

**SIMULAÇÃO DO ENSAIO DE ÂNGULO DE REPOUSO VIA
MÉTODO DOS ELEMENTOS DISCRETOS**

MATHEUS AMARAL MÕES

MONOGRAFIA DE PROJETO FINAL EM ENGENHARIA CIVIL

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL

BRASÍLIA / DF DEZEMBRO 2018

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL

**SIMULAÇÃO DO ENSAIO DE ÂNGULO DE REPOUSO VIA
MÉTODO DOS ELEMENTOS DISCRETOS**

MATHEUS AMARAL MÔES

ORIENTADOR: MÁRCIO MUNIZ DE FARIAS, Ph.D.

MONOGRAFIA DE PROJETO FINAL EM ENGENHARIA CIVIL

BRASÍLIA / DF: DEZEMBRO-2018

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL

**SIMULAÇÃO DO ENSAIO DE ÂNGULO DE REPOUSO VIA
MÉTODO DOS ELEMENTOS DISCRETOS**

MATHEUS AMARAL MÔES

MONOGRAFIA DE PROJETO FINAL SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL DA UNIVERSIDADE DE BRASÍLIA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE BACHAREL EM ENGENHARIA CIVIL.

APROVADA POR:

Márcio Muniz de Farias, Ph.D. (UnB)

(Orientador)

Manoel Porfírio Cordão Neto, Ph.D. (UnB)

(Examinador interno)

Carlos Alexander Recarey Morfa, DSc. (UCLV-Cuba)

(Examinador externo)

BRASÍLIA/DF, 06 DE DEZEMBRO DE 2018

FICHA CATALOGRÁFICA

MÕES, MATHEUS AMARAL

Análise do método dos elementos discretos para determinação numérica do atrito em materiais granulares [Distrito Federal] 2018.

xxv, 69p., 297 mm (ENC/FT/UnB, Bacharel, Engenharia Civil, 2018)

Monografia de Projeto Final - Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Civil e Ambiental.

- | | |
|---------------------|------------------------|
| 1. Micromecânica | 2. Elementos discretos |
| 3. Ângulo de atrito | 4. Ângulo de repouso |
| I. ENC/FT/UnB | II. Título (Bacharel) |

REFERÊNCIA BIBLIOGRÁFICA

MÕES, M.A. (2018). Simulação do Ensaio de Ângulo de Repouso via Método Dos Elementos Discretos. Monografia de Projeto Final em Engenharia Civil, Departamento de Engenharia Civil e Ambiental, Universidade de Brasília, Brasília, DF, 69p.

CESSÃO DE DIREITOS

AUTOR: Matheus Amaral Mões.

TÍTULO: Simulação do Ensaio de Ângulo de Repouso via Método Dos Elementos Discretos

GRAU: Bacharel em Engenharia Civil ANO: 2018

É concedida à Universidade de Brasília a permissão para reproduzir cópias desta monografia de Projeto Final e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta monografia de Projeto Final pode ser reproduzida sem a autorização por escrito do autor.

Matheus Amaral Mões

e-mail: moesmatheus@gmail.com

SUMÁRIO

1	INTRODUÇÃO	1
1.1	IDENTIFICAÇÃO DO PROBLEMA	2
1.2	HIPÓTESE	2
1.3	OBJETIVO	3
1.4	ESCOPO DO TRABALHO	3
2	REVISÃO BIBLIOGRÁFICA	4
2.1	HISTÓRICO DO MÉTODO DOS ELEMENTOS DISCRETOS (DEM)	4
2.2	PERSPECTIVAS DO MÉTODO DOS ELEMENTOS DISCRETOS (DEM)	5
3	FUNDAMENTAÇÃO TEÓRICA DO DEM	6
3.1	EMPACOTAMENTO	7
3.1.1	MÉTODOS DINÂMICOS	7
3.1.2	MÉTODOS CONSTRUTIVOS	10
3.2	MOVIMENTO DAS PARTÍCULAS	12
3.3	FORÇAS DE CONTATO	17
3.4	BUSCA DE CONTATOS	22
4	ENSAIOS EXPERIMENTAIS	25
4.1	ÂNGULO DE REPOUSO	25
5	METODOLOGIA	27
5.1	SOFTWARE ADOTADO	27
5.2	ESTRUTURA DA SIMULAÇÃO	28

5.3	DEFINIÇÃO DOS PARÂMETROS DA SIMULAÇÃO	32
5.3.1	MODELO CONSTITUTIVO DE CONTATO	32
5.3.2	MODELO DE INTEGRAÇÃO	33
5.3.3	FORMATO DAS PARTÍCULAS	33
5.3.4	CONDIÇÕES DE CONTORNO INICIAIS	33
5.4	PARÂMETROS A SEREM ANALISADOS	35
5.5	PROCESSAMENTO DAS SIMULAÇÕES	36
6	APRESENTAÇÃO E ANÁLISE DOS RESULTADOS	40
7	CONCLUSÕES E SUGESTÕES	50

ÍNDICE DE TABELAS

TABELA 5.1 PROPRIEDADES FÍSICAS DE VIDRO POTTERS	27
TABELA 5.2 PROPRIEDADES ADOTADAS NO MODELO CONSTITUTIVO DE CONTATO	32
TABELA 5.3 CONDIÇÕES INICIAIS DE CONTORNO	34
TABELA 6.1 EMPACOTAMENTO CRIADO	40
TABELA 6.2 TEMPO DE PROCESSAMENTO E NÚMERO DE ITERAÇÕES	42
TABELA 6.3 – ÂNGULOS DE REPOUSO OBTIDOS	44
TABELA 6.4 – ÂNGULOS DE REPOUSO OBTIDOS	45

ÍNDICE DE FIGURAS

FIGURA 3.1 – PASSOS DO MÉTODO DOS ELEMENTOS DISCRETOS	7
FIGURA 3.2 – MÉTODO DA EXPANSÃO DE RAIOS	8
FIGURA 3.3 – COMPACTAÇÃO DE CAMADAS DE JIANG	9
FIGURA 3.4 – MÉTODO DE FRENTE DE AVANÇO NO FORMATO ABERTO	10
FIGURA 3.5 – MÉTODO DE FRENTE DE AVANÇO NO FORMATO FECHADO	10
FIGURA 3.6 – MÉTODO DE TRIANGULAÇÃO	11
FIGURA 3.7 – SOBREPOSIÇÃO NA PARTÍCULA	15
FIGURA 3.8 – SIMULAÇÃO COM PASSO DE TEMPO 0,01S	16
FIGURA 3.9 – SIMULAÇÃO COM PASSO DE TEMPO 0,012S	16
FIGURA 3.10 – SIMULAÇÃO COM PASSO DE TEMPO 0,02S	16
FIGURA 3.11 – SOBREPOSIÇÃO DE PARTÍCULAS	18
FIGURA 3.12 – RELAÇÃO TENSÃO DEFORMAÇÃO NO MODELO WALTON-BRAUN	20
FIGURA 3.13 – TENSÃO DEFORMAÇÃO NO MODELO MOLLA AMORTECEDOR	22
FIGURA 3.14 – MODELO DE MALHA	24
FIGURA 4.1 - ÂNGULO DE REPOUSO	25
FIGURA 5.1 – CONDIÇÕES DE CONTORNO DA SIMULAÇÃO	29
FIGURA 5.2 – EMPACOTAMENTO	29
FIGURA 5.3 – MOVIMENTO VERTICAL DO CONE	30
FIGURA 5.4 – PARTÍCULAS EM MOVIMENTO E EM EQUILÍBRIO	31
FIGURA 5.5 – ÂNGULO DE REPOUSO	31
FIGURA 5.6 – CURVAS GRANULOMÉTRICAS UTILIZADAS	35
FIGURA 5.7 – CURVAS GRANULOMÉTRICAS UTILIZADAS	36
FIGURA 6.1 – CURVAS GRANULOMÉTRICAS OBTIDAS E ESPECIFICADAS	41
FIGURA 6.2 – CURVAS GRANULOMÉTRICAS OBTIDAS E ESPECIFICADAS	42
FIGURA 6.3 – POSIÇÃO FINAL DAS PARTÍCULAS	43
FIGURA 6.4 – ALTURAS MÁXIMAS	44
FIGURA 6.5 – ALTURAS MÁXIMAS	45
FIGURA 6.6 – ÂNGULO DE REPOUSO	46
FIGURA 6.7 – TENSÕES NORMAL NAS PARTÍCULAS (PA)	47
FIGURA 6.8 – VELOCIDADE DAS PARTÍCULAS (M/S)	48
FIGURA 6.9 – TENSÃO CISALHANTE (PA)	49

LISTA DE ABREVIATURAS E SÍMBOLOS

ABREVIATURAS

DEM *Discrete Element Method*

SÍMBOLOS

ΔF	Incremento de força
m	Massa
U'	Velocidade
U''	Aceleração
C	Coefficiente de perda de energia
$K(U)$	Rígidez
I_p	Inércia
ω_p	Velocidade de rotacional
ω'_p	Aceleração rotacional
M_p	Momento atuante
a_p	Aceleração da partícula
v_p	Velocidade da partícula
x_p	Posição da partícula
Δt	Incremento de tempo
δ	Distância entre partículas
δ'	Varição na distância entre partículas
δ'_t	Velocidade de deslocamento tangencial entre partículas
F_n	Força normal entre partículas
F_t	Força tangencial entre partículas
k	Rigidez no contato
G	Modulo de elasticidade
R	Raio da partícula
ν	Coefficiente de poisson

1 INTRODUÇÃO

Existem atualmente duas famílias principais de métodos para simulações numéricas na geotecnia, os métodos baseados na mecânica de meios contínuos e os métodos baseados na mecânica de meios descontínuos. O ramo mais utilizado é o dos meios contínuos, representado principalmente pelo método dos elementos finitos. Uma das grandes vantagens deste tipo de método é a velocidade na execução de análise.

Os métodos de meios contínuos, no entanto, apresentam discrepâncias em alguns aspectos, principalmente no que diz respeito à escala micro dos materiais. Esse tipo de método falha em adequar-se a variações da escala micro, como a granulometria, forma e distribuição de materiais granulares. Além disso esse tipo de método encontra dificuldade em simular grandes deformações e faturamento do material, como observa Donzé *et al.* (2009).

Da dificuldade apresentada pelos métodos de meio contínuo em abranger certas situações surge a necessidade de explorar novas alternativas. Os métodos de meio descontínuos, como o próprio método dos elementos discretos (DEM), surgem como uma alternativa para fazer uma transição e compreender melhor os problemas geotécnicos na escala micro.

O método dos elementos discretos surgiu com a publicação de Cundall e Strack (1971) e inicialmente a sua capacidade de processamento impunha um grande obstáculo na aplicação do método. Recentemente, no entanto, com o aumento da capacidade de processamento e a otimização dos algoritmos, a aplicação do método se tornou mais viável.

A crescente aplicabilidade do DEM aliada à crescente facilidade em obter os parâmetros do solo em microescala, com o avanço da microscopia, por exemplo, torna o método cada vez mais difundido. Segundo O'Sullivan (2008), o DEM se estabeleceu como uma das principais ferramentas para complementar as análises de meios contínuos e suprir as falhas desses métodos.

1.1 IDENTIFICAÇÃO DO PROBLEMA

A motivação do presente trabalho surge da incapacidade dos métodos numéricos baseados em mecânica do meio contínuo, o mais famoso deles o método dos elementos finitos, de simular problemas geotécnicos de modo adequado em uma escala micro.

Os métodos baseados em mecânica de meios contínuos são de grande valia para a geotécnica devido a qualidade e velocidade com que são geradas as análises, principalmente na escala macro. Características da escala micro, no entanto, são inseridas nestes métodos como ajustes matemáticos de modo a simular o comportamento via modelos constitutivos fenomenológicos, ou seja, baseados em observações na escala macroscópica.

Desse modo esses métodos não são de grande valia para compreender o que ocorre diretamente no grão e como essas mudanças interferem no comportamento final do solo. Além disso, não é possível simular como mudanças na microescala do solo podem interferir no seu comportamento.

1.2 HIPÓTESE

A hipótese do presente trabalho é que o método dos elementos discretos é um método adequado para estudar o comportamento do solo em uma escala micro, especialmente na simulação do ensaio de ângulo de repouso.

O método permite a variação adequada dos parâmetros da escala micro que dependem diretamente do grão. Como exemplo desses parâmetros tem-se o ângulo de atrito dos grãos de solo, a distribuição granulométrica e o índice de vazios. Desse modo é possível simular numericamente o comportamento do solo em função dos parâmetros obtidos no grão de modo a complementar outros métodos numéricos utilizados atualmente.

1.3 OBJETIVO

O objetivo do presente trabalho é verificar a validade do método dos elementos discretos (DEM) como uma alternativa para simular o comportamento do solo a partir de sua microescala, em especial no que concerne ensaio de ângulo de repouso. Para realizar essa validação do método serão feitas simulações numéricas utilizando-o e variando os parâmetros do solo.

A variação dos parâmetros será observada com relação ao atrito no solo, simulando o ensaio de ângulo de repouso, o qual será realizado com diferentes curvas granulométricas do material.

Os resultados obtidos serão comparados com os resultados experimentais observados por Girardi (2016), com a finalidade de observar a confiabilidade das simulações numéricas.

1.4 ESCOPO DO TRABALHO

O presente trabalho está dividido em 7 Capítulos. No presente Capítulo 1 é feita uma breve introdução dos assuntos tratados no trabalho. No Capítulo 2 é feita uma breve revisão bibliográfica abordando alguns fundamentos teóricos e trabalhos relevantes no assunto. No Capítulo 3 são mostradas as bases teóricas do método dos elementos discretos e sua implementação. No Capítulo 4 são mostrados aspectos do ensaio de ângulo de repouso a ser simulado numericamente. No Capítulo 5 é descrita a metodologia adotada no trabalho, com parâmetros e procedimentos adotados na simulação. No Capítulo 6 são mostrados os resultados obtidos e uma análise dos resultados. Por fim, no Capítulo 7 são mostradas conclusões e sugestões do trabalho realizado.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo será apresentado um breve histórico do método dos elementos discretos, partindo dos desafios que levaram ao desenvolvimento do método até suas primeiras implementações. Em seguida serão mostradas algumas perspectivas em relação ao método e suas aplicações.

2.1 HISTÓRICO DO MÉTODO DOS ELEMENTOS DISCRETOS (DEM)

O método dos elementos discretos (DEM) é uma técnica de simulação numérica que utiliza meios particulados. O método teve seu início no estudo do comportamento de rochas, principalmente na dificuldade proposta pelo estudo de conjuntos rochosos com fraturas.

Inicialmente esse tipo de estudo era feito com a utilização de métodos baseados em meios contínuos, como o método dos elementos finitos. Foi na dificuldade de se modelar meios fraturados e descontínuos por esse tipo de método que surgiu a necessidade de propor novas soluções.

Goodman, Taylor e Brekke (1968) deram início às bases do método dos elementos discretos buscando desenvolver um modelo para simular juntas em maciços rochosos. Foi proposto o desenvolvimento de um elemento de ligação para a junta da rocha com a finalidade de modelar o sistema. Por meio de estudos eles conseguiram demonstrar a influência entre o tipo de ligação na falha e as tensões, deslocamentos e padrões de falha.

O método foi desenvolvido como um acréscimo para modelagens utilizando o método dos elementos finitos (FEM). Desse modo a modelagem poderia ser feita por meio de um método de elementos contínuos como o método dos elementos finitos e as juntas eram modeladas a partir do método de ligação proposto por Goodman (1968), gerando um conjunto de blocos e ligações.

Essa nova proposta passou a exigir um novo modelo simulação para juntas. Desse modo passa a ser necessária a categorização da junta de acordo com sua rigidez normal e tangencial, utilizando a tensão cisalhante como variável na modelagem da junta. A modelagem proposta por Goodman deu origem ao método dos elementos distintos

(Distinct Element Method). Posteriormente Cundall e Strack (1971) deram continuidade ao trabalho desenvolvido por Goodman (1968).

Cundall e Strack (1979) estabeleceram as bases do método e esse é considerado o seu marco inicial. Cundall e Strack (1979) buscaram validar o método desenvolvendo um programa chamado BALL. O programa criado gerava vetores de forças resultantes da interação entre partículas. Posteriormente o resultado do programa foi comparado com o resultado de uma análise fotoelástica, chegando à validação do método desenvolvido.

2.2 PERSPECTIVAS DO MÉTODO DOS ELEMENTOS DISCRETOS (DEM)

O grande avanço na capacidade computacional recentemente tornou o método dos elementos discretos mais aplicável. Segundo Sullivan (2011) o método não deve substituir experimentos físicos como modo de compreender as respostas do solo.

Para Sullivan (2011) o método se estabeleceu como uma das ferramentas disponíveis para engenheiros geotécnicos. Entretanto ainda é necessária uma aplicação do método em uma gama maior de problemas e também maiores desenvolvimentos nos algoritmos utilizados. Esse desenvolvimento é necessário para atingir uma boa confiabilidade do método e uma boa aplicabilidade, segundo a autora.

Sullivan (2011) complementa que o método deve continuar sendo uma importante ferramenta na compreensão da micromecânica do solo de modo a auxiliar no avanço de modelos utilizando métodos de meios contínuos.

