

Master thesis

DEM-simulations and theory of the HCP

Author: M.P.W. Drummen

Supervised by: Dr. S. Luding
O. Mouraille

Faculty: NSM, DCT, TU Delft

Keywords: Strain definitions, Granular media,
DEM simulations

February 2006



Delft University of Technology

Summary

This thesis deals with the description of the static and kinematic properties of a structured mono-dispersed frictionless 3D granular packing by means of both theoretical derivations and DEM simulations. The packing used for this research is the Hexagonal Close Packing (HCP), which is a common crystal structure in nature. The properties of interest are the volume fraction of the packing, the contact network of the spheres, the Elastic Modulus Tensor (EMT) and the potential energy density. The aim is to evaluate how well the properties of a granular packing can be estimated with DEM simulations and what methods are required to improve the estimations. The simulations were performed with both periodic and wall boundary systems.

For the simulations in the periodic boundary system the estimated properties of the HCP correspond very well with the theoretically derived properties. For the wall boundary systems the deviations from the theoretical predictions are generally large. However, it is shown that with some simple methods reasonable estimations can be made of the volume fraction and EMT of a ‘perfect’ HCP from a relatively small HCP in a wall boundary system.

The second goal of this thesis is to evaluate the performance of three strain definitions described in literature that are claimed to be valid for 3D granular assemblies. This must lead to a recommendation for which definition is the most suitable to implement in the DEM data processing software. The first strain definition was proposed by Bagi and is based on the replacement of the grains in the system by equivalent continua. The second method uses the best-fit approach, where the average displacement of all particles is used to define the strain tensor. The third and last method was proposed by Satake and like the first method, the definition is based on the division of the granular assembly in smaller sub-volumes.

All three strain definitions are capable of accurately describing the strain tensor based on DEM simulations with uniaxial compression tests with HCP’s in periodic boundary systems. For the wall boundary systems, the deviations are generally very large. The main reason is that it was not possible to base the description of the strain tensor on the displacements of all the spheres in the packing; a small selection of spheres was used instead, which are in the case of wall boundary systems not representative for the entire packing, in contrast with periodic boundary systems. Since the best-fit method is the fastest and most straightforward of the three strain definitions, its implementation in the DEM software is preferred over the other two methods. However, more research on more complex packings should be done before it can be judged which strain definition performs best under all conditions.

Contents

Summary	i
1 Introduction	1
1.1 Background and motivation of the research	1
1.2 Goal of the thesis	2
1.3 Approach and outline of the thesis	2
2 Stress and strain definitions for granular media	4
2.1 Introduction	4
2.2 Strain definitions based on an equivalent continuum	7
2.2.1 Bagi's definition of stress and strain	7
2.2.2 Kruyt & Rothenburg's definition of strain	16
2.2.3 Kuhn's definition of strain	18
2.2.4 Cambou's and Dedecker's definition of strain	20
2.3 Strain definitions based on best-fit approach	22
2.3.1 Definition based on particle translations	23
2.3.2 Definition based on contact deformations	25
2.3.3 Definition including particle rotations	27
2.4 The Satake strain	28
2.5 Conclusions and discussion	31
3 Theory of the HCP	33
3.1 Introduction	33
3.2 The structure	33
3.2.1 The contact network	34
3.3 The elastic modulus tensor	37
3.4 The unit cell	41
3.5 The volume fraction (density)	41
3.6 Potential energy density	44
3.6.1 Contact type 1	45
3.6.2 Contact type 2	47
3.6.3 Contact type 3	48
3.6.4 Total potential energy of the unit cell	49
4 Discrete Element Model simulations	51
4.1 Introduction to DEM	51
4.1.1 Numerical time-step	51
4.1.2 Physical interpretation	52
4.1.3 The compression tests	53
4.2 Preparation of the packing	54
4.2.1 Periodic boundaries	56
4.2.2 Wall boundaries	59

4.3	The simulations	62
4.4	Tools for extracting results	64
4.4.1	Basic tools	64
4.4.2	Advanced tools	64
5	Results	66
5.1	Introduction	66
5.2	Properties of the HCP	66
5.2.1	The volume fraction (density)	66
5.2.2	The contact network	70
5.2.3	The kinetic and potential energy	73
5.2.4	The elastic modulus tensor	75
5.3	The strain tensor for an HCP	86
5.3.1	Bagi's strain definition	86
5.3.2	The best-fit strain based on particle translations	98
5.3.3	Satake's strain definition	102
5.3.4	Overview of results	110
6	Conclusions and discussion	112
6.1	The properties of the HCP	112
6.1.1	The volume fraction	112
6.1.2	The contact network	113
6.1.3	Kinetic and potential energy	113
6.1.4	The elastic modulus tensor	113
6.2	The strain tensor for an HCP	114
7	Recommendations	116
	References	118
	Appendices	A
A	Derivation of u for one contact	A
B	Algorithms for defining an HCP in Fortran	G
B.1	Periodic boundaries	G
B.2	Wall boundaries	H

1 Introduction

This master thesis deals with the research that has been done within the framework of the Chemical Engineering Master program. The research was done at the NanoStructured Materials (NSM) section (former Particle Technology section) of the Delft University of Technology, over the period of March 2005 - February 2006, under the supervision of Dr. S. Luding and O. Mouraille. This chapter gives an introduction to the background and motivation of the research. The goals are defined and the approach is explained, together with the outline of this thesis.

1.1 Background and motivation of the research

The description of the behaviour of granular assemblies under applied forces and deformations has been a strongly growing field of research in physics and material science over roughly the past two decades. This proved to be a difficult and challenging task, since granular assemblies are generally highly inhomogeneous. Variables like size distribution, particle shape and the randomness of the packing, among many other parameters, all have their effect on the kinematics of a granular medium. Using the basic principles from continuum mechanics as a reference point, constitutive relations were derived for discrete media. Because of the complexity of granular solids, the modelled granular systems were greatly simplified and idealized, aiming at a more basic description of stress-strain relations. Such simplifications can be obtained by making the system two dimensional, keeping the shape of the particle simple (disks in 2D and spheres in 3D, but some extensions to slightly more distorted shapes have also been modelled) and making the system mono-dispersed and regularly structured.

An important tool in modelling the mechanical behaviour of granular assemblies is the Discrete Element Model (DEM), developed in the 70's by Cundall and Strack [4]. It is a numerical model where the motion of all particles, usually disks in 2D and spheres in 3D, is determined for every time interval. In principle, DEM simulations can be used to determine all kinds of properties of granular assemblies. Such properties are:

- The volume fraction
- The potential energy density
- The contact network
- The stress and strain (rate) tensor
- The elastic modulus tensor

However, these properties cannot directly be obtained from the data output of the simulations and have to be derived. So far, these properties have mainly been derived for 2D granular assemblies with DEM. Simulations with 3D assemblies have also been performed, but the derived properties are often not accurate enough and methods to calculate these properties have to be improved. Especially the definition of strain for granular media has been a subject of debate over the past two decades. Various suggestions for strain definitions have been proposed in literature ([3], [5], [7]-[12], [14], [15]). Most of them focused on 2D assemblies, but some of the definitions apply to 3D systems as well.

1.2 Goal of the thesis

This thesis deals with the questions how well properties of 3D granular assemblies in DEM simulations are estimated and which strain definitions that have been proposed in literature are the most accurate and reliable. In order to be able to evaluate whether the obtained results from the simulations are correct, the simulations are performed with a system consisting of frictionless equi-sized spheres forming a Hexagonal Close Packing (HCP). For such regular structures, it is possible to theoretically derive the properties of interest, like the volume fraction, potential energy density, contact network and the elastic modulus tensor. Moreover, two of the three strain definitions that are tested in this thesis strongly rely on geometrical properties of the assembly for the division of the system in smaller cells. By using a structured packing, it is possible to theoretically derive the shapes of these cells and base the fundamental characteristics of these two definitions on these theoretically derived cell structures. If agreement is obtained between the theoretical predictions and the results obtained from the DEM simulation for the structured packing, the simulated granular assemblies can be made more complex and less ideal in a next step, such that the properties cannot be derived from theory any more. The long-term goal is to be able to rely on DEM simulations for the derivation of the properties of realistic (non-ideal, random) granular assemblies.

1.3 Approach and outline of the thesis

First, in the chapter ‘*Stress and strain definitions for granular media*’ an overview is given of the most relevant stress and especially strain definitions for granular media that have been proposed in literature over approximately the past ten years. The definitions that are also valid for 3D granular media will be tested for their accuracy in this thesis.

In the chapter ‘*Theory of the HCP*’, properties of the HCP will be derived theoretically. First, the structure of the packing will be introduced, from which the contact network can be derived. This contact network forms the

basis for the description of the elastic modulus tensor, which relates stress to strain. For deriving the volume fraction and potential energy density a unit cell will be defined. This unit cell is an elementary volume that is representative for the properties of the entire packing.

Next, a general introduction to the Discrete Element Model will be given in chapter 4, followed by a description of how the HCP was prepared for the simulations, both for periodic and wall boundary conditions. Furthermore, it will be explained how the simulations were performed and methods to retrieve results from the simulations will be discussed in this chapter.

The results are covered in two sections. In the first part, the properties that were theoretically derived in the chapter '*Theory of the HCP*' are now obtained from the DEM simulations. The effect of the size of the packing on its volume fraction will be determined and the contact network will be retrieved from the data output. The kinetic stability of a static HCP is evaluated, as well as its potential energy (density). The elastic modulus tensor is obtained in two different ways:

- Using the contact network from a static snapshot.
- Calculating the slope of the stress against strain graph, obtained from several snapshots.

In the second part of the results, the three strain definitions that are valid for 3D assemblies, as described in chapter 2, will be tested for the HCP. The three corresponding strain tensors were derived from the same simulations in order to be able to compare the results.

The conclusions and discussions follow in chapter 6. The packing properties obtained from the simulations will be compared to the theoretical predictions and the accuracy of the three strain definitions will be judged.

This thesis finalizes with some recommendations for improvements for processing information from DEM simulations. Furthermore, a recommendation will be given for which strain definition should be implemented in the DEM programme in order to get the most reliable and accurate approximation of the macroscopic strain tensor.

2 Stress and strain definitions for granular media

2.1 Introduction

In continuum mechanics and material science, deformations of solid bodies are described by stress-strain relations. These relations are called ‘*constitutive equations*’, which should contain all relevant information about the kinematic and mechanical characteristics of the material. The relation between stress loads and the strain field is given by the equilibrium equations of continuum mechanics. The relation between displacements and the strain field is given by the constitutive equations. These equilibrium and geometrical equations are well-defined in continuum mechanics. However this is not the case for granular assemblies. A granular medium is composed of distinct particles which displace independently from each other and interact only at contact points. The discrete and inhomogeneous character of such a medium thus results in a complex discontinuous behavior under conditions of loading and unloading.

While the definition for stress is relatively straightforward and well-accepted by researchers, strain is more complicated to describe. Over the past decades, several strain definitions have been proposed. Most of these definitions can be categorized into one of two groups:

- Strains based on equivalent continua.
- Strains based on best-fit approaches.

Both groups can be subdivided into various types of definitions. These versions are usually strongly related and under specific conditions even identical.

In chapter 2.2, the definitions that are based on equivalent continua will be discussed. These versions were proposed by Bagi [2], Kruyt & Rothenburg [8], Kuhn [11] and Cambou & Dedecker [3], [5]. Chapter 2.3 deals with definitions that are based on the best-fit approach. These definitions are based on particle translations [7], contact deformations [3] [12] or the relative translations of neighbouring grains [3]. A third type of strain definition proposed by Satake in [15] that does not fall in the former two categories is discussed in chapter 2.4. The conclusions and discussion are finally presented in chapter 2.5.

The nomenclature used differs very much from publication to publication and therefore it occurs that similar symbols have different meanings. To avoid confusion, a table is presented at the beginning of every (sub)chapter that gives an overview of the nomenclature used in the section indicated in the caption of the table.

In the derivation of the strain definitions described in chapters 2.2 and 2.4, the granular system is divided into subdomains using tessellation methods. Three frequently used tessellation methods are:

- Voronoi tessellation
- Dirichlet tessellation
- Delaunay network

A brief introduction to these methods will be given first to point out the similarities and differences.

Voronoi tessellation

The Voronoi tessellation method is named after Georgy Voronoi and it can be used for the decomposition of a metric space determined by distances to a specified discrete set of objects in the space, for example by a discrete set of points. For any set of points S in Euclidian space and for almost any point x , there is one point of S to which x is closer than to any other point of S . If S contains only two points, a and b , then the set of all points equidistant from a and b is a hyperplane. That hyperplane is the boundary between the set of all points closer to a than to b , and the set of all points closer to b than to a . It is the perpendicular bisector of the line segment from a and b . The set of all points closer to point c of set S than to any other point of S is the interior of a space called the ‘Voronoi cell’. The Voronoi tessellation can also be applied to a granular medium, but only when all the grains have the same size.

Dirichlet tessellation

The Voronoi and Dirichlet tessellations are often regarded as the same tessellation methods. This may be true for a set of points in Euclidian space, but for a granular medium the equality does not hold. The advantage of the Dirichlet tessellation over the Voronoi tessellation method for a granular assembly is that the grains do not necessarily have to be of the same size. Consider a set of non-intersecting circular grains in 2D or non-intersecting spherical grains in 3D. A domain can be assigned to each grain, consisting of those points which have a shorter or equal tangent to that grain than to any other grain. The common faces of the domains are the power planes in 3D, or power lines in 2D, of neighbouring grains. For a monosized granular assembly, the Dirichlet and Voronoi tessellations give the same result.

The Delaunay network

The Delaunay network is constructed by using the Voronoi or Dirichlet cells as a basis. Therefore, the Voronoi/Delaunay and Dirichlet/Delaunay methods are dual. Consider an assembly of circular or spherical grains. If the

Voronoi or Dirichlet cell of two grains have a common side, the two grain centers are connected by a straight line. These connecting lines form the Delaunay network of the assembly. The Delaunay cells are triangles in 2D and tetrahedra in 3D assemblies. Therefore this method is also frequently called the ‘Delaunay triangulation method’.

The definition can be modified by only connecting the centers of contacting grains by a straight line. In this version, the lines in the network correspond to the internal supports in the microstructure. However, the duality between the Voronoi and Dirichlet tessellation does not necessarily hold in this case.

2.2 Strain definitions based on an equivalent continuum

In this section, strain definitions based on an equivalent continuum are described. The procedure of this method is to replace a system of particles by an equivalent continuum and assume that all particles move and deform according to the equivalent continuum. To this continuum an appropriate translation field is assigned. Then, the deformation gradient of this continuum is determined and averaged over the volume. This deformation gradient is expressed in terms of micro-variables, like the displacement of individual particles and geometrical properties of particles and contacts. The strain tensor is the symmetric part of the translation gradient tensor. The various strain definitions that follow this procedure differ from each other in the way how an equivalent continuum is defined and how a suitable translation field can be assigned to it. In the following paragraphs, the definitions proposed by Bagi, Kruyt & Rothenburg, Kuhn and Cambou & Dedecker will be discussed.

2.2.1 Bagi's definition of stress and strain

The aim of Bagi's research [2] was to find macro-level state-variables through the proper averaging of micro-variables. The mechanical state of a granular assembly and its dynamics due to applied forces could exactly be described and predicted if the following characteristics are fully given:

- position and geometry of each grain;
- displacement (translations and rotations) of each grain;
- contact forces;
- material properties of the individual grain.

However, such a detailed description is not necessary and too complicated from a practical point of view. The challenge is to describe the stress and strain relations for a representative elementary volume of a granular system. The question is which and how many micro-variables should be used, how the representative elementary volume should be chosen and how the average has to be performed.

Geometrical representation

Before Bagi elaborates on the definitions of stress and strain for granular media, two different geometrical representations for discrete materials are introduced. These are the *material cell system* and the *space cell system*. These concepts will be described in more detail first.

Table 1: Nomenclature used in section 2.2.1

Symbol	Description
a	vector that forms the basis of the definition of d
b	vector whose direction is normal to a face of a space cell and whose magnitude is equal to the area of the face
d	complementary area vector
D	dimension of the system
e	displacement gradient tensor
F	concentrated force acting on the boundary of a (sub)domain
l	branch vector connecting the centers of two contacting grains
n	outward unit normal vector
p	load acting on a system
\overline{PG}	distance between a point P and a grain G
S	the surface area of a (sub)domain
u	continuous displacement field tensor
v	vector connecting the grain center to one of its contact points with another grain
V	volume of a (sub)domain
x	coordinate of a grain center or contact point
σ	stress tensor

The material cell system

Bagi applies a modified version of the Dirichlet tessellation to define the material cells. The difference lies in the fact that the grains are allowed to have any arbitrary shape, while the Dirichlet tessellation can only be applied to a system consisting of circular or spherical grains. Assume an assembly of grains of arbitrary shape in 2D or 3D Euclidean space. The distance \overline{PG} is defined as the distance between a point P and a grain G . For a specific grain G_0 , collect all those points P whose distance from G_0 is less or equal than from any other grain:

$$\overline{PG_0} \leq \overline{PG_k} \quad (k \neq 0).$$

These points P form a domain around the G_0 grain, as indicated in figure 1.

The material cell has the following properties:

- There is exactly one grain in each domain.
- The domains are contiguous.
- Grains on the boundary of an open assembly have infinitely large domains while the domains are finite in the inside of the assembly.

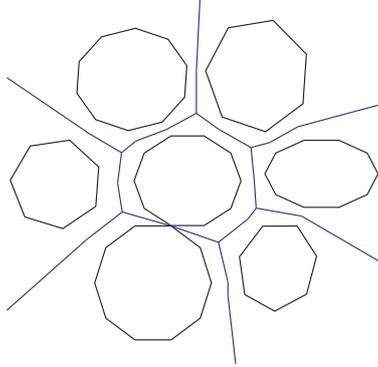


Figure 1: *The material cell geometry. The boundaries of the cells are indicated by the lines. Particles on the boundary of an open assembly have infinitely large domains.*

- The common face of neighbouring domains (belonging to the grains G_1 and G_2) is the set of P points for which $\overline{PG_1} = \overline{PG_2} \leq \overline{PG_k}$ for all $k \neq 1$ and $k \neq 2$.
- If two grains have a contact point, the corresponding domains must have a common face that contains the contact point itself.

The space cell system

The construction of the space cell system is strongly based on the above mentioned definitions and characteristics of the material cell system. Using the material cell system as a starting point, the space cell system for a 2D assembly can be defined as follows:

1. *Nodes* of the space cell system are the grain centers (each material cell contains one node).
2. When two material cells have a common face, the corresponding grain centers should be connected by a straight line. These lines will be the *edges* in the space cell system.
3. The edges form closed cells, which are triangles in 2D and tetrahedrons in 3D.

Figure 2 illustrates the space cell system of a 2D assembly.

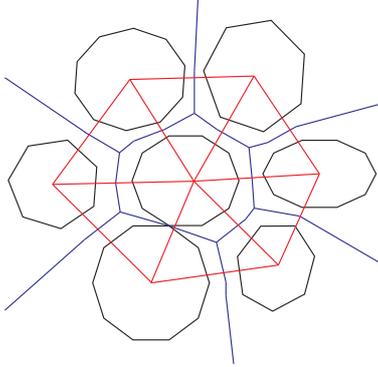


Figure 2: The space cell system. The boundaries of the space cells are indicated by the red lines (forming triangles)

Stress

Bagi argues that for describing the stress tensor of a granular system, the material cell system is the preferred geometrical representation. The definition of the stress tensor is based on contact forces between grains and material cells divide the space into subdomains in such a way that contact forces act between them.

First, the stress tensor for a closed continuous domain with volume V will be defined. This body is loaded on its boundary S by a distributed force $p_i(x_j)$. Depending on the loads a $\sigma_{ij} = \sigma_{ij}(x_k)$ stress tensor belongs to every point of the domain satisfying the boundary conditions

$$\sigma_{ij}n_j = p_i, \quad (1)$$

where n_i is the outwards unit normal vector on S . The volume average of the stress tensor can be expressed as a surface integral:

$$\bar{\sigma}_{ij} = \frac{1}{V} \iiint_{(V^L)} \sigma_{ij} dV = \frac{1}{V} \iint_{(S)} x_i p_j dS. \quad (2)$$

If the domain is divided into subdomains, the average stress tensor can be calculated separately for each subdomain:

$$\bar{\sigma}_{ij}^L = \frac{1}{V^L} \iint_{(S^L)} x_i p_j dS, \quad (3)$$

where V^L and S^L are the volume and boundary of the L -th subdomain, distributed forces $p_i(x_j)$ act on S^L from the neighbouring subdomains and the external boundary.

In the cases where there are concentrated forces acting on the boundary of the domain and between the subdomains, equation (3) can be written in a discrete form. When writing the forces acting from outside as point forces $F_i^1, F_i^2, \dots, F_i^k, \dots$; acting at boundary points $x_i^1, x_i^2, \dots, x_i^k, \dots$, equation (3) can be transformed to:

$$\bar{\sigma}_{ij} = \frac{1}{V} \sum_{(k)} x_i^k F_j^k, \quad (4)$$

where the index k runs over the external point forces.

Now consider the L -th subdomain; the forces $F_i^1, F_i^2, \dots, F_i^c, \dots$ act on its boundary at the points $x_i^1, x_i^2, \dots, x_i^c, \dots$, partly from the neighbouring subdomains and partly from outside. The average stress here is

$$\bar{\sigma}_{ij}^L = \frac{1}{V} \sum_{(c)} x_i^c F_j^c. \quad (5)$$

Since the forces inside cancel other out each other the in the sum, the volume-weighted average for the whole domain is (again):

$$\begin{aligned} \bar{\sigma}_{ij} &= \frac{1}{V} \sum_{(L)} V^L \bar{\sigma}_{ij}^L = \frac{1}{V} \sum_{(L)} \left(\sum_{(c)} x_i^c F_j^c \right) \\ &= \frac{1}{V} \sum_{(k)} x_i^k F_j^k. \end{aligned} \quad (6)$$

By using the derived equation in (6), the stress tensor for a granular medium will be defined. The equation will be transformed into a form that only contains micro-variables for a discrete sysem. The x_i^c vectors can be decomposed into two parts, as shown in figure 3,

$$x_i^c = x_{0i}^L + v_i^c \quad (7)$$

where x_{0i}^L is the coordinate of the center of the L -th grain. In lack of body forces the equilibrium equation

$$\sum_{(c)} F_j^c = 0 \quad (8)$$

is valid and therefore

$$\sum_{(c)} x_i^c F_j^c = \sum_{(c)} v_i^c F_j^c. \quad (9)$$

The branch vector can be defined as

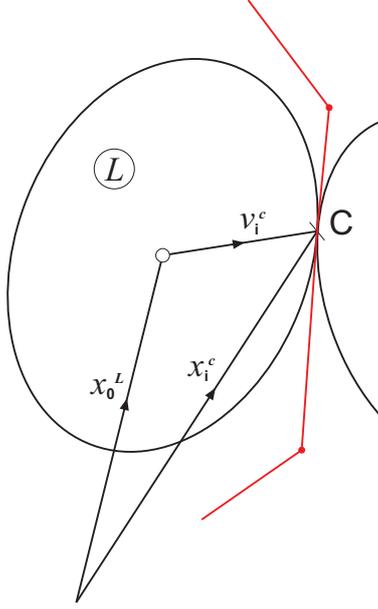


Figure 3: Illustration of the position vectors \mathbf{x}

$$l_i^c = v_i^{1c} - v_i^{2c}. \quad (10)$$

In the double sum in equation (6) each contact is considered twice, except from the boundary contacts. Instead of the v_i^c vectors the branch vectors can be applied. This means that the average stress tensor for a granular material can be written as

$$\bar{\sigma}_{ij} = \frac{1}{V} \sum_{(c)} l_i^c F_j^c. \quad (11)$$

Strain

To define a strain tensor for a granular assembly, Bagi replaced each space cell in the granular assembly by a continuum, so relations from continuum theory could be applied. Formulating for an arbitrary surface S , this reads:

$$\bar{e}_{ij} = \frac{1}{V} \iint_{(S)} u_i n_j dS, \quad (12)$$

where u_i is a continuous displacement field of the subdomain and n_j is the outwards unit normal vector on the surface area S of the cell volume V . After discretizing the system the author could formulate an expression of the displacement gradient tensor that contained discrete microvariables only.

Figure 4 illustrates why the space cell system is the chosen geometrical model to describe strain. For simple compression and simple shear, the global deformation of the assembly is very well represented by the deformations of the space cells, since they characterise the distortions of the internal structures itself, instead those of an individual grain or contact.

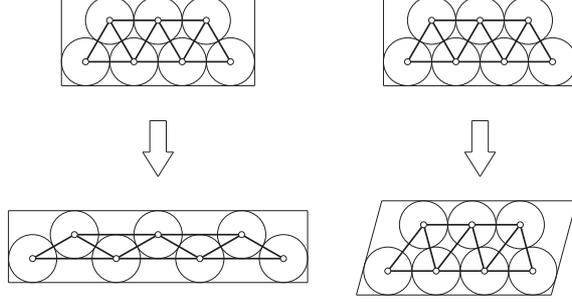


Figure 4: Compression and shear of a 2D assembly of particles. The thick solid lines indicate the space cells.

Taking the L -th space cell, its average displacement gradient tensor is in the continuous form:

$$\bar{e}_{ij}^L = \frac{1}{V^L} \iint_{(S^L)} u_i n_j dS. \quad (13)$$

Using the fact that n_j is constant along edges and assuming that $u_i(x_j)$ is linear along the edges, equation (13) can be written in the following discrete form:

$$\bar{e}_{ij}^L = \frac{1}{V^L} \sum_{k=1}^{D+1} u_i^k a_j^k, \quad (14)$$

where D is the dimension of the system. The meaning of a_j^k needs some more explanation. Consider a space cell and number its nodes as 1, 2, ..., $D+1$. Denote the faces of the cell by the number of that node, which is not contained by the face (i.e. the k -th face contains all the nodes except the k -th node). Assign a vector b_i^k to each face in the following way:

- The magnitude of b_i^k is equal to the area of the face (or length in 2D).
- The direction of b_i^k is normal to the face, pointing outwards:

$$b_i^k = |b| n_i^k. \quad (15)$$

The vector a_i^k is then defined as

$$a_i^k = -\frac{1}{D} b_i^k = -\frac{b}{D} n_i^k. \quad (16)$$

The introduced vectors \mathbf{a} and \mathbf{b} are illustrated in figure 5 in 2D.

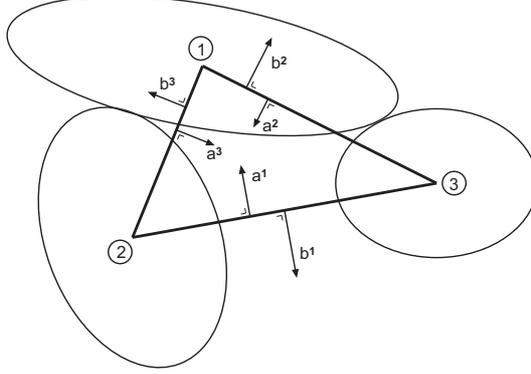


Figure 5: Illustration of the vectors \mathbf{a} and \mathbf{b} . The magnitude of \mathbf{a} equals the branch length and the magnitude of \mathbf{b} equals the area of the face in 3D or the length of the face in 2D (in 2D a face is a one dimensional line and in 3D a face is an area)

Since the sum over all a_i^k vectors is zero, equation (14) can be modified by subtracting the same u_i^0 vector from each nodal displacement of the cell:

$$\bar{e}_{ij} = \frac{1}{VL} \sum_{k=1}^{D+1} (u_i^k - u_i^0) a_j^k. \quad (17)$$

This expression says that the rigid-body translations do not change the deformation of the cell. Bagi chose u_i^0 to be the average translation of the nodes:

$$u_i^0 = \frac{1}{D+1} \sum_{k=1}^{D+1} u_i^k, \quad (18)$$

which gives that the average deformation gradient of the L -th cell is

$$\bar{e}_{ij}^L = \frac{1}{D+1} \frac{1}{VL} \sum_{m < n} (u_i^m - u_i^n) (a_j^m - a_j^n). \quad (19)$$

The vector a_i^k can be used to define a geometrical micro-variable of the space cell system: the complementary area vector. This vector can be constructed in the following way: Consider a pair of grains, G_1 and G_2 , that have a either a real or virtual contact¹, so the two grain centers, 1 and 2, are

¹Two particles have a virtual contact when they do not touch each other, but their respective material cells do have a common face and the particle centers are connected by an edge of a space cell.

connected in the space cell system. Collect now all those space cells that contain this edge. Assume that altogether T cells were found; denote them as $\text{cell}(1), \text{cell}(2), \dots, \text{cell}(t), \dots, \text{cell}(T)$. In the next step calculate the difference $a_i^{1(t)} - a_i^{2(t)}$ separately in each cell from $t = 1$ to T ; after summation over all space cells containing the 1 – 2 edge, the complementary area vector

$$d_i^{12} = \frac{1}{D+1} \sum_{t=1}^T \left(a_i^{1(t)} - a_i^{2(t)} \right) \quad (20)$$

is obtained. Now going back to equation (19); if the term $(u_i^m - u_i^n)$ is written as Δu_i^{mn} and the result in equation (20) is used, equation (19) can be written as

$$\bar{e}_{ij} = \frac{1}{V} \sum_{m < n} \Delta u_i^{mn} d_j^{mn}. \quad (21)$$

The definition contains micro-variables only, namely the relative displacements of neighbouring nodes (Δu_i^{mn}) and the corresponding complementary area vector (d_j^{mn}). The symmetric part of this tensor expresses the deformations of the space cells and it is suggested to be the strain tensor of granular assemblies. The anti-symmetric part of this strain tensor reflects the average rigid-body rotation of the space cells.

