

# Algoritmos de Detecção de Contato: Avaliação de Desempenho em Diferentes Cenários

Marcos Antônio Pereira da Silva Júnior<sup>1,2</sup>, Bruno Normande Lins<sup>1,2</sup>,  
Leonardo Viana Pereira<sup>1,2</sup>

<sup>1</sup> Laboratório de Computação Científica e Análise Numérica (LaCCAN)  
Instituto de Computação (IC) – Universidade Federal de Alagoas (UFAL)  
Maceió – AL – Brasil

<sup>2</sup>Laboratório de Computação Científica e Visualização (LCCV)  
Universidade Federal de Alagoas (UFAL)  
Maceió – AL – Brasil

{marcos\_antonio, bruno\_normande, lpviana}@lccv.ufal.br

## 1. Introdução

As formulações físicas e matemáticas desenvolvidas ao longo da história são ferramentas utilizadas para descrever e prever os comportamentos de mecanismos regidos pelas leis da natureza. Muitos desses fenômenos podem ser descritos em termos de equações diferenciais. Entretanto, em geral, solucionar essas equações por meio de métodos analíticos clássicos para geometrias arbitrárias é quase impossível. Sendo assim, torna-se necessário o uso de técnicas de aproximação. Um dos métodos frequentemente utilizados pela comunidade científica são os métodos numéricos, cujo uso tem sido bastante difundido principalmente pelo aumento dos recursos computacionais.

Porém, mesmo com os avanços alcançados nos métodos numéricos e na computação, algumas técnicas encontram gargalos que podem inviabilizar seu uso. Alguns desses casos estão presentes em três métodos: Método dos Elementos Discretos (MED), Método dos Elementos Finitos (MEF) e a combinação de ambos (MED-MEF).

O Método dos Elementos Discretos é uma técnica de modelagem numérica utilizada na solução de problemas que podem ser representados como um conjunto de corpos ou partículas discretas. O Método dos Elementos Finitos é uma técnica numérica de resolução de equações diferenciais parciais. Já combinação desses métodos é destinada a solucionar problemas que envolvam a dinâmica transitória de sistemas que incluem um grande número de corpos deformáveis que interagem uns com os outros e que podem, em geral, sofrer fragmentações ou fraturas.

Nessas três técnicas a detecção de contato se apresenta como um dos principais gargalos, podendo chegar a mais de 60% do tempo de computação [Munjiza 2004] e, no caso de partículas tridimensionais e irregulares, facilmente ultrapassa os 80% [Nezami et al. 2004]. Sendo assim, o presente trabalho visa analisar alguns dos principais algoritmos de detecção de contato presentes na literatura a fim de caracterizá-los quanto ao desempenho em diversos cenários.

## 2. Algoritmos de Detecção de Contato

A detecção do contato destina-se a encontrar pares de elementos discretos que estão próximos uns dos outros, ou seja, esta etapa visa eliminar pares de elemen-

tos que não irão interagir em determinado momento da computação do problema. O principal objetivo da detecção é evitar o processamento da interação do contato (etapa seguinte a detecção) quando este não existe [Munjiza 2004, Hogue 1998, Perkins and Williams 2001]. Estes algoritmos usualmente são divididos em duas classes baseadas no tipo de discretização espacial empregada, os de árvore hierárquica e os de grade. Neste trabalho, trataremos apenas os algoritmos baseados em grid.

## 2.1. Algoritmos Baseados em Grid

### Mapeamento Direto

O algoritmo de Mapeamento Direto utiliza como base mapeamentos espaciais, dividindo o espaço em células de mesmo tamanho. Esse tamanho é escolhido de forma que o maior dos elementos discretos seja contido pelas células, ou seja, o lado de um célula é igual ao diâmetro do círculo que circunscreve o maior elemento [Munjiza 2004]. Elementos mapeados na mesma célula ou em células vizinha estes podem estar em contato. Sua complexidade computacional é linear.

### Mapeamento por Célula

O algoritmo de Mapeamento por Células surgiu como uma metodologia alternativa para algoritmo de Mapeamento Direto devido à perda de eficiência observada nos casos em que a diferença entre as dimensões das partículas cresce [Munjiza 2004, Dem 2009]. Nesse algoritmo, a dimensão da célula é igual à dimensão da menor partícula. Verifica-se células que contêm duas ou mais partículas.