Segundo Simpson e Tatsuoka (2008) e Yu (2004), o desenvolvimento do método necessita dos seguintes pontos: (I) Mais estudos paramétricos para compreender melhor a influência dos parâmetros sobre as respostas observadas; (II) Modelos de constitutivos de contatos e morfologias de partículas mais realísticas; (III) Número maior de partículas nas simulações (IV) Modelos mais completos e algoritmos mais eficientes; (V) Melhor quantificação das forças inter-partículas e entre partículas e fluido; (VI) Teorias melhores estimando parâmetros constitutivos; (VII) Desenvolvimentos na ligação entre partícula e fluido e simulação de fluidos multifásicos.

3 FUNDAMENTAÇÃO TEÓRICA DO DEM

O método dos elementos discretos (DEM, *Discrete Element Method*, em inglês) usualmente passa pelas etapas descritas a seguir. Inicialmente deve ser criado o empacotamento de grãos com a definição da geometria das partículas e sua posição dentro do domínio de interesse do problema. Em seguida deve ser estabelecido um modelo constitutivo com a finalidade de obter as forças geradas pela interação entre partículas no problema.

Posteriormente deve ser definido um procedimento para definir quais partículas estão em contato entre si e um método para otimizar o processo de busca de contatos. O número de potenciais contatos é muito grande e por isso é inviável testar todas as possibilidades.

Uma vez determinado o esforço entre partículas, esse valor deve ser utilizado para determinar o movimento entre as partículas. O movimento ocorre por meio de translação e rotação das partículas e é definido por meio da mecânica Newtoniana para partículas.

O movimento de partículas é calculado para um intervalo de tempo incremental de modo, com intervalos de tempo suficiente pequenos para evitar a propagação de erros localizados. Uma vez calculado o movimento para um intervalo incremental de tempo suficientemente pequeno retorna-se para a etapa de identificação de contatos, depois para o cálculo dos esforços gerados e novamente para a determinação dos movimentos.

O processo é repetido até atingir o tempo desejado e posteriormente os dados são analisados. A seguir serão mostradas com maior detalhe cada uma das etapas envolvidas no processo usual do método dos elementos discretos. A Figura 3.1 ilustra as etapas envolvidas no método.

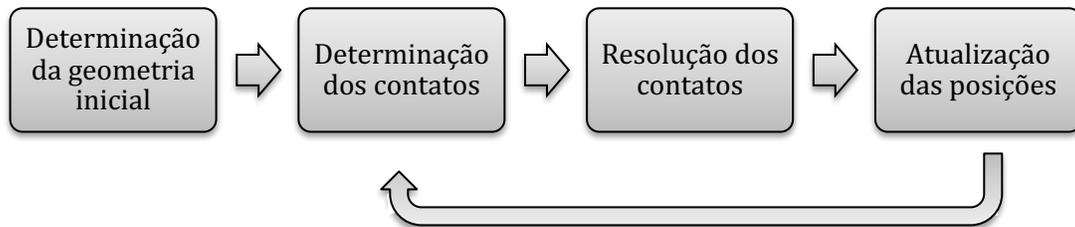


Figura 3.1 – Passos do método dos elementos discretos

3.1 EMPACOTAMENTO

A definição das condições iniciais do problema no método dos elementos discretos é uma importante etapa do processo. Diferentemente de outros métodos, a criação das condições iniciais pode ser dispendiosa tanto do ponto de vista de execução de código quanto do ponto de vista de processamento.

Bagi (2005) define duas principais famílias de métodos para se obter as condições iniciais das partículas, os métodos dinâmicos e construtivos. Nos métodos construtivos a geometria é calculada sem a utilização de etapas de cálculo do método ao passo que nos métodos dinâmicos ao menos uma das etapas de cálculo é utilizada na definição da geometria inicial.

3.1.1 MÉTODOS DINÂMICOS

Os métodos dinâmicos requerem menos complexidade na criação da geometria já que muitas vezes incorporam elementos do próprio método. De acordo com Pérez (2015) os métodos dinâmicos podem demandar maior capacidade de processamento e requererem mais tempo. A seguir serão destacados alguns métodos dessa vertente.

3.1.1.1 CRIAÇÃO DE PARTÍCULAS DE MODO ALEATÓRIO

Este processo consiste na geração de partículas de raio e posição aleatória dentro da área em questão. O processo envolve a etapa de detecção de contato que será abordada no presente trabalho. Ao gerar partículas aleatoriamente muitas são geradas sobrepostas umas às outras desse modo é necessário detectar as partículas sobrepostas e removê-las.

Apesar da sua facilidade do ponto de vista da execução e de processamento, o processo de criação de partículas aleatórias tende a gerar empacotamentos com um número alto de vazios. Desse modo em muitas situações esse método não se mostra adequado devido às propriedades geradas no sistema de partículas.

Esse método também é utilizado como uma das etapas ao longo de métodos mais complexos de empacotamento.

3.1.1.2 EXPANSÃO DE RAIOS

O método de expansão dos raios é utilizado em conjunto com o método anterior. A expansão de raios surge como uma maneira de diminuir o teor de vazio no empacotamento. As partículas geradas pelo método anterior têm o seu raio expandido gradativamente até entrar em contato com uma partícula vizinha. A Figura 3.2 ilustra o processo do método.

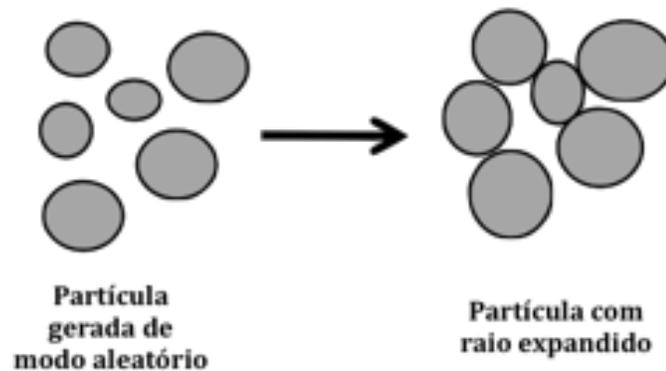


Figura 3.2 – Método da expansão de raios
(O’SULLIVAN 2011 – modificado)

3.1.1.3 COMPACTAÇÃO DE CAMADAS

Para estudar certos fenômenos geotécnicos há a necessidade de se obter certas propriedades no empacotamento como tensão inicial e índice de vazios. O método de compactação de camadas propostos por Jiang (2003) surge como um método para facilitar a obtenção dessas condições iniciais.

A Figura 3.3 ilustra o processo de compactação de camadas. Primeiramente é gerado um grupo de partículas aleatórias em uma região definida, etapa 1. Posteriormente as partículas são compactadas verticalmente até atingir as condições desejadas, etapa 2. Depois a região é expandida, etapa 3, e mais partículas são adicionadas no espaço vazio, etapa 4, e em seguida há uma nova compactação. As etapas 3, 4 e 5 são repetidas até se atingir a espessura desejada.

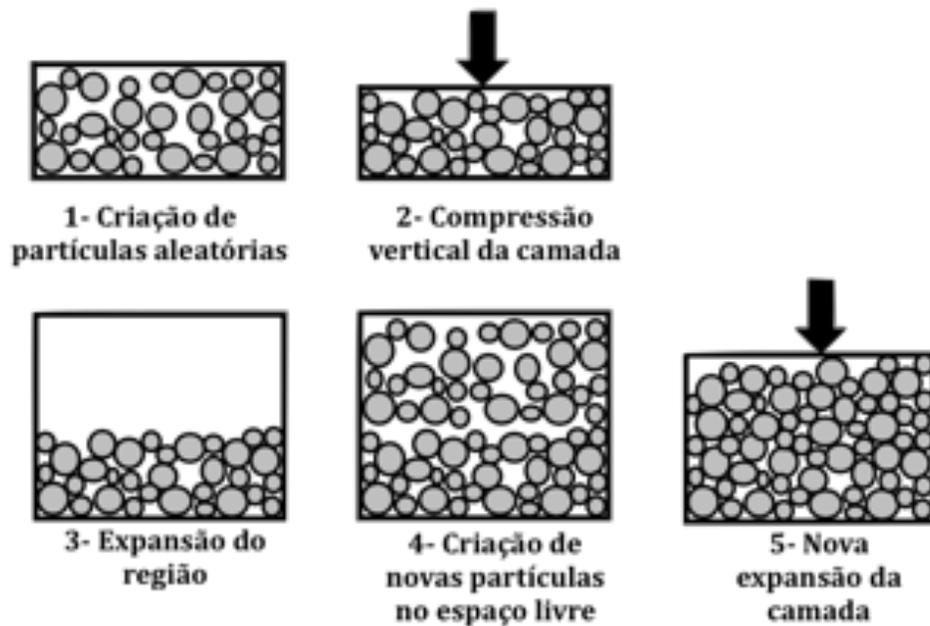


Figura 3.3 – Compactação de camadas de Jiang
(O’SULLIVAN 2011 – modificado)

3.1.2 MÉTODOS CONSTRUTIVOS

Segundo Sullivan (2011) os métodos construtivos podem ser eficientes e conceitualmente bem compreensíveis, no entanto sua implementação do ponto de vista de código computacional pode ser difícil para se obter eficiência acima dos métodos dinâmicos. Alguns métodos construtivos serão mostrados a seguir.

3.1.2.1 MÉTODO DE FRENTE DE AVANÇO

Proposto por Feng (2003) o método consiste na criação de novos discos a partir de um conjunto inicial de discos. Conforme mostrado na Figura 3.4 um grupo inicial de três discos é formado e o conjunto se expande a partir deles. Um novo disco adicionado deve ser inserido de modo a estar em contato com o último disco gerado até que um disco atinja o limite do domínio.

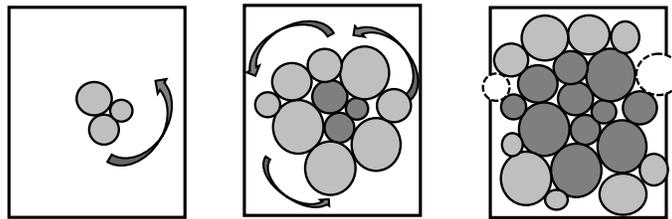


Figura 3.4 – Método de frente de avanço no formato aberto
(O’SULLIVAN 2011)

Bagi (2005) expressa preocupação com o método possuir a tendência de deixar vazios nas quinas do espaço. Bagi (2005) propôs um método alternativo em que as partículas são preenchidas de dentro para fora por meio de anéis conforme mostrado na Figura 3.5. Esse processo alternativo evita os vazios deixados nos extremos pelo método anterior.

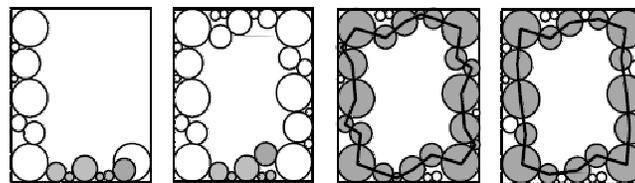


Figura 3.5 – Método de frente de avanço no formato fechado
(O’SULLIVAN 2011)

Os dois métodos descritos anteriormente possuem uma implementação consideravelmente mais simples em simulações bidimensionais. Nos casos tridimensionais a implementação possui considerável complexidade. Uma grande dificuldade na implementação é a utilização de partículas não esféricas devido a dificuldade em situar esferas em contato com as já geradas. Pérez et Al (2016), no entanto propôs uma técnica de minimização para superar este obstáculo.

3.1.2.2 MÉTODO DE TRIANGULAÇÃO

No método de triangulação proposto por Cui e Sullivan (2003) inicialmente é gerada uma nuvem de pontos no espaço da simulação. Posteriormente os pontos da nuvem são ligados formando triângulos no caso bidimensional e tetraedros no caso tridimensional.

Posteriormente são inscritos círculos nos triângulos formados e esferas no caso dos tetraedros. Posteriormente são inseridos círculos ou esferas nos vértices de modo a tangencias pelo menos um dos círculos ou esferas inscritas. A Figura 3.6 ilustra o processo.

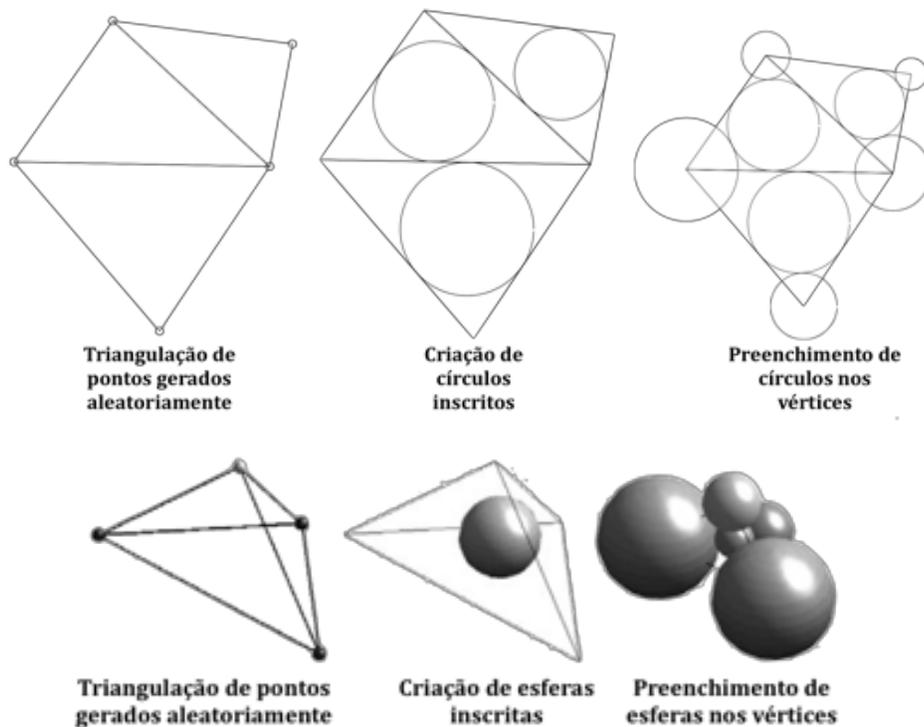


Figura 3.6 – Método de triangulação
(O’SULLIVAN 2011 – modificado)

3.2 MOVIMENTO DAS PARTÍCULAS

O processo usual para solucionar a movimentação de elementos dentro de um sistema se dá pela criação de uma matriz de rigidez e uma posterior inversão do sistema com o objetivo de encontrar os deslocamentos. Esse processo é fundamental no método dos elementos finitos, por exemplo, o método mais utilizado em simulações numéricas na engenharia civil. O processo usual é dado pela seguinte expressão:

$$\Delta F = m \cdot U'' + C \cdot U' + K(U) \quad (3.1)$$

Onde,

ΔF : incremento de força no sistema

m : massa do sistema

U'' : aceleração

C : Coeficiente de perda de energia devido ao movimento

$K(U)$: matriz de rigidez global do sistema

Esse método, no entanto, não seria eficiente caso fosse aplicado há um problema onde usualmente é aplicado o método DEM. Segundo Sullivan (2011) um sistema relativamente pequeno com 1000 partículas em tridimensional resultaria em uma matriz de rigidez com 36 milhões de elementos que posteriormente precisaria ser invertida. Esse processo, neste método consumiria grande quantidade de memória e demandaria um grande tempo de cálculo.

Cundall e Strack (1979) propuseram um método alternativo e mais eficiente para o problema. Ao invés de considerar um sistema global composto pelo conjunto das partículas cada partícula passa a ser considerado o equilíbrio de cada partícula individualmente. Esse processo requer menor requisito de memória e pode ser implementado de modo mais rápido para um sistema com grande quantidade de partículas. Zhu (2007) elaborou a seguinte expressão para generalizar o equilíbrio de uma partícula:

$$m_p \cdot u''_p = \sum_{c=1}^{N_{c,p}} F_{pc}^{con} + \sum_{j=1}^{N_{nc,p}} F_{pj}^{non-con} + F_p^f + F_p^g + F_p^{apl} \quad (3.2)$$

Onde,

m_p : massa da partícula

u''_p : aceleração da partícula

F_{pc}^{con} : forças atuantes devido ao contato entre partículas

$F_{pc}^{non-con}$: força de contato entre a partícula e os elementos de contorno do sistema

F_p^f : forças geradas por fluídos

F_p^g : força gerada pela aceleração da gravidade

F_p^{apl} : força externa aplicada

Para a obtenção das forças entre partículas e com o contorno do sistema é necessário a adoção de um modelo de contato. Existem diversas opções de modelos de contato podendo tornar o modelo mais próximo da realidade e demandar maior processamento. Os modelos computacionais serão abordados mais à frente no presente trabalho.

Outra observação importante é que o coeficiente de perda de energia não estava presente na equação anterior. O coeficiente é adicionado no modelo de constitutivo de contato entre as partículas.

Há ainda a velocidade angular da partícula que depende da sua inércia e do somatório dos momentos gerados pelas forças de contato em cada partícula, conforme:

$$I_p \cdot \frac{d\omega_p}{dt} = \sum_{j=1}^{N_{mon}} M_{pj} \quad (3.3)$$

Onde,

I_p : inércia da partícula

ω_p : velocidade angular da partícula

M_{pj} : momentos gerados pelas forças de contato

O momento usualmente é gerado por forças tangenciais ou forças normais não alinhadas com o centro da partícula, modelos de contato que consideram momento diretamente no contato da partícula, no entanto, também já foram criados.

Com a obtenção da aceleração da partícula resultante das forças de contato na partícula existe a necessidade de se adotar um processo de integração para se obter a velocidade e o deslocamento com a aceleração.