Duality of the state variables

The proposed definitions of strain and stress in equations (21) and (11) have some similarities; the summations run over the same contacts (except for boundaries), the contact forces belong to the same pair of particles as the relative displacements. However, there are some drawbacks of this method. The material cells and space cells represent different volumes (or areas in 2D), as can be seen in figure 2. The relative difference in this area or volume can be decreased by making the averaging volume large enough by including more grains.

The definitions presented by Bagi are not yet sufficient to reflect all peculiarities of granular systems. Such features are the distributions of micro-level variables (contact forces, relative displacements, geometrical micro-variables) and the ability of individual grains (and clusters of grains as well) to rotate, roll and slide.

2.2.2 Kruyt & Rothenburg's definition of strain

In a publication in 1996, Kruyt and Rothenburg suggested a strain definition for two dimensional assemblies. The modelled system consisted of disks over a small size distribution. As is shown in figure 6, only contacting grains are connected, forming sets of polygons. These polygons are replaced by continuous subdomains and for each subdomain the deformation gradient is defined.

Table 2: Nomenclature used in section 2.2.2

Symbol	Description
A	area of a continuous domain
e	deformation gradient tensor
g	vector connecting the centers of two polygons
h	vector perpendicular to g
Δu	relative translation of two particle centers

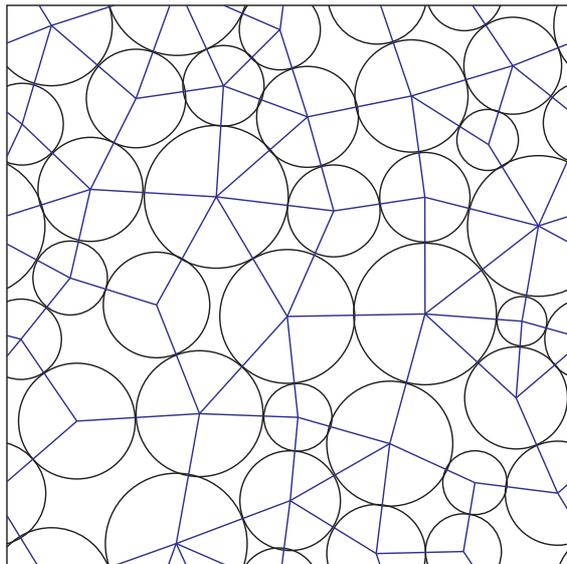


Figure 6: Polygons are formed by connecting contacting disks

The average deformation gradient of this continuous domain with area ‘ A ’ can be expressed by

$$\bar{e}_{ij} = \frac{1}{A} \sum_c \Delta u_j^c h_i^c, \quad (22)$$

where the summation with index ‘ c ’ runs over all contacting pairs of grains; Δu_j^c is the relative translation of the two particle centers forming pair ‘ c ’ and

h_i^c is a vector that is perpendicular to the g_i^c vector connecting the centers of the two polygons neighboring branch 'c':

$$h_i^c = -e_{ij}g_j^c \quad (23)$$

If pair 'c' is on the boundary, g_i^c connects the center of the polygon with the middle of branch 'c'. In the special case where all the polygons of the assembly are triangles, the definition proposed by Kruyt and Rothenburg in equation (22) is equivalent to the definition suggested by Bagi, see equation (21).

2.2.3 Kuhn's definition of strain

The definition of strain that Kuhn proposed in reference [11] is a combination of the definitions suggested by Bagi [2] and by Kruyt and Rothenburg [8]. The difference with Bagi's approach lies in the description of the space cells. Like Kruyt & Rothenburg, Kuhn only connected the centers of grains if they are in contact. This creates void cells that are not necessarily triangles, but can also be polygons. Kuhn called his equivalent to the space cell the *Particle graph*, where A^i represent void cells that are each surrounded by branch vectors of contacting particles. An example is shown in figure 7.

Table 3: Nomenclature used in section 2.2.3

Symbol	Description
A	material region
b	vectors that define the geometry of a polygon
l	branch vector connecting the centers of two contacting particles
L	velocity gradient tensor
\mathcal{L}	number of void cells
\mathcal{M}	total number of contacts
$\overline{\mathcal{M}}$	portion of the total number of contacts
\mathcal{N}	number of particles
$\overline{\mathcal{N}}$	portion of the total number of particles
Q	matrix that serves as a sort of prefactor in the calculation of L
\hat{u}	relative velocity between two particles

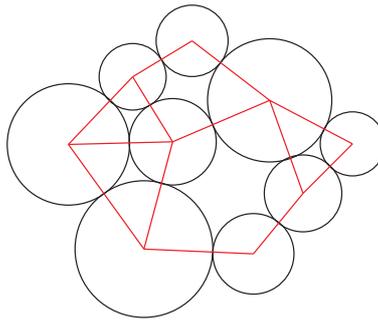


Figure 7: Particle graph

The modified particle graph is represented by its \mathcal{L} void cells (faces), $\overline{\mathcal{M}}$ contacts (edges or branch vectors) and $\overline{\mathcal{N}}$ particles (vertices). The overbars designate the portion of the total \mathcal{M} and \mathcal{N} contacts and particles that remain after non-participating particles (particles with no contact) have

been neglected. Superscripts i , j and k will be used as indices to represent void cells, contacts and particles, respectively.

Kuhn defined the deformation rate within a material region A as a velocity gradient \mathbf{L} , a function of position \mathbf{x} . The author distinguished between the velocity gradient at macroscale; the spatial average $\bar{\mathbf{L}}$ within the entire system, and the average velocity gradient at microscale $\bar{\mathbf{L}}^i$, within each of the \mathcal{L} void cells, such that

$$\bar{\mathbf{L}} = \frac{1}{A} \sum_{i=1}^{\mathcal{L}} A^i \bar{\mathbf{L}}^i. \quad (24)$$

Where Bagi defined triangular regions [2], Kuhn defined polygonal regions. The author made the equivalent assumption that the velocity varies linearly along the edges of the polygons with m^i edges. This leads to the expression:

$$\bar{\mathbf{L}}^i = \frac{1}{6A^i} \sum_{j_1, j_2 \in \{0, 1, \dots, m^i - 1\}} Q_{j_1, j_2}^{m^i} \hat{\mathbf{u}}^{i, j_1} \otimes \mathbf{b}^{i, j_2}. \quad (25)$$

Here, the vector $\hat{\mathbf{u}}^{i, j}$ is the relative velocity vector between two particles on edge j of the i th void cell A^i . The \mathbf{b} vectors define the geometry of polygon i (see figure 8).

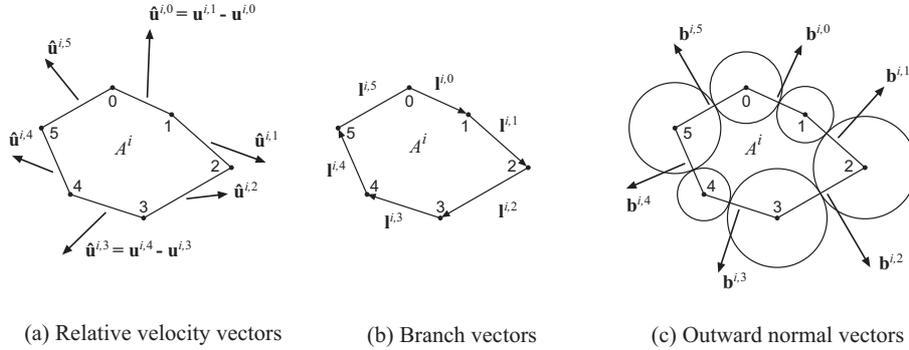


Figure 8: Vectors associated with edges of the i th void cell.

The dyadic product of the vectors $\hat{\mathbf{u}}^{i, j_1}$ and \mathbf{b}^{i, j_2} is weighted by the elements of an $m \times m$ matrix Q^m , where m is the number of edges of a void cell i . For the case where the void cell is a triangle, the matrix Q^3 is

$$Q^3 = \begin{bmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix}. \quad (26)$$

The Q^m matrices for polygons with $m \geq 4$ edges can be found in [10].

2.2.4 Cambou's and Dedecker's definition of strain

Another definition for strain that is only valid for two dimensional assemblies was presented by Cambou and Dedecker in references [3] and [5]. It was basically the same as the two dimensional version suggested by Bagi in [2]. The subsystems were made by Delaunay-triangularization (see Fig. 9).

Table 4: Nomenclature for section 2.2.4

Symbol	Description
a	displacement gradient tensor
u	displacement vector
x	spatial variable
ϵ	strain tensor
ω	rotation of the equivalent continuum

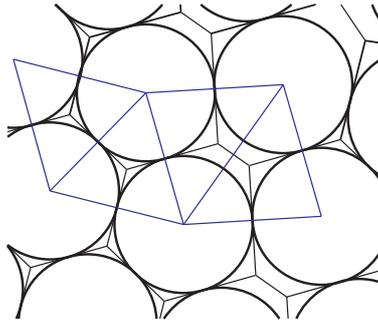


Figure 9: Delaunay-triangularization of a part of an assembly of disks

By using the Delaunay-triangularization method to discretize the medium, it is possible to derive a mean value of the global strain:

$$\bar{\epsilon}_{ij} = \langle \epsilon_{ij}^e \rangle, \quad (27)$$

where ϵ^e is the strain tensor defined on each triangular element e (see Fig. 10).

Cambou used the assumption that the local strain is constant in each local triangular element. This means that in any one element, the displacement is linear with respect to a spatial variable x :

$$u_i^e(x_j) = a_{ij}^e x_j + u_{0i}^e, \quad (28)$$

in which \mathbf{a} is the displacement gradient tensor and \mathbf{u}_0 is a constant vector. The local strain of the equivalent continuum and the asymmetric displacement gradient, which contains rotations of the system, can be calculated:

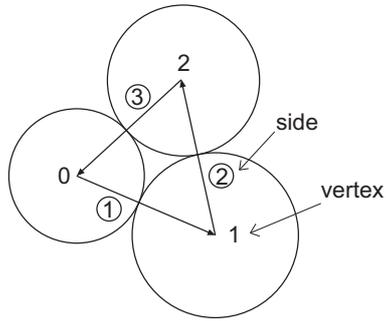


Figure 10: Definition of the notations for vertices and sides.

$$\begin{cases} \epsilon_{ij}^e = \frac{1}{2} (a_{ij}^e + a_{ji}^e) \\ \omega_{ij}^e = \frac{1}{2} (a_{ij}^e - a_{ji}^e) . \end{cases} \quad (29)$$

2.3 Strain definitions based on best-fit approach

This section describes strain definitions that are based on the best-fit approach. The aim of this method is to find a deformation gradient that is the best approximate of the real displacement of the system. The various definitions that are proposed in literature differ from each other in the way that the type of displacement is described. The definitions can be based on the following type of displacements:

- Particle translations
- Contact deformations
- Relative translations of neighbouring grains

Since describing the displacement seemed to be insufficient to describe the kinematics, the best-fit strains based on contact deformations and on the relative translation of neighbouring grains were proposed to improve the description of the kinematics of a granular system [12] [3].

Table 5: Nomenclature used in chapter 2.3

Symbol	Description
a	tensor used for shorthand notation
b	tensor used for shorthand notation
d	tensor used for shorthand notation
e	average deformation gradient tensor
f	tensor used for shorthand notation
l	branch vector connecting the centers of two contacting spheres
u	translation vector of a particle
x	position vector of particle
w	the inverse of f
z	square sum of the deviation between average displacement of the system and the individual displacement of a particle
z	inverse of a
α	deformation gradient tensor
Δ	relative displacement of the contact
Δu	relative displacement of the centers of two contacting particles
φ	rotation of a grain

2.3.1 Definition based on particle translations

This best-fit strain definition is based on the translations of individual particles [7]. If the average translation of all particles is non-zero, then u_i^p is the difference between the translation of particle ‘ p ’ and the average translation of all grains. The position of particle ‘ p ’ is indicated by x_i^p . It also holds here that if the average of the particle positions is non-zero, then x_i^p is the difference of position of this particle from the average position (center of mass).

A deformation gradient tensor α_{ij} can be defined that translates a particle ‘ p ’ from a position x_i^p to a new position, such that

$$u_i^p = \alpha_{ij} x_j^p \quad (30)$$

is the translation. In most cases not every particle translates exactly according to this tensor. In these cases, an α_{ij} that is valid for every particle in the system does not exist. The purpose of the best-fit approach is to define an α_{ij} that gives the best approximation of the real displacement of the system. How this procedure works will be explained in more detail here.

Because equation (30) does not hold for every particle the difference

$$u_i^p - \alpha_{ij} x_j^p$$

is non-zero. The α_{ij} must be found that makes the square sum of these differences the smallest:

$$z(\alpha_{ij}) = \sum_p \left(u_i^p - \alpha_{ij} x_j^p \right) \left(u_i^p - \alpha_{ij} x_j^p \right) \rightarrow \min! \quad (31)$$

The minimum of the function $z(\alpha_{ij})$ can be found by determining the derivative with respect to α_{ij} :

$$\frac{\partial z}{\partial \alpha_{ij}} = 0,$$

for any i and j . This leads to two systems of linear equations (for a two dimensional system), each of them consisting of two scalar equations:

$$\begin{bmatrix} \sum_{(p)} x_1^p x_1^p & \sum_{(p)} x_2^p x_1^p \\ \sum_{(p)} x_1^p x_2^p & \sum_{(p)} x_2^p x_2^p \end{bmatrix} \cdot \begin{bmatrix} \alpha_{1i} \\ \alpha_{2i} \end{bmatrix} = \begin{bmatrix} \sum_{(p)} u_i^p x_1^p \\ \sum_{(p)} u_i^p x_2^p \end{bmatrix}. \quad (32)$$

The shorthand notation for this expression is

$$a_{jk} \alpha_{ki} = b_{ji}, \quad \text{where} \quad a_{jk} = \sum_{(p)} x_k^p x_j^p \quad \text{and} \quad b_{ji} = \sum_{(p)} u_i^p x_j^p.$$

The α_{ij} that follows from the equation above is the average translation gradient tensor e_{ij} :

$$e_{ij} := \alpha_{ij} = z_{ik} \sum_{(p)} u_j^p x_k^p, \quad (33)$$

where z_{ik} is the inverse of a_{mn} . The symmetric part of e_{ij} is the average strain of the system of particles.

2.3.2 Definition based on contact deformations

The best-fit strain definition based on contact deformations was proposed by Liao et al. in 1997 [12]. The relative displacement of the contact is denoted by Δ_i^c . If every particle in the system is displaced by the same deformation gradient tensor α_{ij} , then the displacement of contact c shared by particles p and q can be described by

$$\Delta_i^c = \alpha_{ji} l_j^c = \alpha_{ji} (x_j^p - x_j^q), \quad (34)$$

where x_j^p and x_j^q are the positions of particles p and q and l_j^c is the branch vector connecting the centers of the two contacting particles. Usually, not every particle in the assembly is displaced exactly by α_{ij} so

$$\Delta_i^c - \alpha_{ji} l_j^c \neq 0. \quad (35)$$

As was the case with the particle translations in the previous paragraph, there usually is a discrepancy between the average displacement of the system and the individual displacement of contacts. Also here, the aim is to find the α_{ij} that gives the best approximation of the average contact displacements. This can be done by calculating the square sum function of the deviations and determining its minimum:

$$z(\alpha_{ij}) = \sum_{(c)} (\Delta_i^c - \alpha_{ji} l_j^c) (\Delta_i^c - \alpha_{ji} l_j^c) \rightarrow \min! \quad (36)$$

The function $z(\alpha_{ij})$ has a minimum when the derivative with respect to α_{ji} is zero:

$$\frac{\partial z}{\partial \alpha_{ji}} = 0 \quad (37)$$

for any i and j , which leads to two systems of linear equations for a 2D assembly and to three systems of linear equations for a 3D assembly:

$$\begin{bmatrix} \sum_{(c)} l_1^c l_1^c & \sum_{(c)} l_2^c l_1^c \\ \sum_{(c)} l_1^c l_2^c & \sum_{(c)} l_2^c l_2^c \end{bmatrix} \cdot \begin{bmatrix} \alpha_{1i} \\ \alpha_{2i} \end{bmatrix} = \begin{bmatrix} \sum_{(c)} \Delta_i^c l_1^c \\ \sum_{(c)} \Delta_i^c l_2^c \end{bmatrix}. \quad (38)$$

The shorthand notation for this expression is

$$f_{jk} \alpha_{ki} = d_{ji} \quad \text{where} \quad f_{jk} = \sum_{(c)} l_k^c l_j^c \quad \text{and} \quad d_{ji} = \sum_{(c)} \Delta_i^c l_j^c.$$

The average contact displacement gradient tensor can now be written as

$$e_{ij} := \alpha_{ij} = w_{ik} \sum_{(c)} \Delta_j^c l_k^c, \quad (39)$$

where w_{ik} is the inverse of f_{jl} . The symmetric part of e_{ij} is the average strain of the analyzed system.

Cambou et al. [3] proposed two modified versions of equation (39). In the first modified version, the description of the relative translation is based on the centers of contacting particles, Δu_j^c , instead of describing the translation of the contact point of the two contacting particles. Following the same procedure as Liao did in [12], Cambou came to the following definition based on the relative displacement of the centers of contacting particles:

$$e_{ij} := w_{ik} \sum_{(c)} \Delta u_j^c l_k^c \quad (40)$$

where w_{ik} is the same as in equation (39). The definitions presented by Liao and Cambou in equations (39) and (40), respectively, give the same result when the particles don't have rotations.

The second modification of equation (39) that was proposed by Cambou in [3] is that not only the displacements of pairs of contacting grains are described, but also pairs of particles which grain centers are connected with an edge in a Delaunay triangulation network. Such pairs of particles are indicated with the superscript 'e'. The resulting average translation gradient tensor becomes, similar to equation (40):

$$e_{ij} = w_{ik} \sum_{(e)} \Delta u_j^e l_k^e \quad (41)$$

The difference between the two modified definitions that Cambou proposed with equations (40) and (41) comes into play when pairs of particles are close together but not in contact. When every edge in the Delaunay triangulation network corresponds to a contact between two grains, then the two equations give the same result.

2.3.3 Definition including particle rotations

The difference between the strain definition of Liao et al. in equation (39) and the definition of Cambou in equation (40) can be used to express the effect of particle rotations:

$$e_{ij}^{rot} = w_{ik} \sum_{(c)} (\Delta_j^c - \Delta u_j^c) l_k^c, \quad (42)$$

where the term $(\Delta_j^c - \Delta u_j^c)$ is the difference between the relative translation in the contact point and the relative translation between the two centers of two contacting particles. This difference originates from the rotations φ_i^{1c} and φ_i^{2c} of the contacting grains. This can be expressed as:

$$\Delta_j^c - \Delta u_j^c = e_{jmn} \varphi_m^{2c} (x_n - x_n^{2c}) - e_{jmn} \varphi_m^{1c} (x_n^c - x_n^{1c}). \quad (43)$$

In figure 11 the notation in the equation above is explained.

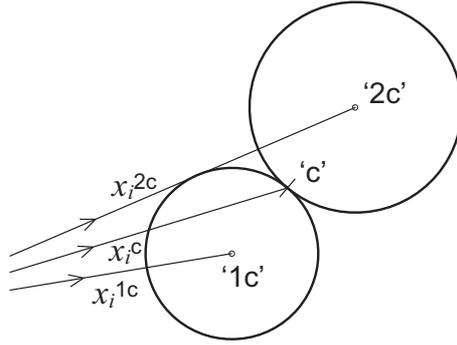


Figure 11: Explanation of the notation used in this section. The x_i vectors are the position vectors.

The symmetric part of e_{ij}^{rot} is called the rotational best-fit strain.

2.4 The Satake strain

Satake proposed a new definition of strain in [15]. The author defined stress and strain of granular assemblies using the micro-mechanical quantities at contact points, like the contact force and contact displacement. The sub-systems were constructed using Dirichlet tessellation. The definition is applicable for assemblies of disks in 2D and assemblies of spheres in 3D.

The procedure starts with the Dirichlet tessellation of a system of grains, creating polyhedra. Every grain in the system corresponds to one polyhedron and each face of a Dirichlet polyhedron corresponds to a contact with a neighbouring grain. Such a contact can be a virtual contact as well, which defines the corresponding virtual branch vectors. This means that the grains are not in contact, but their corresponding polyhedra share a face. From the Dirichlet tessellation, the Delaunay network can be constructed by connecting the centers of neighbouring polyhedra. In 3D, a Delaunay simplex is a tetrahedron and in 2D a triangle. These simplexes form the basis for the micro-mechanical description of the granular assembly. The author uses the following notation in his publication:

- *Dirichlet polyhedron (polygon) i* : Polyhedron (polygon) corresponding to grain i .
- *face (side) i* : a face (side) of a Dirichlet polyhedron (polygon) corresponding to contact C . Face C is a polygon.
- *Simplex*: A cell obtained by Delaunay triangulation.
- C^{ij} : contact of grains i and j .
- D (D^{ijkl} in 3D, D^{ijk} in 2D): Dirichlet center of a simplex (tetrahedron $ijkl$, triangle ijk).
- R_c : region accompanied by contact C , or simply region of contact C .
- \mathbf{l}^{ij} : branch vector which connects vertices i and j of a triangle in this order, which must be taken counter-clockwise (when the triangle is looked at from outside, in 3D). If the side ij corresponds to contact C , i.e. C^{ij} , \mathbf{l}^{ij} is also written as \mathbf{l}_c .
- \mathbf{b}^i : face (side) vector whose magnitude is the area of the face (length of the side) i and whose direction is outward normal to the face (side).

Consider a point D for a side ij of a simplex, which satisfies the following equation:

$$\overline{D_i}^2 - \overline{D_j}^2 = r_i^2 - r_j^2, \quad (44)$$

where r_i and r_j are the radii of grains placed at i and j , respectively. Applying equation (44) to all sides of a simplex, the position vector \mathbf{D} of a point D satisfies the relation

$$\mathbf{i}^{ij} \cdot \mathbf{D} = c^i - c^j, \quad (45)$$

where

$$c^i = \frac{1}{2} (r_i^2 - \mathbf{i} \cdot \mathbf{i}), \quad (46)$$

where \mathbf{i} is a position vector of vertex i . For c_j a similar expression applies. The equations obtained for every side of a simplex meet at only one point D (D^{ijkl} in 3D and D^{ijk} in 2D), which is called the Dirichlet center of the simplex. The simultaneous solution of the equations of (45) lead to the following expression of the position vector:

$$\mathbf{D}^{ijkl} = \frac{1}{3V} (c^i \mathbf{b}^i + c^j \mathbf{b}^j + c^k \mathbf{b}^k + c^l \mathbf{b}^l) \quad (47)$$

$$\mathbf{D}^{ijk} = \frac{1}{2A} (c^i \mathbf{b}^i + c^j \mathbf{b}^j + c^k \mathbf{b}^k), \quad (48)$$

where V or A denotes the volume or area of the concerned simplex, in 3D and 2D, respectively, and \mathbf{b}^i is a face vector in 3D or side vector in 2D whose magnitude is the area of the face (or length of the side) i and whose direction is outward normal to the face (side). Next, the region R_c of a contact C will be defined. R_c is a region of a quadrilaterally shaped polyhedron formed by points i and j and vertices of face C (side of C in 2D). Such a region is called the region of contact C . As R_c in 3D consists of some Delaunay tetrahedrons having a common side ij , a part of R_c belongs to one of such tetrahedrons.

The dual branch vector \mathbf{h}_c is an important vector in the micro-mechanical definition of the strain tensor for granular assemblies. The author defines \mathbf{h}_i as

$$\mathbf{h}_i = S_c \bar{\mathbf{n}}_c \quad (\text{in 3D}) \quad (49)$$

$$\mathbf{h}_i = s_c \mathbf{n}_c \quad (\text{in 2D}), \quad (50)$$

where S_c is the area vector of face C , s_c is the length of side C and \mathbf{n}_c is the outward normal vector of face C of polyhedron (side C of polygon) i , which is the same as the direction of the branch vector \mathbf{l}_c . V_c or A_c , the volume or area of R_c , is expressed respectively as

$$V_c = \frac{1}{3} \mathbf{l}_c \cdot \mathbf{h}_c = |\mathbf{b}| \quad (51)$$

$$A_c = \frac{1}{2} \mathbf{l}_c \cdot \mathbf{h}_c = |\mathbf{b}|. \quad (52)$$

For the definition of stress, the author introduced the stress at contact C , which is expressed as

$$\sigma_c = \frac{1}{3V_c} \mathbf{l}_c \mathbf{f}_c, \quad (53)$$

where V_c is the volume of R_c and satisfies the relation $\sum V_c = V$, and \mathbf{f}_c is the contact force.

The author argues that the micro-mechanical definition of the strain tensor should also start from that of strain at a contact C , which is written as

$$\epsilon_c = \frac{1}{3V_c} \mathbf{h}_c \mathbf{u}_c. \quad (54)$$

This definition is based on the property

$$\mathbf{u}_c = \mathbf{l}_c \cdot \epsilon_c, \quad (55)$$

which is analogous to the generalized continuum mechanics strain definition. The author calls ϵ_c the micro-mechanical strain at contact C . The overall strain tensor of a granular assembly is defined in the following form:

$$\epsilon = \frac{1}{V} \sum \mathbf{h}_c \mathbf{u}_c = \frac{1}{V} \sum 3V_c \epsilon_c. \quad (56)$$

Satake points out that although a micro-mechanical strain definition was also proposed by Bagi in [2] and by Kruyt & Rothenburg in [8], there are some important differences between the definitions:

1. No effect of the particle radii is considered in the definition of the vector \mathbf{d}_c (see chapter 2.2.1), because that is only defined by the shape and size of the concerned simplex.
2. The vector \mathbf{h}_c has a reasonable relation

$$\mathbf{f} = S_c \mathbf{n}_c \cdot \boldsymbol{\sigma}_c \quad (= \mathbf{h}_c \cdot \boldsymbol{\sigma}_c), \quad (57)$$

which is an analogous form to the Cauchy's theorem in continuum mechanics, but \mathbf{d}_c does not have such a meaningful relation.

3. During a deformation, a virtual branch sometimes makes a sudden change. In such a case, the change of R_c defined by \mathbf{h}_c is natural and continuous, but that by \mathbf{d}_c is not.

2.5 Conclusions and discussion

This literature survey discussed several techniques to describe strain for granular assemblies. These methods were divided in three categories:

- Definitions based on an equivalent continuum.
- Definitions based on the best-fit approach.
- The Satake strain.

The differences between the various strain definitions based on an equivalent continuum are rather small. Under generalized conditions, they usually give the same result. When for instance every polygon in Kruyt & Rothenburg's description is a triangle, the resulting strain definition is equivalent to Bagi's strain definition. In 2D, all four definitions based on equivalent continua describe the average translation gradient of an equivalent continuum whose boundaries are formed by the lines that go through the centers of the boundary particles. Therefore, all four definitions can be generalized to the following expression:

$$\bar{e}_{ij} = \frac{1}{A} \oint_{(B)} u_j n_i dl, \quad (58)$$

where A is the area within the boundaries, (B) denotes the boundary line, u_i is the translation of a point somewhere on a straight part of a boundary line. It is given by linear interpolation of the translation of the two corresponding grain centers; n_i is the outwards normal vector in the boundary points and the integration with respect to length dl is carried out along (B) .

Of the definitions based on equivalent continua, Bagi's definition is the only one that can also be applied to 3D systems. The space cells in such systems are tetrahedra, which are relatively easy to describe mathematically. For the other three definitions, it was not clearly stated in the articles if and how their strain definitions could be extended to 3D systems. In the case of Kruyt & Rothenburg's and Kuhn's definitions, the constructed space cells would be complexly shaped polyhedra. It can be imagined that it would be quite cumbersome to define translations fields, the space volume etcetera for such cells.