### Detecção de Contato por Ordenação

Por sua vez, a Detecção de Contatos por Ordenação (*Sorting Contact Detection*) [Munjiza 2004] diminui o total de memória gasto nos métodos anteriores usando vetores com a localização dos elementos, no lugar da representação matricial do grid. Apesar de uma menor necessidade de memória, tal algoritmo é restrito à complexidade dos algoritmos de ordenação e de busca, e por isso sua complexidade é  $O(N \log N)$

## 3. Metodologia

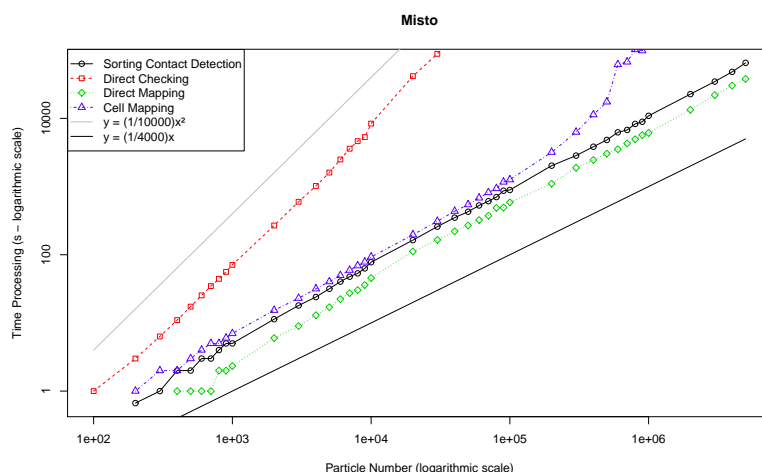
Neste trabalho pretendemos analisar e validar os algoritmos implementados no DEMOOP (Discrete Element Method Object Oriented Programming), um software desenvolvido pelo Laboratório de Computação Científica e Visualização (LCCV) para simulação de partículas e utilizado pela indústria do petróleo. Utilizamos o *cluster* computacional GradeBR/UFAL da Rede Galileu.

Cada simulação foi executada usando 8 *threads* OpenMP em um nó com dois processadores *Intel Xeon Quad-Core X5560* com 48 GB de memória RAM. Nossas simulações utilizam partículas esféricas com tamanhos próximos e confinadas em uma caixa de lados paralelos e submetidos a gravidade (campo de força constante na direção vertical). A distância entre as partículas são medidas em unidades arbitrárias (UA).

Algumas comparações de desempenho de diferentes algoritmos de detecção de contato estão presentes na literatura [Han et al. 2007]. Medimos o tempo de processamento da CPU em simulações de um sistema fechado de  $N$  elementos. Nosso objetivo é medir o efeito de diferentes densidades de partículas sobre o desempenho dos algoritmos estudados.

## 4. Resultados

Três cenários com diferentes “densidades” foram ajustados para testar os limites de algumas situações realistas: espaço denso, espaço esparsos e espaço misto. No primeiro cenário mostramos a simulação do espaço denso, começando com uma rede densa quase regular que lentamente se transforma em um ambiente compacto, devido a ação da força gravitacional. No segundo cenário, mostramos a simulação do espaço esparsos, começando com uma configuração quase regular de partículas soltas que se espalham dentro da caixa devido a gravidade. Por fim, é testado o espaço misto, começando com uma rede densa quase regular que desliza em um plano inclinado.



**Figura 1. Medição da complexidade algorítmica da busca de vizinhos baseada em grid espacial no cenário misto, onde a densidade de partículas é variável. As curvas  $y = k_1x^2$  e  $k_2y = x$ , onde  $k_1$  e  $k_2$  são constantes, servem apenas como apoio visual.**

Na figura 4 mostramos parte dos resultados obtidos até o momento. É apresentado a avaliação no cenário misto, onde a densidade de partículas é variável.

## 5. Conclusões

Estudamos o comportamento de algoritmos de busca de vizinhos em função da densidade de partículas esféricas de raio constante em uma caixa. O tempo de execução destes algoritmos varia de acordo com esta densidade. Devemos implementar algoritmos de árvore hierárquica e medir seu comportamento. A partir destes resultados iremos sugerir o melhor algoritmo de busca de contato baseado na distribuição de partículas ou o emprego algoritmos mistos/adaptativos para otimizar esta tarefa. Esta classe de algoritmos é importante em várias aplicações, de simulações numéricas utilizando DEM ou FEM a computação gráfica e jogos.

## Referências

- (2009). *Demoop - Manual Teórico*. LCCV.
- Han, K., Feng, Y. T., and Owen, D. R. J. (2007). Performance comparisons of tree-based and cell-based contact detection algorithms. *Engineering Computations*, 24(1-2):165–181.
- Hogue, C. (1998). Shape representation and contact detection for discrete element simulations of arbitrary geometries. *Engineering Computations*, 15(2-3):374+.
- Munjiza, A. (2004). *The Combined Finite-Discrete Element Method*. John Wiley & Sons.
- Nezami, E., Hashash, Y., Zhao, D., and Ghaboussi, J. (2004). A fast contact detection algorithm for 3-d discrete element method. *Computers and Geotechnics*, 31(7):575 – 587.
- Perkins, E. and Williams, J. (2001). A fast contact detection algorithm insensitive to object sizes. *Engineering Computations*, 18(1-2):48–61.