O processo de integração mais utilizado é constituído pelas seguintes equações mostradas a seguir. Rapaport (2004) adota o nome de *Leap Frog* ao processo e Munjiza (2004) denomina o processo como sistema de integração Verlet. As etapas do processo são mostradas a seguir:

$$a_p^t = \frac{1}{\Delta t} \cdot (v_p^{t+\Delta t/2} - v_p^{t-\Delta t/2}) \quad (3.4)$$

$$v_p^{t+\Delta t/2} = v_p^{t-\Delta t/2} + \Delta t \cdot \frac{F_p^t}{m_p} = v_p^{t-\Delta t/2} + \Delta t \cdot a_p^t \quad (3.5)$$

$$x_p^{t+\Delta t} = x_p^t + \Delta t \cdot v_p^{t+\Delta t/2} \quad (3.6)$$

Onde:

a_p : aceleração da partícula

Δt : passo de tempo adotado no processo de integração

v_p : velocidade da partícula

F_p : força atuante na partícula

x_p : posição da partícula

m_p : massa da partícula

O método consiste em calcular a aceleração da partícula com base na força atuante na mesma. Com a aceleração é possível calcular a velocidade da partícula e conseqüente sua posição após o passo de tempo adotado.

Vale ressaltar que segundo Cundall e Strack (1979) o método adotado é condicionalmente estável e por isso pequenos passos de tempo devem ser utilizados. Essa instabilidade do método se deve em parte ao fato de que a sobreposição entre as partículas e o contorno pode ser defasada caso não seja adotado um passo de tempo adequado.

Nos modelos de contato utilizados para determinar a força na partícula a principal variável é a sobreposição das partículas (δ) mostrada na Figura 3.7. Com um passo de tempo muito pequeno essa sobreposição pode se tornar muito alta levando a uma força muito alta no contato.

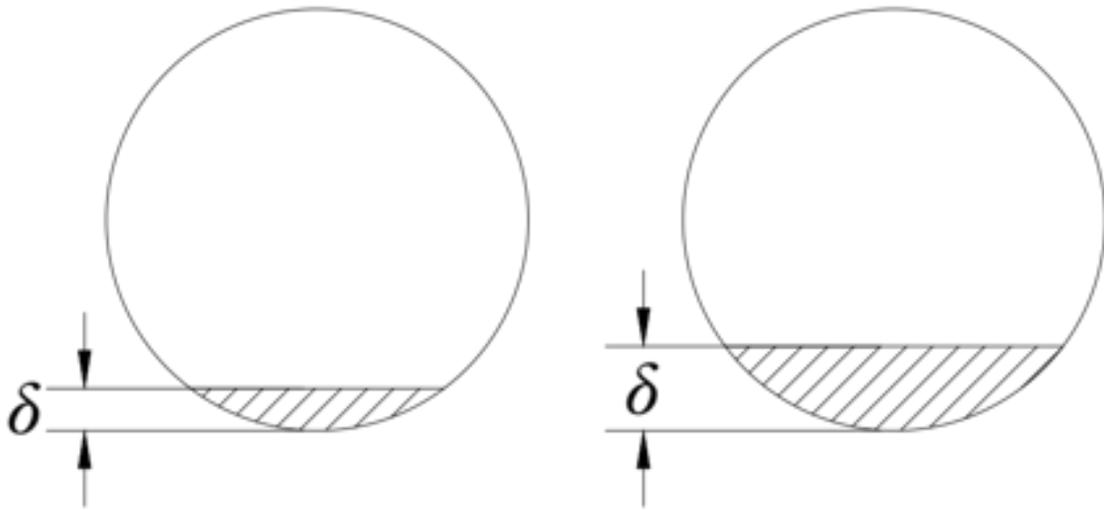


Figura 3.7 – Sobreposição na partícula

Outra possível fonte de erro é o processo de integração numérico adotado. O processo de integração não leva em consideração a não linearidade do movimento das partículas gerando um erro conhecido como erro de truncamento no processo. Com um passo de tempo elevado esse erro aumenta e é criada uma instabilidade no processo.

Para mostrar o efeito dessa instabilidade foi feita uma simulação utilizando uma esfera de raio 0,1m com peso de 1kg e rigidez de 10.000N/m sendo solta de uma altura de 5m. Foi adotado um modelo de contato simples em que a força no contato é igual à sobreposição vezes a rigidez da bola. Variou se o passo de tempo e a altura da bola, no eixo das ordenadas, é mostrada em função do tempo, no eixo das abscissas. Os resultados são mostrados nas Figura 3.8, Figura 3.9, Figura 3.10.

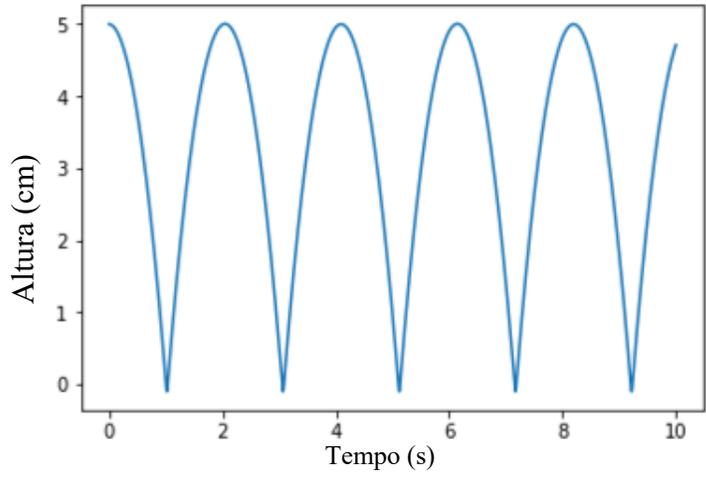


Figura 3.8 – Simulação com passo de tempo 0,01s

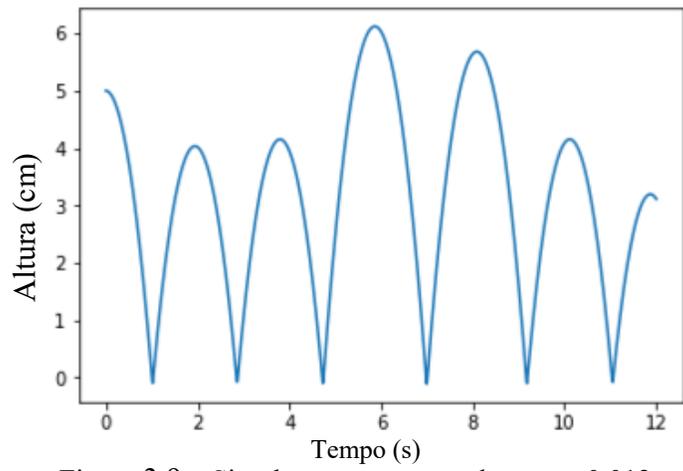


Figura 3.9 – Simulação com passo de tempo 0,012s

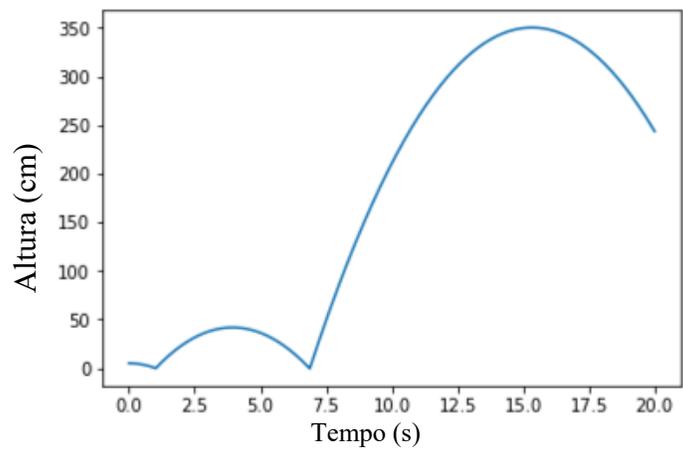


Figura 3.10 – Simulação com passo de tempo 0,02s

Foi possível notar que o passo de tempo elevado inicialmente gerou uma instabilidade no sistema e posteriormente uma força muito elevada e deslocamento muito elevados.

A velocidade de rotação angular pode ser determinada em processo semelhante ao descrito anteriormente para a posição da partícula. Com o momento da partícula obtido por meio dos modelos de contato é determinada a aceleração angular e por meio do processo de integração e obtida a velocidade angular da partícula. O processo é mostrado a seguir:

$$I_p \cdot \omega'_p = M_p \quad (3.7)$$

$$\omega_p^{t+\Delta t/2} = \omega_p^{t+\Delta t/2} + \Delta t \cdot \frac{M_p^t}{I_p} \quad (3.8)$$

Onde,

I_p : inércia da partícula

ω_p : velocidade angular da partícula

M_p : momentos gerados pelas forças de contato

Δt : passo de tempo

3.3 FORÇAS DE CONTATO

O processo de determinação das forças de contato é o que mais demanda tempo dentro do método. Segundo estimativas de Stutman (2002) esse processo gasta 90% do tempo da simulação. Existem variáveis que influenciam nesse tempo. A adoção de um modelo de contato mais complexo pode elevar o tempo e um empacotamento de partículas mais denso também. Com uma maior densidade de partículas há uma maior quantidade de contatos e conseqüentemente um maior tempo resolvendo-os.

Zhu (2007) destacou que o contato entre partículas é um fenômeno extremamente complexo, dependendo de sua geometria, propriedades do material constituinte e movimento da partícula. Segundo Sullivan (2011) duas simplificações quase sempre são adotadas nos modelos de contato. A primeira delas é descrever a superfície da partícula como lisa. A segunda é considerar que o contato entre as partículas ocorre em único ponto.

No método, segundo Sullivan (2011), os corpos são considerados rígidos, uma simplificação do que ocorre na realidade onde os corpos se deformam, além disso é permitida uma sobreposição (δ) entre os corpos, conforme mostrado na Figura 3.11. Essa sobreposição ou distância entre corpos é a utilizada na determinação da força de contato por meio da utilização de um modelo constitutivo de contato. JERZY (2008) no entanto, propõe um modelo adotando corpos flexíveis.

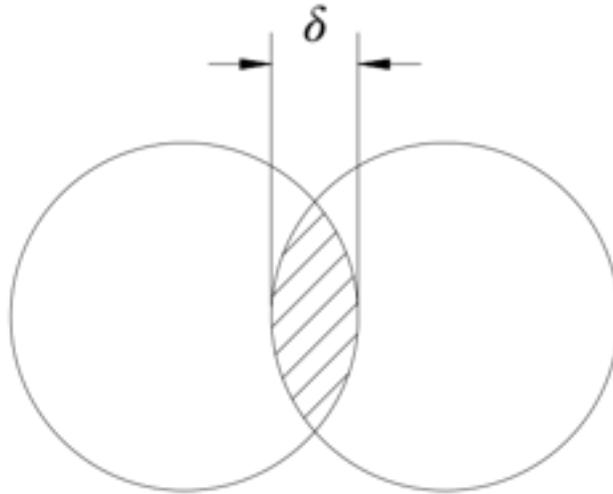


Figura 3.11 – Sobreposição de partículas

As forças de contato no método se dividem em duas categorias, as forças normais à direção do contato, podendo ser de tração ou compressão, e as forças tangenciais ao contato. As duas forças possuem diversos modelos de contato incorporando diferentes elementos no modelo. A maior proximidade da realidade de um modelo mais complexo se traduz em maior necessidade de capacidade de processamento e maior tempo de execução da simulação já que esta etapa é a que mais demanda tempo dentro do método.

Entre os modelos de contato para a força normal Sullivan (2011) destaca o modelo linear de molas, o modelo Hertziano, o modelo linear de Walton-Braun e o modelo de mola e amortecedor.

O modelo de molas é o mais utilizado no método, principalmente devido a sua simplicidade. O modelo segue a seguinte equação:

$$F = k \cdot \delta \quad (3.9)$$

onde a força de contato (F) é função apenas de uma rigidez (k) e da separação entre as partículas (δ). Segundo Sullivan (2011) uma grande dificuldade na aplicação deste modelo é que a rigidez no modelo não está claramente relacionada a uma propriedade do material. Uma possibilidade, no entanto, é calibrar o modelo de acordo com resultados observados experimentalmente.

O modelo Hertziano se apresenta como um modo de corrigir a falta de correlação entre a rigidez do modelo linear de molas e a natureza física do material. A força entre os contatos é calculada de acordo com a mesma expressão mostrada anteriormente para o modelo de molas.

A diferença para o modelo anterior reside no fato de que a rigidez de um contato passa a depender da natureza das duas partículas do contato, resultando em uma rigidez conforme:

$$k = \frac{2 \cdot G \cdot \sqrt{2 \cdot R}}{3 \cdot (1 - \nu)} \cdot \sqrt{\delta} \quad (3.10)$$

Onde a rigidez (k) depende do raio (R), do módulo elástico (G) e do coeficiente de Poisson (ν) das duas partículas (a) e (b) e também da interpenetração das partículas (δ). Raio, módulo elástico, e o coeficiente de Poisson são determinados conforme:

$$R = \frac{2 \cdot R_a \cdot R_b}{R_a + R_b} \quad (3.11)$$

$$G = 0,5 \cdot (G_a + G_b) \quad (3.12)$$

$$\nu = 0,5 \cdot (\nu_a + \nu_b) \quad (3.13)$$

Onde (R_a) e (R_b) são os raios da partículas a e b respectivamente, (G_a) e (G_b) são os módulos de elasticidade, (ν_a) e (ν_b) são os coeficientes de Poisson das partículas a e b .

O modelo de Walton-Braun busca trazer a perda de energia do contato ao mesmo tempo em que preserva uma relação linear entre a distância das partículas e a força no contato. Proposto por Walton e Braun (1986) o modelo segue as seguintes relações para carregamento e descarregamento:

$$F = k_1 \cdot \delta \quad (3.14)$$

(carregamento)

$$F = k_2 \cdot (\delta - \delta_p) \quad (3.15)$$

(descarregamento)

Onde, (k_1) é a rigidez para o círculo de carregamento, (k_2) é a rigidez para o ciclo de descarregamento, (δ_p) é a deformação plástica no contato e (δ) é a interpenetração entre as partículas.

A deformação plástica depende da força máxima estabelecida na ligação de modo segue:

$$\delta_p = \frac{F_{max}}{k_2} \quad (3.16)$$

Onde, F_{max} é a máxima força estabelecida na partícula durante a etapa de carregamento.

O comportamento resultante desse modelo é mostrado na Figura 3.12. É possível notar que há uma maior energia sendo absorvida na fase de carregamento do que é liberado na fase de descarregamento, gerando uma perda de energia o processo. Além disso é possível notar que há uma deformação residual plástica no processo.

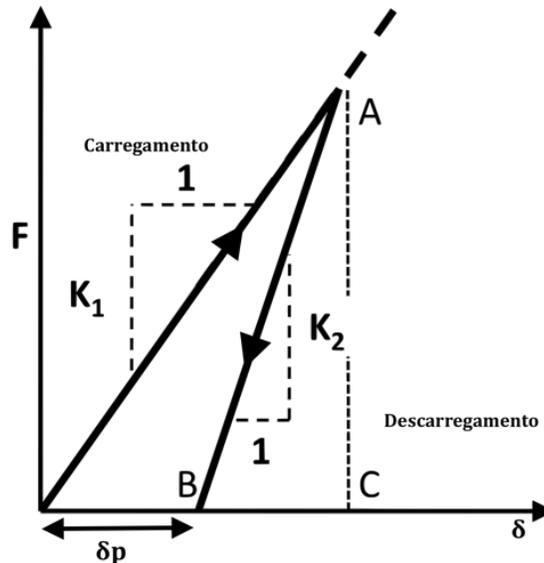


Figura 3.12 – Relação tensão deformação no modelo Walton-Braun
(O’SULLIVAN 2011 – modificado)

O modelo de Walton-Braun pode ser correlacionado com o coeficiente de restituição de uma colisão entre duas partículas. O coeficiente de restituição se relaciona com as duas medidas de rigidez de Walton-Braun conforme:

$$e = \frac{v'_a - v'_b}{v_a - v_b} \quad (3.17)$$

Onde (v) é a velocidade antes da colisão de duas partículas (a) e (b) e (v') é a velocidade após a colisão dessas duas partículas, (e) é o coeficiente de restituição, essa medida relaciona a perda de energia na colisão.

O modelo de mola e amortecedor por sua vez incorpora a perda de energia por meio da adoção de um amortecedor viscoso. O amortecedor modifica a força no contato em função da variação entre a distância das duas partículas acrescentando um termo dissipativo, conforme:

$$F = K \cdot \delta + C \cdot \delta' \quad (3.18)$$

Onde, (F) é a força de contato entre as partículas, (δ') é a variação de distância entre as partículas e (C) é um coeficiente de dissipação e (K) é a rigidez do contato.

A adoção deste modelo resulta em uma relação de força deformação diferenciada para o carregamento e o descarregamento conforme a figura mostra a seguir. A diferença entre as curvas gera uma perda de energia na colisão entre as partículas já que a energia em cada ciclo é representada pela integração de cada curva. Desse modo a perda de energia é representada pela área entre as duas curvas mostradas na Figura 3.13.