The differences between the various versions of the strains based in the best-fit approach are more fundamental in most cases. These definitions are based on different aspects of the material behaviour. In contrast to the definitions based on equivalent continua, the various versions of the best-fit strains cannot be expressed in a more generalized equation like in equation (58). The best-fit strain based on individual particle translations can directly be applied to 3D granular assemblies. For the other best-fit

strain, the applicability for 3D systems was not discussed by the respective authors.

The Satake strain described in chapter 2.4 is valid for both 2D and 3D granular assemblies. It's definition is based on the Dirichlet tessellation and the subsequent Delaunay triangularization of the granular system. There is some unclarity among researchers on how to interpret the contact deformation for virtual contacts. Moreover, the Satake strain is computationally expensive and therefore difficult to implement in Discrete Element Model (DEM) software.

For describing strain for three dimensional granular assemblies with (DEM) simulations, the equivalent continuum based strain definition proposed by Bagi, the best-fit strain based on individual particle translations and the Satake strain are the options to consider. However, these models are not capable of accurately describing more complex behaviour of the system, like rotations of individual grains or sudden changes in the contact network. Still, a lot of improvements have to be realised to correctly describe strain for 3D granular systems.

3 Theory of the HCP

3.1 Introduction

In this section, the properties of a hexagonal close packing will be described from a theoretical point of view. First the structure is introduced, based on the contact network of each sphere. From this contact network, the elastic modulus tensor can be derived. Next, a unit cell will be defined. The remaining properties of the HCP can be derived using this unit cell; the density (volume fraction) and the potential energy density. The same properties will also be determined from DEM-simulations (see section 5). It can then be verified if theory and simulations show agreement.

3.2 The structure

The hexagonal close packing (HCP) consists of alternating layers ‘A’ and ‘B’ of equi-sized spheres. A sphere in a layer ‘B’ lies in the hole that three contacting spheres in the layer ‘A’ below (and above) make with each other (see figure 12).

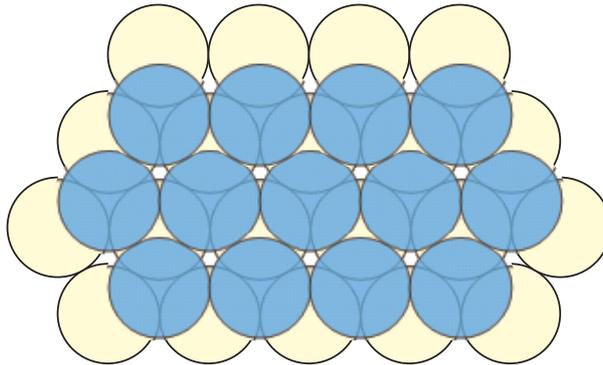


Figure 12: Topview on 2 layers of an HCP. The blue spheres in the top layer (layer ‘B’) are made transparent to indicate their position relative to the yellow spheres in the layer ‘A’ below.

Except for spheres at the boundary of a packing, each sphere has twelve contacting neighbouring spheres:

- Six contacting spheres are in the same layer. Connecting the centers of these six spheres with lines forms a hexagon.
- Three contacting spheres are in the layer above, forming a tetrahedron with the center sphere.
- Three contacting spheres are in the layer below, also forming a tetrahedron with the center sphere.

A cluster of these thirteen contacting spheres is shown in figure 13.

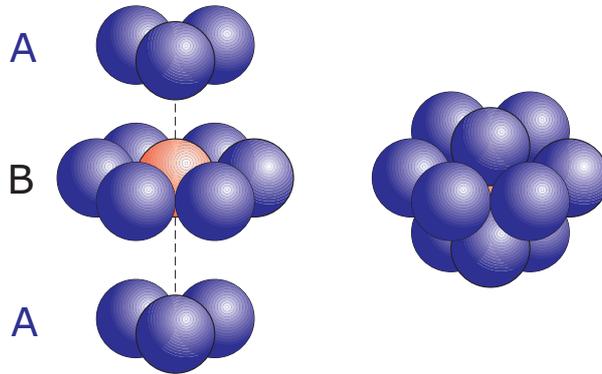


Figure 13: A cluster of contacting spheres. The red sphere in the center has six contacts in the same layer, three contacts above and three contacts below the plane.

3.2.1 The contact network

If the positions of two contacting spheres in the packing is described by the vectors \mathbf{p}_1 and \mathbf{p}_2 , the branch vector \mathbf{l} is defined as the difference between these two points:

$$\mathbf{l} = \mathbf{p}_2 - \mathbf{p}_1 . \quad (59)$$

This branch vector describes the direction and the distance of the line that connects the centers of the spheres and points from sphere 1 to sphere 2. The overlap δ of the two particles is defined as:

$$\delta = (2a) \hat{\mathbf{n}} - \mathbf{l} , \quad (60)$$

assuming equi-sized spheres. Here, a is the radius of the spheres and $\hat{\mathbf{n}}$ is the normal unit vector, defined by the direction of the branch vector:

$$\hat{\mathbf{n}} = \frac{\mathbf{l}}{l} , \quad (61)$$

where the scalar l is the length of the vector \mathbf{l} . The contact network of a sphere in the HCP is described by the twelve normal unit vectors that this particle has with its neighbours. While the contact networks of spheres within a layer ‘A’ or ‘B’ are identical, the contact network of a sphere in layer ‘A’ is similar but not equal to that of a sphere in layer ‘B’. The reason for this will be explained later in this section. From the cluster shown on the righthand side of figure 13, a ball-and-stick model can be created, giving more insight in the orientation of the spheres (see figure 14).

To show how the normal unit vectors can be obtained, the tetrahedron formed by spheres 1, 2, 3 and 8 in figure 14 will be used, as shown in figure 15.

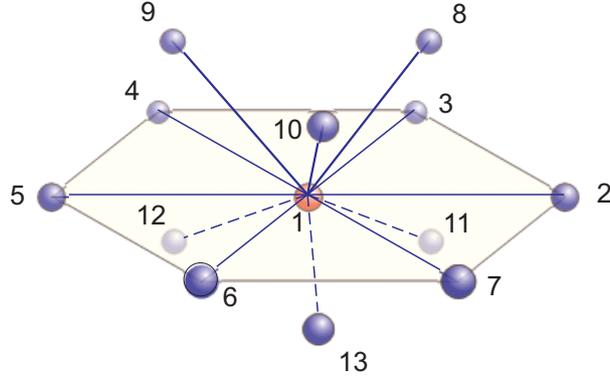


Figure 14: The contact network of a sphere in an HCP. Thick solid lines indicate contacts with spheres in the layer above, thin solid lines with spheres in the same layer and dashed lines with spheres in the layer below. The spheres are numbered from 1 to 13.

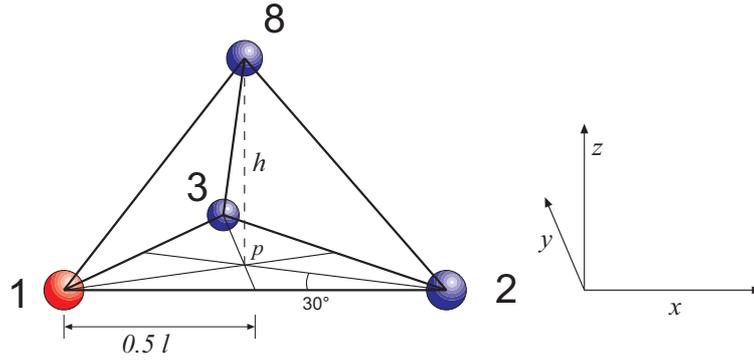


Figure 15: Four spheres form a tetrahedron

The center of the red sphere 1 in figures 14 and 15 is chosen as the origin of a Cartesian coordinate system. The centers of spheres 1, 2 and 3 are in the $x - y$ plane, with sphere 2 lying on the x -axis with a distance l from the origin. The normal vector for the contact between spheres 1 and 2 only has an x component and therefore this vector can be written as:

$$\hat{\mathbf{n}}_{(c1-2)} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \quad (62)$$

The normal vector referring to the contact between spheres 1 and 3 has x and y components. Superposing the center of sphere 3 on the x - and y -axis shows that the x -component is 0.5 and the y -component is $\frac{1}{2} \tan 60^\circ = \frac{1}{2} \sqrt{3}$, so that:

$$\hat{\mathbf{n}}_{(c1-3)} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2}\sqrt{3} \\ 0 \end{pmatrix}. \quad (63)$$

The normal vector belonging to the branch vector between spheres 1 and 8 has three non-zero components. The x -component is again $\frac{1}{2}$. From figure 15 it follows that the y -component equals $\frac{1}{2} \tan 30^\circ = \frac{1}{2\sqrt{3}}$. The z -component can be found in two ways. One way is to calculate the height h of the triangle that is formed with points 1, 8 and p . The easier way is to use the knowledge that the square sum of the elements of a normal unit vector must be unity:

$$h = \left(\sqrt{1 - (0.5)^2 - \left(\frac{1}{2\sqrt{3}} \right)^2} \right) l = \sqrt{\frac{2}{3}} l. \quad (64)$$

The normal unit vector $\hat{\mathbf{n}}_{c1-8}$ can therefore be written as

$$\hat{\mathbf{n}}_{c1-8} = \begin{pmatrix} 0.5 \\ \frac{1}{2\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix}. \quad (65)$$

The other normal unit vectors can also be obtained by forming tetrahedrons and choosing the center of the red sphere (no. 1) as the origin of a Cartesian coordinate system. The set of all normal unit vectors connected to all contacts for an arbitrary sphere in an HCP is summarized below:

$$\begin{array}{cccccc} \hat{\mathbf{n}}_{c1-2}^A & \hat{\mathbf{n}}_{c1-3}^A & \hat{\mathbf{n}}_{c1-4}^A & \hat{\mathbf{n}}_{c1-5}^A & \hat{\mathbf{n}}_{c1-6}^A & \hat{\mathbf{n}}_{c1-7}^A \\ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} & \begin{pmatrix} 0.5 \\ 0.5\sqrt{3} \\ 0 \end{pmatrix} & \begin{pmatrix} -0.5 \\ 0.5\sqrt{3} \\ 0 \end{pmatrix} & \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix} & \begin{pmatrix} -0.5 \\ -0.5\sqrt{3} \\ 0 \end{pmatrix} & \begin{pmatrix} 0.5 \\ -0.5\sqrt{3} \\ 0 \end{pmatrix} \\ \\ \hat{\mathbf{n}}_{c1-8}^A & \hat{\mathbf{n}}_{c1-9}^A & \hat{\mathbf{n}}_{c1-10}^A & \hat{\mathbf{n}}_{c1-11}^A & \hat{\mathbf{n}}_{c1-12}^A & \hat{\mathbf{n}}_{c1-13}^A \\ \begin{pmatrix} 0.5 \\ \frac{1}{2\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} -0.5 \\ \frac{1}{2\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} 0 \\ -\frac{1}{\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} 0.5 \\ \frac{1}{2\sqrt{3}} \\ -\sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} -0.5 \\ \frac{1}{2\sqrt{3}} \\ -\sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} 0 \\ -\frac{1}{\sqrt{3}} \\ -\sqrt{\frac{2}{3}} \end{pmatrix} \end{array} \quad (66)$$

The contact network for a particle in layer B is slightly different from that of a particle in layer A . If the configuration shown in figure 14 belongs to a

sphere in layer A , than the spheres 8 to 13 are rotated 180° if the red sphere is in a B -layer. The consequence is that the signs of the y -components of the normal unit vectors for these contacts is flipped. So for an arbitrary sphere in layer B , the normal unit vector for contacts 8 to 13 are (the contacts with spheres in the same layer is identical as for a sphere in layer A):

$$\begin{aligned}
& \hat{\mathbf{n}}_{c1-8}^B & \hat{\mathbf{n}}_{c1-9}^B & \hat{\mathbf{n}}_{c1-10}^B & \hat{\mathbf{n}}_{c1-11}^B & \hat{\mathbf{n}}_{c1-12}^B & \hat{\mathbf{n}}_{c1-13}^B \\
& \begin{pmatrix} 0.5 \\ -\frac{1}{2\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} -0.5 \\ -\frac{1}{2\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} 0 \\ \frac{1}{\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} 0.5 \\ -\frac{1}{2\sqrt{3}} \\ -\sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} -0.5 \\ -\frac{1}{2\sqrt{3}} \\ -\sqrt{\frac{2}{3}} \end{pmatrix} & \begin{pmatrix} 0 \\ \frac{1}{\sqrt{3}} \\ -\sqrt{\frac{2}{3}} \end{pmatrix} \\
& & & & & & (67)
\end{aligned}$$

This difference in the contact network between spheres in layers A and B shows that there is an anisotropy in the structure of the HCP. The question whether this has consequences for the elastic modulus tensor will be answered in the next section.

3.3 The elastic modulus tensor

In continuum theory, the elastic response of a continuous medium on an applied force is described by the generalized Hooke's law, by relating stress ($\boldsymbol{\sigma}$) to strain ($\boldsymbol{\epsilon}$) with the Elastic Modulus Tensor (EMT) \mathbf{C} , which is a fourth order tensor:

$$\sigma_{\alpha\beta} = C_{\alpha\beta\gamma\delta}\epsilon_{\gamma\delta} . \quad (68)$$

For a 3D system, the EMT has 81 components. However, they are not independent. This is due to the symmetries of the stress and strain tensors, which cause the following equalities for $C_{\alpha\beta\gamma\delta}$:

$$C_{\alpha\beta\gamma\delta} = C_{\beta\alpha\gamma\delta} = C_{\alpha\beta\delta\gamma} = C_{\beta\alpha\delta\gamma} . \quad (69)$$

There are also cases where there is an additional symmetry under exchange of the first and second pair of indices:

$$C_{\alpha\beta\gamma\delta} = C_{\gamma\delta\alpha\beta} . \quad (70)$$

Because of these symmetries, it is possible to write the EMT as a two-dimensional matrix. With the matrix below, all the relevant information is provided by the at most 21 independent entries:

$$\mathbf{C} = \begin{pmatrix} C_{1111} & C_{1122} & C_{1133} & C_{1123} & C_{1113} & C_{1112} \\ & C_{2222} & C_{2233} & C_{2223} & C_{2213} & C_{2212} \\ & & C_{3333} & C_{3323} & C_{3313} & C_{3312} \\ & & & C_{2323} & C_{2313} & C_{2312} \\ & & & & C_{1313} & C_{1312} \\ & & & & & C_{1212} \end{pmatrix}. \quad (71)$$

The entries in the lower left triangle are not written down because they have the same value as the entry on the mirrored position in the matrix (e.g. $C_{1122} = C_{2211}$). Even though the EMT is a four dimensional object, it has a physical interpretation. The first two indices indicate the direction of the applied stress. When $\alpha = \beta$, a force is acting perpendicular on the x -, y -, or z -plane. When the indices α and β are not equal, a shear force is acting on a plane. The second pair of indices, γ and δ , refer to the direction of strain. When $\gamma = \delta$, there is a deformation in the x -, y - or z -direction. The indices $\gamma \neq \delta$ refer to a shape change and possibly a rotation of the system if non-symmetric strain would be allowed.

The generalized Hooke's law can also be used for discrete media to relate stress to strain. Since the stress and strain tensor can be defined in terms of micro-mechanical variables of the system, the EMT can also be written in terms of these variables. When only normal contributions are considered, the strain tensor can be written as (see [13]):

$$\boldsymbol{\epsilon}^c = \frac{\delta}{l} \hat{\mathbf{n}} \hat{\mathbf{n}}, \quad (72)$$

and the stress tensor can be written as

$$\boldsymbol{\sigma}^c = \frac{kl\delta}{V_c} \hat{\mathbf{n}} \hat{\mathbf{n}}. \quad (73)$$

The variables l , δ and $\hat{\mathbf{n}}$ have already been introduced and discussed above. The stiffness k represents the 'softness' of the sphere contacts. The higher the stiffness, the more a sphere resists to deformation (in the model, spheres do not deform but overlap each other when pressed together). Here, V_c is the appropriate volume related to the contact c over which the stress is averaged.

Because the stress and strain tensors can be derived from a static contact network, the EMT can also be derived from this contact network. It is the partial derivative of the stress tensor with respect to the deformation:

$$C_{\alpha\beta\gamma\delta} = \frac{\partial \sigma_{\alpha\beta}}{\partial \epsilon_{\gamma\delta}} = \frac{1}{V} \sum_{p \in V} \left(k \sum_{c=1}^C \left(\frac{l^2}{2} \right) n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c \right). \quad (74)$$

This equation says that the EMT for a specific system can be determined by calculating the tensor for every contact ‘ c ’ and summing all these individual tensors up. This procedure is repeated for every particle within the volume ‘ V ’ and again summed up. The term $\frac{l^2}{2}$ accounts for the size of the particles and the fact that every contact is calculated twice.

With the definitions for the elastic modulus tensor given above, the tensor will be derived for the contact network of a single particle in an HCP. The contact network for a particle in layer ‘ A ’ is drawn in figure 14 and the related normal unit vectors are given in equation (66). To illustrate how the EMT can be calculated for a single contact c , the contact between particles 1 and 8 in figure 14 will be taken as an example. The normal unit vector is

$$\hat{\mathbf{n}}_{c1-8}^A = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix}. \quad (75)$$

To find the C_{1111} entry of the EMT, the first entry of $\hat{\mathbf{n}}$ must be taken to the fourth power and multiplied by the prefactor:

$$\begin{aligned} (C_{1111})_{c1-8}^A &= \frac{kl^2}{V_c} ((n_1)_{c1-8}^A * (n_1)_{c1-8}^A * (n_1)_{c1-8}^A * (n_1)_{c1-8}^A) \\ &= \frac{kl^2}{V_c} (0.5 * 0.5 * 0.5 * 0.5) \\ &= \frac{kl^2}{V_c} \left(\frac{1}{16} \right). \end{aligned} \quad (76)$$

Every one of the four indices of the EMT refers to the same index of a normal unit vector. So, to take a second example, the $(C_{3212})_{c1-8}^A$ entry is:

$$\begin{aligned} (C_{3212})_{c1-8}^A &= \frac{kl^2}{V_c} ((n_3)_{c1-8}^A * (n_2)_{c1-8}^A * (n_1)_{c1-8}^A * (n_2)_{c1-8}^A) \\ &= \frac{kl^2}{V_c} \left(\sqrt{\frac{2}{3}} * \frac{1}{2\sqrt{3}} * 0.5 * \sqrt{\frac{2}{3}} \right) \\ &= \frac{kl^2}{V_c} \left(\frac{\sqrt{3}}{18} \right). \end{aligned} \quad (77)$$

Doing the calculation for every entry of the EMT for this contact gives the following result (see equation (71) for explanation of the indices):

$$\mathbf{C}_{c1-8}^A = \frac{kl^2}{V_p} \begin{pmatrix} \frac{1}{16} & \frac{1}{48} & \frac{1}{6} & \frac{\sqrt{3}\sqrt{6}}{72} & \frac{\sqrt{6}}{24} & \frac{\sqrt{3}}{48} \\ & \frac{1}{144} & \frac{1}{18} & \frac{\sqrt{3}\sqrt{6}}{216} & \frac{\sqrt{6}}{72} & \frac{\sqrt{3}}{144} \\ & & \frac{4}{9} & \frac{\sqrt{3}\sqrt{6}}{27} & \frac{\sqrt{6}}{9} & \frac{\sqrt{3}}{18} \\ & & & \frac{1}{18} & \frac{\sqrt{3}}{18} & \frac{\sqrt{6}}{72} \\ & & & & \frac{1}{6} & \frac{\sqrt{3}\sqrt{6}}{72} \\ & & & & & \frac{1}{48} \end{pmatrix}. \quad (78)$$

Repeating these calculations for every contact and summing up the resulting twelve matrices gives:

$$\mathbf{C}^A = \frac{kl^2}{V_p} \begin{pmatrix} \frac{5}{2} & \frac{5}{6} & \frac{2}{3} & 0 & 0 & 0 \\ & \frac{5}{2} & \frac{2}{3} & 0 & 0 & 0 \\ & & \frac{8}{3} & 0 & 0 & 0 \\ & & & \frac{2}{3} & 0 & 0 \\ & & & & \frac{2}{3} & 0 \\ & & & & & \frac{5}{6} \end{pmatrix}, \quad (79)$$

using the rule that all equal entries of the matrices can be summed up directly:

$$(\mathbf{C}_{1111})^A = (\mathbf{C}_{1111})_{c1-2}^A + (\mathbf{C}_{1111})_{c1-3}^A + \dots + (\mathbf{C}_{1111})_{c1-13}^A \quad (80)$$

As said before, in section 3.2.1, a sphere in layer ‘*B*’ has a slightly different contact network from a sphere in layer ‘*A*’ (the differences are shown in equations (66) and (67)). In order to see whether or not this has consequences for the elastic modulus tensor, the EMT was also calculated for a sphere in layer ‘*B*’. Using the contact network given in equation (67), together with the first six contacts from equation (66), the same EMT as for a sphere in layer ‘*A*’ was found, i.e.:

$$\mathbf{C}^B = \mathbf{C}^A. \quad (81)$$

This means that no additional averaging procedure is needed to account for the effect of the different contact networks in different layers.

3.4 The unit cell

Before the volume fraction (density) and the potential energy density can be derived for an HCP, a proper unit cell with volume V_p must be defined. A unit cell is a small volume of a packing, but it represents all features of the total system. A complete space-filling packing can be constructed by ‘glueing’ unit cells repeatedly together in all the x -, y - and z -directions. A suitable unit cell must contain all possible types of normal unit vectors. However, it does not necessarily have to contain complete contacts; only an appropriate fraction of the contact is already sufficient to describe the properties of the full contact. Naturally, the size of the unit cell should be kept as small as possible. This reduces the efforts to determine, e.g., the (potential energy) density. A unit cell that fulfills the above mentioned requirements for an HCP has the shape of a hexagonal prism. The hexagonally shaped base lies in the x - y plane, halfway in layer ‘ A ’ (the center of an arbitrary sphere in this layer lies on the same z -coordinate as the base of this cell). From this base, the prism is oriented straight up, parallel to the z -axis. Exactly on halfway the prism lies the center of an arbitrary sphere in layer ‘ B ’. The edges of the hexagonal cross-section cut all the contacts this sphere has in halves. The center of this sphere lies exactly in the center of the hexagon. The upper face of the prism also lies exactly halfway towards another layer ‘ A ’. Figure 16 visualizes what the unit cell looks like by giving a topview and a front view. Figure 17 shows that merging unit cells together forms an HCP and that the cells are repeatable.

3.5 The volume fraction (density)

With the aid of the unit cell defined in section 3.4 the volume fraction of an HCP can be calculated. This can be done by determining the volume of the unit cell and the volume of (parts of) the spheres that this cell contains. The volume fraction is the ratio between this total sphere volume ($V_{s \in cell}$) and cell volume (V_{cell}):

$$\eta_{HCP} = \frac{V_{s \in cell}}{V_{cell}} . \quad (82)$$

The volume of the unit cell will be calculated first. It will be expressed in terms of the micro-variable l , the branch vector. The cell has the geometry of a hexagonal prism. The volume of this prism is the area of its base times its height. The top view in figure 16 shows that the width of the hexagon is exactly one branch vector length. The hexagon can be divided in 12 triangles. One of such a triangle is drawn in figure 18.

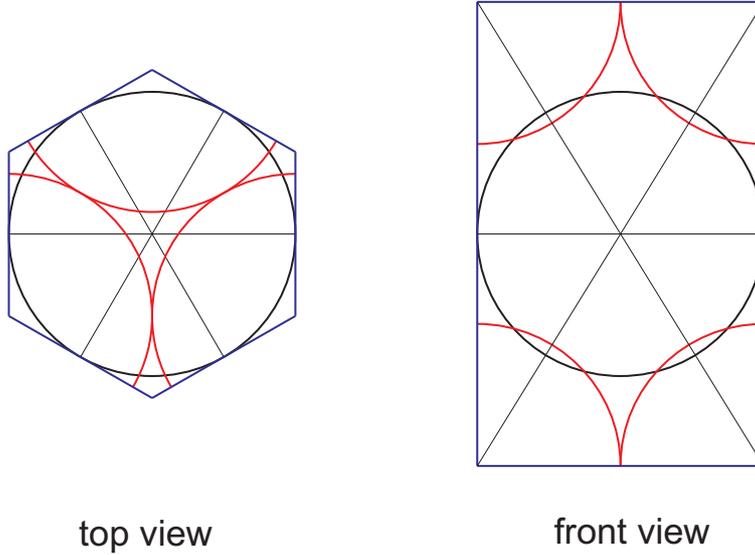


Figure 16: Top view and front view on the unit cell of an HCP. The boundaries of the cell are indicated by the thick, blue lines. If it is assumed that the black-lined sphere is in a layer 'A', then the red lines (arcs) are parts of spheres in 'B'-layers. The thin black lines are branch vectors, connecting the centers of contacting spheres.

The area of this triangle is

$$A_{triangle} = \frac{1}{2} * 0.5l * b . \quad (83)$$

The height b can be expressed in terms of l :

$$b = \tan 30^\circ * 0.5l = \frac{l}{2\sqrt{3}} , \quad (84)$$

giving for $A_{triangle}$:

$$A_{triangle} = \frac{1}{2} * 0.5l * \frac{l}{2\sqrt{3}} = \frac{l^2}{8\sqrt{3}} . \quad (85)$$

The total area of the hexagon is thus:

$$A_{hex} = 12 * A_{triangle} = \frac{12l^2}{8\sqrt{3}} = \frac{\sqrt{3}}{2} * l^2 . \quad (86)$$

In section 3.2.1 the normal unit vector for (among others) contacts between particles in neighbouring layers was determined. From this unit vector it is known that the shortest distance between the centers of two layers is $\sqrt{\frac{2}{3}}l$. The front view in figure 16 shows that the unit cell spans exactly two layers of spheres. The height of the prism is therefore:

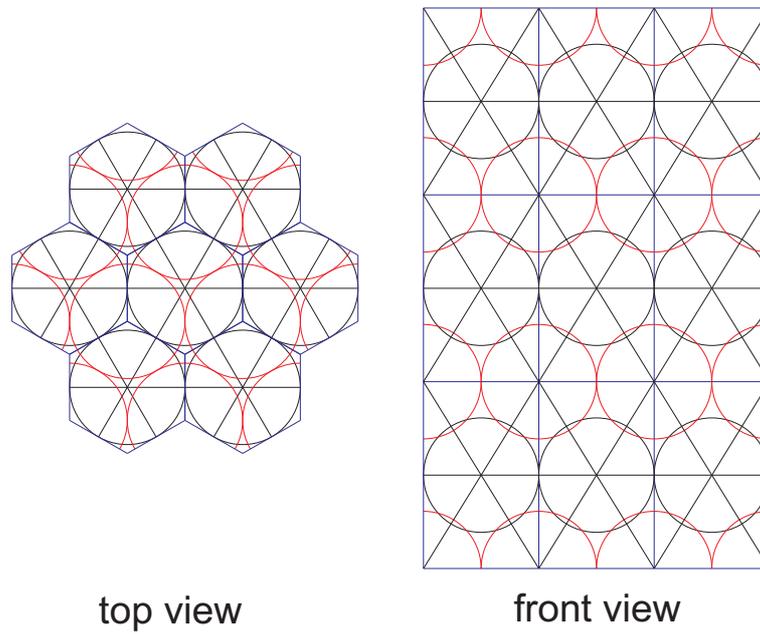


Figure 17: Merging unit cells together forms an HCP.

$$h_{hex} = 2\sqrt{\frac{2}{3}} * l . \quad (87)$$

Finally, the volume of the unit cell is:

$$\begin{aligned} V_{cell} &= A_{hex} * h_{hex} \\ &= \frac{\sqrt{3}}{2} * l^2 * 2\sqrt{\frac{2}{3}} * l \\ &= \sqrt{2} * l^3 . \end{aligned} \quad (88)$$

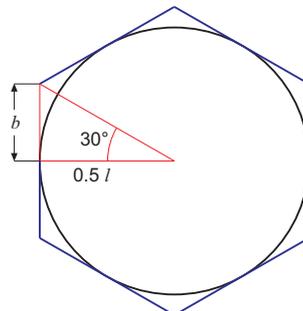


Figure 18: Dimensions of the hexagonal base of the unit cell

The next step is to determine the volume of the spheres that one unit cell contains. From figure 16 it can be seen that a unit cell contains one complete sphere and six fractions of different spheres. These six fractions together form one complete sphere, so a unit cell contains the volume of 2 spheres. Assuming that the overlapping volume between spheres is negligible compared to the total volume of a sphere, the volume of the spheres can be expressed in terms of l , which is two times the sphere radius. The volume of the two spheres is therefore:

$$\begin{aligned} V_{s \in cell} &= 2 * \frac{4\pi}{3} * \left(\frac{1}{2}l\right)^3 \\ &= \frac{\pi}{3} * l^3 . \end{aligned} \tag{89}$$

Using equation (82) it follows that the density of an HCP is:

$$\begin{aligned} \eta_{HCP} &= \frac{V_{s \in cell}}{V_{cell}} \\ &= \frac{\pi * l^3}{3 * \sqrt{2} * l^3} \\ &= \frac{\pi}{3\sqrt{2}} \approx 0.7405 . \end{aligned} \tag{90}$$

This density is known to be the highest that a packing of equi-sized spheres can reach.

