de Recursos Humanos da ANP para o Setor Petróleo e Gás (PRH-ANP/MCT).

Referências Bibliográficas

ARAÚJO, C. N.; LAGES, E. N.; LIRA, W. W. M. Uma estratégia híbrida para a geração de sistemas particulados bidimensionais. **XXX CILAMCE**, Búzios, 2009.

FACCIO, C. J.; FRERY, A. C.; MATOS LIRA, W. W. Estratégia para distribuição espacial de discos utilizando processos pontuais. **XXXI CILAMCE**, Buenos Aires, 2010.

LABRA, C.; OÑATE, E. **HIGH-DENSITY SPHERE PACKING TO DISCRETE ELEMENT METHOD SIMULATIONS**. [S.l.]: [s.n.], 2008.

MAFFRA, F. A.; GATTAS, M. **Uma implementação do método de Levenberg-Marquardt em C**. Rio de Janeiro: Tec-Graf, PUC-Rio, 2008.