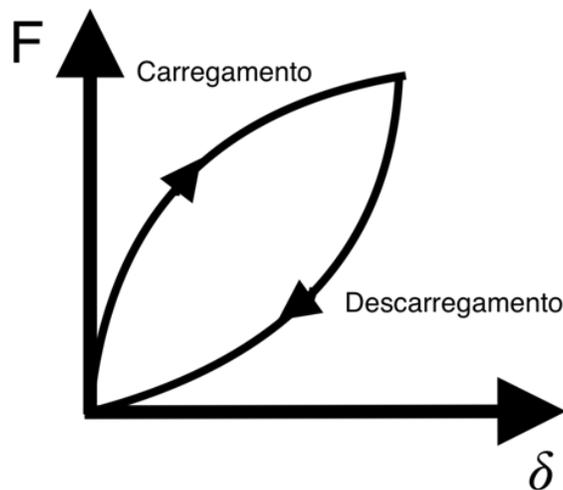


Figura 3.13 – Tensão deformação no modelo mola amortecedor

Segundo Pöschel e Schwager (2005) existe uma aparente discrepância no modelo utilizado para calcular força tangencial no contato. Na maioria dos casos as partículas simuladas são completamente lisas o que faria com que não houvesse forças tangenciais nos contatos. Entretanto, em quase todos os modelos é adotado um parâmetro de fricção no contato resultando em uma força tangencial.

Os modelos de contato tangencial se dividem em duas etapas diferentes. Uma primeira etapa consiste no momento antes das partículas deslizarem entre si e um segundo após eles deslizarem.

Segundo O’Sullivan (2011) o modo mais simples de calcular a força tangencial é conforme:

$$F_t(\delta_t, \delta_t') = k \cdot \int_{t_c}^t \delta_t' dt \approx k_t \cdot \sum_{t_c}^t \delta_t' \Delta t \quad (3.19)$$

Onde, a força tangencial (F_t) antes do deslizamento é relacionada à movimentação tangencial acumulada e com uma rigidez (k), em um modelo linear. O deslocamento tangencial é calculado por meio de integração numérica utilizando a velocidade de deslocamento (δ_t') e o passo de tempo (Δt)

Na etapa pós deslizamento o modelo mais simplista é utilizar um modelo de atrito de Coulomb conforme:

$$F_t = |\mu \cdot F_n| \cdot \frac{F_t}{|F_t|} \quad (3.20)$$

Onde a força tangencial (F_t) depende de um coeficiente de atrito (μ), maior que 0 e usualmente menor que 1, e da força normal atuando na partícula (F_n). A direção da força é dada pela força tangencial calculada anteriormente (F_t).

3.4 BUSCA DE CONTATOS

Para determinar os esforços entre duas partículas, primeiro é necessário identificar se as duas partículas estão em contato entre si. Existem diversos métodos para identificar se duas partículas estão em contato. Os métodos mais usuais consistem na identificação de uma distância mínima entre as partículas e a posterior comparação entre essa distância

mínima e a distância efetiva entre as partículas. Caso a distância seja menor ou igual a mínima há um contato entre as partículas.

Pöschel e Schwager (2005) definem essa distância mínima como *Verlet distance*. Quando a distância efetiva é menor que a distância Verlet a as partículas são consideradas em contato e devem ser acrescentadas a lista Verlet, das partículas em contato. Depois que as partículas são acrescentadas a lista devem ser calculados os esforços no contato entre as duas partículas. A lista deve ser atualizada a cada incremento de tempo ao longo da execução do programa.

Vale ressaltar que o contorno da partícula nem sempre é o real contorno da partícula e muitas vezes o contorno é simplificado utilizando um quadrado ou esfera com a finalidade de otimizar o algoritmo. Desse modo a distância efetiva entre as partículas pode ser menor do que a mínima calculada entre elas.

Do ponto de vista do algoritmo o caminho mais fácil, para identificar o contato entre partículas, seria realizar a verificação entre cada uma das partículas a cada incremento de tempo na execução do programa. Essa perspectiva, no entanto, se mostra inviável do ponto de vista da capacidade de processamento e tempo de execução. O número de potenciais contatos entre partículas cresce proporcionalmente ao quadrado do número de partículas, se tornando extremamente alto para um número ainda pequeno de partículas. Desse modo é necessário otimizar o programa do ponto de vista da verificação de contato entre partículas para tornar o processo viável.

Munjiza (2004) destacou a importância de algoritmos de detecção de contato na implementação do método dos elementos discretos. O autor destacou ainda os requerimentos para esse tipo de algoritmo. Segundo o autor é necessário que o programa seja confiável, de fácil implementação e eficiente do ponto de vista da utilização da capacidade de processamento.

Os tipos mais comuns de mecanismo utilizados para detectar contatos são os baseados na utilização de malhas. Pöschel e Schwager (2005), Munjiza (2004), e Bobet (2009) descrevem com detalhes diversos algoritmos para detectar contatos.

Nos mecanismos utilizando malhas, uma malha é definida ao longo do domínio do problema usualmente de modo que a maior das partículas possa ser incluída em apenas um elemento da malha. Para a obtenção dos potenciais elementos em contato é utilizada a

malha de modo que uma partícula no elemento da malha número um pode estar potencialmente em contato com uma partícula em cada um dos elementos de número 2 a 8 circulando o local da partícula e o próprio elemento de número 1 onde está a partícula. A Figura 3.14 ilustra o exemplo.

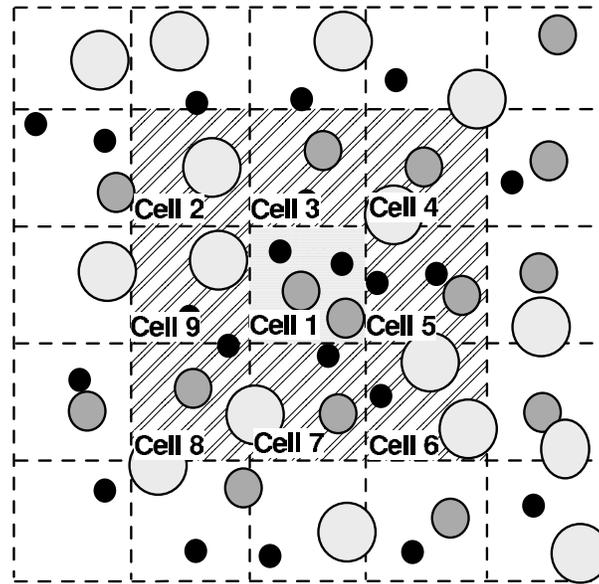


Figura 3.14 – Modelo de malha

(O’SULLIVAN 2011)

No caso tridimensional a partícula pode estar em contato com cada partícula nos 26 elementos de malha vizinhos e o elemento de malha em que está inserido, totalizando 27.

No passo seguinte existem divergências entre duas opções. A primeira delas é testar o contato das partículas conforme descrito anteriormente para em seguida inserir a partícula na lista de partículas em contato. A segunda é inserir a partícula diretamente na lista sem que antes seja feito o teste se está em contato ou não.

4 ENSAIOS EXPERIMENTAIS

Com a finalidade de validar o método, optou-se por analisar um ensaio experimental relativo ao atrito no solo, sendo adotado o ensaio de ângulo de repouso. Esse ensaio permite obter o ângulo de repouso para um dado material granular.

A seguir serão mostrados alguns aspectos sobre o ensaio e a execução. Os resultados experimentais obtidos por Girardi (2016) posteriormente serão utilizados como base para comparar resultados numéricos e experimentais e analisar a efetividade do método dos elementos discretos.

4.1 ÂNGULO DE REPOUSO

Segundo Guzmán (2008) o ângulo de atrito é definido como o ângulo sob o qual se assenta um material submetido à esforço nulo ou próximo disso. A Figura 4.1 ilustra o ângulo de repouso de um material.

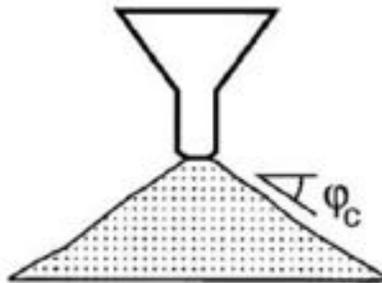


Figura 4.1 - Ângulo de repouso

(Gúzman, 2008)

Segundo Taylor (1962), o ângulo de repouso é uma boa aproximação para o ângulo de atrito. Dessa particularidade surge o grande interesse pelo ângulo de repouso do solo e o grande número de maneiras para se obter ele.

Um dos ensaios é o proposto por Miura, Maeda e Toki (1997). Neste o material é inserido por meio de um funil em um aro metálico. O funil deve ser posicionado próximo da pilha de material de modo a possuir uma pequena altura de queda e não influenciar no

estado de tensões do material. Uma vez realizado o experimento, o ângulo pode ser medido a partir da pilha formada.

O ensaio proposto por Miura Maeda e Toki(1997) serviu como base para os ensaios realizados por Girardi (2016) e por isso são de grande importância no presente trabalho.

5 METODOLOGIA

A metodologia do trabalho foi dividida em duas etapas. Primeiro foram realizadas as simulações numéricas e posteriormente foram comparados os resultados numéricos com os resultados experimentais, a fim de determinar a validade do método. Os resultados numéricos utilizados foram obtidos por Girardi (2016).

Os resultados foram obtidos por meio da utilização de esferas padronizadas de vidro Potters em ensaios experimentais. As propriedades das esferas são mostradas na Tabela 5.1.

Tabela 5.1 Propriedades físicas de vidro Potters

Propriedades físicas	
Densidade aparente (g/cm ³)	1,5
Densidade de massa (g/cm ³)	2,5
Índice de refração	1,50 - 1,52
Resistência a compressão (psi)	14.000 - 36.000
Dureza MOH	5 - 6
Dureza Knoop 100g carga (kg/mm ²)	515
Coefficiente estático de fricção	0,9 - 1,0
Propriedades químicas	
Composição	Vidro tipo soda-cal
Sílica livre	Nenhuma

5.1 SOFTWARE ADOTADO

A primeira etapa do processo de simulação foi definir o modo como serão feitas as simulações. Existem diferentes opções de softwares comerciais (PFC, EDM e STAR-CCM+) e não comerciais (LIGGGTS, LAMMPS e YADE) capazes de realizar simulações utilizando o método dos elementos discretos. Além disso, é possível desenvolver um algoritmo e implementá-lo com base nos aspectos teóricos do método descritos anteriormente.

Foi feita a opção por realizar as simulações utilizando o *software* livre YADE (*Yet Another Dynamic Engine*) e a escolha se deu por alguns fatores. O *software* YADE consiste

em um programa desenvolvido utilizando C++, em que os comandos são feitos por meio de *Scripts* utilizando linguagem de programação Python.

Essa combinação de linguagens leva à algumas vantagens na implementação. A linguagem C++, apesar de ser mais complexa e demandar mais do programador, possui maior velocidade de processamento, algo vital devido ao fato já explicitado anteriormente do DEM demandar grande capacidade de processamento. Já a linguagem Python é de implementação mais fácil e demanda menos do programador, facilitando o processo de escrever rotinas a serem executadas.

Outra característica que pesou nessa escolha foi a versatilidade do *software* YADE. O programa é extremamente versátil no sentido de permitir um grande número de configurações e variações em relação a maneira de se implementar o método, como variação no modelo constitutivo a ser adotado.

Além disso pesou na decisão a presença de uma documentação extensa, auxiliando na utilização do programa. A presença de uma grande base de utilizadores no programa facilita a solução de problemas que venham a aparecer ao utilizá-lo, sendo também um fator relevante na escolha. Também é de se considerar que o *software* YADE surgiu para resolução de problemas de geotecnia.

5.2 ESTRUTURA DA SIMULAÇÃO

Para facilidade de compreensão optou-se por dividir o script para execução da simulação em dois, um principal com as instruções para execução e um auxiliar para definição de funções a serem utilizadas no script principal. Essa opção foi feita exclusivamente para facilidade de compreensão durante a execução e os dois scripts poderiam ser feitos em um único arquivo. O script principal e o auxiliar são mostrados no anexo I e anexo II.

A simulação foi realizada de modo a se aproximar ao ensaio de ângulo de deposição realizado por Girardi (2016) experimentalmente. A primeira etapa constitui na criação das condições de contorno da simulação, composta por um prato para deposição de partículas e um cone para a deposição inicial das partículas e potencial transferência para o prato por gravidade. As condições são mostradas na Figura 5.1 a seguir.

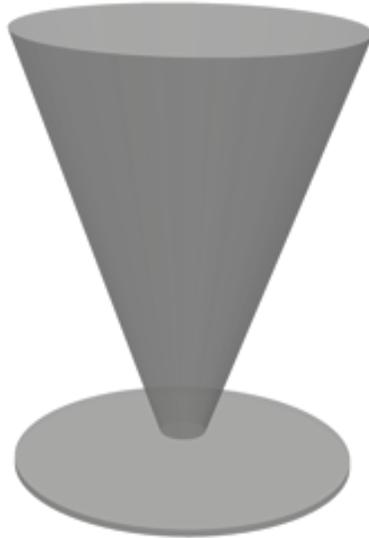


Figura 5.1 – Condições de contorno da simulação

A etapa seguinte consiste na geração do empacotamento de modo a preencher o cone citado anteriormente. O empacotamento foi gerado utilizando função própria do YADE, a qual busca gerar o máximo número possível de partículas em uma dada região. Na função de empacotamento é utilizada como argumento de entrada a curva granulométrica desejada, para isso foram utilizadas as curvas ensaiadas por Girardi (2016). A Figura 5.2 ilustra o empacotamento gerado.

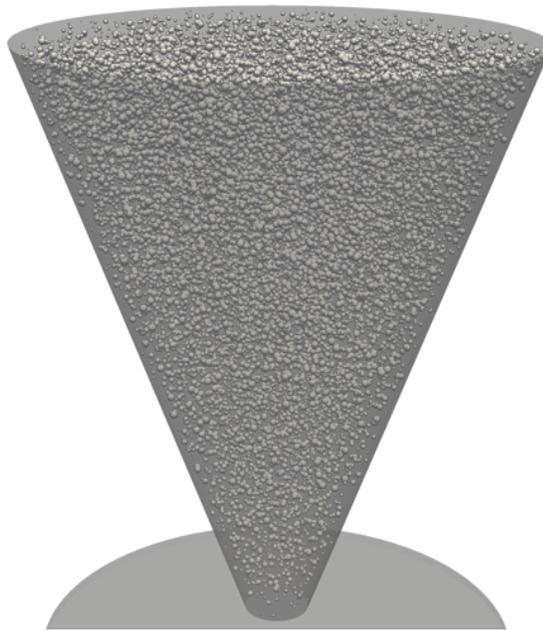


Figura 5.2 – Empacotamento

Em seguida são atribuídas as propriedades físicas aos materiais gerados e foi definido o modelo constitutivo de contato. Tanto os elementos de contorno como as partículas tiveram o mesmo modelo de contato constitutivo e propriedades físicas. As propriedades físicas foram adotadas de acordo com as fornecidas pelo fabricante das esferas ensaiadas experimentalmente e o modelo de contato adotado foi o Hertziano, adiante serão mostrados maiores detalhes de ambos.

Posteriormente foi atribuída uma velocidade vertical para o cone subir em relação ao prato e acompanhar o acúmulo das partículas. Girardi (2016) movimentava o cone de modo a manter a sua base o mais próximo possível do material acumulado, sendo que nas simulações não foi possível replicar esse conceito. Para se aproximar disso foi definida uma velocidade vertical constante e foram realizados testes variando a velocidade para se aproximar do proposto por Girardi (2016). A Figura 5.3 a seguir ilustra a movimentação do cone.

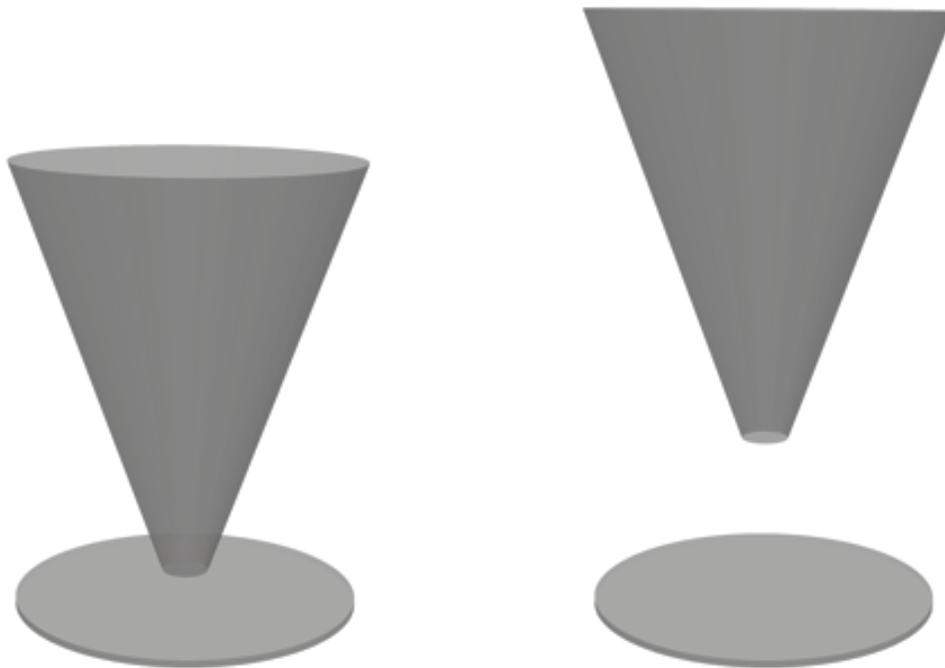


Figura 5.3 – Movimento vertical do cone

Para a definição do passo de tempo foi utilizada uma ferramenta do próprio YADE baseada nas propriedades dos materiais utilizados. Após a definição das condições iniciais da simulação foi dado início ao processo iterativo por meio do incremento de passo de

tempos nos moldes descritos anteriormente. As partículas fluem pelo cone e se acumulam no prato e a simulação é executada até o sistema atingir um nível de equilíbrio de forças mínimo. A Figura 5.4 ilustra as partículas fluindo e após atingir o equilíbrio de forças.