3.6 Potential energy density

In the following, the potential energy of an HCP will be expressed with continuum quantities. However, single contact properties are not ‘real’ continuum quantities. The goal is to show that local, single contact properties can be expressed in continuum notation.

The potential energy density of the system is the sum over the potential energies of every single contact contained within the unit cell, divided by the volume of the unit cell:

$$u = \frac{1}{V_{cell}} \sum_{c=1}^C u^c . \tag{91}$$

The potential energy as a function of the stress and strain tensor is defined as

$$u^c = \frac{1}{2} (\boldsymbol{\sigma}^c : \boldsymbol{\epsilon}^c) , \tag{92}$$

where $\boldsymbol{\sigma}^c$ is the stress tensor for a single contact (see equation (73)) and $\boldsymbol{\epsilon}^c$ is the strain tensor for a single contact (see equation (72)). The definition in equation (92) is based on the assumption that the stress and strain tensor are symmetric. The colon in equation (92) denotes the double dot product between $\boldsymbol{\sigma}^c$ and $\boldsymbol{\epsilon}^c$. A double dot product of two tensors equals the trace of the dot product between these two tensors. The proof for equation (92) (for 2D) is given in Appendix A. When only deformations in the normal direction are considered, the strain tensor can be written as

$$\boldsymbol{\epsilon}^c = \frac{\delta}{l} \hat{\mathbf{n}}^c \hat{\mathbf{n}}^c, \quad (93)$$

where δ is the overlap between the contacting particles and l is the branch vector, connecting the centers of the contacting spheres. The stress tensor is defined as

$$\boldsymbol{\sigma}^c = \frac{kl\delta}{V_c} \hat{\mathbf{n}}^c \hat{\mathbf{n}}^c. \quad (94)$$

The term $\hat{\mathbf{n}}^c \hat{\mathbf{n}}^c$ denotes the dyadic product of the unit vector in normal direction.

A unit cell of an HCP contains three types of contacts:

1. The contacts of the center sphere with spheres that are in the same $x - y$ plane. These are contacts $c1 - 2$ to $c1 - 6$ in figure 14. Only half the lengths of these contacts lie in the unit cell.
2. The contacts of the center sphere with the spheres in the layers above and below. These are contacts $c1 - 7$ to $c1 - 13$ in figure 14. The full lengths of these contacts lie in the unit cell.
3. The contacts between spheres whose centers lie at the bottom or top plane of the unit cell. These are contacts $c8 - 9$, $c9 - 10$, $c8 - 10$, $c11 - 12$, $c12 - 13$ and $c11 - 13$ in figure 14. The full lengths of these contacts lie in the unit cell, but only half of their volumes lie inside the cell.

It is assumed here that all the contacts (in their complete form) have the same stiffness k . Contacts of the same type therefore have the same potential energy under identical deformation, i.e. for identical δ . For every type of contact, one contact will be used as an example to determine the potential energy.

3.6.1 Contact type 1

For the first type, the potential energy of contact $c1 - 3$ will be determined. Rewriting the definition of the potential energy density of equation (92) for this contact gives:

$$u^{(c1-3)} = \frac{1}{2} \left(\boldsymbol{\sigma}^{(c1-3)} : \boldsymbol{\epsilon}^{(c1-3)} \right) . \quad (95)$$

Since only half of the length of the contact lies in the unit cell, the stress and strain tensor must be expressed as a function of δ' , l' and k' , where the prime indicates that it must be taken for a half contact, so:

$$\begin{aligned} \delta' &= \frac{1}{2} \delta \\ l' &= \frac{1}{2} l \\ k' &= 2k . \end{aligned}$$

It may be counter-intuitive that the spring stiffness doubles when the spring length is halved, but it follows from the definition of the spring constant, which says that this constant is found by dividing the force acting on the spring by its compression (expressed in length units). In this problem, the force that acts on the spring is the same, but the compression of the spring is halved, which results in the doubled spring constant. The strain tensor for contact $c1 - 3$ is:

$$\begin{aligned} \boldsymbol{\epsilon}^{(c1-3)} &= \frac{\delta'}{l'} \hat{\mathbf{n}}^{(c1-3)} \hat{\mathbf{n}}^{(c1-3)} \\ &= \frac{\frac{1}{2} \delta}{\frac{1}{2} l} \hat{\mathbf{n}}^{(c1-3)} \hat{\mathbf{n}}^{(c1-3)} \\ &= \frac{\delta}{l} \hat{\mathbf{n}}^{(c1-3)} \hat{\mathbf{n}}^{(c1-3)} . \end{aligned} \quad (96)$$

The stress tensor for contact $c1 - 3$ is:

$$\begin{aligned} \boldsymbol{\sigma}^{(c1-3)} &= \frac{k' l' \delta'}{V_c} \hat{\mathbf{n}}^{(c1-3)} \hat{\mathbf{n}}^{(c1-3)} \\ &= \frac{2k \frac{1}{2} l \frac{1}{2} \delta}{V_c} \hat{\mathbf{n}}^{(c1-3)} \hat{\mathbf{n}}^{(c1-3)} \\ &= \frac{k l \delta}{2V_c} \hat{\mathbf{n}}^{(c1-3)} \hat{\mathbf{n}}^{(c1-3)} . \end{aligned} \quad (97)$$

Both tensors are written as a function of a prefactor times a matrix, which is the dyadic product of the normal unit vector $\hat{\mathbf{n}}^{(c1-3)} = [\frac{1}{2}, \frac{1}{2}\sqrt{3}, 0]$:

$$\hat{\mathbf{n}}^{(c1-3)} \hat{\mathbf{n}}^{(c1-3)} = \begin{bmatrix} \frac{1}{4} & \frac{\sqrt{3}}{4} & 0 \\ \frac{\sqrt{3}}{4} & \frac{3}{4} & 0 \\ 0 & 0 & 0 \end{bmatrix} . \quad (98)$$

The dot product $\boldsymbol{\sigma} \cdot \boldsymbol{\epsilon}$ is the product

$$\boldsymbol{\sigma} \cdot \boldsymbol{\epsilon} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix} \cdot \begin{bmatrix} \epsilon_{11} & \epsilon_{21} & \epsilon_{31} \\ \epsilon_{12} & \epsilon_{22} & \epsilon_{32} \\ \epsilon_{13} & \epsilon_{23} & \epsilon_{33} \end{bmatrix}. \quad (99)$$

The potential energy density is a half times the trace of this product. This can be written as (see Appendix A for more details):

$$\begin{aligned} u &= \frac{1}{2}([\sigma_{11}\epsilon_{11} + \sigma_{21}\epsilon_{21} + \sigma_{13}\epsilon_{13}] \\ &\quad + [\sigma_{21}\epsilon_{21} + \sigma_{22}\epsilon_{22} + \sigma_{23}\epsilon_{23}] \\ &\quad + [\sigma_{31}\epsilon_{31} + \sigma_{32}\epsilon_{32} + \sigma_{33}\epsilon_{33}]). \end{aligned} \quad (100)$$

For contact (c1 – 3) the result is

$$\begin{aligned} u &= \frac{1}{2} \left(\frac{kl\delta}{2V_c} \right) \left(\frac{\delta}{l} \right) \left[\left(\left(\frac{1}{4} \right)^2 + \left(\frac{\sqrt{3}}{4} \right)^2 + (0)^2 \right) \right. \\ &\quad \left. + \left(\left(\frac{\sqrt{3}}{4} \right)^2 + \left(\frac{3}{4} \right)^2 + (0)^2 \right) + \left((0)^2 + (0)^2 + (0)^2 \right) \right] \\ &= \left(\frac{k\delta^2}{4V_c} \right) \left(\frac{1}{4} + \frac{3}{4} + 0 \right) \\ &= \frac{k\delta^2}{4V_c}. \end{aligned} \quad (101)$$

All the contacts of this type have the same potential energy density as (c1 – 3), so together this gives

$$u^{(c1-2 \rightarrow 7)} = 6 * \frac{k\delta^2}{4V_c} = \frac{3k\delta^2}{2V_c}. \quad (102)$$

3.6.2 Contact type 2

Now the potential energy for the second type of contacts will be determined. These contacts lie completely in the unit cell, so the stress and strain tensor are as described in equations (94) and (93), respectively. Let's take contact (c1 – 8) as an example for deriving the potential energy. Its normal unit vector is $\hat{\mathbf{n}}^{(c1-8)} = [\frac{1}{2}, \frac{1}{2\sqrt{3}}, \sqrt{\frac{2}{3}}]$, so the dyadic product ($\hat{\mathbf{n}}^{(c1-8)}\hat{\mathbf{n}}^{(c1-8)}$) is:

$$\hat{\mathbf{n}}^{(c1-8)}\hat{\mathbf{n}}^{(c1-8)} = \begin{bmatrix} \frac{1}{2} & \frac{1}{4\sqrt{3}} & \frac{1}{2}\sqrt{\frac{2}{3}} \\ \frac{1}{4\sqrt{3}} & \frac{1}{12} & \frac{1}{\sqrt{18}} \\ \frac{1}{2}\sqrt{\frac{2}{3}} & \frac{1}{\sqrt{18}} & \frac{2}{3} \end{bmatrix}, \quad (103)$$

using equation (100), the potential energy density for this contact is

$$\begin{aligned}
u &= \frac{1}{2} \left(\frac{k l \delta}{V_c} \right) \left(\frac{\delta}{l} \right) \left[\left(\left(\frac{1}{4} \right)^2 + \left(\frac{1}{4\sqrt{3}} \right)^2 + \left(\frac{1}{2\sqrt{3}} \right)^2 \right) \right. \\
&\quad + \left(\left(\frac{1}{4\sqrt{3}} \right)^2 + \left(\frac{1}{12} \right)^2 + \left(\frac{1}{\sqrt{18}} \right)^2 \right) \\
&\quad \left. + \left(\left(\frac{1}{2\sqrt{3}} \right)^2 + \left(\frac{1}{\sqrt{18}} \right)^2 + \left(\frac{2}{3} \right)^2 \right) \right] \\
&= \left(\frac{k \delta^2}{2V_c} \right) \left(\frac{1}{4} + \frac{1}{12} + \frac{2}{3} \right) \\
&= \frac{k \delta^2}{2V_c} .
\end{aligned} \tag{104}$$

There are six contacts of this type, so the result is multiplied by six:

$$u^{(c1-8 \rightarrow 13)} = 6 * \frac{k \delta^2}{2V_c} = \frac{3k \delta^2}{V_c} . \tag{105}$$

3.6.3 Contact type 3

Finally, the potential energy for the third type of contact will be calculated. Contact (c8 – 9) will be taken as an example. The normal unit vector for this contact is $\hat{\mathbf{n}}^{(c8-9)} = [-1, 0, 0]$, giving for the dyadic product with itself:

$$\hat{\mathbf{n}}^{(c8-9)} \hat{\mathbf{n}}^{(c8-9)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} . \tag{106}$$

The full length of the contact lies in the unit cell, but the contact is cut in halves over the $x - y$ plane, so only half of its volume is within a unit cell. The overlap δ and the branch vector l remain the same as for a full contact, but the stiffness constant k is two times smaller, giving for the stress and strain tensor:

$$\boldsymbol{\sigma}^{(c8-9)} = \frac{\frac{1}{2} k l \delta}{V_c} \hat{\mathbf{n}}^{(c8-9)} \hat{\mathbf{n}}^{(c8-9)} \tag{107}$$

$$\boldsymbol{\epsilon}^{(c8-9)} = \frac{\delta}{l} . \tag{108}$$

The potential energy for this contact can immediately be written down, since there is only one non-zero element in the dyadic product tensor of equation (106):

$$\begin{aligned}
u^{(c8-9)} &= \frac{1}{2} \left(\frac{k\delta^2}{2V_c} \right) (1^2) \\
&= \frac{k\delta^2}{4V_c} .
\end{aligned} \tag{109}$$

Also for this type there are six contacts:

$$\begin{aligned}
u^{(c8-9) \rightarrow (c12-13)} &= 6 * \frac{k\delta^2}{4V_c} \\
&= \frac{3k\delta^2}{2V_c} .
\end{aligned} \tag{110}$$

3.6.4 Total potential energy of the unit cell

The total potential energy that one unit cell contains is the sum of the potential energy of every contact of that cell, so the obtained results for the three types of contacts can be summed up to give:

$$\begin{aligned}
u^{cell} &= u^{(c1-2 \rightarrow 7)} + u^{(c1-8 \rightarrow 13)} + u^{(c8-9) \rightarrow (c12-13)} \\
&= \frac{3k\delta^2}{2V_c} + \frac{3k\delta^2}{V_c} + \frac{3k\delta^2}{2V_c}
\end{aligned} \tag{111}$$

$$\boxed{u^{cell} = \frac{6k\delta^2}{V_c}} . \tag{112}$$

In section 3.5 it was calculated that the volume of the unit cell is, expressed as a function of the branch vector:

$$V_{cell} = \sqrt{2} * l^3 . \tag{113}$$

The potential energy density for an HCP is thus:

$$u^{cell} = \frac{6k\delta^2}{\sqrt{2}l^3} = \frac{3\sqrt{2}k\delta^2}{l^3} . \tag{114}$$

This result can also be derived with the following reasoning: A perfect periodic HCP has an average of six full contacts per particle. The unit cell used here contains two spheres, so this should correspond to twelve full contacts. Therefore, the potential energy density should equal twelve times the potential energy of a full contact, divided by the volume of the unit cell:

$$\begin{aligned}
u^{cell} &= \frac{12 * \frac{1}{2}k\delta^2}{V_c} \\
&= \frac{6k\delta^2}{\sqrt{2}l^3} \\
&= \frac{3\sqrt{2}k\delta^2}{l^3} .
\end{aligned}
\tag{115}$$

This is indeed the same result as found in equation (114).

In this chapter, some properties of the HCP have been derived and expressed as a function of single contact properties. These expressions will be used to evaluate how accurately the properties of the HCP can be predicted from DEM simulations. The following chapter will first give an introduction to DEM and details on how the DEM simulations were performed will be presented. After that, in chapter 5, the obtained properties of the HCP from the DEM simulations will be given.

4 Discrete Element Model simulations

This chapter deals with the Discrete Element Model (abbreviated as DEM) simulations that were performed with the HCP. First, a general introduction to DEM will be given. Next, the preparation of the HCP will be explained, both for periodic and wall boundary conditions. Finally, methods will be described to extract and process information from the simulation. The results from the simulations will then be discussed in the next chapter.

4.1 Introduction to DEM

The Discrete Element Model (DEM) as applied to rocks and other granular materials like sand was introduced by Cundall and Strack in a publication in 1979 [4]. It was developed to model the mechanical behaviour of disks and spheres. It is a numerical model, where the motion of all the particles is determined for every time interval. The particles move independently from one another and interact only at contact points. Since DEM simulations are processor intensive, there is a limitation in the time duration of the simulation and number of discrete particles in the system. Advances in processor technology and software expand these limits continuously and the possibility of running simulations on distributed systems increase the capacity even further.

A DEM simulations starts with a system of particles, with the initial positions and initial linear (and angular) velocities of every particle specified. Then the forces which act on each particle are computed for every time interval from the data available using relevant physical laws. The most important types of forces that can be modelled with DEM are recoil (when particles collide), friction and gravity. Friction will not be considered in the following. All these forces are added up to find the total force acting on a particle. The new position and velocity is calculated with Newtons equations of motion for every particle, over the duration of a time step:

$$m_i \frac{d^2}{dt^2} \vec{r}_i = \vec{f}_i, \quad (116)$$

where m_i is the mass of particle i , \vec{r}_i is its position vector and the total force $\vec{f}_i = \sum_c \vec{f}_i^c$ acting on it due to contacts with other particles or with the walls. When the new positions and velocities have been calculated, the force balances are set up again and the cycle is repeated.

4.1.1 Numerical time-step

It is very important to choose the right time step $\Delta t_{\mu D}$ for the simulation. When the time-step is much too large, the errors make the simulation irrelevant, typically leading to a dramatic increase of energy. When the time-step is too small, either the simulation has to run too long, or the numerical

accuracy of 16-bit numbers is not sufficient any more. A useful guideline is to take a look at the contact time of a collision between two particles. To accurately describe a collision, it must take around forty to fifty time steps to complete a collision, from the time of the first overlap to the time that the particles overlap for the last time (more on the overlap of particles will follow later in this section). The contact time of a collision is described by the following relation:

$$t_c = \frac{\pi}{\omega}, \quad \text{with} \quad \omega = \sqrt{\frac{k}{m_{12}} - \eta_0^2}, \quad (117)$$

where k is the stiffness and ω is the eigenfrequency of the contact. The rescaled damping coefficient is $\eta_0 = \gamma_0/(2m_{12})$, with the reduced mass

$$m_{12} = \frac{m_1 m_2}{m_1 + m_2}. \quad (118)$$

When the simulated particles have a radius of $10^{-3} m$, a density of $2000 kg/m^2$ and a stiffness constant $k = 10^5 N/m$, the contact time of a collision is $t_c = 2.3 \cdot 10^{-5}$ seconds, so the time step of the simulation should be around $\Delta t_{\mu D} = 4 \cdot 10^{-7}$ seconds. To put it in another way, it takes 2.5 million time steps to simulate one second. Even with powerful processors, it may take weeks or even months to make a one minute simulation of a system of say 40,000 particles.

4.1.2 Physical interpretation

In reality, particles deform when they collide. At the contact point, there is a slight compression and the surfaces are flattened. The higher the impact velocity of the particles, the more the particles are compressed and deformed and this leads to a higher recoil force. However, it is too complex to accurately describe the deformation of a particle during a collision with DEM. This problem is solved by allowing the particles to overlap. As soon as the particles start to overlap, a virtual spring is activated, that wants to push (recoil) the particles away from each other (see figure 19). The greater the overlap, the higher the repulsive force.

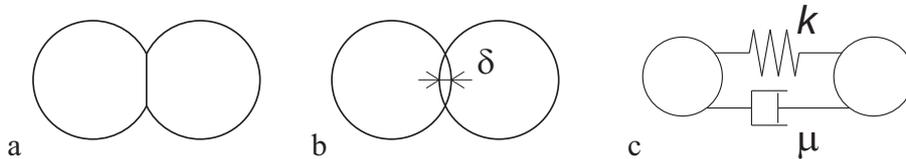


Figure 19: a) In reality, particles deform when they collide and at the contact the surfaces flatten. b) In DEM, particles are allowed to overlap to model deformations. c) A collision is modelled with a spring (the repulsive force) and a dashpot (damping).

There are various contact models that can be used to relate the overlap to the repulsive force. The most important ones are the linear contact model, where $f_{recoil} \propto \delta$, and the Hertz contact law, where $f_{recoil} \propto \delta^{3/2}$. The former is typically applied as the most simple approach or when plastic deformations must be modelled (in that case, the spring stiffness for loading is different from the spring stiffness for unloading). The latter describes the deformation behaviour of perfectly elastic spheres [6]. For the simulations that were done for this project, the simplest linear contact model was used in combination with a linear damping coefficient.

4.1.3 The compression tests

In some of the DEM simulations one wall is moved inwards, compressing the packing. This movement is strain-controlled. This means that the position of this wall is defined as a function of time t . The shape of this function is a cosine and it is activated for a half period. After that half period, the wall stops moving. The position of the wall ($z(t)$) is (see also figure 20):

$$z(t) = z_f + \frac{z_0 - z_f}{2} (1 + \cos \omega t) \quad \text{for} \quad 0 \leq \omega t \leq \pi \quad (119)$$

where z_0 is the initial wall position, z_f is the final wall position and ω is the rate of deformation ($\omega = 2\pi f$).

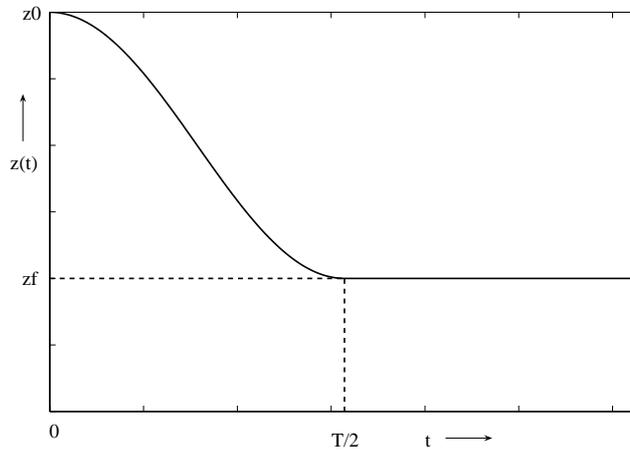


Figure 20: The position of the wall as a function of time t . After a half period, the wall stops moving.

The shape of the function that describes the wall motion is arbitrary, as long as it starts slowly and it stops gradually, in order to reduce shock and inertia effects.

4.2 Preparation of the packing

In order to do a DEM simulation, a system of discrete particles must be defined first. First the initial position of every particle has to be provided. When the system contains only a couple of particles, the positions can be determined manually, but for larger systems it is more efficient to make a script that computes the initial positions. Since the performed simulations for this project contain of the order of 10^3 spheres, a Fortran code was written that calculates the initial coordinates of every particle in the system (see appendix B). This code uses the following properties of the system:

- Sphere radius
- The overlap δ between spheres
- Sphere density
- The number of particles in the system
- Initial velocities; linear, in x -, y - and z -direction and angular, around the x , y and z axis

and provides:

- The size of the packing, from the number of particles in the x -direction, the number of rows (in the y -direction) and the number of layers (in the z -direction)
- The position of the walls, which can be calculated from the sphere radius, overlap and size of the packing.

The code is constructed in such a way that parameters can easily be adjusted. If for example the sphere radius has to be changed, only one number has to be redefined to calculate the new coordinates of the spheres in the packing.

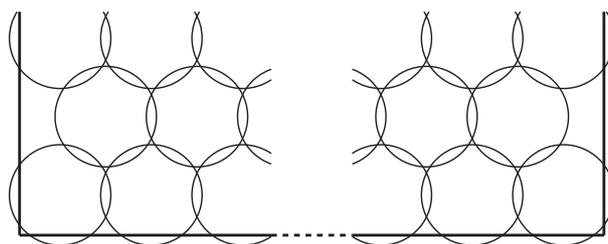
A DEM simulation can be done with two types of boundaries:

- Periodic boundaries
- Wall boundaries

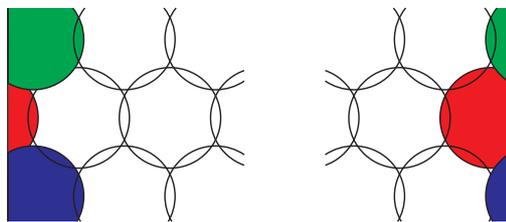
In a system with periodic boundaries, the particles that travel through, for example, the right side of the system, reappear on the left side. If the system would contain one particle with a certain velocity v , the particle could travel an infinitely long time and distance without ever hitting a boundary, but instead jumping again and again from one side of the system to another. With a periodic boundary system, it is possible to make a ‘perfect’ dense

packing. Particles whose centers are at a virtual boundary at the edge of the system, touch the particles that are at the other side of the system (as it appears on the screen).

With a wall boundary simulation, on the other hand, the system contains ‘real’ walls. A single particle with a velocity v can only travel for a limited time and distance before it hits a boundary and it bounces of. With a wall boundary system, it is not possible to make a ‘perfect’ packing. If the spheres in the bottom layer completely cover the bottom, then the spheres at the boundaries in the second layer do not touch a wall and an empty space is created between the spheres and the walls (see figure 21).



wall boundaries



periodic boundaries

Figure 21: Packings with wall and periodic boundaries. The additional empty space near the walls is clearly visible in the system with walls. In the figure of the periodic system, the equally colored pieces belong to the same sphere. Note that the overlaps are exaggerated.

Since a perfect packing can be made with a small periodic boundary system, it is possible to obtain properties of the packing that match with theoretical predictions for a huge system, a crystal for example (with of the order of 10^{23} atoms). Every particle has the same (or similar) contact network (as discussed in chapter 3; the contact network is not identical but similar for spheres in A -layers with respect to spheres in B -layers). As a result, the potential energy density and the elastic modulus tensor should also match theoretical predictions. However, in ‘real life’, periodic boundaries do not exist. If one wants to compare results from real experiments with simula-

tions, it is preferred to use wall boundaries in the simulations. The system size must be sufficiently large to minimize the effect of the walls.

In the following, it will be explained how the packings with periodic and wall boundaries were created with the Fortran code.

4.2.1 Periodic boundaries

The HCP with periodic boundary conditions is created in several steps. These steps consist of several loops. The fastest way to write a script for the packing would be to create all the odd numbered rows of a layer in one loop, followed by the even numbered rows in another loop and in the same manner with the odd and even numbered layers. Although the HCP can easily be made this way, it has the disadvantage that spheres are numbered in a non-continuous way. The number sequence skips a row and a layer every time. When it is desired to pick out a specific sphere in the system, it is difficult to deduce its number. To solve this problem, one odd and one even row is defined in one loop, so the number sequence is continuous. Also one ‘A’- and one ‘B’-layer are made in one loop. For the periodic packing, it is important to keep in mind that the spheres at the boundaries touch the spheres at the other sides of the system. It is for instance not desirable to make a packing where the top and bottom layers are both either ‘A’- or either ‘B’-type, because these layers are contiguous in the simulation. For a perfect HCP, every layer must alternate in type. The used algorithm in the Fortran script to create an HCP for a system with periodic boundary conditions is given in Appendix B.1.

There is some freedom to choose the coordinates of the spheres in the packing, as long as the first sphere near a boundary is no more than one sphere radius away from the boundary. The position of the (virtual) walls must be chosen such that spheres on opposite sides of the screen touch each other. Figure 22 shows two layers of spheres and the position of the (virtual) walls for the periodic boundary system.

The spacing between the centers of spheres in the x -direction is always the length of the branch vector ($2a - \delta = l$). In the packing made with the Fortran code, the center of the first sphere of the first row of the first layer is a half sphere radius away from the three nearest virtual boundaries (left wall, front wall and bottom wall), as indicated in figure 22. The x -coordinate $p_{x,odd}$ of the i th sphere of a total of n spheres in the first row of the first layer can be calculated with:

$$p_{x,odd} = (2a - \delta)(i - 1) + \frac{1}{2}a \quad \text{for} \quad i = 1 \dots n . \quad (120)$$

A sphere in the second row of the first layer lies exactly between two spheres of the row below, so the x -coordinates of spheres in this row can be described by:

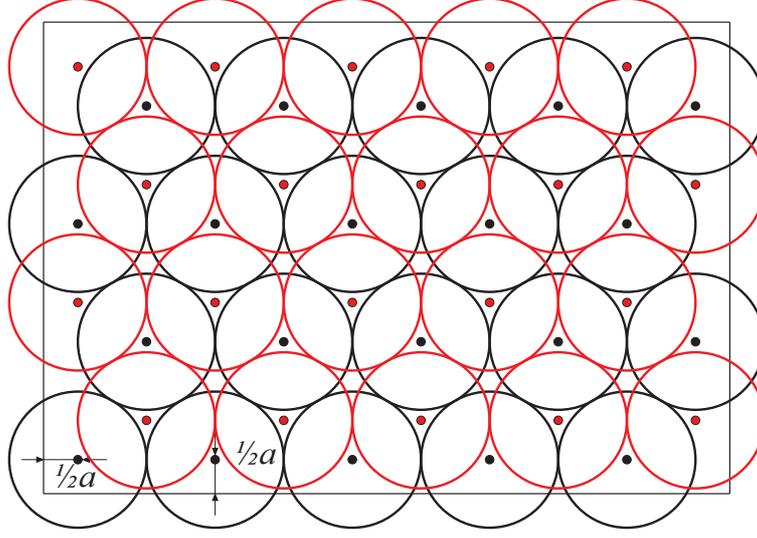


Figure 22: A top view on two layers of spheres with radius a in a system with periodic boundaries. The black-lined spheres form the bottom layer (layer ‘A’) and the red-lined spheres form the next layer ‘B’.

$$p_{x,even} = (2a - \delta)(i - 1) + \frac{1}{2}(2a - \delta) + \frac{1}{2}a \quad \text{for } i = 1 \dots n. \quad (121)$$

For the second layer, the x -coordinates of the first and second rows are just the reverse from coordinates of the the layer below. The spacing between the centers of spheres in the y -direction is $\frac{1}{2}\sqrt{3}l$. As was explained above, there is a loop in the Fortran code that repeats after an odd and an even numbered row have been defined. Therefore, the spacing between two rows must be taken to calculate the y -coordinates. For the odd rows in the first layer, this reads:

$$p_{y,odd} = \frac{1}{2}\sqrt{3}(2a - \delta)(j - 1) * 2 + \frac{1}{2}a \quad \text{for } j = 1 \dots p, \quad (122)$$

where j is the number of odd/even row pairs in a layer, with a total of p pairs. Of course the terms $\frac{1}{2}$ and 2 cancel each other, but it is left in the equation to keep it clear where the contributions come from. The y -coordinates of the spheres in the even rows of the first layers are:

$$p_{y,even} = \frac{1}{2}\sqrt{3}(2a - \delta)(j - 1) * 2 + \frac{1}{2}\sqrt{3}(2a - \delta) + \frac{1}{2}a \quad \text{for } j = 1 \dots p. \quad (123)$$

For the second layer, the y coordinates are not just simply the reverse of the coordinates of the first layer. With respect to the first layer, the second layer

is shifted $\frac{1}{2\sqrt{3}}(2a - \delta)$ in the y -direction. Finally, the spacing between the centers of spheres between neighbouring layers is $\sqrt{\frac{2}{3}}$. The layers defined in A/B -pairs. The z -coordinate $p_{z,A}$ of a sphere in the k th layer A , with a total of q layer pairs, is:

$$p_{z,A} = \sqrt{\frac{2}{3}}(2a - \delta)(k - 1) * 2 + \frac{1}{2}a \quad \text{for} \quad k = 1 \dots q . \quad (124)$$

For B -layers, the z -coordinates $p_{z,B}$ are:

$$p_{z,A} = \sqrt{\frac{2}{3}}(2a - \delta)(k - 1) * 2 + \sqrt{\frac{2}{3}}(2a - \delta) + \frac{1}{2}a \quad \text{for} \quad k = 1 \dots q . \quad (125)$$

In the Fortran code, one of the two equations for every coordinate direction (x , y and z) is combined in one command line. Since there are two types of rows (odd and even) and two types of layers (' A ' and ' B '), there are four command lines needed to define the HCP. For instance, the line that defines the position of the spheres in an odd row of an A -layer, contains equations (120), (122) and (124).