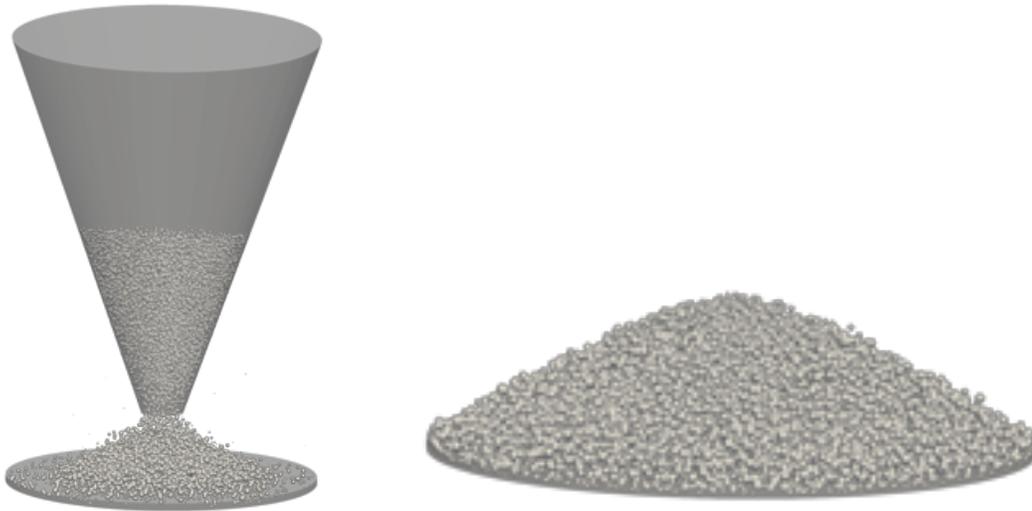


Figura 5.4 – Partículas em movimento e em equilíbrio

O ângulo de repouso foi calculado utilizando o mesmo procedimento realizado por Girardi (2016):

$$\phi = \arctan(H/R) \quad (5.1)$$

Onde o ângulo de repouso (ϕ) depende da altura das partículas acumuladas no prato (H) e do raio do prato onde elas se acumularam (R). A Figura 5.5 ilustra o ângulo de repouso.

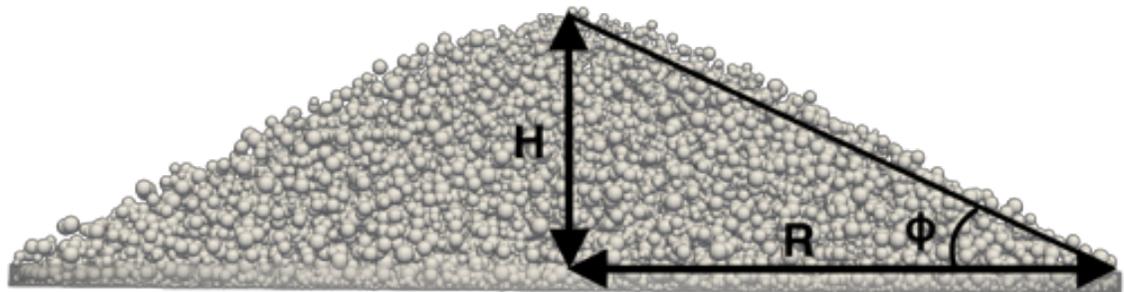


Figura 5.5 – Ângulo de repouso

5.3 DEFINIÇÃO DOS PARÂMETROS DA SIMULAÇÃO

Ao longo da simulação podem ser adotadas variações dentro das etapas de cálculo do DEM. Assim como na geração de partículas essas condições interferem no tempo de processamento e no resultado final da simulação.

As diferentes opções usualmente consistem em uma troca entre menor necessidade de processamento e maior proximidade com os fenômenos que ocorrem na prática nas partículas.

5.3.1 MODELO CONSTITUTIVO DE CONTATO

O modelo constitutivo possui grande influência no tempo da simulação pois é utilizado diversas vezes ao longo da simulação e pode interferir no tempo de cálculo destas. Além disso, o modelo de contato tem interferência direta no comportamento da simulação, sendo que um modelo mais sofisticado e com parâmetros mais próximos da realidade tende a aumentar a exatidão da simulação.

Como as propriedades do material ensaiado experimentalmente foram claramente especificadas pelo fabricante, optou-se por utilizar um modelo em que essas fossem levadas em consideração. Além disso, as partículas foram produzidas por processo industrial, de modo a ter as propriedades próximas do especificado, fato não muito frequente na utilização de matérias naturais.

Optou-se pelo modelo constitutivo de contato Hertziano, mostrado anteriormente no trabalho. Essa escolha se deu pelo fato de o modelo utilizar o módulo de Young da partícula, o coeficiente de Poisson e o ângulo de atrito do material, todos eles fornecidos pelo fabricante. Os valores adotados são mostrados na Tabela 5.2 Propriedades adotadas no modelo constitutivo de contato.

Tabela 5.2 Propriedades adotadas no modelo constitutivo de contato

Módulo de Young* (kPa)	17,2
Coefficiente de Poisson	0,22
Densidade (kg/m ³)	2.500
Ângulo de Atrito (°)	43,5

Foi feita uma adaptação com relação ao módulo de Young do material. Devido ao tipo de experimento, o módulo de Young não gera grande interferência no ângulo de repouso obtido, segundo Taylor (1962) o ângulo de repouso é uma aproximação para o ângulo de atrito no material. Entretanto, essa propriedade tem grande influência no tempo de execução da simulação.

Grandes valores de rigidez levam a um tempo de contato reduzido entre as partículas e desse modo é necessário um menor passo de tempo na simulação para realizá-la sem erro numérico considerável. Devido ao número elevado de partículas dado pela utilização das partículas em escala real, realizar as simulações com a rigidez do material se tornou uma opção inviável do ponto de vista computacional e foi feita uma redução significativa do seu valor.

5.3.2 MODELO DE INTEGRAÇÃO

Foi adotado o modelo *Verlet* de integração, descrito anteriormente no trabalho. Este modelo de integração é o de mais fácil implementação e o mais utilizado. Além disso, o modelo já possui ferramentas no YADE para sua implementação.

5.3.3 FORMATO DAS PARTÍCULAS

Existe a possibilidade de adotar diferentes formatos de partículas em uma simulação utilizando o DEM. A utilização de diferentes formatos permite analisar como estas influem no comportamento final. A utilização de partículas em formato diferente do esférico, no entanto, impõe desafios de implementação e processamento muito maiores. Optou-se por realizar as simulações somente utilizando partículas esféricas conforme foi realizado nos ensaios de Girardi (2016)

5.3.4 CONDIÇÕES DE CONTORNO INICIAIS

As condições de contorno utilizadas foram baseadas no trabalho experimental realizado por Girardi (2016), com pequenas adaptações. Assim como Girardi (2016) foi

utilizado também um conjunto composto por cones, onde eram depositadas as partículas, e um prato onde as partículas se acumulavam.

Em relação ao realizado por Girardi (2016) foi feita uma redução na altura do prato, de 1,05 cm para 1,5mm. O principal motivo dessa redução foi a redução na quantidade de partículas necessárias para o experimento diminuindo o tempo de processamento da simulação.

No ensaio realizado por Girardi (2016), assim como proposto por Miura, Maeda e Toki (1997), a subida do cone era regulada de modo a manter sua base o mais próximo o possível do material granular no prato, de modo a minimizar a altura de queda e reduzir o aparecimento de tensões. Isso, no entanto, não se mostrou passível de ser executado na simulação e foi necessário definir uma velocidade constante de subida do cone. Desse modo, foram realizados testes com diferentes velocidades até se obter uma velocidade adequada de 7,5mm/s.

Outras condições de contorno definidas foram a altura do cone (9 cm), o diâmetro da base do cone (1,5 cm) e o diâmetro do topo do cone (7,5cm). A Tabela 5.3 Condições iniciais de contorno e a Figura 5.6 a seguir mostram as condições iniciais adotadas.

Tabela 5.3 Condições iniciais de contorno

Diâmetro do prato (cm)	7,5
Altura do prato (mm)	1,5
Altura do cone (cm)	9
Diâmetro na base do cone (cm)	1
Diâmetro no topo do cone (cm)	7,5
Velocidade do cone (mm/s)	7,5

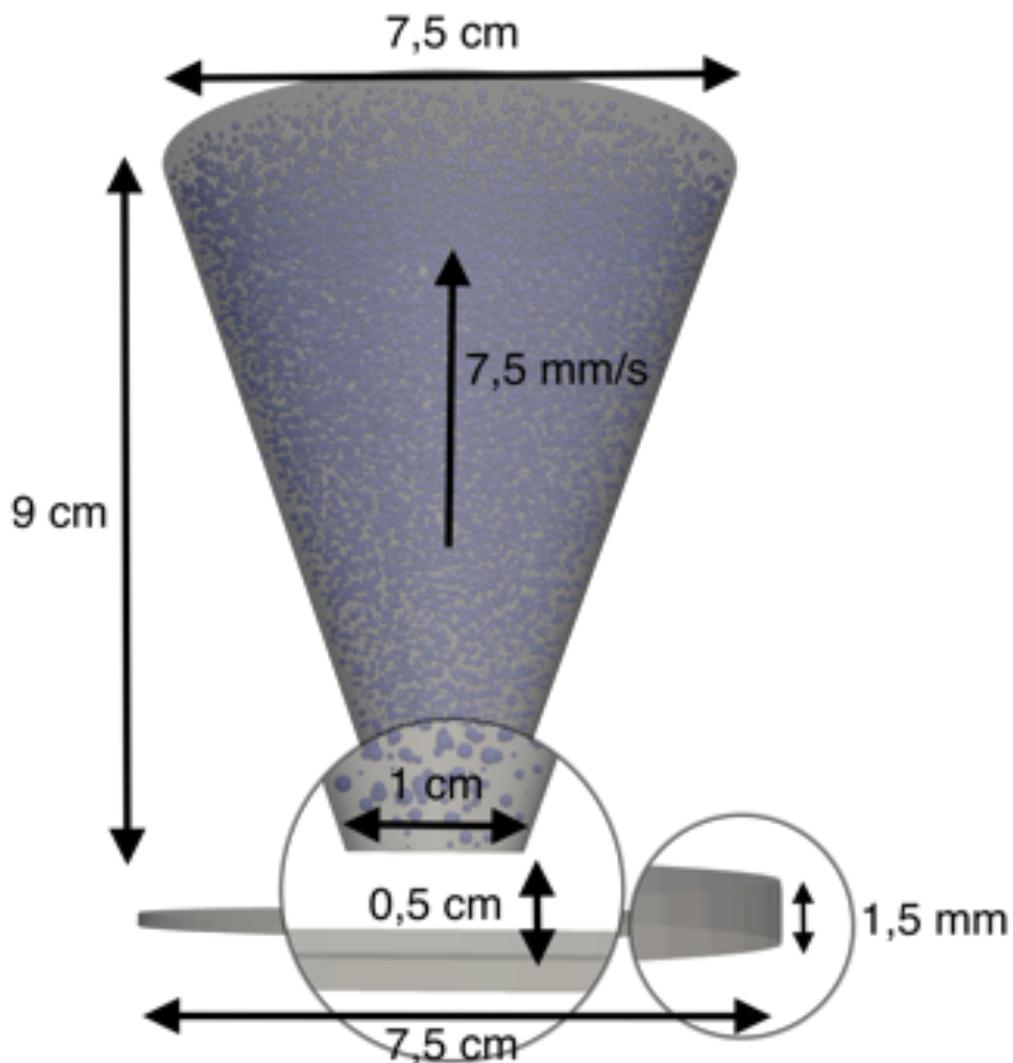


Figura 5.6 – Curvas granulométricas utilizadas

5.4 PARÂMETROS A SEREM ANALISADOS

Serão adotadas quatro curvas granulométricas diferentes, conforme foram ensaiadas experimentalmente por Girardi (2016), com a finalidade de se comparar os resultados obtidos experimentalmente com os resultados obtidos numericamente. A Figura 5.7 ilustra as curvas a serem utilizadas. As curvas granulométricas correspondem às esferas de vidro Potter citadas anteriormente.

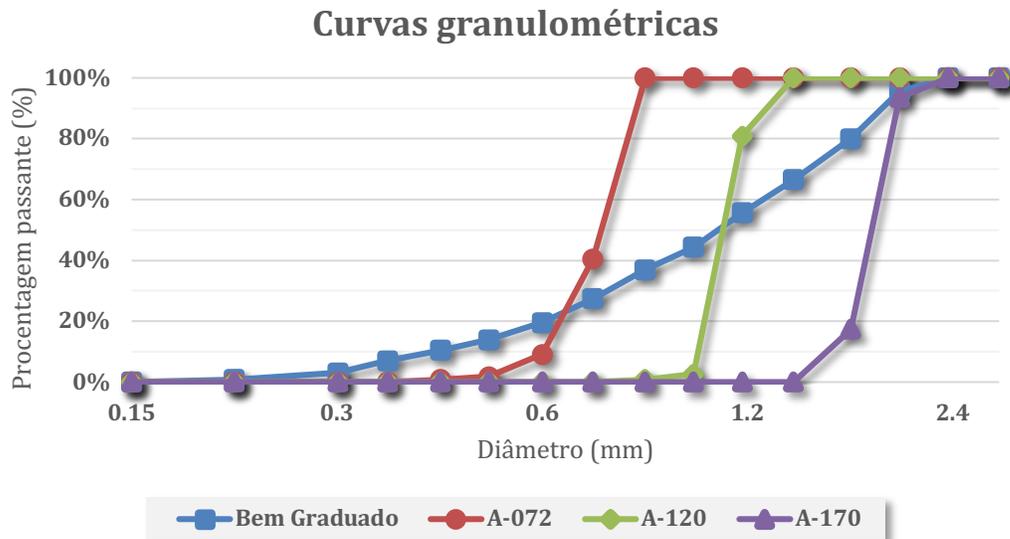


Figura 5.7 – Curvas granulométricas utilizadas
(Girardi, 2016)

Essa variação da curva granulométrica deve permitir a observação de como o DEM se comporta com a variação dos parâmetros, de modo a observar se a mudança nos parâmetros gera o resultado esperado pelas observações experimentais. Além disso, a variação deve permitir a observação de fenômenos que não podem ser vistos experimentalmente, como a tensão nos grãos do material fora da superfície de ruptura.

5.5 PROCESSAMENTO DAS SIMULAÇÕES

Devido à grande demanda computacional imposta pelas condições e geometria das simulações, optou-se por realizar o processamento utilizando máquinas virtuais em ambiente *Cloud*. Essa opção, apesar de incorrer em custo de processamento, gera algumas vantagens como possibilidade de maior capacidade computacional e também processadores desenhados para realizar processamento em tempo integral, sem aquecimento excessivo ou queda de produtividade.

A opção, no entanto, incorre em algumas barreiras. Entre elas estão custos de processamento e necessidade de programação em ambiente Linux/UNIX, ainda que não seja necessário conhecimento extensivo desse.

A transição para ambiente *Cloud* é facilitada até certo ponto pelo fato de o *software* YADE só ser executável a partir de sistemas operacionais Linux como Ubuntu. Sistemas

operacionais Linux e UNIX como o *MacOS* possuem maior facilidade em acessar máquinas virtuais de provedores de Cloud utilizando o comando SSH no terminal do dispositivo. Já computadores com sistema operacional Windows e preciso utilizar um software livre chamado PuTTY para simular um terminal. Além disso utilizar sistemas operacionais Linux tem menores custos pois o sistema operacional é livre.

Ao todo foi utilizado um conjunto de três comandos adicionais para possibilitar a execução do framework YADE em ambiente *Cloud* adequadamente. O comando SSH, um protocolo para acessar dispositivos remotos, o comando SCP, protocolo para transferência de arquivos entre os dispositivos, e outro comando chamado *Disown* para realizar comandos sem um a presença de um terminal ativo.

Os comandos SSH e SCP são executados entre diferentes computadores e por isso possuem procedimentos de segurança adicionais, e por requisito dos provedores de Cloud só podem ser executados mediante chave de 2048 *bits* ou usuário e senha. A utilização da chave de 2048 *bits*, constituída por um arquivo permite maior nível de automatização pois pode ser inserida juntamente com o comando, ao passo que o segundo pede o input manual do usuário, por esse motivo essa foi a solução adotada e mostrada nos exemplos a seguir.

O comando SSH tem a seguinte sintaxe:

```
ssh -i "key" "user@ip"
```

É necessário inserir o caminho para chave de acesso de 2048 *bits* (*key*), nome do usuário da máquina virtual (*user*) cadastrado no momento da criação da mesma e o IP da máquina virtual (*ip*). É necessário também permitir acesso ao dispositivo pela porta número 22, utilizada para este protocolo, alguns provedores tem como opção padrão a porta aberta. Após a execução adequada do comando é concedido acesso ao terminal e o *software* YADE pode ser instalado e executado.

Já o comando SCP, utilizado na transferência de arquivos tem a seguinte sintaxe para transferir arquivos do computador para a máquina virtual:

```
scp -i key -r origin user@ip:destination
```

E a seguinte sintaxe para transferir da máquina virtual de volta para o computador:

```
scp -i key -r user@$ip:origin destination
```

Onde *key* é a chave de acesso de 2048 *bits*, *user* é o usuário, *ip* é o endereço de IP da máquina virtual, *origin* é o caminho do arquivo a ser transferido, para transferir todos os

arquivos de um diretório é necessário inserir o caminho do diretório seguido de um asterisco (*) e *destination* é o caminho do diretório de destino dos arquivos transferidos.