The position of the walls also has to be defined in the script. For the left, front and bottom walls, the x , y and z coordinates respectively are simply zero. The x -coordinate of the right wall can be determined by imaginarily extending the first row of the first layer with one sphere (this sphere could be seen as the first sphere of the layer), the position of the right wall is then a half sphere radius on the left of the center of this sphere. The x -coordinate $p_{x,ima}$ of this imaginary sphere would be

$$p_{x,ima} = (2a - b)n + \frac{1}{2}a , \quad (126)$$

so subtracting $\frac{1}{2}a$ simply gives for the right wall position $p_{rightwall}$:

$$p_{rightwall} = (2a - b)n . \quad (127)$$

Following the same procedure, the positions of the back and top wall are

$$p_{backwall} = \frac{1}{2}(2a - b)p * 2 , \quad (128)$$

and

$$p_{topwall} = \sqrt{\frac{2}{3}}(2a - \delta)q * 2 . \quad (129)$$

Finally, the number of spheres that the packing contains must be provided by the Fortran script. Since every row runs from 1 to n spheres, every

layer has p row pairs and there are q pairs of A/B layers, the number of spheres N_{HCP} is:

$$N_{HCP} = 4 * n * p * q . \quad (130)$$

An HCP with periodic boundaries made with this code must have a minimum size. It is not allowed for a sphere to have two contact points with a neighbouring sphere, because the software tools that process information from the simulation cannot handle this. Such a contact could happen when a row contains only 2 spheres; there is the obvious contact, but there would also be a contact with the part of the sphere that goes through the boundary and appears on the other side, as is shown in figure 23.

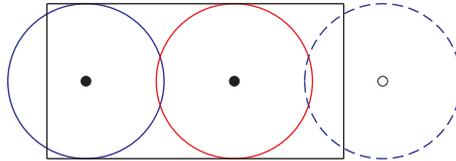


Figure 23: With periodic boundaries, a row with only two spheres would have two contact points between the same two spheres, while no sphere is allowed to contact another sphere more than once.

The minimum number of spheres in one row is thus three. For the same reason, the minimum number of row pairs is two and for the A/B layer pairs it is also 2. Therefore, the smallest HCP that can be used for simulations with DEM contains $4 * 3 * 2 * 2 = 48$ spheres.

4.2.2 Wall boundaries

The Fortran code that defines an HCP with wall boundary conditions has a similar structure to the code for periodic boundaries. However, there are a couple of important differences:

- There is no freedom to choose the position within a certain range; they always have a fixed position from the walls which is determined by the sphere radius and the desired overlap.
- In the ‘ A ’-layers, the odd rows have one sphere more than the even rows. In the ‘ B ’-layers, it is the other way around.
- Layer ‘ B ’ has one row less than layer ‘ A ’. As a result, an additional command line is needed to define the last row of ‘ B ’-layers
- As a result of the above two mentioned differences, the calculation of the number of spheres in the packing is less straightforward.

The used algorithm to define the positions of the spheres for a wall boundary simulation is given in Appendix B.2. A top view on two layers of an HCP with wall boundary conditions is given in figure 24.

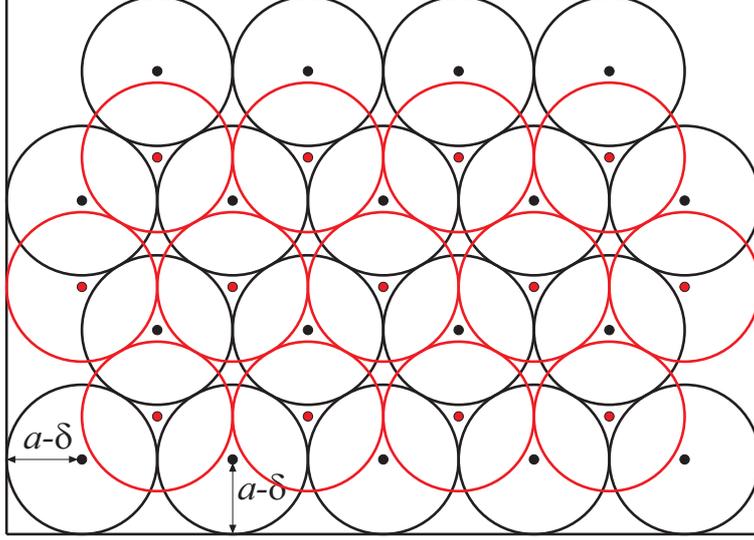


Figure 24: A top view on two layers of an HCP with wall boundary conditions. The black spheres form the bottom layer and the red spheres the layer above. The relatively large empty spaces near the walls are clearly visible.

The spacings between spheres in x -, y - and z -direction are the same as for an HCP with periodic boundaries. For the odd and even rows in ‘A’-layers, the x -coordinate of the i th sphere in a row with n (for odd rows) or $n - 1$ (for even rows) reads:

$$p_{x,odd} = (2a - \delta)(i - 1) + a - \delta \quad \text{for} \quad i = 1 \dots n \quad (131)$$

$$p_{x,even} = (2a - \delta)(n - 1) + \frac{1}{2}(2a - \delta) + a - \delta \quad \text{for} \quad i = 1 \dots n - 1. \quad (132)$$

The x -coordinates of the spheres in the odd and even rows in ‘B’-layers are the reverse of as for the coordinates of odd and even rows in ‘A’-layers. The y -coordinate for the j th sphere in an odd row in an ‘A’ layer is:

$$p_{y,odd} = \frac{1}{2}\sqrt{3}(2a - \delta)(j - 1) * 2 + a - \delta \quad \text{for} \quad j = 1 \dots p \quad (133)$$

and for the even rows of an ‘A’-layer:

$$p_{y,even} = \frac{1}{2}\sqrt{3}(2a - \delta)(j - 1) * 2 + \frac{1}{2}\sqrt{3}(2a - \delta) + a - \delta \quad \text{for} \quad j = 1 \dots p. \quad (134)$$

Layer ‘*B*’ is shifted $\frac{1}{2\sqrt{3}}(2a - \delta)$ in the *y*-direction with respect to layer ‘*A*’. However, this layer has one row less, so the loop is repeated for $j = 1, p - 1$. The last row of a ‘*B*’-layer must be defined with a separate command line. This last row is an odd row. The *y*-coordinate for the spheres in this row is:

$$p_{y,last} = \frac{1}{2}\sqrt{3}(2a - \delta)(p - 1) * 2 + \frac{1}{2\sqrt{3}}(2a - \delta) + a - \delta . \quad (135)$$

Finally, the *z*-coordinates for spheres in ‘*A*’- and ‘*B*’-layers are respectively:

$$p_{z,A} = \sqrt{\frac{2}{3}}(2a - \delta)(k - 1) * 2 + a - \delta \quad \text{for } k = 1 \dots q \quad (136)$$

and

$$p_{z,B} = \sqrt{\frac{2}{3}}(2a - \delta)(k - 1) * 2 + \sqrt{\frac{2}{3}}(2a - \delta) + a - \delta \quad \text{for } k = 1 \dots q . \quad (137)$$

Next, the positions of the walls have to be defined in the Fortran script. Again, the coordinates of the left, front and bottom walls are set to zero. The coordinates of the right, back and top walls become, respectively:

$$p_{rightwall} = (2a - \delta)(n - 1) + 2a - 2\delta \quad (138)$$

$$p_{backwall} = \frac{1}{2}\sqrt{3}(2a - \delta)(p - 1) * 2 + \frac{1}{2}\sqrt{3}(2a - \delta) + 2a - 2\delta \quad (139)$$

$$p_{topwall} = \sqrt{\frac{2}{3}}(2a - \delta)(q - 1) * 2 + \sqrt{\frac{2}{3}}(2a - \delta) + 2a - 2\delta . \quad (140)$$

Finally, the number of spheres that the HCP contains must be specified in the Fortran script. This is a bit less straightforward for a packing with wall boundaries, because the number of spheres is different for odd and even rows and for ‘*A*’- and ‘*B*’-layers. In an ‘*A*’-layer, the odd rows have n spheres and the even rows $n - 1$. An odd/even pair has thus $2n - 1$ spheres. There are p of such pairs, so an ‘*A*’-layer contains $(2n - 1) * p$ spheres. An odd/even pair in a ‘*B*’-layer also contains $2n - 1$ spheres, but there are $p - 1$ of such pairs. The last row has an additional number of $n - 1$ spheres, so a ‘*B*’-layer contains $(2n - 1) * (p - 1) + n - 1$ spheres. Summing this up to the number of spheres in an ‘*A*’-layer and multiplying this for a q number of *A/B* layer pairs gives for the total number of spheres in the packing:

$$\begin{aligned}
N_{HCP} &= ((2n - 1) * p + (2n - 1) * (p - 1) + n - 1) * q \\
&= (4np - 2p - n) * q .
\end{aligned}
\tag{141}$$

It must be noted that when the same values for n , p and q are used for periodic and wall boundary simulations, the respective packings have a different number of spheres. This is due to the fact that about half of the rows in an HCP for wall boundary conditions have one sphere less than the other rows, while with periodic boundaries, the packing has the same number of spheres in every row (and layer). The volume fraction of an HCP in a wall boundary simulation depends on the size of the packing. A small packing has a lower volume fraction than a larger packing. The relation between the volume fraction and the size of the packing is analyzed in section 5.2.1.

4.3 The simulations

This section describes the simulations that were performed for both periodic and wall boundary conditions. The aim of the simulations is to assess how well DEM models the behaviour and characteristics of an HCP. Since the packing properties could be derived from theory (see chapter 3), the reliability of the results from DEM simulations can be evaluated. The most important properties are:

- The contact network of a static packing.
- The potential energy density of a static packing.
- The elastic modulus tensor (EMT) of a static packing.
- The EMT obtained from uniaxial compression tests.

However, obtaining the properties of an HCP is not the main reason why the DEM simulations are performed. A more interesting question is to examine how well a small part of the assembly represents the characteristics of the total packing. Especially for packings in simulations with wall boundary conditions it is desired to know if reliable results can be obtained from a small part of the packing. As was explained before, for a small packing, the wall-effects have a significant influence on the properties of the packing. However, it could still be possible to obtain properties of the packing that match predictions from theory and from simulations with periodic boundary conditions, by taking so to speak ‘samples’ from the packing (more on this will be discussed in the next part).

In a DEM-simulation, there are several parameters that can be varied. Going through all possible types of combinations to determine their effect on the results would be a very time consuming if not impossible task, since the

simulations can take between 1 and 15 hours (the time is strongly related to the size of the packing, but also on the available processor speed for the user). The parameters that were not varied are:

- The sphere radius
- The initial overlap between spheres
- The density of the spheres
- The gravitational acceleration; this was set to zero in all simulations
- The background damping factor
- The period of the sine function that describes the wall movement
- The friction constant; the spheres are always frictionless

The adjustment of the above mentioned parameters would certainly give quantitatively different result, but these variations are not of interest for this thesis. This does not mean that the parameters were chosen randomly. For the ratio between the overlap and the sphere radius a small value was taken (the overlap is 10^3 smaller than the sphere radius). Since only small deformations were examined, the dynamics of the packing are more obvious when the overlaps are small as well. The reason why the deformations were kept small is to be sure that the stress-strain relation is in the linear regime and the generalized Hooke's law can be applied. For small deformations, the system behaves elastically, while for large deformations a real material would permanently deform (plastic deformation). In a DEM simulation, large deformations would lead to the breaking of contacts and the subsequent rearrangement of the contact network, which is not desired.

The parameters that were varied in the performed DEM simulations are:

- The size of the packing. This was done in two ways:
 - By extending the packing in only one direction, by adjusting only either n , p or q
 - By extending the packing in all three directions, by adjusting all n , p and q .
- The amplitude of the sine function that describes the wall movement.
- The spring stiffness k .

The influence of the size of the packing on its characteristics is especially of interest for simulations with wall boundary conditions, since the effect of the walls is expected to diminish as the packing size increases.

4.4 Tools for extracting results

4.4.1 Basic tools

During a simulation, after a certain number of time steps information about the system is written into files. Some quantities that describe the packing can directly be obtained from these files, like the contact network of an arbitrary sphere in the packing, while other properties have to be post-processed from the stored information. The most basic information is stored in the so-called ‘*ene*’, ‘*c3d*’ and ‘*fstat*’ files. The *c3d*-files contain information about the positions and velocities of all the spheres in the packing at a certain time. With this information, the simulation can be visualized with the software tool ‘*xballs*’. The *ene*-files contain information about (a.o.) the total potential and kinetic energies of the system and the position of the walls as a time-series. The *fstat* finally contains information about all the contacts.

4.4.2 Advanced tools

For the data-acquisition and averaging of more intricate characteristics of the simulated system, the so-called *TAV*-tool was developed. With this tool, advanced properties can be derived for a complete packing, but also for a specific part of the system. One can for instance divide the packing in 27 ($= 3 \times 3 \times 3$) cells and extract information from only one of these cells. In this way, a cell can be selected that is in the center of the packing, where the spheres are less influenced by wall effects. This is especially an interesting feature for simulations with wall boundary conditions.

An important characteristic of a packing is its volume fraction (or density). When this property must be determined for a specific part of the system, it cannot be provided directly by the basic data files, but it must be computed. The problem is to correctly determine the volume of the part of a sphere that lies in the cell. For a large cell, a reasonable estimate of the volume fraction can be made by completely including the volumes of the spheres whose centers lie in the cell and completely excluding the ones whose centers fall outside. However, for smaller cells, the error made by this method becomes significant. A more sophisticated method is to take the branch vector between two spheres, divide it in a certain number of fractions and determine which fractions are inside or outside the cell. A different contribution to the volume fraction can be assigned to every fraction to account for the non-linear change in volume (non-linear mass distribution) along the branch vector length. Other properties that are provided by the *TAV*-program for the entire system or only one specific cell are, among others, the elastic modulus tensor and the number of spheres and contacts.

Since these properties cannot be determined directly and have to be estimated, a more accurate prediction can be made by estimating these

properties for different small ‘samples’ of the packing and calculating the average of the individual results. This sampling was done by dividing a packing in 27 equal cells by making 2 divisions in all the x -, y - and z -directions, and shifting this cell diagonally through the packing, as is shown in figure 25.

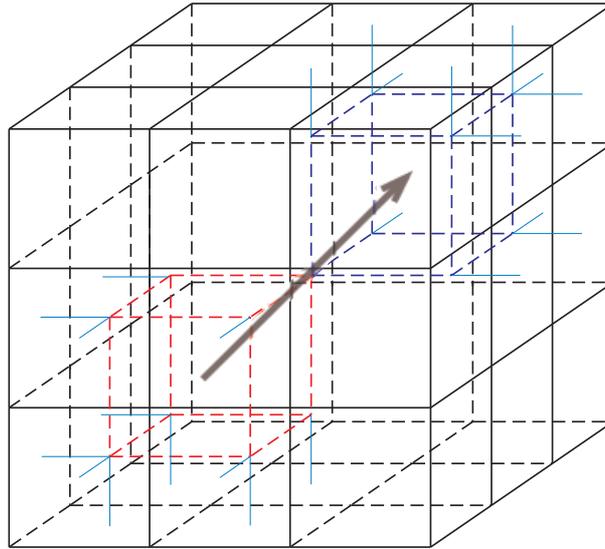


Figure 25: The so called moving average: The average cell is shifted from the bottom-left-front part (the red box) to the top-right-back part (blue box) of the packing in e.g. ten steps. The light blue lines project the corners of the boxes onto the walls of the system.

The reason to make a diagonal shift of the cell and not, for instance, a shift only in the x -direction is because in the last case, only samples from the same layers would be taken. If an error in the estimation of the volume fraction would be caused in the top or bottom area of the cell, it would be repeated for every sample. For a diagonal shift, the composition of the different sample cells is more diversified and there is a higher probability that the individual errors cancel each other (partially) out during averaging. As long as one shift is not exactly in the range of a (multiple of the) sphere radius or layer width, a more reliable approximation can presumably be achieved. The more samples are taken, the more reliable the results become. However, it is also time-consuming, because the positions of the boundaries of the subcell have to be calculated for every step and have to be specified with sufficient accuracy in the TAV-program. It is shown in section 5.2.1 that eleven samples are sufficient to get a reasonable estimate.

5 Results

5.1 Introduction

In this chapter the results from the DEM simulations are presented. In the first part, the properties of the HCP that were theoretically derived in chapter 3 will now be determined from DEM simulations. In the second part, three strain definitions that are valid for 3D granular assemblies will be tested for the HCP, combining the theoretical knowledge with data from simulations.

5.2 Properties of the HCP

5.2.1 The volume fraction (density)

In this paragraph, the density of HCP's in systems with wall and periodic boundary conditions will be determined from DEM simulations.

Packing with wall boundaries

To determine the effect of the size of an HCP on its volume fraction in a system with wall boundary conditions, the packings described in table 6 were created.

Table 6: *Dimensions of the packing*

n	p	q	$\#N$
4	2	2	48
6	3	3	180
8	4	4	448
10	5	5	900
12	6	6	1584
14	7	7	2548
16	8	8	3840
20	10	10	7600
26	13	13	16900
30	15	15	26100
34	17	17	38148

The packing was increased in size by extending in all three dimensions every time, keeping the shape close to cubic. It is clear from the table that the number of spheres ($\#N$) increases rapidly per step. The volume fractions of the packings were obtained from the *ene*-files. Figure 26 shows the volume fraction of the HCP against the number of spheres in the packing.

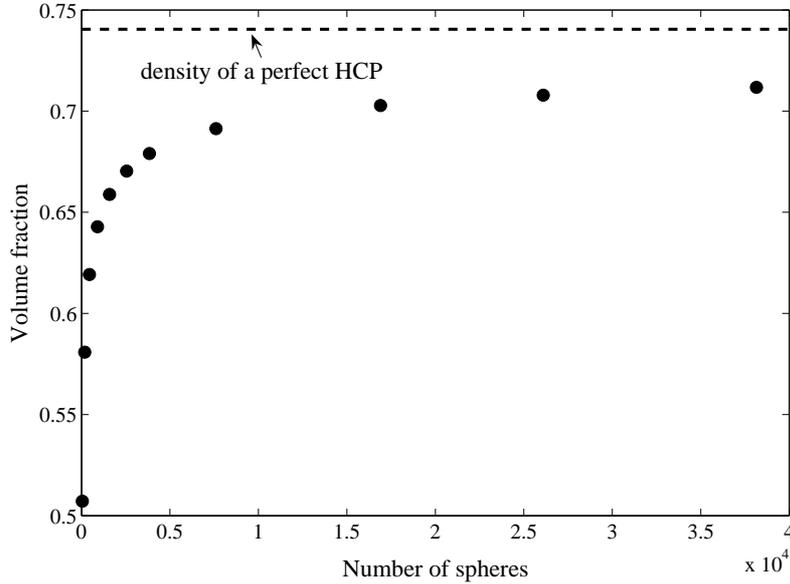


Figure 26: The volume fraction against the number of spheres in the packing.

The plot in figure 26 indicates that the volume fraction will approach a maximum density asymptotically that is significantly lower than the density of a ‘perfect’ HCP. In order to determine this maximum, a relation between the number of spheres and the volume fraction must be found. Figure 27 shows that multiplying the volume fraction with the number of spheres yields a straight line when it is set against the number of spheres.

The function of the trendline presented in figure 27 can be rewritten as a function of the density by dividing both sides by the number of spheres:

$$\eta_{HCP} = \frac{0.712N - 80.112}{N} . \quad (142)$$

From equation (142) follows that when the number of spheres in the packing goes to infinity, the volume fraction η_{HCP} approaches 0.712, which is significantly lower than the volume fraction of a perfect packing ($\eta_{HCP} = 0.7405$). This indicates that the effect of the walls cannot be nullified by making the packing infinitely large. Therefore, it seems that it will always be necessary to divide a packing in a system with wall boundary conditions in smaller cells and derive the packing properties from this cell. However, as a side remark it must be mentioned that such a conclusion may not be drawn by extrapolating the derived trend line, even though the fit seems to be perfect. It could very well be that the density will still increase when the system gets larger than 40,000 spheres, but that cannot be verified by the DEM simulation, because it is not able to handle more spheres.

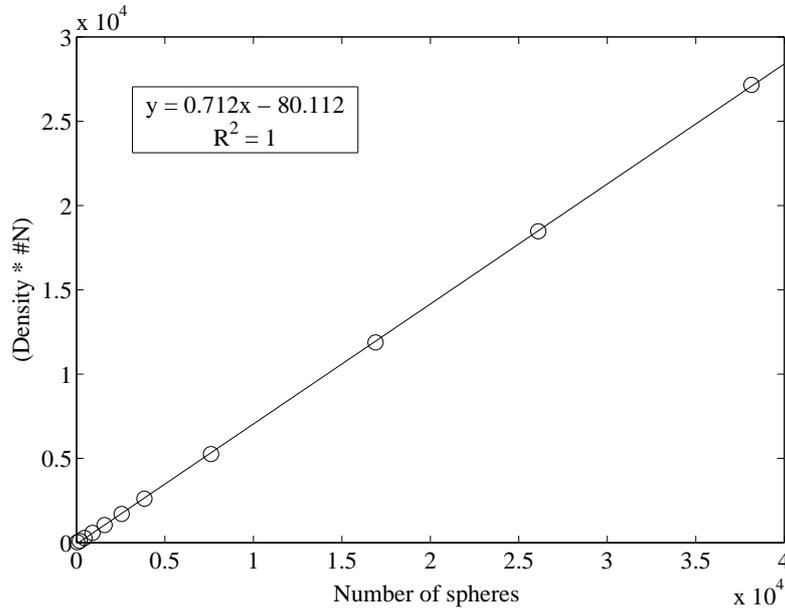


Figure 27: Plotting the packing density times the number of spheres ($\#N$) against the number of spheres yields a straight line. The function of the line is given in the box in the plot area. The R^2 is 1, indicating that the fit is perfect.

Since the method used above does not lead to an answer to the question of what the volume fraction of an infinitely large HCP in a wall boundary system is, a second approach was used to answer this question. In the Fortran code, there are three command lines that calculate the positions of the right, back and top walls of the system (the positions of the other three walls are all zero), see equations (138)-(140). With these equations, the volume of the system can be calculated. Furthermore, there is a command line that calculates the number of spheres in the packing, using equation (141). Knowing that the volume of a single sphere is $\frac{4}{3}\pi a^3$, the total volume of spheres is simply this volume times the total number of spheres in the packing. Dividing this volume by the total volume of the system gives then the volume fraction of the packing. In this way, the size of the packing has virtually no restrictions and therefore the volume fraction of an infinitely large HCP can now be calculated. This results in a volume fraction of $\eta_{HCP} = 0.7415$, which is within 0.15% agreement with the theoretically derived volume fraction.

It can be concluded that a ‘perfect’ HCP could be created in a wall boundary system, but due to the restraints of the current processor technology, the DEM simulation is not yet capable of handling a system that is so large that the wall effects hardly effect the properties of a granular packing.

Moving-average method

The volume fraction in the center of a packing can be estimated using the so-called ‘moving-average’-method (see section 4.4.2). The volume fraction is averaged over several sub-volumes of the packing, to account for the error made in the estimation of the volume fractions of individual sub-volumes. This method is tested on an HCP of 1584 spheres ($n = 12$, $p = 6$, $q = 6$) in a wall boundary system. The overall volume fraction is 0.657, but away from the walls, inside the packing, the density should correspond to the density of a ‘perfect’ HCP, i.e. a volume fraction of 0.7405. Over all three axes, the packing is cut through two times, so 27 ($3 \times 3 \times 3$) equally sized sub-volumes are obtained. The one sub-cell in the middle is selected to determine its volume fraction. This cell is shifted eleven times and for each position, the volume fraction of this cell is retrieved. This test is repeated four times; each time, the center sub-cell is moved in a different direction. These directions are straight in the x -, y - and z -directions and diagonally, as visualized in figure 25. The results are summarized in table 7 below.

Table 7: Volume fraction of the center sub-cell obtained with the moving-average method. The error given in the last row is the deviation of the average volume fraction from that of a ‘perfect’ HCP, i.e. a density of 0.7405.

	Shift direction			
shift no.	<i>x</i> -axis	<i>y</i> -axis	<i>z</i> -axis	diagonally
1	0.745	0.760	0.760	0.758
2	0.745	0.759	0.741	0.724
3	0.724	0.724	0.746	0.728
4	0.745	0.767	0.748	0.737
5	0.745	0.764	0.740	0.728
6	0.761	0.761	0.761	0.761
7	0.745	0.758	0.740	0.722
8	0.745	0.771	0.748	0.744
9	0.745	0.765	0.746	0.733
10	0.745	0.771	0.741	0.736
11	0.745	0.761	0.758	0.742
average	0.747	0.763	0.748	0.738
error	0.83%	3.05%	1.05%	0.39%

These results show that with especially the diagonal shift, a fairly good approximation of the actual volume fraction can be obtained. The deviation here is only 0.39%, which is significantly better than the average obtained for the other shift directions. The column with the results for the shift in the x -direction clearly shows that systematically the same error is made. This is probably due to the fact that the sub-cell is only shifted along the the same

layers and rows, while with the diagonal shift, samples are taken from both different layers and rows, preventing that the same error is systematically made for every shift.

Packing with periodic boundaries

The influence of the size of the packing on its volume fraction was also analyzed for a system with periodic boundaries. Like was done with the wall boundary tests, a number of packings was made with the properties listed in table 8 below.

Table 8: *Dimensions of the packing*

n	p	q	#N
4	2	2	64
6	3	3	216
8	4	4	512
10	5	5	1000
12	6	6	1728
14	7	7	2744
16	8	8	4096

Comparing tables 6 and 8 clearly shows that a packing with in a system with periodic boundaries contains more spheres than a packing created with the same values for the parameters n , p and q in a system with wall boundaries. This already suggests that the volume fraction of the periodic packing will be higher. The volume fractions obtained from the *ene*-files confirm this presumption, as can be seen in figure 28.

The volume fraction of an HCP in a system with periodic boundaries is independent of the size of the packing. Every sphere has the same number of contacts with the same (or similar) orientations. The volume fraction found with the DEM simulation is $\eta_{HCP} = 0.7451$, which is slightly higher than the theoretically derived value of 0.7405 (see chapter 3.5). This can be explained by the fact that in the theoretical derivation, the effect of the overlap on the volumes of the unit cell and the spheres was neglected. In the simulation, the spheres are slightly pressed together, densifying the packing. The error made with neglecting the overlap in the theoretical derivation of the density is 0.62%, which seems to justify the simplification.

5.2.2 The contact network

To verify if a created packing for a DEM simulation is indeed an HCP, it is useful to check the contact network of an arbitrary sphere in the packing. The *fstat*-files contain this information for every sphere in the packing.