O comando é de grande importância para transferir os códigos a serem executados na máquina virtual e também para transferir os resultados obtidos das simulações.

Por meio do acesso utilizando o protocolo SSH mencionado anteriormente a máquina virtual executa os comandos fornecidos apenas mediante a continuidade da conexão entre os dois dispositivos. O comando *Disown* é utilizado para contornar essa situação, ele desassocia o comando fornecido à máquina virtual ao computador que solicitou o comando desse modo o comando passa a ser executado independentemente da conexão entre os dois.

Para executar o comando são necessárias duas ações. Primeiro é necessário executar o comando a ser desassociado com “&” adicionado no seu término, desse modo o comando é executado em segundo plano e outros comandos podem ser executados em paralelo. Em seguida é necessário executar o comando *Disown* com a seguinte sintaxe:

```
disown -h
```

O comando desassocia o ultimo item da lista de comandos sendo executados em paralelo, removendo o vinculo com o computador de modo a continuar a execução após o término da conexão.

No anexo III do presente trabalho é mostrado um exemplo de *script* escrito utilizando a linguagem Bash (*Bourne-again shell*) utilizada em sistemas operacionais Linux e UNIX. O *script* foi utilizado para fazer upload de arquivos para a máquina virtual, realizar as simulações e posteriormente baixar o resultado. Os três comandos citados anteriormente são utilizados no *script*.

No presente projeto optou-se por utilizar o provedor *Google Cloud Platform*. Foram experimentadas também soluções oferecidas pela *Amazon Web Services (AWS)* e *Microsoft Azure*. Todas elas apresentaram recursos similares com funcionamento adequado dos procedimentos citados anteriormente. A escolha final se deu pela melhor oferta de preço e disponibilidade.

A capacidade de processamento em máquinas virtuais é alocada na forma de CPUs virtuais, sendo o seu comportamento ao longo da execução semelhante ao de um núcleo. Desse modo o *software* YADE, desenvolvido com tecnologia *Threading* para dividir a carga entre diferentes núcleos em paralelo, distribui a carga entre diferentes CPUs virtuais.

Segundo o provedor um CPU virtual tem capacidade de processamento próxima a um computador comum. Essa relação foi verificada em testes na implementação da simulação observando desempenho relativamente próximo entre um computador e um CPU virtual.

Optou-se por utilizar uma máquina virtual com 24 CPUs virtuais, localizada nos EUA. A escolha permitiu ganho considerável na velocidade de processamento auxiliando principalmente na fase de teste e desenvolvimento de simulações, uma vez que erros e falhas poderiam ser corrigidos mais rapidamente.

6 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Utilizando a funcionalidade do YADE de gerar empacotamentos foram obtidos sistemas com grande quantidade de partículas, variando entre 10 mil e 170 mil esferas conforme mostrado na Tabela 6.1. A grande variação no número de partículas é ocasionada pela variação na granulometria do solo. A amostra de menor granulometria gerou (A-072) gerou o maior número e a de maior granulometria (A-170) o menor número.

Outro fator considerável é a massa gerada. Girardi (2016) utilizou 300g de material por ensaio, valor consideravelmente maior que o utilizado nas simulações numéricas. Essa variação se deu por dois fatores principais, o primeiro deles foi a utilização de um prato com profundidade reduzida de 1,05cm para 0,15cm resultando em um menor acúmulo de partículas no fundo deste e a realização de um ensaio com menor sobra de material, visto que um grande número de partículas adicionais gera grande custo computacional. A Tabela 6.1 mostra o número de esferas e a massa gerada.

Tabela 6.1 Empacotamento criado

Curva	Nº de Partículas	Massa (g)
Bem Graduado	19.807	63,30
A-072	173.263	87,89
A-120	44.450	84,19
A-170	10.440	84,53

Devido ao grande número de partículas foi possível gerar um empacotamento com curva granulométrica bem próxima da especificada pelo fabricante do material ensaiado experimentalmente. A Figura 6.1 mostra as curvas granulométricas obtidas pelo processo de empacotamento.

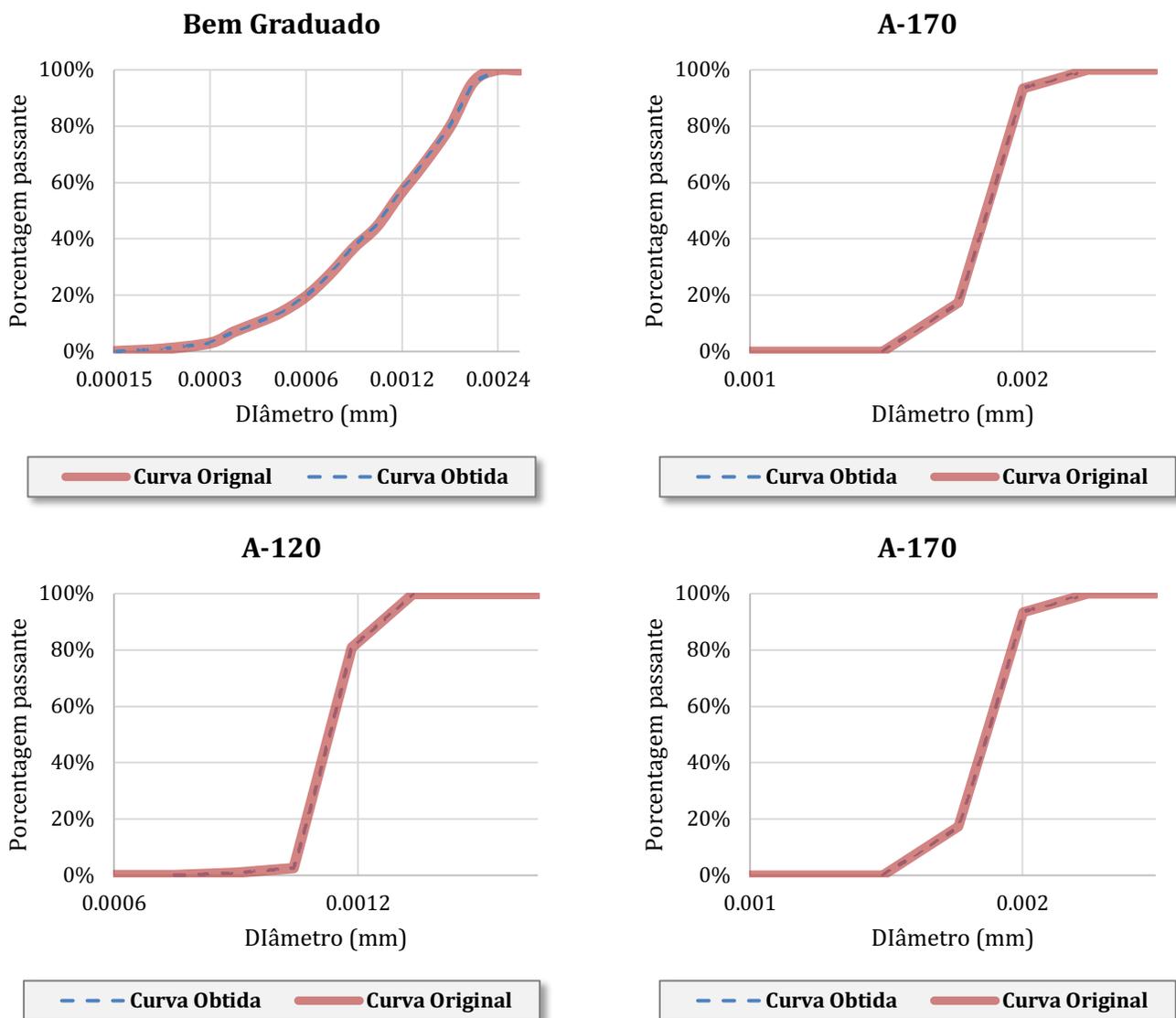


Figura 6.1 – Curvas granulométricas obtidas e especificadas

O número de partículas geradas teve influência no tempo de processamento para realizar a simulação. Apesar de todas as simulações terem requerido número de iterações similar, variando entre 100.000 e 110.000 o tempo de processamento teve grande variação, sendo o maior tempo 16 vezes superior ao menor tempo, conforme mostrado na Tabela 6.2. Esse fato mostra como um maior número de partículas tem grande influência no tempo por iteração.

Tabela 6.2 Tempo de processamento e número de iterações

Curva	Nº de Partículas	Tempo de processamento (h)	Iterações
Bem Graduado	19.807	2,11	110.000
A-072	173.263	18,07	100.000
A-120	44.450	4,19	100.000
A-170	10.440	1,12	100.000

A Figura 6.2 ilustra o tempo de processamento em função do número de partículas nas simulações executadas.

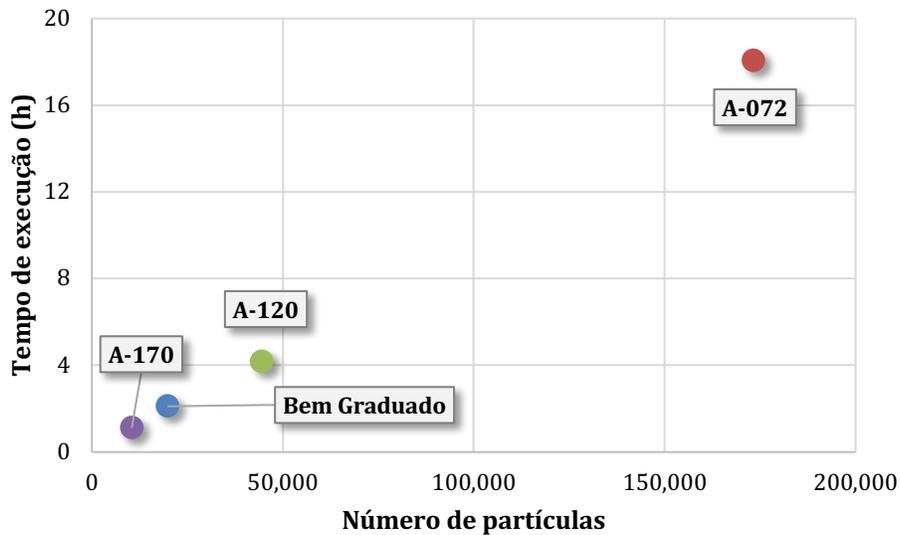


Figura 6.2 – Curvas granulométricas obtidas e especificadas

A utilização do equilíbrio das forças como instrumento de parada da simulação por meio da ferramenta oferecida pelo YADE forneceu um critério adequado. A Figura 6.3 demonstra a altura final de cada partícula em função da distância ao eixo central do prato para as simulações realizadas. É possível observar o formato do acúmulo de partículas ao fim da simulação.

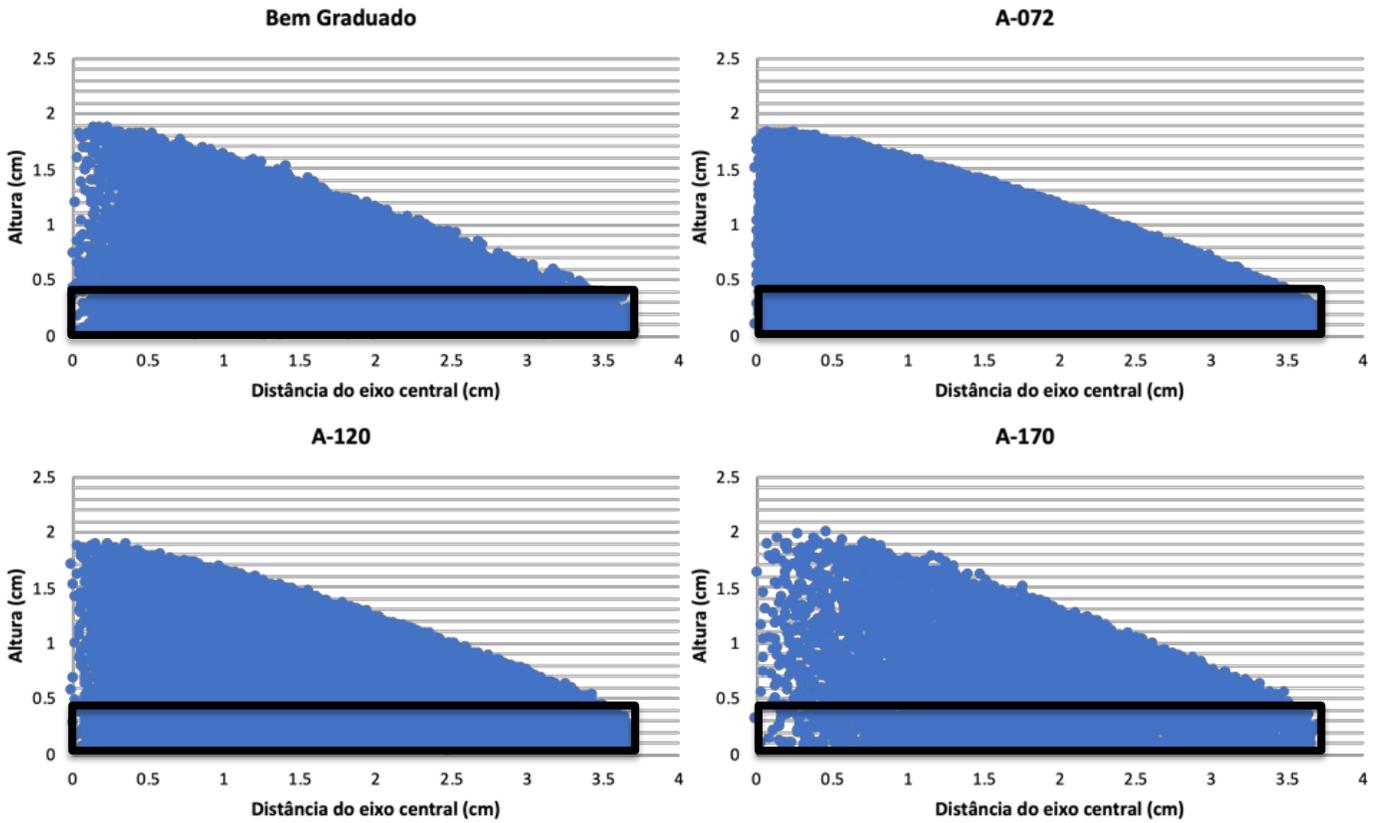


Figura 6.3 – Posição final das partículas

Os valores obtidos para as alturas máximas ao final da simulação foram consideravelmente próximos com uma variação de apenas 1,7mm entre o maior e o menor valor. A Figura 6.4 ilustra os resultados obtidos.

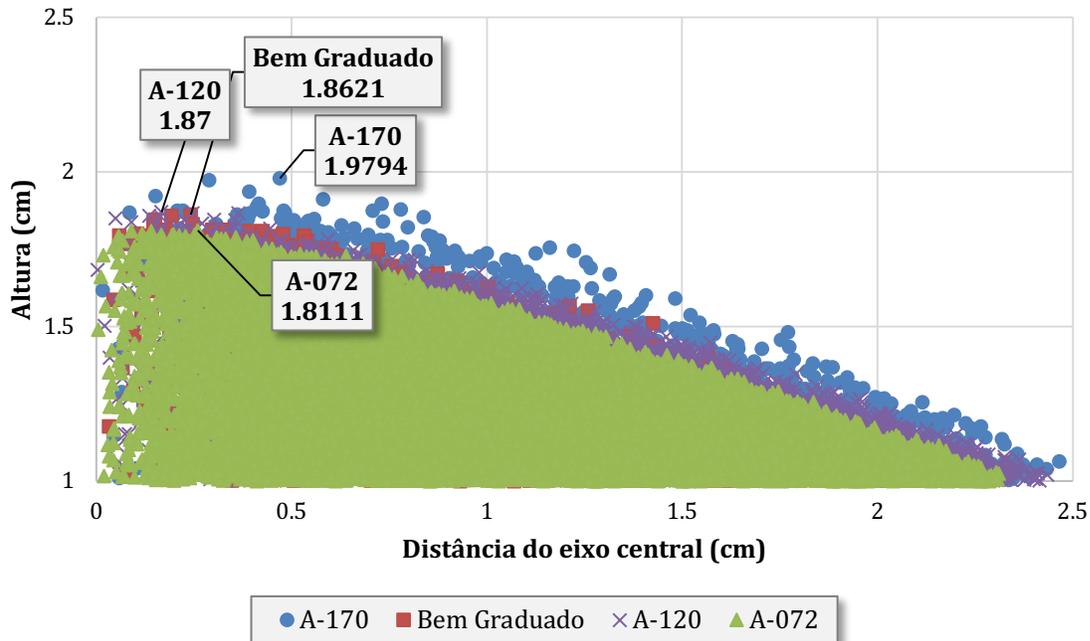


Figura 6.4 – Alturas máximas

Com as alturas máximas foi subtraída a altura do prato e foi calculada a altura acima do prato somente (H). Com isso foi calculado o ângulo de repouso (Φ) conforme a equação 5.1. Os resultados obtidos são mostrados na Tabela 6.3.

Tabela 6.3 – Ângulos de repouso obtidos

Nome	Altura máxima (cm)	H (cm)	Φ (°)
Bem Graduado	1,862	1,712	24,63°
A-072	1,661	1,661	23,98°
A-120	1,720	1,720	24,73°
A-170	1,829	1,829	26,10°

Assim como foi utilizada a equação 5.1 para calcular o ângulo de repouso (Φ) a partir das alturas obtidas a mesma equação foi adotada para obter a altura necessária para obter o ângulo de repouso calculado por Girardi (2016). Isso permitiu uma comparação tanto em relação a altura como ao ângulo de repouso.