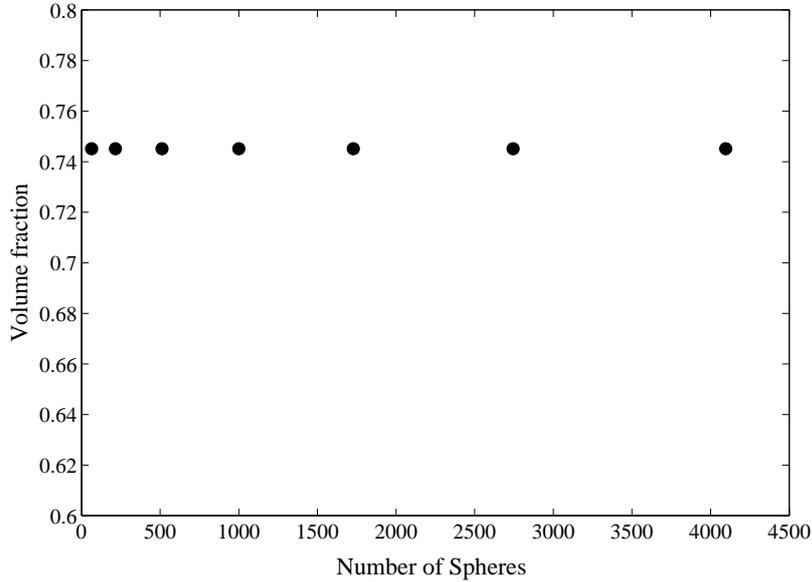


Figure 28: The volume fraction of an HCP in a periodic boundary system is 0.7451 and independent of the size of the packing.

Packing with wall boundaries

An HCP created with the Fortran code for a system with wall boundary conditions is not completely stable under the initial configuration of the sphere positions. The relatively large empty spaces between the spheres near the walls (as was shown in figure 21) cause an uneven force distribution and as a result, the sum over all forces acting on a particle does not equal zero initially. This problem can be solved by allowing the system to ‘relax’; this means that the DEM simulation is run while no external forces are acting on the system, until all the spheres are in equilibrium and the total kinetic energy of the system is zero. In the simulations a linear dashpot constant of $4 * 10^{-2}$ is used to bring the system to rest. When the system is stable, the actual compression tests can be started. Table 9 shows the contact network of a randomly selected sphere (no. 422) in a wall boundary packing consisting of 900 spheres. The normal vectors presented in this table correspond very well to the theoretically derived contact network given in equation (66), so the relaxation hardly affects the contact network.

Although relaxation of the system does not lead to a significant change in the contact network, it does affect the overlap length δ between contacting spheres. The overlap lengths of the twelve contacts of sphere no. 422 are given on the left-hand side of table 10. It clearly shows that the overlaps are much smaller than the initially specified value of 10^{-6} in the Fortran

Table 9: Contact network of sphere no. 422 in a wall boundary packing of 900 spheres after relaxation, as presented in the fstat-file. (for clarity; $\frac{1}{2}\sqrt{3} \approx 0.866$, $\sqrt{\frac{2}{3}} \approx 0.816$, $\frac{1}{2\sqrt{3}} \approx 0.289$, $\frac{1}{\sqrt{3}} \approx 0.577$)

Contact	$\hat{\mathbf{n}}_1$	$\hat{\mathbf{n}}_2$	$\hat{\mathbf{n}}_3$
423 422	1	5.9355783e-07	3.713674833e-07
432 422	0.4999894385	0.866031501371	-4.545007359e-06
431 422	-0.499982059478	0.866035761489	-4.776842608e-06
421 422	-1	-1.864954276e-14	-1.88927976e-13
412 422	-0.499977008765	-0.866038677373	8.436414479e-07
413 422	0.499971940373	-0.866041603411	9.810536115e-07
517 422	0.500021114204	0.28875070633	0.816456927795
516 422	-0.500021786139	0.288749573175	0.816456917037
507 422	-3.639945637e-06	-0.577387117121	0.81647052425
337 422	0.500023529091	0.28873288195	-0.816461752464
336 422	-0.500023247653	0.288731989869	-0.816462240298
327 422	-3.048540530e-06	-0.577393459563	-0.816466039003

code. Moreover, there is much difference in the overlap length from contact to contact, while initially, they were all the same. The average overlap is now $6.32 * 10^{-7}$, which is a decrease of 36.8%. The sphere overlaps after relaxation were also determined for a bigger HCP, containing 3840 spheres. The results are given on the right-hand side of table 10. There is still a large spread from overlap to overlap, but the average has now increased to $7.85 * 10^{-7}$, which is a decrease of 21.5% from the initial overlap length of 10^{-6} before relaxation. This shows that the size of the packing must still be increased significantly to suppress the wall effects.

Packing with periodic boundaries

Also for a 1000 sphere periodic packing a sphere is randomly selected to determine its contact network; in this case it is sphere no. 445. This sphere has contacts with spheres 335, 344, 345 (the layer below sphere 445), 434, 435, 444, 446, 454, 455 (in the same layer) and 535, 544, 545 (the layer above). The entries of each normal unit vector are given in table 11. It shows that the contact network of sphere no. 445 corresponds to the contact in equation (66), which was positioned in a layer arbitrarily labelled ‘A’. A sphere in the layer above sphere no. 445, for instance sphere no. 545, must have the same contact network as the the sphere described in equation (67). The difference between the spheres in the different layers is that the sign of the y -components of the seventh to twelfth normal unit vectors is flipped. Table 12 shows that this is indeed the case.

Table 10: The left table shows the length of the overlaps of the contacts between sphere no. 422 and its neighbours. This packing consisted of 900 spheres. The right table shows the overlaps of the contacts between sphere no. 1466 and its neighbours. This packing consisted of 3840 spheres. For both packings, the overlaps were determined after relaxation and both systems had wall boundary conditions.

Contact	overlap	Contact	overlap
423 422	$6.137 * 10^{-7}$	1467 1466	$7.114 * 10^{-7}$
432 422	$5.196 * 10^{-7}$	1482 1466	$7.515 * 10^{-7}$
431 422	$5.335 * 10^{-7}$	1481 1466	$7.447 * 10^{-7}$
421 422	$6.103 * 10^{-7}$	1465 1466	$7.120 * 10^{-7}$
412 422	$5.119 * 10^{-7}$	1450 1466	$1.036 * 10^{-6}$
413 422	$5.196 * 10^{-7}$	1451 1466	$1.026 * 10^{-6}$
517 422	$7.035 * 10^{-7}$	1714 1466	$8.770 * 10^{-7}$
516 422	$7.035 * 10^{-7}$	1713 1466	$8.785 * 10^{-7}$
507 422	$7.288 * 10^{-7}$	1698 1466	$4.637 * 10^{-7}$
337 422	$7.093 * 10^{-7}$	1234 1466	$8.877 * 10^{-7}$
336 422	$7.098 * 10^{-7}$	1233 1466	$8.823 * 10^{-7}$
327 422	$7.167 * 10^{-7}$	1218 1466	$4.491 * 10^{-7}$

5.2.3 The kinetic and potential energy

Packing with wall boundaries

As explained before, an HCP created for a wall boundary system is initially not stable and a relaxation time is needed to bring the system to rest. Figure 29 shows the decay of the kinetic energy during such a relaxation period. The kinetic energy drops about 20 orders in magnitude to a stable value of the order of 10^{-25} .

Due to the relaxation, the potential energy decreases as well. From figure 30 it follows that the initial total potential energy of the system (containing 900 spheres) is $2.5 * 10^{-4}$ and drops to a stable value of $1.9 * 10^{-4}$, which is a decrease of 23.5%.

Packing with periodic boundaries

An HCP in a periodic boundary system created with the Fortran code should be kinetically stable when external are acting on the system. When running a simulation where the (virtual) walls are not moving and the initial velocity of all the spheres is zero, the total kinetic energy should remain zero as well and the potential energy should remain a constant (non-zero) value. Figure 31 shows the kinetic energy against time of a periodic HCP ($N = 216$, $n = 6$, $p = 3$ and $q = 6$). The kinetic energy is of the order of 10^{-27} . The noise

Table 11: Contact network of sphere no. 445, as presented in the fstat-file. (for clarity; $\frac{1}{2}\sqrt{3} \approx 0.866$, $\sqrt{\frac{2}{3}} \approx 0.816$, $\frac{1}{2\sqrt{3}} \approx 0.289$, $\frac{1}{\sqrt{3}} \approx 0.577$)

Contact pair	$\hat{\mathbf{n}}_1$	$\hat{\mathbf{n}}_2$	$\hat{\mathbf{n}}_3$
446 445	1	-0	-0
455 445	0.500000000001	0.866025403784	-0
454 445	-0.500000000001	0.866025403784	-0
444 445	-1	0	0
434 445	-0.499999999999	-0.866025403785	-0
435 445	0.499999999999	-0.866025403785	-0
545 445	0.500000000001	0.288675134596	0.816496580927
544 445	-0.500000000001	0.288675134596	0.816496580927
535 445	-0	-0.577350269192	0.816496580926
345 445	0.499999999999	0.288675134595	-0.816496580928
344 445	-0.499999999999	0.288675134595	-0.816496580928
335 445	0	-0.577350269189	-0.816496580928

in the plot is caused by the numerical errors that are made by the DEM simulation. Comparing this energy with the potential energy of the static packing (see figure 32), it can be said that the kinetic energy is indeed zero. The potential energy is constant at $6.48 * 10^{-5}$, which is of the order of 10^{22} higher.

The potential energy of the static packing determined from the DEM simulation should correspond with the derived theoretical model in chapter 3.6. The potential energy in figure 32 is an absolute value, so it must first be converted to a potential energy density by dividing it by the volume of the packing. The volume can be derived from the position of the virtual walls given in the *c3d*-files. These positions are:

$$p_{rightwall} = 0.0128$$

$$p_{backwall} = 0.0111$$

$$p_{topwall} = 0.0104$$

Knowing that the positions of the other three walls are zero for their respective orientations, the volume of the system is:

$$V_{HCP} = 0.0128 * 0.0111 * 0.0104 = 1.49 * 10^{-6} . \quad (143)$$

The potential energy density ($u_{density}$) is the absolute potential energy of the system (u_{abs}) divided by V_{HCP} :

Table 12: Contact network of sphere no. 545, as presented in the fstat-file. (for clarity; $\frac{1}{2}\sqrt{3} \approx 0.866$, $\sqrt{\frac{2}{3}} \approx 0.816$, $\frac{1}{2\sqrt{3}} \approx 0.289$, $\frac{1}{\sqrt{3}} \approx 0.577$)

Contact pair	$\hat{\mathbf{n}}_1$	$\hat{\mathbf{n}}_2$	$\hat{\mathbf{n}}_3$
546 545	1	-0	-0
556 545	0.4999999999995	0.866025403787	-0
555 545	-0.4999999999995	0.866025403787	-0
544 545	-1	0	0
535 545	-0.4999999999999	-0.866025403785	-0
536 545	0.4999999999999	-0.866025403785	-0
646 545	0.5000000000005	-0.288675134598	0.816496580924
645 545	-0.5000000000005	-0.288675134598	0.816496580924
655 545	-0	0.577350269193	0.816496580925
446 545	0.5000000000001	-0.288675134596	-0.816496580927
445 545	-0.5000000000001	-0.288675134596	-0.816496580927
455 545	0	0.577350269189	-0.816496580928

$$\begin{aligned}
 u_{density} &= \frac{u_{abs}}{V_{HCP}} \\
 &= \frac{6.48 * 10^{-5}}{1.49 * 10^{-6}} \approx 43.35 .
 \end{aligned}
 \tag{144}$$

In section 3.6 the following expression for the potential energy density for a unit cell of an HCP was derived:

$$u^{cell} = \frac{3\sqrt{2}k\delta^2}{l^3} ,
 \tag{145}$$

where l equals $2a - \delta$. Filling in the same values as were specified for the simulation ($k = 10^5$, $\delta = 10^{-6}$ and $a = 1.07 * 10^{-3}$) gives

$$u^{cell} = \frac{3\sqrt{2} * 10^5 * (10^{-6})^2}{(2 * 1.07 * 10^{-3} - 10^{-6})^3} \approx 43.35 ,
 \tag{146}$$

which is in agreement with the potential energy density found with the simulation.

5.2.4 The elastic modulus tensor

The Elastic Modulus Tensor (EMT) described in section 3.3 can also be obtained from DEM simulations. There are two ways to retrieve this information:

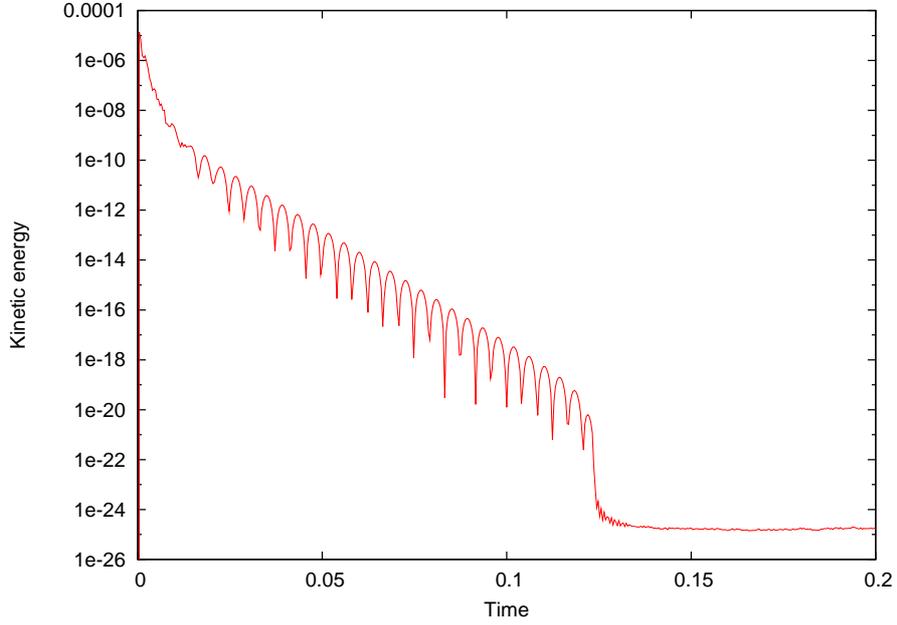


Figure 29: The kinetic energy decreases until the system with wall boundaries is in equilibrium. There are no external forces acting on the system at this point. Number of spheres is 900, linear damping constant is $4 * 10^{-2}$.

1. Using the contact network from a static snapshot of the simulation as a basis, similar to the method used in section 3.3.
2. Determining the slope of the stress against strain graph, obtained from several snapshots.

Packing with wall boundaries

For both methods, the EMT for a packing in a periodic boundary system was obtained from the same simulation and will be described below. The specified conditions for the simulation are summarized in table 13.

The EMT from a static snapshot

The EMT obtained from a static snapshot can be obtained from the *TAV*-file. The provided values are calculated with an expression similar to equation (74):

$$C_{\alpha\beta\gamma\delta} = \frac{N}{V} \left(\frac{l^2}{4} \right) n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c . \quad (147)$$

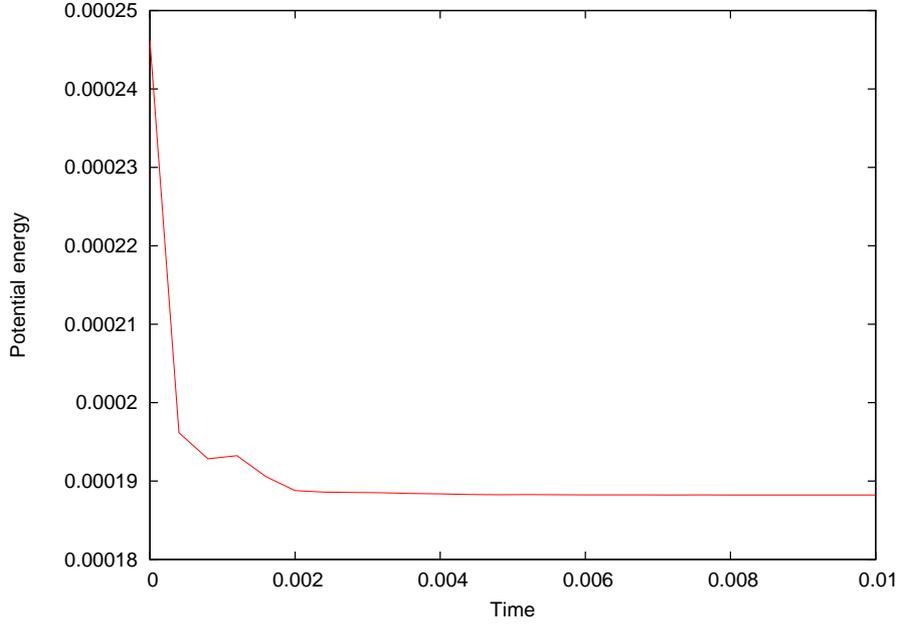


Figure 30: The decrease in the total potential energy of the wall boundary system during relaxation. The drop is about 23.5%. Number of spheres: 900, linear damping constant: $4 * 10^{-2}$.

This expression is only valid for a packing where the EMT is equal for every individual sphere. The summations in equation (74) are replaced by N , the number of spheres in the packing. The stiffness constant k is not included in this expression and the value provided in the *TAV*-file must therefore still be multiplied by k . Because of that, changing the stiffness constant has no effect on the values of the EMT provided by the *TAV*-file. The obtained tensor $C_{\alpha\beta\gamma\delta}^*$ is (* is used to indicate that this tensor is divided by k):

$$C_{\alpha\beta\gamma\delta}^* = \begin{pmatrix} 329.15 & 100.62 & 100.62 & 3.03 & 0 & 0 \\ & 318.26 & 79.70 & -3.23 & 0 & 0 \\ & & 342.23 & -0.40 & 0 & 0 \\ & & & 79.70 & 0 & 0 \\ & & & & 77.16 & 3.03 \\ & & & & & 100.62 \end{pmatrix}. \quad (148)$$

For the conditions specified in table 13, the volume of the packing is $V = 7.21 * 10^{-6}$ and the branch vector l is $2a - \delta = 2 * 1.07 * 10^{-3} - 1 * 10^{-6} = 2.139 * 10^{-3}$. Rewriting equation (147) to

$$n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c = C_{\alpha\beta\gamma\delta}^* \frac{4V}{Nl^2} \quad (149)$$

results in the following matrix:

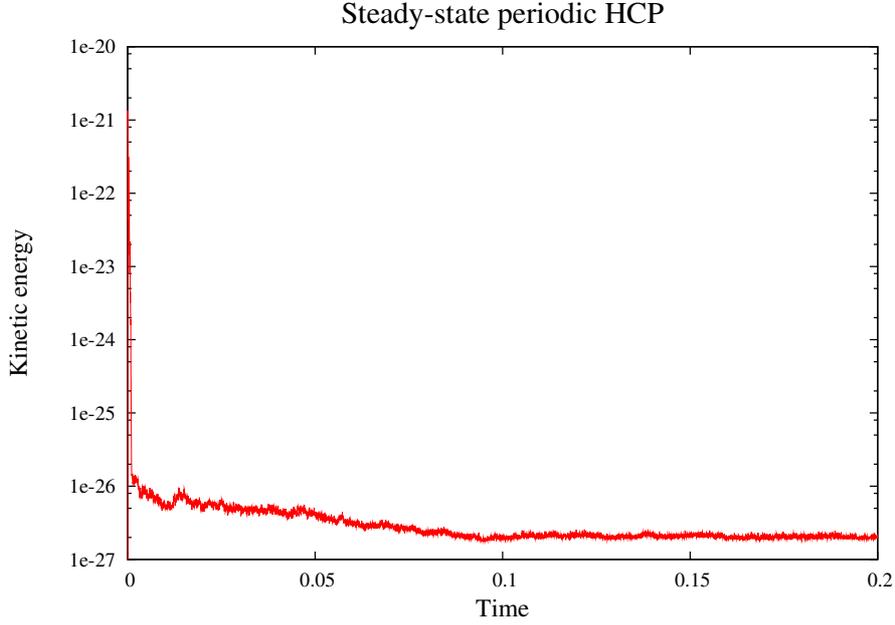


Figure 31: The kinetic energy of the HCP packing as a function of time when there are no external forces acting on the system and the initial velocity of the spheres is zero. The noise is caused by the numerical errors. ($k = 10^5$, $a = 1.07 * 10^{-3}$, $\delta = 10^{-6}$)

$$n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c = \begin{pmatrix} 2.304 & 0.704 & 0.540 & 0.021 & 0 & 0 \\ & 2.228 & 0.558 & -0.023 & 0 & 0 \\ & & 2.396 & -0.003 & 0 & 0 \\ & & & 0.558 & 0 & 0 \\ & & & & 0.540 & 0.021 \\ & & & & & 0.704 \end{pmatrix}. \quad (150)$$

The entries of this matrix deviate significantly from the theoretically derived entries shown in equation (79) in section 3.3. The results can be improved by correcting for the volume fraction of the packing, which is 0.6408. A ‘perfect’ HCP has a density of 0.7405, so the entries in equation (150) are multiplied by $0.7405/0.6408 = 1.155$, which results in:

$$n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c = \begin{pmatrix} 2.662 & 0.814 & 0.624 & 0.025 & 0 & 0 \\ & 2.574 & 0.645 & -0.026 & 0 & 0 \\ & & 2.768 & -0.003 & 0 & 0 \\ & & & 0.645 & 0 & 0 \\ & & & & 0.624 & 0.025 \\ & & & & & 0.814 \end{pmatrix}, \quad (151)$$

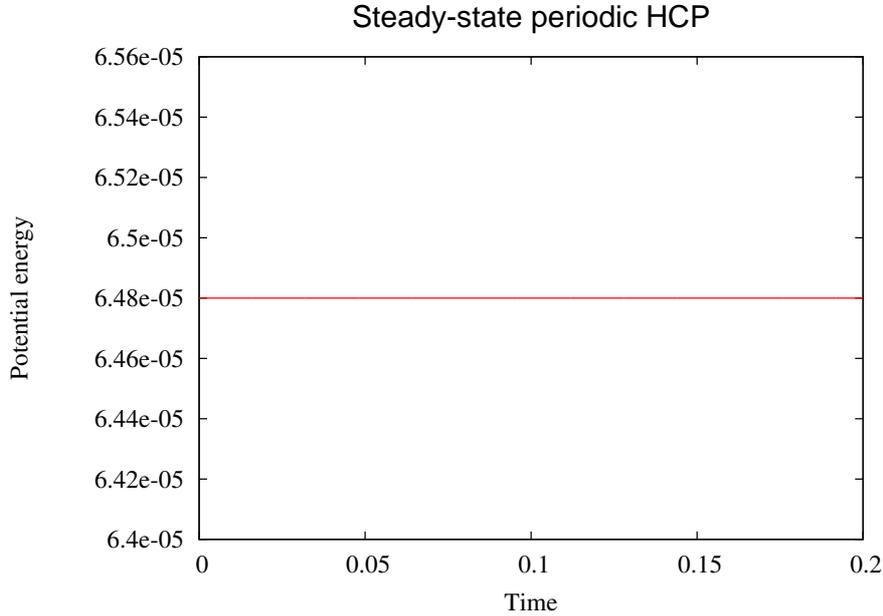


Figure 32: The potential energy of the HCP packing as a function of time when there are no external forces acting on the system. ($k = 10^5$, $a = 1.07 * 10^{-3}$, $\delta = 10^{-6}$)

which is a better approximation of the theoretically derived values in equation (79).

The EMT from a static snapshot obtained from sub-volumes

The EMT from a single snapshot was also determined by looking at smaller sub-volumes of the same packing. Over all three axes, the packing was cut through two or more (up to 8) times, creating equally sized and shaped pieces of the HCP. One of these cells, located in the center of the packing, was selected every time and from this cell, the EMT was determined. Since the spheres in these cells in the center of the packing don't 'feel' any walls, it is an interesting question whether or not this improves the approximation of the EMT of an 'ideal' packing. From the selected sub-volumes, the volume fraction, number of spheres, the volume and the entries of the EMT were determined. Table 14 gives an overview of the obtained results.

From this table it can be concluded that the estimations made of the EMT of an HCP from sub-cells have hardly improved. The density is generally overestimated (the density of a perfect HCP is 0.7405) and n_{3333} is in a number of cases smaller than n_{1111} and n_{2222} , while it should be larger. The maximum deviations between the predicted and theoretically derived values of the single sphere EMT are quite large as well. For cell divisions

Table 13: Specified parameters in the DEM simulation

Parameter	Specified value
Geometrical characteristics	
boundary conditions	wall
n	10
p	5
q	5
number of spheres	900
sphere radius	$1.07 * 10^{-3}$
overlap	10^{-6}
Material characteristics	
normal stiffness	10^5
linear dashpot constant	$4 * 10^{-2}$
sphere density	2000
friction coefficient	0
Other conditions	
gravity	0
amplitude moving wall	10^{-5}
period moving wall	0.1

of 6 per axis and higher, the results can be considered as irrelevant. It is, for example, predicted that number of spheres in a sub-cell obtained with 7 divisions per axis is smaller than for 8 divisions per axis, which is of course not true.

It must be noted that the density correction was not performed here, because in practice, the ‘real’ density in the center of the packing is not known. However, the moving-average method could then be applied to obtain a better estimation of this density.

The EMT from several snapshots

With this method, the entries of the EMT were obtained from three different simulations. In the first one, only the right wall was moved inwards, in the second the back wall and in the third the top wall. With each single simulation, three components of the EMT could be obtained. With the simulation where the right wall was moved, the first three entries in the first column of the EMT were found. The simulation with the compression in the y -direction provided the first three entries of the second column and the simulation with the compression in the z -direction the first three entries of the third column. With the conditions specified in table 13, the obtained EMT is:

Table 14: Properties of the center sub-cell for several numbers of divisions in all the x -, y - and z -directions. The columns n_{iiii} show the first three diagonal elements of the average EMT of one contact (comparable to equation (150); the density correction is not included). The last column shows the maximum deviation between the obtained values of the entries of the EMT (for a single sphere) and the theoretically derived values.

div.	dens.	N	V_{cell}	n_{1111}	n_{2222}	n_{3333}	% dev.
0	0.641	900	$7.21 * 10^{-6}$	2.30	2.23	2.40	19.0%
2	0.819	42.6	$2.67 * 10^{-7}$	2.69	2.60	2.49	7.1%
3	0.756	16.6	$1.13 * 10^{-7}$	2.61	2.88	2.31	19.4%
4	0.767	8.61	$5.77 * 10^{-8}$	2.35	2.44	2.81	-14.2%
5	0.758	4.93	$3.34 * 10^{-8}$	2.52	2.80	2.41	10.9%
6	0.823	3.37	$2.10 * 10^{-8}$	3.11	2.68	2.20	23.5%
7	0.570	1.56	$1.41 * 10^{-8}$	2.13	2.45	2.93	-22.8%
8	1.072	2.06	$9.89 * 10^{-9}$	3.77	2.75	1.82	-50.7%

$$C_{\alpha\beta\gamma\delta} = \begin{pmatrix} 4.457 * 10^7 & 1.048 * 10^7 & 0.857 * 10^7 & - & - & - \\ 1.051 * 10^7 & 3.968 * 10^7 & 0.835 * 10^7 & - & - & - \\ 0.857 * 10^7 & 0.848 * 10^7 & 5.512 * 10^7 & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \end{pmatrix}. \quad (152)$$

These values look quite different from the results obtained from a static snapshot, given in equation(148). This stems from the fact that not the same conversion term is valid to go from the matrix formed by $n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c$ to the EMT $C_{\alpha\beta\gamma\delta}$. In the first place, the stiffness constant k is now included in the prefactor. Moreover, the full branch vector, from one sphere center to the other, is taken instead of half of the branch vector, from the sphere center to the contact point. Due to the square term, the contribution of the branch vector becomes four times higher. With these considerations, there is still a factor two difference, which can be explained by the knowledge that for the values of the entries in equation (150) it was already accounted for the fact that the contributions of all the contacts are counted twice ². So to trace back the matrix $n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c$ from the results given in equation (148), the following expression has to be used:

$$n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c = C_{\alpha\beta\gamma\delta} \frac{2V}{Nl^2}. \quad (153)$$

²One sphere has twelve contacts, but each contact is shared with two spheres. Dividing the total number of spheres in a (periodic boundary) packing by the total number of spheres yields an average of six contacts per sphere.

However, since the wall effects make the system highly non-ideal, the result do not resemble the EMT obtained from a static snapshot:

$$n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c = \begin{pmatrix} 1.555 & 0.366 & 0.299 & - & - & - \\ 0.367 & 1.385 & 0.291 & - & - & - \\ 0.299 & 0.296 & 1.923 & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \end{pmatrix}. \quad (154)$$

The wall effects can be reduced by increasing the size of the packing. Therefore, the EMT determined from several snapshots was also determined for an HCP consisting of 3840 spheres. This resulted in the following matrix:

$$n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c = \begin{pmatrix} 1.792 & 0.486 & 0.385 & - & - & - \\ 0.486 & 1.639 & 0.374 & - & - & - \\ 0.384 & 0.375 & 2.139 & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \end{pmatrix}. \quad (155)$$

This is an improvement with respect to the results of the previous 900 sphere packing, but still the deviations from the theoretically derived values are very large. It is expected that it is possible to obtain sufficiently accurate predictions for the EMT of a system with wall boundary conditions by this method, but then the system should contain of the order of 50,000 spheres or more. However, such a large system cannot be handled yet by the DEM simulation. Therefore, obtaining the EMT for a wall boundary system from several snapshots of DEM simulations is not a suitable method, because the results are not reliable.

Packing with periodic boundaries

For both methods, the EMT for a packing in a periodic boundary system was obtained from the same simulation and will be described below. The specified conditions for the simulation are summarized in table 15.

The EMT from a static snapshot

With the periodic packing and system conditions described in table 15 the obtained EMT $C_{\alpha\beta\gamma\delta}^*$ from a static snapshot is (*' is used to indicate that this tensor is divided by k):

Table 15: Specified parameters in the DEM simulation

Parameter	Specified value
Geometrical characteristics	
boundary conditions	periodic
n	10
p	5
q	5
number of spheres	1000
sphere radius	$1.07 * 10^{-3}$
overlap	10^{-6}
Material characteristics	
normal stiffness	10^5
linear dashpot constant	$4 * 10^{-2}$
sphere density	2000
friction coefficient	0
Other conditions	
gravity	0
amplitude moving wall	10^{-5}
period moving wall	0.1

$$C_{\alpha\beta\gamma\delta}^* = \begin{pmatrix} 411.15 & 136.25 & 109.76 & 0 & 0 & 0 \\ & 409.06 & 109.86 & 0 & 0 & 0 \\ & & 439.45 & 0 & 0 & 0 \\ & & & 109.86 & 0 & 0 \\ & & & & 109.76 & 0 \\ & & & & & 136.25 \end{pmatrix}. \quad (156)$$

The volume of this specific packing is $V = 6.942 * 10^{-6}$ and the branch vector l is $2a - \delta = 2 * 1.07 * 10^{-3} - 1 * 10^{-6} = 2.139 * 10^{-3}$. Again, after rewriting equation (147) to equation (149), the following matrix is obtained:

$$n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c = C_{\alpha\beta\gamma\delta}^* \frac{4V}{Nl^2}, \quad (157)$$

which results in the following matrix:

$$n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c = \begin{pmatrix} 2.495 & 0.827 & 0.666 & 0 & 0 & 0 \\ & 2.483 & 0.667 & 0 & 0 & 0 \\ & & 2.667 & 0 & 0 & 0 \\ & & & 0.6667 & 0 & 0 \\ & & & & 0.666 & 0 \\ & & & & & 0.827 \end{pmatrix}. \quad (158)$$

The entries of this matrix correspond quite well with the theoretically derived entries given in the matrix in equation (79) in section 3.3. The maximum deviation between the entries is 0.77%, which is acceptable.

The EMT from several snapshots

To determine the EMT from several snapshots for a periodic boundary packing, the HCP and system conditions described in table 15 were used again. Similar to the approach used with the wall boundary system, three compression tests were performed; one for every x -, y and z -direction. The obtained entries are shown in the following matrix:

$$C_{\alpha\beta\gamma\delta} = \begin{pmatrix} 7.986 * 10^7 & 3.041 * 10^7 & 2.213 * 10^7 & - & - & - \\ 3.040 * 10^7 & 7.982 * 10^7 & 2.213 * 10^7 & - & - & - \\ 2.212 * 10^7 & 2.211 * 10^7 & 8.811 * 10^7 & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \end{pmatrix}. \quad (159)$$

Also here, equation (153) can be used to determine the EMT for a single sphere. This results in the following matrix:

$$n_{\alpha}^c n_{\beta}^c n_{\gamma}^c n_{\delta}^c = \begin{pmatrix} 2.4157 & 0.9199 & 0.6695 & - & - & - \\ 0.9197 & 2.4147 & 0.6695 & - & - & - \\ 0.6691 & 0.6689 & 2.6654 & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \end{pmatrix}. \quad (160)$$

Comparing this results with the theoretically derived entries given in the matrix in equation (79) in section 3.3 shows that these results are acceptable, as opposed to the EMT obtained from several snapshots of the DEM simulations with wall boundary systems:

$$\begin{pmatrix} 3.37\% & -10.38\% & -0.42\% & - & - & - \\ -10.37\% & 3.41\% & -0.42\% & - & - & - \\ -0.36\% & -0.34\% & 0.05\% & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \end{pmatrix}. \quad (161)$$

The first two diagonal entries deviate around 3.4%, which is not really good but still acceptable. The deviations of the entries in the third column and third row are much smaller, but for the C_{1122} and C_{2211} components it is

too high. Deriving the components of the EMT by constructing the stress against strain graph from several snapshots is therefore only useful if a rough approximation is sufficient.

5.3 The strain tensor for an HCP

In this section, some strain definitions that were described in chapter 2 will be tested on an HCP by combining theoretically derived properties with results from DEM simulations. As was explained in chapter 2, three of the discussed strain definitions were claimed to be valid for 3D granular assemblies, namely:

1. Bagi's strain definition, based on an equivalent continuum (see section 2.2.1).
2. The best-fit strain based on particle translations (see section 2.3.1).
3. Satake's strain definition (see section 2.4).

With the obtained results it can be evaluated which of the definitions is the most accurate and reliable under the conditions used in the DEM simulations.

5.3.1 Bagi's strain definition

In short, Bagi's approach to define a strain tensor for a granular assembly consists of the following steps:

- Construction of a material cell system, using an algorithm that is strongly related to the Voronoi and Dirichlet tessellation methods.
- Construction of a space cell system, using the Delaunay triangulation method.
- Defining the complementary area vectors for all the edges of the space cell system.
- Determining the relative displacement of neighbouring nodes.
- Calculating the strain tensor from the results of the previous two steps.

These steps will be worked out in detail for an HCP, using knowledge from both theory and simulations.

Construction of the material cell system

Bagi generalized the Voronoi and the Dirichlet tessellation methods to be able to define cells in systems with grains of arbitrary shape. However, the assembly discussed here consists of equi-sized spheres and therefore all three methods lead to the same system of cells. Moreover, since an HCP is

a regularly structured packing, all the material cells have the same shape; only their orientations are different from layer to layer, as will be shown later.

The faces of the material cells are collections of points in the system where the distance between the two closest spheres is equal. Since an HCP has only ‘real’ contacts (thus no virtual contacts), the contact point between two spheres lies in the plane of a face of the material cell. The edges of the cells are collections of points where the distance between the three closest spheres is equal. Finally, the corners of the cells are points in the system where the distance between the four (sometimes even more) closest spheres is equal. It is this last property that will be used to construct the material cells, using a schematic representation of an arbitrary sphere in an ‘A’-layer with its twelve contacts (see figure 33).

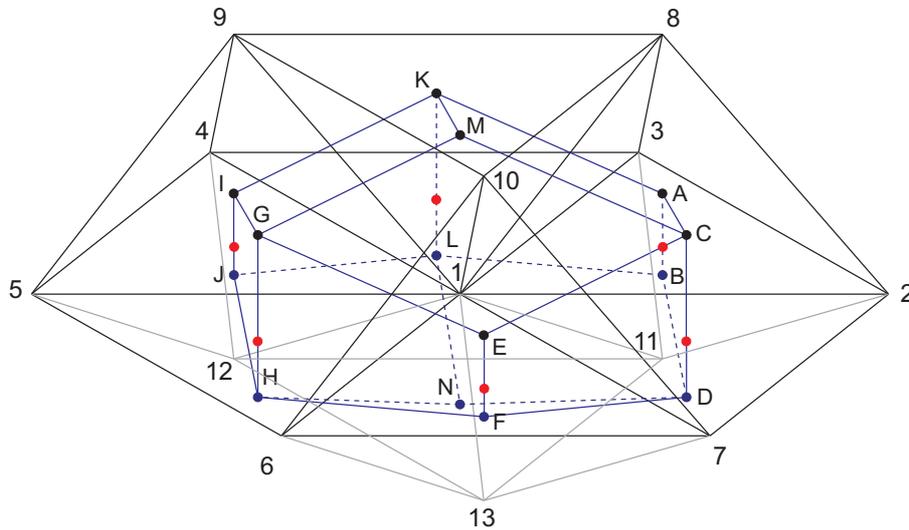


Figure 33: The material cell for an arbitrary sphere in an ‘A’-layer. The black dots indicate the positions of the corners of the cell above the plane of the layer, the red dots are the positions where edges of the cell intersect the plane and the blue dots indicate the positions of the cell corners below the plane. The sphere centers are numbered from 1 to 13 and the corners of the material cell are labelled from **A** to **N**.

Corner **A** in figure 33 lies in the center of the tetrahedron formed by spheres 1, 2, 3 and 8. The corners **B**, **E**, **F**, **I**, **J**, **M** and **N** can also be found by determining the centers of tetrahedrons. Corner **K** is the center of the plane formed by spheres 3, 4, 8 and 9. The corners **C**, **D**, **G**, **H** and **L** are also the centers of similar planes. The red dots indicate the positions where the edges of the material cell intersect the horizontal plane formed by the spheres in the same layer as center sphere 1. The red dot between corners **A** and **B** for example is the center of the triangle formed by spheres 1, 2

and 3.

The material cell for an arbitrary sphere in a 'B'-layer has the same shape as the material cell of an 'A'-layer sphere, only its orientation is different; it is rotated 180° around the z -axis. The two types of material cells are shown in figure 34 below.

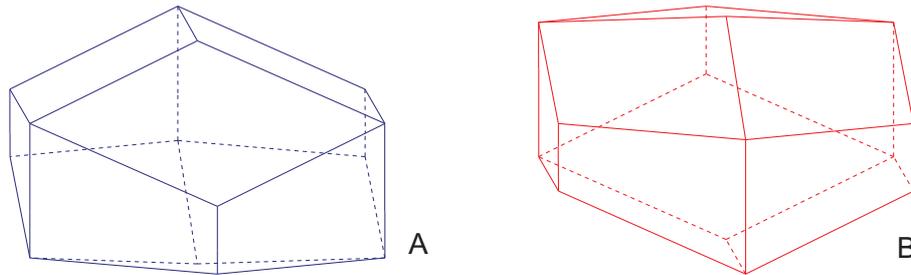


Figure 34: The left picture shows the material cell for a sphere in an 'A'-layer and the right picture for a sphere in a 'B'-layer.

Linking the material cells together will completely fill the space in an HCP system. It shows that the material cells of two contacting spheres share a common face. The complete material cell system for the thirteen spheres of figure 33 is shown in figure 35. To improve the visibility, the layers are separated from each other and black dotted lines are added to indicate some positions where corners of cells fall together.

Construction of the space cell system

The constructed material cell system forms the basis for the construction of the space cell system. When two material cells share a common face, the centers of the corresponding spheres are connected with a straight line. This simply gives back the contact network of the HCP. However, not all the formed space cells are tetrahedrons yet. For the coming steps, it is required that the space is completely triangularized. This can be done in the following way: Imagine that for instance sphere 9 in figure 33 would be slightly shifted to the left and sphere 3 to the right, then the shape of the respective material cells, and the cells around it, would also slightly change. This may result in a common face of the cells of spheres 4 and 8. The sphere centers can then be connected by a straight line, even though the spheres are not in contact. In this way, the system can completely be triangularized. The result is shown in figure 36.

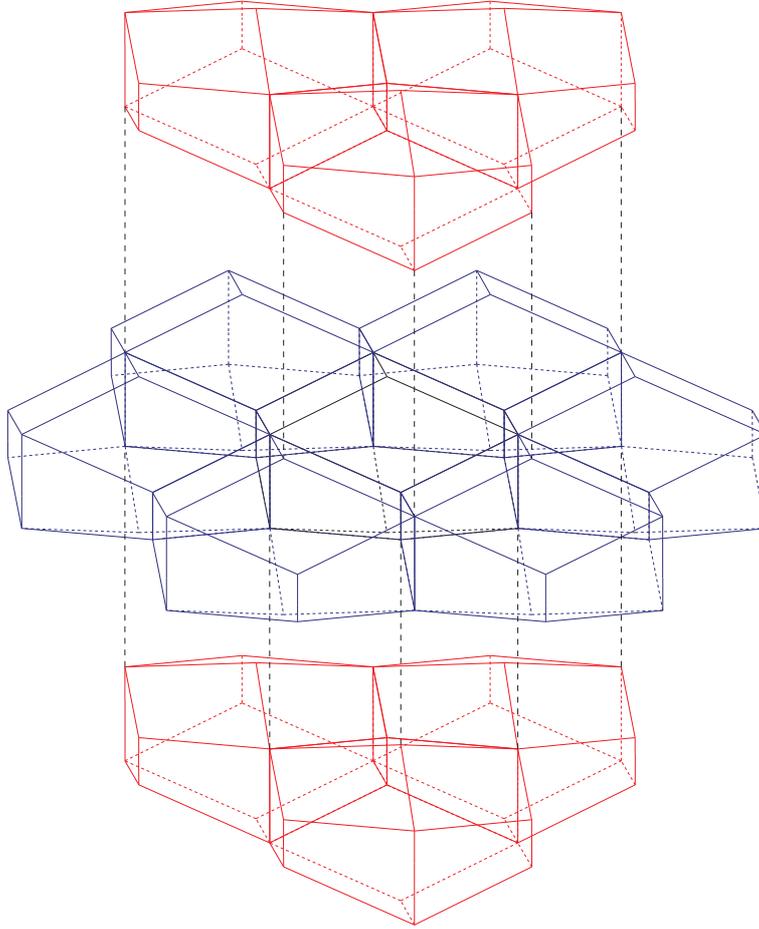


Figure 35: The material cell system formed by thirteen spheres. The black cell in the center corresponds to sphere one in figure 33. To improve the visibility, the layers are separated from each other and black dotted lines are added to indicate some positions where corners of cells fall together.

The complementary area vector

For every edge of the space cell system, the complementary area vector given with equation (20) must be determined. The calculations needed to obtain this vector will be worked out in detail for one edge. For the remaining edges, only the results will be given. The selected edge is the edge formed by the line connecting the centers of spheres 1 and 8 in figure 36; the corresponding complementary area vector is d_i^{1-8} . The following steps have to be performed in order to calculate this vector:

- Find all the space cells that contain the 1-8 edge and label them $t_1, t_2 \dots t_T$, where T is the number of cells that contain this edge.

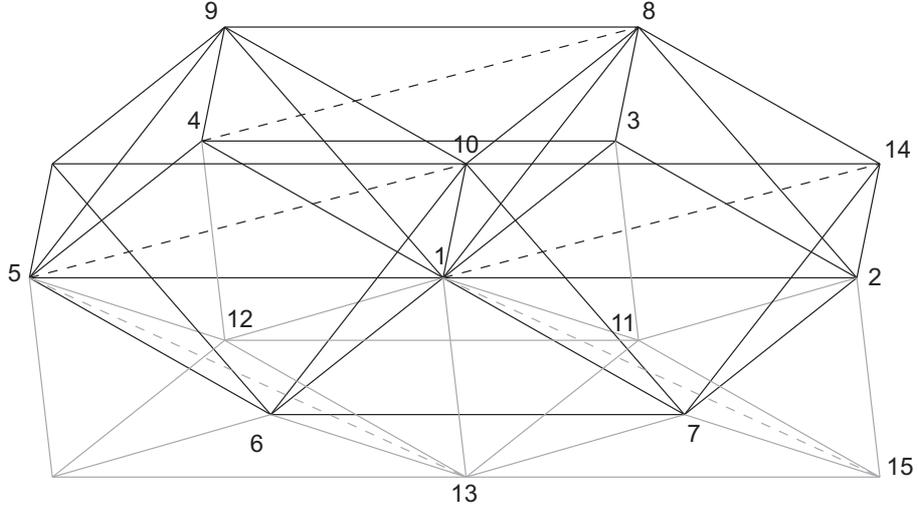


Figure 36: The space cell system for an HCP. The solid lines correspond to ‘real’ contacts and the dashed lines correspond to ‘virtual’ contacts obtained by slightly shifting some spheres.

- For every face of every cell t , assign a vector $b_i^{k(t)}$, where k is the number of the node that is not contained in the respective face. The direction of the vector is perpendicular to the face, pointing outwards and its magnitude equals the area of the face.
- From vector $b_i^{k(t)}$, vector $a_i^{k(t)}$ is calculated, using equation (16).
- Calculate the complementary vector for edge 1-8 with equation (20).

For the 1-8 edge, six space cells can be found that contain this edge. These cells are given in table 16 below.

Table 16: The space cells that contain the 1-8 edge. The numbering used in figure 36 is adopted here.

Cell number t	Corresponding spheres
1	1-2-3-8
2	1-8-9-10
3	1-2-8-14
4	1-8-10-14
5	1-4-8-9
6	1-3-4-8

The next step is the assignment of the $b_i^{k(t)}$ vectors to each face of each cell. Actually, to determine the complementary area vector for the 1-8 edge,

only the vectors $b_i^{1(t)}$ and $b_i^{8(t)}$ are needed. However, the other two vectors of each cell will also be calculated, because they will be needed anyway when the complementary area vectors for the other edges are calculated. Again, one cell will be taken as an example to calculate the $b_i^{k(t)}$ vectors; in this case cell 4 of table 16, formed by lines connecting the centers of spheres 1, 8, 10 and 14 of figure 36. This cell with the corresponding vectors is shown in figure 37.

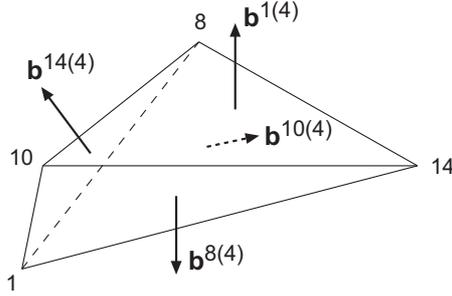


Figure 37: Space cell no. 4 with the corresponding $b_i^{k(4)}$ vectors.

The direction of vector $b_i^{1(4)}$ can directly be found, using the knowledge that spheres 8, 10 and 14 have the same z -coordinate. The vector perpendicular to this plane points in the positive z -direction, since sphere 1 lies below the plane. The unit vector corresponding to $b_i^{1(4)}$ is thus $[0, 0, 1]$. Since the face is an equilateral triangle with sides that have the branch length l , its area $A^{8-10-14}$ equals $\frac{1}{4}\sqrt{3}l^2$. Vector $b_i^{1(4)}$ can therefore be written as:

$$b_i^{1(4)} = \frac{1}{4}\sqrt{3}l^2 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \quad (162)$$

The other three $b_i^{k(4)}$ vectors require a bit more work to derive. In general, when two vectors are perpendicular, the inner product between these two vectors equals zero. So if a vector is perpendicular to a plane, then this vector is perpendicular to every vector that lies in this plane. Since two non-parallel vectors define a plane, the direction of these two vectors provides enough information to construct a vector that is perpendicular to this plane. Vector $b_i^{8(4)}$ must be perpendicular to the 1-10-14 plane, thus also to all the vectors that lie in this plane. Let's take the vector that connects the centers of spheres 1 and 14, and the vector that connects the centers of spheres 10 and 14 to construct $b_i^{8(4)}$. The (relative) position of sphere 1 is $[0, 0, 0]$, of sphere 10 is $[0, -\frac{1}{\sqrt{3}}, \sqrt{\frac{2}{3}}]$ and the (relative) position of sphere 14 is $[1, -\frac{1}{\sqrt{3}}, \sqrt{\frac{2}{3}}]$. The vector connecting spheres 1 and 14 (l_i^{14-1}) is thus:

$$l_i^{14-1} = \begin{pmatrix} 1 \\ -\frac{1}{\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ -\frac{1}{\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} \quad (163)$$

and the vector l_i^{14-10} is:

$$l_i^{14-10} = \begin{pmatrix} 1 \\ -\frac{1}{\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} - \begin{pmatrix} 0 \\ -\frac{1}{\sqrt{3}} \\ \sqrt{\frac{2}{3}} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \quad (164)$$

The inner product between vectors $b_i^{8(4)}$ and l_i^{14-1} and between $b_i^{8(4)}$ and l_i^{14-10} must be zero. This can be written in the following way:

$$\begin{pmatrix} l_1^{14-1} & l_2^{14-1} & l_3^{14-1} \\ l_1^{14-10} & l_2^{14-10} & l_3^{14-10} \end{pmatrix} \cdot \begin{pmatrix} b_1^{8(4)} \\ b_2^{8(4)} \\ b_3^{8(4)} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \quad (165)$$

Filling in the known values for the branch vectors in an augmented matrix and row-sweeping it to an echelon matrix gives a solution for the $b_i^{8(4)}$ vector, where one entry is free to chose:

$$\begin{aligned} \left(\begin{array}{ccc|c} 1 & -\frac{1}{\sqrt{3}} & \sqrt{\frac{2}{3}} & 0 \\ 1 & 0 & 0 & 0 \end{array} \right) &\rightarrow \left(\begin{array}{ccc|c} 1 & -\frac{1}{\sqrt{3}} & \sqrt{\frac{2}{3}} & 0 \\ 0 & \frac{1}{\sqrt{3}} & -\sqrt{\frac{2}{3}} & 0 \end{array} \right) \\ &\rightarrow \left(\begin{array}{ccc|c} 1 & -\frac{1}{\sqrt{3}} & \sqrt{\frac{2}{3}} & 0 \\ 0 & 1 & -\sqrt{2} & 0 \end{array} \right). \end{aligned}$$

The last matrix tells that

$$b_1^{8(4)} - \frac{1}{\sqrt{3}}b_2^{8(4)} + \sqrt{\frac{2}{3}}b_3^{8(4)} = 0$$

and

$$b_2^{8(4)} - \sqrt{2}b_3^{8(4)} = 0.$$

If $b_3^{8(4)}$ is set to 1, then it follows from the above two equalities that $b_2^{8(4)} = \sqrt{2}$ and $b_1^{8(4)} = 0$, so for now, $b_i^{8(4)} = [0, \sqrt{2}, 1]$. The third component, representing the z -direction is now positive, pointing upwards. However, looking at figure 37 reveals that it must point downwards, so its direction must be reversed. Moreover, the vector must be normalized and then be multiplied by the area of the corresponding plane. The magnitude of the vector is currently $\sqrt{(\sqrt{2})^2 + 1^2} = \sqrt{3}$, so the vector must be divided by a

factor $\sqrt{3}$. The area of the plane is $A^{1-10-14} = \frac{1}{2}l^2$. Vector $b_i^{8(4)}$ therefore becomes:

$$b_i^{8(4)} = \frac{1}{2}l^2 \begin{pmatrix} 0 \\ -\sqrt{\frac{2}{3}} \\ -\frac{1}{\sqrt{3}} \end{pmatrix}. \quad (166)$$

Using the same method as described above, the vectors $b_i^{10(4)}$ and $b_i^{14(4)}$ can be calculated; this results in:

$$b_i^{10(4)} = \frac{1}{2}l^2 \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{6}} \\ -\frac{1}{\sqrt{3}} \end{pmatrix} \quad (167)$$

and

$$b_i^{14(4)} = \frac{1}{4}\sqrt{3}l^2 \begin{pmatrix} -\sqrt{\frac{2}{3}} \\ \frac{1}{3}\sqrt{2} \\ \frac{1}{3} \end{pmatrix}. \quad (168)$$

It can be verified that the sum of the above four $b_i^{k(4)}$ vectors yields zero.

Next, the $a_i^{k(4)}$ vectors for the same cell must be determined, using equation (16):

$$a_i^{k(4)} = -\frac{1}{D}b_i^{k(4)},$$

where D is the dimension of the system, in this case 3D. This simply gives the following four vectors:

$$a_i^{1(4)} = -\frac{1}{4\sqrt{3}}l^2 \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad a_i^{8(4)} = -\frac{1}{6}l^2 \begin{pmatrix} 0 \\ -\sqrt{\frac{2}{3}} \\ -\frac{1}{\sqrt{3}} \end{pmatrix}$$

$$a_i^{10(4)} = -\frac{1}{6}l^2 \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{6}} \\ -\frac{1}{\sqrt{3}} \end{pmatrix} \quad a_i^{14(4)} = -\frac{1}{4\sqrt{3}}l^2 \begin{pmatrix} -\sqrt{\frac{2}{3}} \\ \frac{1}{3}\sqrt{2} \\ \frac{1}{3} \end{pmatrix}.$$

To calculate the complementary area vector for the 1-8 edge, only the vectors $a_i^{1(t)}$ and $a_i^{8(t)}$ from every space cell that contain this edge are needed. The above performed calculations must also be performed for the other five space cells mentioned in table 16 to obtain the respective $a_i^{1(t)}$ and $a_i^{8(t)}$ vectors. The results are summarized in table 17 below.

Table 17: The $a_i^{k(t)}$ vectors corresponding to the 1-8 edge. the superscript k refers to the node of the space cell t that is not included in the face where the vector is perpendicular to.

cell number	vector	prefactor	$a_1^{k(t)}$	$a_2^{k(t)}$	$a_3^{k(t)}$
1	$a_i^{1(1)}$	$-\frac{1}{4\sqrt{3}}l^2$	$\sqrt{\frac{2}{3}}$	$\frac{1}{3}\sqrt{2}$	$\frac{1}{3}$
	$a_i^{8(1)}$	$-\frac{1}{4\sqrt{3}}l^2$	0	0	-1
2	$a_i^{1(2)}$	$-\frac{1}{4\sqrt{3}}l^2$	0	0	1
	$a_i^{8(2)}$	$-\frac{1}{4\sqrt{3}}l^2$	$-\sqrt{\frac{2}{3}}$	$-\frac{1}{3}\sqrt{2}$	$-\frac{1}{3}$
3	$a_i^{1(3)}$	$-\frac{1}{4\sqrt{3}}l^2$	$\sqrt{\frac{2}{3}}$	$\frac{1}{3}\sqrt{2}$	$\frac{1}{3}$
	$a_i^{8(3)}$	$-\frac{1}{6}l^2$	0	$-\sqrt{\frac{2}{3}}$	$-\frac{1}{\sqrt{3}}$
4	$a_i^{1(4)}$	$-\frac{1}{4\sqrt{3}}l^2$	0	0	1
	$a_i^{8(4)}$	$-\frac{1}{6}l^2$	0	$-\sqrt{\frac{2}{3}}$	$-\frac{1}{\sqrt{3}}$
5	$a_i^{1(5)}$	$-\frac{1}{6}l^2$	0	$\sqrt{\frac{2}{3}}$	$\frac{1}{\sqrt{3}}$
	$a_i^{8(5)}$	$-\frac{1}{4\sqrt{3}}l^2$	$-\sqrt{\frac{2}{3}}$	$-\frac{1}{3}\sqrt{2}$	$-\frac{1}{3}$
6	$a_i^{1(6)}$	$-\frac{1}{6}l^2$	0	$\sqrt{\frac{2}{3}}$	$\frac{1}{\sqrt{3}}$
	$a_i^{8(6)}$	$-\frac{1}{4\sqrt{3}}l^2$	0	0	-1

Finally, equation (20) can be applied to calculate the complementary area vector for the 1-8 edge:

$$d_i^{18} = \frac{1}{3+1} \sum_{t=1}^6 (a_i^{1(t)} - a_i^{8(t)}) . \quad (169)$$

With the values specified in table 17, the solution is:

$$d_i^{18} = \frac{1}{4}l^2 \begin{pmatrix} -\frac{1}{3}\sqrt{2} \\ -\sqrt{\frac{2}{3}} \\ -\frac{2}{\sqrt{3}} \end{pmatrix} \quad (170)$$

The complementary area vector must be calculated for every edge in the space cell system. However, an HCP has a regular structure and therefore the space cells are regular as well, limiting the number of differently oriented edges. Looking at the space cell system given in figure 36 shows that the 6-10 edge has the same orientation and the same types of space cells (also with the same relative positions) that contain this edge as the 1-8 edge. It can easily be verified that the complementary area vectors corresponding to the two edges are identical. In total, the space cell system of an HCP contains

fourteen different types of edges, so for every type, the complementary area vector must be calculated one time. The results are presented in table 18 below.