De modo geral os resultados obtidos foram próximos com uma diferença máxima de 7,8% na altura final da pilha e 6,8% no ângulo obtido, essa diferença foi obtida na curva

Bem Graduada. Essa diferença representa 1,44mm e 1,8° na altura e ângulo de repouso respectivamente. A Tabela 6.4 mostra a comparação entre os resultados.

Tabela 6.4 – Ângulos de repouso utilizando DEM e Experimentais

Curva	DEM		Experimental		Δ		Δ (%)	
	H (cm)	Φ (°)	H (cm)	Φ (°)	H (cm)	Φ (°)	H	Φ
Bem Graduado	1,712	24,63°	1,857	26,43°	0,144	1,80°	7,8%	6,8%
A-072	1,661	23,98°	1,549	22,53°	0,112	1,45°	7,2%	6,4%
A-120	1,720	24,73°	1,613	23,36°	0,107	1,37°	6,6%	5,8%
A-170	1,829	26,10°	1,752	25,13°	0,077	0,97°	4,4%	3,8%

Na curva Bem Graduada, que apresentou maior diferença entre resultado experimental e simulações, a maior fração granulométrica (2mm) se mostrou superior a diferença nas alturas finais da pilha (1,44m). As Figura 6.5 e Figura 6.6 ilustram as alturas máximas e ângulos de repouso obtidos nas simulações em relação aos resultados experimentais juntamente com a diferença percentual entre os resultados.

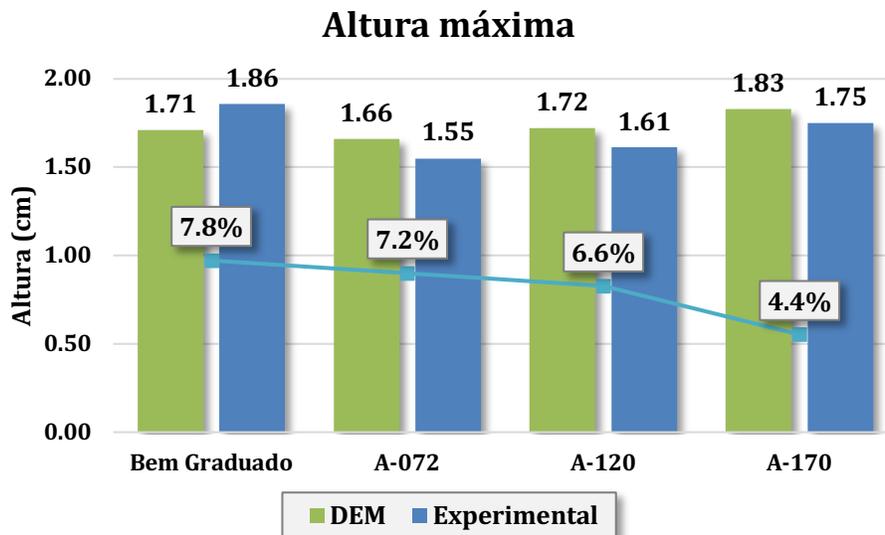


Figura 6.5 – Alturas máximas

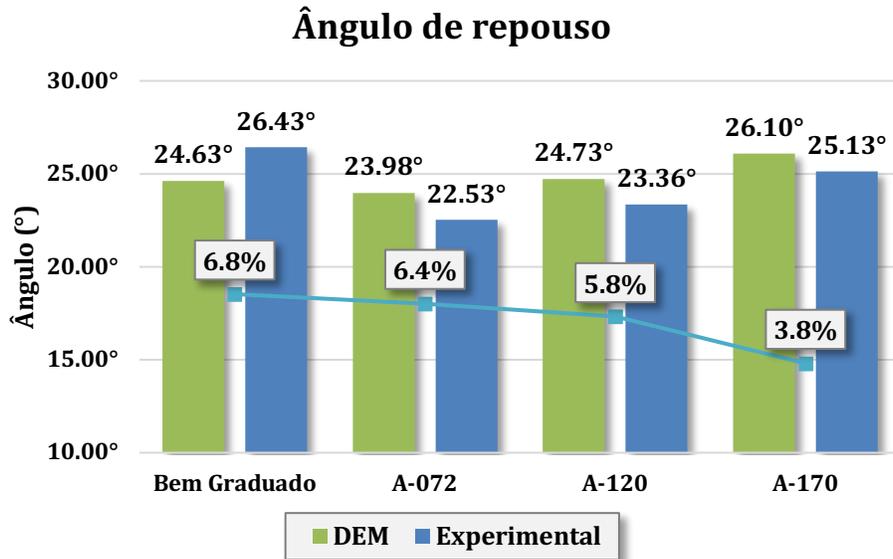


Figura 6.6 – Ângulo de repouso

A proximidade entre os resultados numéricos e experimentais demonstram que as adaptações realizadas na simulação numérica em relação ao ensaio experimental não tiveram grande interferência no resultado. As duas adaptações realizadas foram a diminuição no módulo de elasticidade das partículas e a diminuição na altura do prato para acúmulo das partículas.

Foram analisados também o comportamento das partículas ao longo das simulações quanto a tensão normal, velocidade, e tensão cisalhante, o comportamento das partículas é mostrado nas Figura 6.7, Figura 6.8 e Figura 6.9 respectivamente.

A maior concentração de tensão normal se deu na base do cone devido à formação de um arco de tensão pela descida das partículas. O maior gradiente de velocidade se concentrou na saída das partículas do cone e a maior concentração de tensão cisalhante foi nas laterais do cone onde ocorria a descida das partículas.

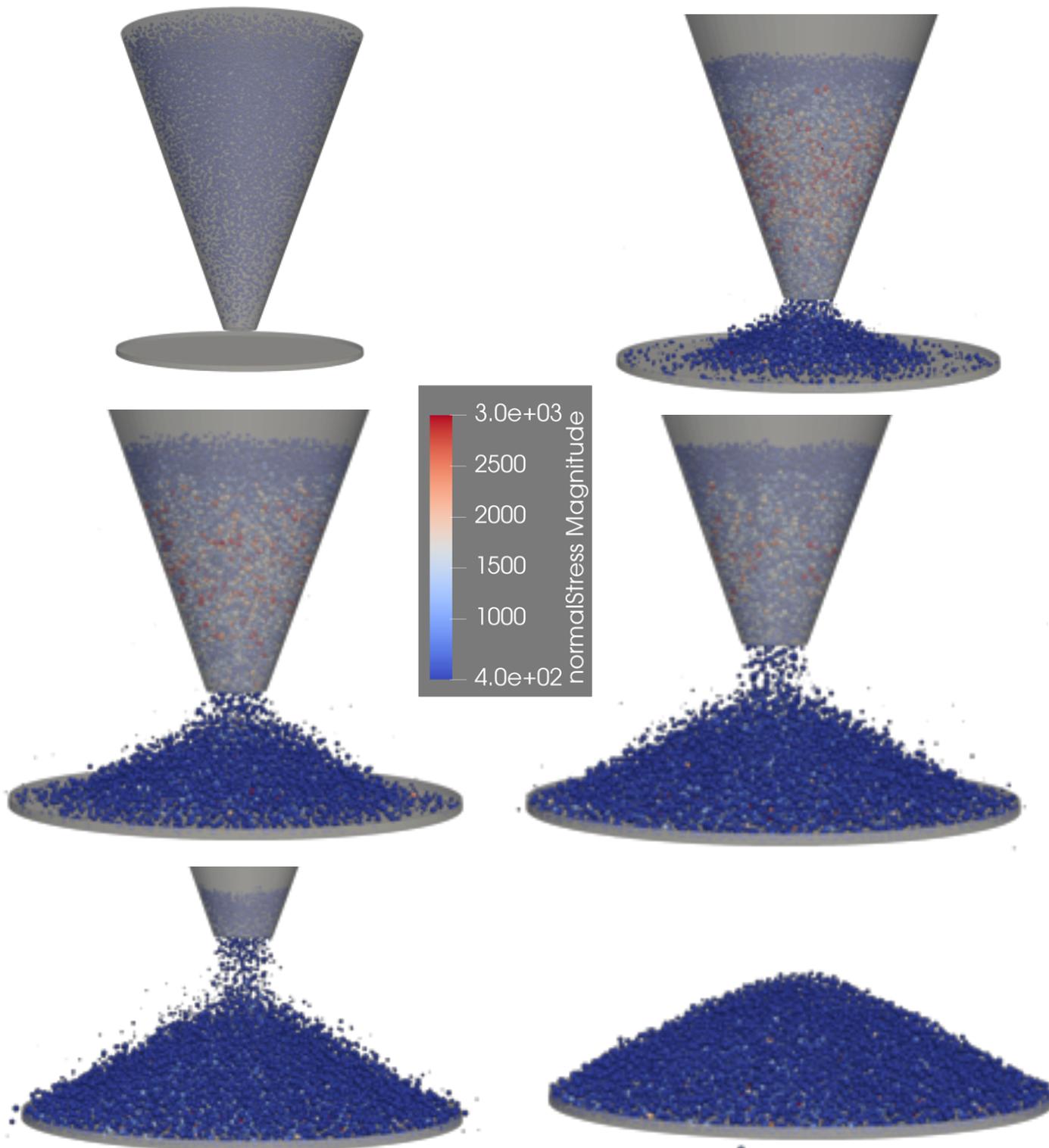


Figura 6.7 – Tensões normal nas partículas (Pa)

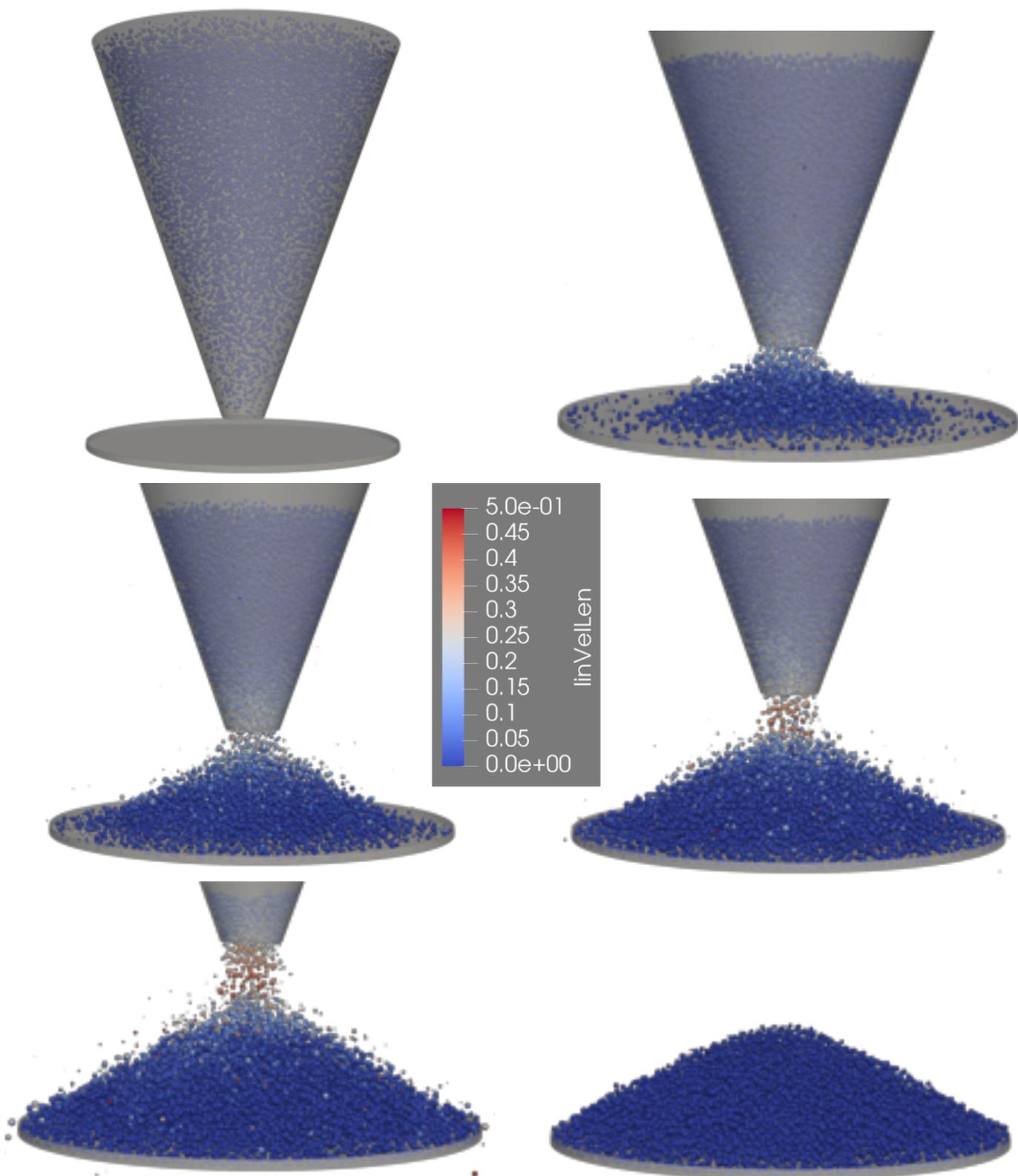


Figura 6.8 – Velocidade das partículas (m/s)

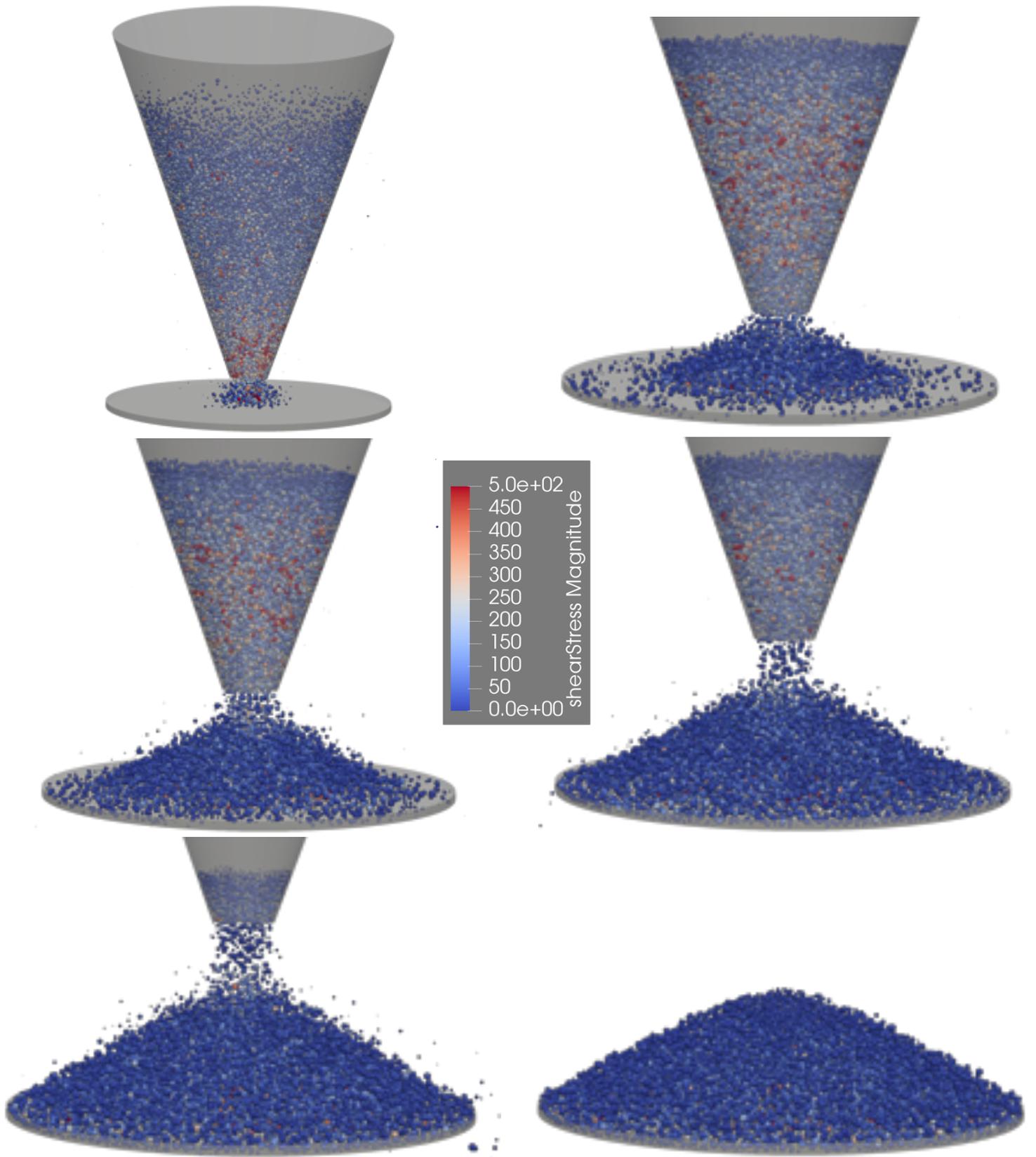


Figura 6.9 – Tensão cisalhante (Pa)

7 CONCLUSÕES E SUGESTÕES

A fim de atingir o objetivo proposto de verificar a validade do método dos elementos discretos (DEM) como uma alternativa para simular o comportamento do solo a partir de sua microescala, em especial no que concerne ao atrito no solo foram realizadas simulações numéricas utilizando o método e os resultados obtidos foram comparados com os resultados experimentais. A seguir são mostradas as conclusões do trabalho executados e sugestões para trabalhos futuros.

7.1 CONCLUSÕES

- O método dos elementos discretos se mostrou como uma ferramenta capaz de simular adequadamente o ensaio experimental proposto. Isso foi observado pela proximidade com o resultado experimental obtido com diferença máxima entre resultado numérico e experimental de 6,8%.
- O material ensaiado experimentalmente situa-se bem próximo de uma situação idealizada, sendo ele esférico e industrializado de modo a manter suas propriedades relativamente constantes. Desse modo a simulação numérica foi facilitada em relação a uma situação com material natural, ainda assim havia uma grande dificuldade imposta pelo elevado número de partículas, chegando a 170 mil. Desse modo pode ser observada a capacidade do método em ensaiar situações próximas do usual, mas ainda no escopo de materiais idealizados.
- Outro fato relevante foi a necessidade de grande capacidade computacional para realizar simulações. Foi utilizado grande capacidade computacional por meio de provedores de máquinas virtuais e ainda assim certas simulações ainda demandaram grande tempo de processamento, chegando a 18 horas para seu completo processamento. Desse modo ficou evidente a necessidade de utilização de máquinas com grande capacidade para realizar simulações robustas utilizando o DEM.
- Foi possível analisar grandezas não mensuráveis experimentalmente como tensão normal e tensão cisalhante. Por meio dessas foi possível notar o aparecimento de arco de tensão na base do cone e a concentração de tensão

cisalhante na lateral do cone. Além disso foi possível notar uma baixa velocidade das partículas na saída do cone, indicando uma altura de queda adequada.

- Sentiu-se certa dificuldade em implementar simulações iniciais. Diferentemente de outros métodos mais estabelecidos o DEM conta com uma menor comunidade de usuários e soluções para implementar as simulações. Desse modo simulações frequentemente requerem criação de Scripts de certa complexidade e conhecimento de linguagens de programação, tornando o método menos acessível e com maior necessidade de tempo para ser implementado.
- Tanto devido à demanda de capacidade computacional como à dificuldade de implementação do DEM, o mesmo acaba se mostrando mais difícil que a realização do ensaio experimental em si. Desse modo a principal vantagem do método em relação ao ensaio experimental se mostrou a possibilidade de observar grandezas não mensuráveis no ensaio experimental como tensão normal e cisalhante.

7.2 SUGESTÕES

- Foi realizado apenas uma simulação para cada curva granulométrica ensaiada experimentalmente devido à dificuldade de processamento imposta pela simulação. Apesar de o ensaio numérico não contar com o erro experimental a realização de um maior número de simulações daria maior validade estatística aos resultados obtidos.
- Devido à configuração geométrica do experimento a altura da pilha ao fim do experimento variou próxima de 1,7 cm na curva com maior granulometria a maior fração granulométrica possuía 2mm, valor superior a 10% da altura final da pilha. Desse modo aumentar o diâmetro do prato colaboraria com uma maior precisão do ensaio apesar de requerer em grande aumento no número de partículas e conseqüentemente no tempo computacional. Outra possibilidade seria realizar esse aumento de diâmetro somente com as frações de maior granulometria que possuem menor quantidade de partículas.

- Segundo Taylor (1962) o ângulo de repouso é uma aproximação para o ângulo de atrito no material. Essa característica foi utilizada como justificativa para diminuir o módulo de elasticidade no material, visto que esse não possui grande influência no ensaio. Como pode ser visto os resultados se aproximaram do experimental e essa propriedade não aparentou ter grande influência. A realização de ensaios adicionais com variação no módulo de elasticidade pode ser uma maneira de verificar a validade dessa premissa utilizada.

REFERÊNCIAS BIBLIOGRÁFICAS

GOODMAN, R. E.; TAYLOR, R. L.; BREKKE, T. L. *A model for the mechanics of jointed rock*, 1968

CUNDALL, P. A.; STRACK O. D. L. *A discrete numerical model for granular assemblies*. Geotechnique, 1979

CUNDALL P. A. *A computer model for simulating progressive, large scale movement in blocky rock systems*, 1971

PANDE G. N.; BEER J. R. *Numerical Methods in Rock Mechanics*, 1990

O'SULLIVAN C. *Particulate Discrete Element Modeling*, 2011

PÖESCHEL T.; SCHWAGER T. *Computational Granular Dynamics*, 2004

MATUTTIS H. G.; CHEN J. *Understanding the Discrete Element Method*, 2014

MUNJIZA A. *The Combined Finite-Discrete Element Method*, 2004

BOBET A. *et. al. Numerical models in discontinuous media: Review of advances for rock mechanics applications*, 2009

BAGI K. *An algorithm to generate random dense arrangements for discrete element simulations of granular assemblies*, 2005

F. V. DONZÉ *et. al. Numerical simulation of drained triaxial test using 3d discrete element modeling*, 2009

SIMPSON B., TATSUOKA F. *Geotechnics: the next 60 years*, 2008

YU A. *Discrete element method an effective way for particle scale research of particulate matter*, 2004

JIANG M.; LEROUEIL S.; KORAD J. *Insight into shear strength functions of unsaturated granulates by DEM analyses*, 2004

FENG Y. T.; HAN K.; OWEN D. *Filling domains with disks: An advancing front approach*, 2003

CUI L.; O'SULLIVAN C. *Analysis of a triangulation-based approach for specimen generation for discrete element simulations*, 2003

ZHU H.; ZHOU R.; YANG R.; YU A. *Discrete particle simulation of particulate systems: Theoretical developments*, 2007

RAPAPPORT D. *The art of molecular dynamics simulation*, 2004

WALTON O.; BRAUN R. *Viscosity, granular-temperature and stress calculations for shearing assemblies of inelastic, frictional disks*, 1986

VALERA R., PERÉS I.; VANMAERCKE S.; MORFA C.; CORTÉS L.; CASAÑAS H. *Modified algorithm for generating high volume fraction sphere packing*, 2015

PERÉS I; MUNIZ M.; MORFA C.; CORTÉS L.; CASAÑAS H.; CARVAJAL H.; VALERA R. *Contributions to the generalization of advancing front particle packing algorithms*, 2015

MORFA C.; PERÉS I; MUNIZ M.; CORTÉS L.; VALERA R.; OÑATE E. *Systemic characterization and evaluation of particle packings as initial sets for discrete element simulations*, 2017

MORFA C.; PERÉS I; MUNIZ M.; CORTÉS L.; VALERA R.; OÑATE E.; CASAÑAS H. *General advancing front packing algorithm for the discrete element method*, 2016

PERÉS I; MORFA R.; MUNIZ M.; VALERA R. *Dense packing of general-shaped particles using a minimization technique*, 2016

MORFA C.; PERÉS I; MUNIZ M.; CORTÉS L.; VALERA R.; OÑATE E.; CASAÑAS H. *Advances in particle packing algorithms for generating the medium in the Discrete Element Method*, 2018

ROJEK J.; ZUBELEWIZ A.; MADAN N. NOSEWICZ S. *The discrete element method with deformable particles*, 2018

ANEXO I – Script principal

```
utils.readParamsFromTable(curva=3)

import pandas as pd
import numpy as np
from yade import pack
from matplotlib import pyplot as plt
import os
import sys
import shutil
import imp
import time
from yade.params import table
t_0=time.time()

#Trocar para diretorio do script
os.chdir('/home/matheus/simulation')

#Importar arquivo com funcoes
FunctionsYade=imp.load_source('FunctionsYade','FunctionsYade.py')

#unidades adotadas
#distancia: m
#massa: kg
#densidade: kg/m3
#velocidade: m/s
#modulo de Young: Pa

#Definir precisao para imprimir do np
np.set_printoptions(precision=2)

#Abrir arquivo com a granulometria
dados=pd.read_csv('granulometria.csv',sep=';',index_col='Diametro')

nomes=np.array(dados.columns.values)

diametros=np.array(dados.index.values,dtype='float')

curvas=[np.array(dados[a],dtype='float')/100 for a in nomes]

#Pasta para guardar resultados
destino='Resultados-curva-%i'%(int(table.curva))
FunctionsYade.create_folder(destino)

#Definir a pasta destino como padrao
os.chdir(destino)

#Criar pasta para videos deletando a antiga
FunctionsYade.create_folder('videos')

path_raios='/home/matheus/Desktop/raios.csv'
```

```

# Variaveis
#dados do prato
radius_prato=0.0747/2 #Largura do prato
alt_prato=0.006/4 #Altura do prato

#Dados do cone
radiusTop = 0.0747/2 #Raio do cone no topo
radiusBottom = 0.005#/2 #Raio do cone no base
coneHeight = 0.09#/2 #Altura do cone
zCone = alt_prato+0.005 #Altura na base do cone
vel_cone=0.005*1.5 #Velocidade para levantar o cone

#Dados do empacotamento
#massa=0.3 #Massa desejada no ensaio
#alt_emp=0.05 #Altura do empacotamento
#lado_emp=radiusTop*2**0.5 #Lado do empacotamento gerado
curva_escolhida=int(table.curva)#0 #Numero da curva escolhida de 0 a 4
curva_original=curvas[curva_escolhida]
#numero_sep=11

# Dados do modelo de contato
young=17.2e3 #Valor da bibliografia (original 17.2e9) alterado para
facilitar simulacao
poisson=0.22 #Valor da bibliografia
density=2500 #Segundo o fabricante
frictionAngle=43.5*np.pi/180 #Para resultar em um coeficiente de friccao de
0.95
#passo_tempo=0.000001

def checkUnbalanced():
    if unbalancedForce()<.02:
        #Gravar resultador em arquivo de log
        with open('log.txt','w') as f:
            f.write('particulas: %i\n'%(len(sp)))
            f.write('tempo de execucao: %f s\n'%(time.time()-t_0))
            f.write('iteracoes: %i\n' %(O.iter))
            f.write('unbalanced: %f\n' %(unbalancedForce()))
            f.write('timestep: %f\n' %(O.dt))
            f.write('Virtual Time: %f\n' %(O.dt*O.iter))
            FunctionsYade.position_csv(O,Sphere,'posicoes.csv')
        #Sair do programa
        sys.exit()
        #Gravar posicoes em arquivo CSV
        FunctionsYade.position_csv(O,Sphere,'posicoes.csv')

#Parametros na simulacao
O.engines=[
    #Resetar as forcas no sistema
    ForceResetter(),
    #Ferramenta para deteccao de contato
    InsertionSortCollider([[Bo1_Sphere_Aabb(),Bo1_Facet_Aabb()]),

```

```

InteractionLoop(
    [Ig2_Sphere_Sphere_ScGeom(),Ig2_Facet_Sphere_ScGeom()],
    [Ip2_FrictMat_FrictMat_MindlinPhys()],
    [Law2_ScGeom_MindlinPhys_Mindlin()
    ],
    #Inserir a gravidade
    NewtonIntegrator(damping=0.1,gravity=(0,0,-9.81)),
    #Funcao para subir o cone
    PyRunner(command= "FunctionsYade.liftCone(0,vel_cone,indexCone)",
        virtPeriod=0.01, nDo = 2),
    #Funcao para parar o cone
        PyRunner(command= "FunctionsYade.stopCone(0,indexCone)",
            virtPeriod=5, nDo = 2),
    #Funcao para apagar esferas que sairam do limite
    PyRunner(command=
        "FunctionsYade.remove(0,Sphere,radius_prato,alt_prato)", iterPeriod =
        100000),
    #Funcao para parar pelo criterio de energia
    PyRunner(command= "checkUnbalanced()", iterPeriod = 5000),
    #Gravador
    VTKRecorder(virtPeriod=0.016,recorders=['all'],fileName='videos/video'),
        #Ferramenta do yade para atualizar passo de tempo
    GlobalStiffnessTimeStepper()
]

#Criar Material das esferas
O.materials.append(FrictMat())
mat=FrictMat(young=young, poisson=poisson,
    density=density,frictionAngle=frictionAngle,label='Friction')

#Inserir material no modelo
O.materials.append(mat)

#Inserir limites
indexCone=FunctionsYade.create_limits(0,radius_prato,alt_prato,radiusTop,radiusBottom,coneHeight,zCone,geom)

#Criar empacotamentos
sp=FunctionsYade.create_pack(0,sphere,pack,radius_prato,alt_prato,radiusBottom,radiusTop,zCone,coneHeight,diametros,curva_original)
print('Numero de esferas: %i'%len(sp))

#Gravar diametros do empacotamento em arquivo
FunctionsYade.sphere_diam_csv('diametros.csv',sp)

#Iniciar simulacao
O.run()
utils.waitForBatch()

```

Anexo II – Script Auxiliar

```
import pandas as pd
import numpy as np
from yade import pack
import os
import sys
import shutil

unidade_diametro={'mm':0.001,'cm':0.01,'dm':0.1,'m':1}

def
create_limits(0,radius_prato,alt_prato,radiusTop,radiusBottom,coneHeight,zC
one,geom):
    #Inserir limites
    # Prato para deposicao
    O.bodies.append(geom.facetCylinder(center = (0,0,alt_prato/2), radius =
    radius_prato, height= alt_prato, segmentsNumber=80,
    wallMask=6,material='Friction'))
    # Cone
    indexCone = O.bodies.append(geom.facetCone((0,0,zCone+coneHeight/2),
    radiusTop, radiusBottom, coneHeight,
    wallMask=4,material='Friction',segmentsNumber=80))
    return indexCone

#Funcoes para mover e parar o cone
# Levantar o cone
def liftCone(0,vel_cone,indexCone):
    for i in indexCone:
        O.bodies[i].state.vel = (0, 0, vel_cone)

# Parar o cone
def stopCone(0,indexCone):
    for i in indexCone:
        O.bodies[i].state.vel = (0, 0, 0)

#Funcao para remover particulas que sairam do dominio
def remove(0,Sphere,radius_prato, alt_prato):
    for a in O.bodies:
        if isinstance(a.shape,Sphere):
            pos=a.state.pos
            dist_cent=(pos[0]**2+pos[1]**2)**0.5
            if dist_cent>radius_prato or pos[2] < -alt_prato:
                O.bodies.erase(a.id)

#Criar uma pasta nova removendo a antiga
def create_folder(nome):
    try:
        shutil.rmtree(nome)
        os.mkdir(nome)
    except:
        try:
```

```

        os.mkdir(nome)
    except:
        pass

#Passar esferas para arquivo CSV
def sphere_diam_csv(path,sp):

    with open(path,'w') as fi:
        for c,r in sp:
            fi.write('%f\n'%(2*r))

def
create_pack(O,sphere,pack,radius_prato,height_prato,radius_base_cone,radius
_topo_cone,z_cone,height_cone,diametros,cum_dist):

    output=[]

#Criar empacotamento do cone
sp_cone=pack.SpherePack()
sp_cone.makeCloud((-radius_topo_cone,-radius_topo_cone,z_cone),
(radius_topo_cone,radius_topo_cone,
z_cone+height_cone),psdSizes=diametros,psdCumm=cum_dist,porosity=0.01,s
eed=1)#,num=5000)

for c,r in sp_cone:
    pos=(c[0]**2+c[1]**2)**0.5
    lim_cone=(c[2]-z_cone)*(radius_topo_cone-radius_base_cone)/
    height_cone+radius_base_cone
    if pos+r<lim_cone:
        O.bodies.append(sphere(c,r,material='Friction'))
        output.append([[c[0],c[1],c[2]],r])
        #pass

return output

def position_csv(O,Sphere,path):
    with open(path,'w') as fi:
        for a in O.bodies:
            if isinstance(a.shape,Sphere):
                pos=a.state.pos
                fi.write('%f;%f;%f\n'%(pos[0],pos[1],pos[2]))

#Checar equilibrio de forcas
def checkUnbalanced(O):
    if unbalancedForce(<.05:
        O.pause()
        sys.exit(0)

```

Anexo III – Script BASH

```
ip="" #IP maquina virtual a ser acessada
key="" #Link da chave de seguranca
sim_folder="" #Pasta com arquivo para fazer upload
out_folder="" #Pasta para exportar output
user= #Usuario da maquina virtual

#Pedir input com opcao para atualizar arquivo na maquina virtual
read -p "Copiar script para VM? (y/n) " copiar

if [ $copiar = "y" ];then
#Fazer upload do arquivo na maquina com SCP
  scp -i $key -r $sim_folder* $user@$ip:/home/$user/simulation
fi

#Optar por realizar comando Disown
read -p "Disown file? (y/n)? " disown

if [ $disown = "y" ];then
  #Entrar na maquina virtual com SSH
  ssh -i "$key" "$user@$ip" << EOF
  #Trocar para diretorio da simulacao
  cd simulation
  #Realizar YADE utilizando 24 nucleos
  yade-batch --plot-update -1 --job-threads 24 curvas.txt ini_yade.py&
  #Utilizar comando Disown no arquivo
  disown -h
EOF
elif [ $disown = "n" ];then
  #Entrar na maquina virtual com SSH
  ssh -i "$key" "$user@$ip" << EOF
  #Trocar para diretorio da simulacao
  cd simulation
  #Realizar YADE utilizando 24 nucleos
  yade-batch --plot-update -1 --job-threads 24 curvas.txt ini_yade.py
EOF
fi

#Pedir input com opcao para baixar output da maquina virtual
read -p "Download output? (y/n) " download

if [ $download = "y" ];then
  #Baixar arquivo via SCP
  scp -i $key -r $user@$ip:/home/$user/simulation/ $out_folder
fi
```