Table 18: The complementary area vectors for an HCP. The left side shows the exact values for the entries and the right side gives the numerical values. The values here are all divided by l^2 , so they are independent of the size of the spheres.

vector	exact value			numerical value		
	d_1^{mn}	d_2^{mn}	d_3^{mn}	d_1^{mn}	d_2^{mn}	d_3^{mn}
d_j^{1-2}	$-\frac{1}{2\sqrt{2}}$	$-\frac{1}{6\sqrt{6}}$	0	-0.35355	-0.06804	0
d_j^{1-3}	$-\frac{1}{6\sqrt{2}}$	$-\frac{1}{2\sqrt{6}}$	0	-0.11785	-0.20412	0
d_j^{1-4}	$\frac{1}{6\sqrt{2}}$	$-\frac{5}{6\sqrt{6}}$	0	0.11785	-0.34021	0
d_j^{1-8}	$-\frac{1}{6\sqrt{2}}$	$-\frac{1}{2\sqrt{6}}$	$-\frac{1}{2\sqrt{3}}$	-0.11785	-0.20412	-0.28868
d_j^{1-9}	$\frac{1}{6\sqrt{2}}$	$-\frac{1}{6\sqrt{6}}$	$-\frac{1}{3\sqrt{3}}$	0.11785	-0.06804	-0.19245
d_j^{1-10}	$\frac{1}{6\sqrt{2}}$	$\frac{1}{2\sqrt{6}}$	$-\frac{1}{2\sqrt{3}}$	0.11785	0.20412	-0.28868
d_j^{1-14}	$-\frac{1}{6\sqrt{2}}$	$\frac{1}{6\sqrt{6}}$	$-\frac{1}{6\sqrt{3}}$	-0.11785	0.06804	-0.09623
d_j^{10-14}	$-\frac{1}{2\sqrt{2}}$	$-\frac{1}{6\sqrt{6}}$	0	-0.35355	-0.06804	0
d_j^{8-10}	$\frac{1}{6\sqrt{2}}$	$\frac{1}{2\sqrt{6}}$	0	0.11785	0.20412	0
d_j^{9-10}	$-\frac{1}{8\sqrt{2}}$	$\frac{17}{24\sqrt{6}}$	$-\frac{1}{16\sqrt{3}}$	-0.08839	0.28918	-0.03608
d_j^{1-11}	$-\frac{1}{6\sqrt{2}}$	$-\frac{1}{2\sqrt{6}}$	$\frac{1}{2\sqrt{3}}$	-0.11785	-0.20412	0.28868
d_j^{1-12}	$\frac{1}{6\sqrt{2}}$	$-\frac{1}{6\sqrt{6}}$	$\frac{1}{3\sqrt{3}}$	0.11785	-0.06804	0.19245
d_j^{1-13}	$\frac{1}{6\sqrt{2}}$	$\frac{1}{2\sqrt{6}}$	$\frac{1}{2\sqrt{3}}$	0.11785	0.20412	0.28868
d_j^{1-15}	$-\frac{1}{6\sqrt{2}}$	$\frac{1}{6\sqrt{6}}$	$\frac{1}{6\sqrt{3}}$	-0.11785	0.06804	0.09623

The relative displacement of neighbouring nodes

A DEM simulation with the conditions specified in table 15 was run for a duration of $t = 0.2$. Only the right wall was moved inwards over a distance 10^{-5} (in the negative x -direction). The initial and final positions of selected spheres were extracted from the simulation, so the displacement of these spheres could be determined. The relative displacement of neighbouring nodes can then be calculated by subtracting the displacement of the individual nodes (or spheres). The selected spheres in the packing of the simulation were chosen in such a way that their relative orientations corresponded to the contact network shown in figure 36. It was verified that the position of the sphere corresponding to sphere 1 in figure 36 was in an ‘A’-layer. Furthermore, a sphere in the center of the packing was selected as the center sphere to be sure that none of the neighbouring spheres would

be across the boundary of the packing. The results are summarized in table 19 below.

Table 19: *The relative displacement of neighbouring nodes.*

vector	no. spheres sim.	Δu_1^{mn}	Δu_2^{mn}	Δu_3^{mn}
Δu_i^{1-2}	445 – 446	$1.00 * 10^{-6}$	0.00	0.00
Δu_i^{1-3}	445 – 455	$5.00 * 10^{-7}$	$1.73 * 10^{-18}$	0.00
Δu_i^{1-4}	445 – 454	$-5.00 * 10^{-7}$	$1.73 * 10^{-18}$	0.00
Δu_i^{1-8}	445 – 545	$5.00 * 10^{-7}$	$-2.90 * 10^{-7}$	0.00
Δu_i^{1-9}	445 – 454	$-5.00 * 10^{-7}$	$-2.90 * 10^{-7}$	0.00
Δu_i^{1-10}	445 – 535	0.00	$-2.90 * 10^{-7}$	0.00
Δu_i^{1-14}	445 – 436	$1.00 * 10^{-6}$	$-2.90 * 10^{-7}$	0.00
Δu_i^{10-14}	535 – 536	$1.00 * 10^{-6}$	0.00	0.00
Δu_i^{8-10}	545 – 535	$-5.00 * 10^{-7}$	$-1.00 * 10^{-14}$	0.00
Δu_i^{9-10}	544 – 535	$5.00 * 10^{-7}$	$-1.00 * 10^{-14}$	0.00
Δu_i^{1-11}	445 – 345	$5.00 * 10^{-7}$	$-2.90 * 10^{-7}$	-10^{-14}
Δu_i^{1-12}	445 – 344	$-5.00 * 10^{-7}$	$-2.90 * 10^{-7}$	-10^{-14}
Δu_i^{1-13}	445 – 335	0.00	$-2.90 * 10^{-7}$	-10^{-14}
Δu_i^{1-15}	445 – 336	$1.00 * 10^{-6}$	$-2.90 * 10^{-7}$	-10^{-14}

The strain tensor

With the calculated complementary area vectors and the relative displacements of the corresponding nodes, the average displacement gradient tensor can be determined using equation (21), which read:

$$\bar{e}_{ij} = \frac{1}{V} \sum_{m < n} \Delta u_i^{mn} d_j^{mn} .$$

The edges used in tables 18 and 19 form a representative elementary volume of an HCP. This unit consists of twelve space cells. Copying this unit in all the x -, y - and z -directions creates the entire space cell system of an HCP. Just like with the periodic boundary system, a ‘perfect’ space cell system can be formed if it is treated like a periodic system; the redundant edges at one side of a boundary compensates the missing edges at the boundary on the opposite side of the system. In other words, the sum in equation (21) only has to run over the fourteen edges of one space cell unit and that result only has to be divided by the volume of that unit. For an HCP system that consists of one thousand of such units, the sum must be repeated a thousand time for every edge, but the volume of the system is a thousand times larger as well, cancelling out the effect of multiple units.

It must be mentioned that it is checked that the relative translation of two neighbouring nodes is independent of their positions in the packing in the case of the periodic boundary systems; all the contact pairs that have the same branch vector and are located in similar layers, have the same corresponding complementary area vector and relative displacement. This means that it is justified to use the results in tables 18 and 19 for similar contact pairs.

The volume of one representative elementary unit of space cells can be calculated by summing up the volumes of the individual space cells of which it consists. From figure 36 it can be concluded that there are only two different shapes of space cells; a perfect tetrahedron and a distorted tetrahedron. The distorted tetrahedron shown in figure 37 has actually the same base and height as a perfect tetrahedron, like cell $t = 1$ in table 16, so the all the space cells (t) have the same volume V_t :

$$\begin{aligned} V_t &= \frac{1}{12} \sqrt{2} l^3 \\ &= \frac{1}{12} \sqrt{2} * 0.002139^3 = 1.153 * 10^{-9} . \end{aligned} \quad (171)$$

The volume of the representative elementary unit (V_{reu}) is this thus twelve times V_t :

$$\begin{aligned} V_{reu} &= \sqrt{2} l^3 \\ &= \sqrt{2} * 0.002139^3 = 1.384 * 10^{-8} . \end{aligned} \quad (172)$$

With this volume, the complementary area vectors of table 18 and the relative displacement vectors of table 19, the following average displacement tensor is found:

$$\bar{\epsilon}_{ij} = \begin{pmatrix} -4.626 * 10^{-4} & -8.435 * 10^{-6} & -5.964 * 10^{-6} \\ -9.740 * 10^{-14} & -3.431 * 10^{-12} & 1.193 * 10^{-13} \\ 0.000 & 0.000 & -2.863 * 10^{-12} \end{pmatrix} . \quad (173)$$

The strain tensor ϵ_{ij} is the symmetric part of the average displacement gradient tensor:

$$\epsilon_{ij} = \frac{1}{2} (\bar{\epsilon}_{ij} + \bar{\epsilon}_{ji}) . \quad (174)$$

This results in the following strain tensor:

$$\epsilon_{ij} = \begin{pmatrix} -4.626 * 10^{-4} & -4.217 * 10^{-6} & -2.982 * 10^{-6} \\ -4.217 * 10^{-6} & -3.431 * 10^{-12} & 5.965 * 10^{-14} \\ -2.982 * 10^{-6} & 5.965 * 10^{-14} & -2.863 * 10^{-12} \end{pmatrix} . \quad (175)$$

The first entry, ϵ_{11} , is the strain in x -direction. Multiplying this value with the initial length of the packing (the distance between the left and right boundary, which is 0.02139) gives:

$$-4.626 * 10^{-4} * 0.02139 = -9.896 * 10^{-6} ,$$

which is almost within 1% agreement with the actual macroscopic strain of the system in the x -direction; $\epsilon_{11} = 10^{-5}$. The strains in the y - and z -directions (ϵ_{22} and ϵ_{33} , respectively) are eight orders of magnitude smaller than ϵ_{11} , so it can safely be said that these strains are negligible. Only ϵ_{12} and ϵ_{13} (and the symmetric counterparts) are not completely negligible. They are around a factor hundred times smaller than ϵ_{11} , while they should be completely zero. Still, it can be said that the result obtained with Bagi's method is in fairly good agreement with the expected strain tensor.

5.3.2 The best-fit strain based on particle translations

The best-fit strain based on particle translation can be obtained in the following way:

- Take the initial positions of the spheres from the simulation. If the average initial position of all spheres is non-zero, then x_i^p is the initial position of sphere p minus the average position.
- Determine the displacement of every sphere from the simulation. If the average over all displacements is non-zero, then u_i^p is the difference between the displacement of sphere p and the average displacement of all spheres.
- Solve equation (32) three times; for $i = 1, 2, 3$. Each solution gives one row of the tensor α_{ij} .
- Calculate the strain tensor ϵ_{ij} from α_{ij} .

The procedure will be worked out in more detail for an HCP.

Relative initial positions

For this method, information from the same DEM simulation with the same packing was used as for Bagi's method. In this case, 48 spheres, approximately located in the center of the packing, were selected. This 'subpacking' consists of four layers of each four rows, which in turn contain three spheres

each. For these 48 spheres, the average initial position was calculated, which resulted in:

$$\bar{x}_i^p = \begin{pmatrix} 0.011764 \\ 0.007327 \\ 0.008394 \end{pmatrix}. \quad (176)$$

For every sphere, this average is subtracted, resulting in the relative initial position vectors for every sphere. The average of these positions is in the order of 10^{-18} , which is as good as zero.

Relative displacements

The displacement of a sphere is simply the difference between the final and initial positions of that sphere. For each of the 48 spheres, this displacement was determined and the average was calculated, which gave:

$$\bar{u}_i^p = \begin{pmatrix} -6.8813 * 10^{-7} \\ 5.4050 * 10^{-12} \\ 3.3500 * 10^{-12} \end{pmatrix} \quad (177)$$

Again, for every sphere this average is subtracted from their respective displacement vectors, giving the relative displacement vector u_i^p . The average of all relative displacements is also (close to) zero now.

The deformation gradient tensor

The next step is to solve the following system of equations three times; for $i = 1, 2, 3$:

$$\begin{bmatrix} \sum_{(p)} x_1^p x_1^p & \sum_{(p)} x_2^p x_1^p & \sum_{(p)} x_3^p x_1^p \\ \sum_{(p)} x_1^p x_2^p & \sum_{(p)} x_2^p x_2^p & \sum_{(p)} x_3^p x_2^p \\ \sum_{(p)} x_1^p x_3^p & \sum_{(p)} x_2^p x_3^p & \sum_{(p)} x_3^p x_3^p \end{bmatrix} \cdot \begin{bmatrix} \alpha_{1i} \\ \alpha_{2i} \\ \alpha_{3i} \end{bmatrix} = \begin{bmatrix} \sum_{(p)} u_i^p x_1^p \\ \sum_{(p)} u_i^p x_2^p \\ \sum_{(p)} u_i^p x_3^p \end{bmatrix}. \quad (178)$$

All the relative initial positions and relative displacements are known, so each sum in equation (178) can easily be calculated. For $i = 1$, the following augmented matrix must be solved in order to obtain α_{j1} :

$$\left(\begin{array}{ccc|c} 2.288 * 10^{-4} & -1.268 * 10^{-4} & -1.268 * 10^{-5} & -1.084 * 10^{-7} \\ -1.268 * 10^{-4} & 2.105 * 10^{-4} & -1.294 * 10^{-5} & 5.886 * 10^{-8} \\ 2.241 * 10^{-5} & -1.294 * 10^{-5} & 1.830 * 10^{-4} & -1.226 * 10^{-8} \end{array} \right). \quad (179)$$

In the following steps, this augmented matrix will be swept to a reduced echelon matrix. First, every entry is divided by the first entry of the row it is in:

$$\left(\begin{array}{ccc|c} 1 & -5.543 * 10^{-1} & 9.798 * 10^{-2} & -4.739 * 10^{-4} \\ 1 & -1.660 & 1.021 * 10^{-1} & -4.642 * 10^{-4} \\ 1 & -5.774 * 10^{-1} & 8.165 & -5.469 * 10^{-4} \end{array} \right). \quad (180)$$

Next, the first entries of the second and third row are made zero by subtracting the first row:

$$\left(\begin{array}{ccc|c} 1 & -5.543 * 10^{-1} & 9.798 * 10^{-2} & -4.739 * 10^{-4} \\ 0 & -1.106 & 4.082 * 10^{-3} & 9.662 * 10^{-6} \\ 0 & -2.309 * 10^{-2} & 8.067 & -7.306 * 10^{-5} \end{array} \right). \quad (181)$$

The entries of the second and third row are divided by the first non-zero entry of that row:

$$\left(\begin{array}{ccc|c} 1 & -5.543 * 10^{-1} & 9.798 * 10^{-2} & -4.739 * 10^{-4} \\ 0 & 1 & -3.692 * 10^{-3} & -8.739 * 10^{-6} \\ 0 & 1 & -3.493 * 10^2 & 3.164 * 10^{-3} \end{array} \right). \quad (182)$$

The second row is subtracted from the third row:

$$\left(\begin{array}{ccc|c} 1 & -5.543 * 10^{-1} & 9.798 * 10^{-2} & -4.739 * 10^{-4} \\ 0 & 1 & -3.692 * 10^{-3} & -8.739 * 10^{-6} \\ 0 & 0 & -3.493 * 10^2 & 3.172 * 10^{-3} \end{array} \right). \quad (183)$$

The entries of the third row are divided by the first non-zero entry of that row. This gives an echelon matrix:

$$\left(\begin{array}{ccc|c} 1 & -5.543 * 10^{-1} & 9.798 * 10^{-2} & -4.739 * 10^{-4} \\ 0 & 1 & -3.692 * 10^{-3} & -8.739 * 10^{-6} \\ 0 & 0 & 1 & -9.082 * 10^{-6} \end{array} \right). \quad (184)$$

In the coming steps, the matrix is swept to a reduced echelon matrix. First, the third entries of the first and second row are made zero by adding or subtracting the right fraction of the third row:

$$\left(\begin{array}{ccc|c} 1 & -5.543 * 10^{-1} & 0 & -4.730 * 10^{-4} \\ 0 & 1 & 0 & -8.772 * 10^{-6} \\ 0 & 0 & 1 & -9.082 * 10^{-6} \end{array} \right) . \quad (185)$$

Finally, the second entry of the first row is made zero by adding $-5.543 * 10^{-1}$ times the second row to the first row:

$$\left(\begin{array}{ccc|c} 1 & 0 & 0 & -4.778 * 10^{-4} \\ 0 & 1 & 0 & -8.772 * 10^{-6} \\ 0 & 0 & 1 & -9.082 * 10^{-6} \end{array} \right) . \quad (186)$$

The solution for α_{j1} can directly be read off from the reduced echelon matrix:

$$\alpha_{j1} = \begin{pmatrix} -4.778 * 10^{-4} \\ -8.772 * 10^{-6} \\ -9.082 * 10^{-6} \end{pmatrix} \quad (187)$$

The same procedure as described above is followed to find α_{j2} and α_{j3} . Combining the obtained three vectors gives the deformation gradient tensor α_{ij} :

$$\alpha_{ij} = \begin{pmatrix} -4.778 * 10^{-4} & -8.772 * 10^{-6} & -9.082 * 10^{-6} \\ -1.288 * 10^{-5} & 4.957 * 10^{-7} & -3.155 * 10^{-5} \\ 6.672 * 10^{-13} & -2.568 * 10^{-14} & 1.022 * 10^{-11} \end{pmatrix} . \quad (188)$$

The symmetric part of α_{ij} is the average strain tensor ϵ_{ij} for the system of spheres:

$$\epsilon_{ij} = \frac{1}{2} (\alpha_{ij} + \alpha_{ji}) . \quad (189)$$

This results in:

$$\epsilon_{ij} = \begin{pmatrix} -4.778 * 10^{-4} & -1.083 * 10^{-5} & -4.541 * 10^{-6} \\ -1.083 * 10^{-5} & 4.957 * 10^{-7} & -1.577 * 10^{-5} \\ -4.541 * 10^{-6} & -1.577 * 10^{-5} & 1.022 * 10^{-11} \end{pmatrix} . \quad (190)$$

As explained in the previous section, ϵ_{11} is the strain in the x -direction. Multiplying this value with the initial length of the packing (the distance between the left and right boundary, which is 0.02139) gives:

$$-4.778 * 10^{-4} * 0.02139 = -1.022 * 10^{-5} ,$$

which deviates around 2% from the actual displacement of the right wall, 10^{-5} . The ϵ_{22} entry and the non-diagonal elements are relatively high,

because they should be negligible compared to ϵ_{11} . Therefore, this result can be regarded as a rough approximation of the true macroscopic strain tensor. A better result can presumably be obtained when all the spheres of the packing are included in the calculations, but this is too time-consuming to do manually.

5.3.3 Satake's strain definition

The procedure that Satake follows to determine the strain tensor can be summarized as follows:

- The Dirichlet tessellation of the system of spheres.
- Construction of the Delaunay network from the Dirichlet tessellation.
- Calculating the Dirichlet centers from the Delaunay network. These centers should correspond to the corners of the Dirichlet cells.
- Calculating the dual branch vector for every contact pair, using equation (49).
- Calculating the volume of the contact region with equation (51).
- Determining the relative displacement of contacting spheres.
- Calculating the strain tensor with equation (56).

These steps will be worked out in detail below.

Dirichlet tessellation of the system

As mentioned before, an HCP is a packing of mono-sized spheres and therefore, the Voronoi and Dirichlet tessellation methods and Bagi's method to define material cells give the same result. The Dirichlet cells were already shown in figures 33 and 34 of the previous section (in Bagi's case the same cells were thus called 'material cells').

The Delaunay network

The Delaunay network is obtained by connecting the centers of spheres whose corresponding Dirichlet cells share a common face. Again, the problem arises that not all the Delaunay cells (simplexes) are (distorted) tetrahedrons. In order to determine the Dirichlet centers with Satake's equations (see section 2.4), it is required that the entire system is completely triangularized. However, if the same method of shifting some spheres is used,

as Bagi did, the Dirichlet centers of the additionally formed cells do not correspond to corners of the Dirichlet cells. Satake does not discuss the situation that not all simplexes are tetrahedrons yet. Therefore, the Delaunay network is left as it is now and the Dirichlet centers that are not the centers of tetrahedrons will be determined improvisationally, using the geometrical knowledge of the configuration of an HCP. The used Delaunay network is shown in figure 38 below. Contrarily to Bagi's space cell system in figure 36, there is a big cell formed by the lines connecting spheres 1, 2, 7, 8, 10 and 14. The Dirichlet center of this simplex is the intersection of the two lines connecting the centers of spheres 1-14 and 2-10.

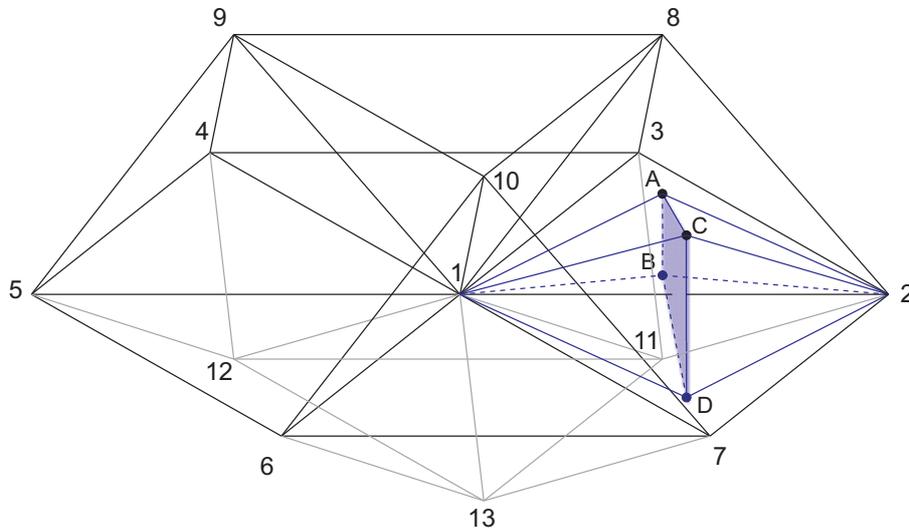


Figure 38: The Delaunay network for an HCP. The blue lines indicate the contact volume of contact 1-2. The blue area is S_c for this contact; it is one of the faces of the Dirichlet cell.

The Dirichlet centers

For the simplexes that have the shape of a tetrahedron, the corresponding Dirichlet centers can be calculated using equations (46) and (47). An example will be given for the Dirichlet center of the simplex formed by the sphere centers 1, 2, 3 and 8 shown in figure 38.

The center of sphere 1 is chosen as the origin of the coordinate system. The radius r_i of every sphere i is $1.07 * 10^{-3}$. The spheres have the following positions i_i :

$$i_1 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad i_2 = \begin{pmatrix} l \\ 0 \\ 0 \end{pmatrix} \quad i_3 = \begin{pmatrix} \frac{1}{2}l \\ \frac{1}{2}\sqrt{3}l \\ 0 \end{pmatrix} \quad i_8 = \begin{pmatrix} \frac{1}{2}l \\ \frac{1}{2\sqrt{3}}l \\ \sqrt{\frac{2}{3}}l \end{pmatrix}.$$

Using equation (46), the factor c^i can be calculated. The \mathbf{b} vectors in equation (47) are the same vectors that Bagi used, so these vectors do not have to be calculated again. The values for c^i and b_i^i for the 1-2-3-8 tetrahedron are summarized in table 20 below:

Table 20: The scalars c^i and b_i^i vectors for the tetrahedron 1-2-3-8 in figure 38.

sphere no.	c^i	b_1^i	b_2^i	b_3^i
1	$5.725 * 10^{-7}$	$\frac{1}{\sqrt{8}}l^2$	$\frac{1}{2\sqrt{6}}l^2$	$\frac{1}{4\sqrt{3}}l^2$
2	$-1.715 * 10^{-6}$	$-\frac{1}{\sqrt{8}}l^2$	$\frac{1}{2\sqrt{6}}l^2$	$\frac{1}{4\sqrt{3}}l^2$
3	$-1.715 * 10^{-6}$	0	$\frac{1}{\sqrt{6}}l^2$	$\frac{1}{4\sqrt{3}}l^2$
8	$-1.715 * 10^{-6}$	0	0	$\frac{\sqrt{3}}{4}l^2$

The volume V of the tetrahedron is $\frac{1}{12}\sqrt{2}l^3$. With these specified values, Dirichlet center becomes:

$$D_i^A = \begin{pmatrix} \frac{1}{2}l \\ \frac{1}{2\sqrt{3}}l \\ \frac{1}{2\sqrt{6}}l \end{pmatrix} \quad (191)$$

The positions of the Dirichlet centers **B**, **E**, **F**, **I**, **J**, **M** and **N** can also be calculated with this method. The other corner positions have to be determined in another way. As mentioned earlier, corner (or Dirichlet center) **C** is the point where the lines 1-14 and 2-14 intersect. It easily follows that the x -coordinate is halfway between sphere centers 1 and 2; the y - and z -coordinates are both half of the y - and z -coordinates of the position of the center of sphere 10:

$$D_i^C = \begin{pmatrix} \frac{1}{2}l \\ -\frac{1}{2\sqrt{3}}l \\ \frac{1}{\sqrt{6}}l \end{pmatrix} \quad (192)$$

The positions of the Dirichlet centers **D**, **G**, **H**, **K** and **L** in figure 33 can be found in a similar way. The positions of all the Dirichlet centers are summarized in table 21.

Table 21: The positions of the Dirichlet centers (corners of the material cells) shown in figure 33. The entries are divided by l^2 .

Dirichlet center	D_1^i	D_2^i	D_3^i
A	$\frac{1}{2}$	$\frac{1}{2\sqrt{3}}$	$\frac{1}{2\sqrt{6}}$
B	$\frac{1}{2}$	$\frac{1}{2\sqrt{3}}$	$-\frac{1}{2\sqrt{6}}$
C	$\frac{1}{2}$	$-\frac{1}{2\sqrt{3}}$	$\frac{1}{\sqrt{6}}$
D	$\frac{1}{2}$	$-\frac{1}{2\sqrt{3}}$	$-\frac{1}{\sqrt{6}}$
E	0	$-\frac{1}{\sqrt{3}}$	$\frac{1}{2\sqrt{6}}$
F	0	$-\frac{1}{\sqrt{3}}$	$-\frac{1}{2\sqrt{6}}$
G	$-\frac{1}{2}$	$-\frac{1}{2\sqrt{3}}$	$\frac{1}{\sqrt{6}}$
H	$-\frac{1}{2}$	$-\frac{1}{2\sqrt{3}}$	$-\frac{1}{\sqrt{6}}$
I	$-\frac{1}{2}$	$\frac{1}{2\sqrt{3}}$	$\frac{1}{2\sqrt{6}}$
J	$-\frac{1}{2}$	$\frac{1}{2\sqrt{3}}$	$-\frac{1}{2\sqrt{6}}$
K	0	$\frac{1}{\sqrt{3}}$	$\frac{1}{\sqrt{6}}$
L	0	$\frac{1}{\sqrt{3}}$	$-\frac{1}{\sqrt{6}}$
M	0	0	$\sqrt{\frac{3}{8}}$
N	0	0	$-\sqrt{\frac{3}{8}}$

The dual branch vectors

For every edge of the Delaunay network, the dual branch vector must be calculated. As was the case with the complementary area vectors with Bagi's method, only a limited number of dual branch vector have to be determined. All other edges are comparable to one of these edges. For the Delaunay network shown in figure 38, there are twelve different types of edges, so to each of these types a dual branch vector must be assigned. As an example, the calculations will be demonstrated for the dual branch vector corresponding to the contact between spheres 1 and 2 in figure 38.

The dual branch vector h_i^{1-2} can be calculated with:

$$h_i^{1-2} = S^{1-2} n_i^{1-2}, \quad (193)$$

where S^{1-2} is the area of the common face shared by the Dirichlet cells corresponding to spheres 1 and 2. The corners of this face are the corners **A**, **B**, **C** and **D** in table 21. The dimensions of the face are given in figure 39.

To determine its surface area, the trapezoid can be divided in a rectangle and two right-angled triangles. The area of the rectangle is

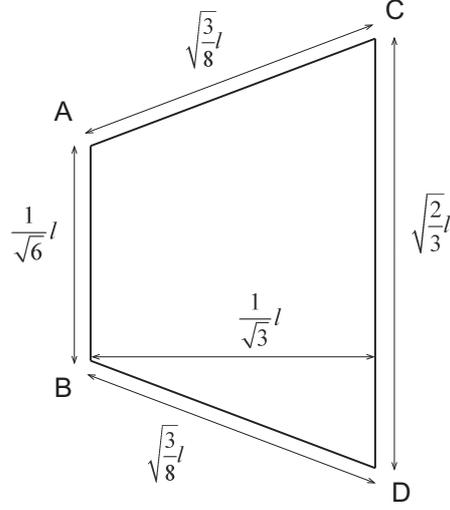


Figure 39: Face $ABCD$ and its dimensions corresponding to the 1-2 contact.

$$\frac{1}{6} * \frac{1}{\sqrt{3}} = \frac{1}{\sqrt{18}}$$

and the area of the two triangles together is

$$\frac{2}{2\sqrt{6}} * \frac{1}{\sqrt{3}} = \frac{1}{2\sqrt{18}} .$$

The surface area of the face is thus

$$\begin{aligned} S^{1-2} &= \frac{1}{\sqrt{18}} l^2 + \frac{1}{2\sqrt{18}} l^2 \\ &= \frac{1}{\sqrt{8}} l^2 . \end{aligned}$$

The outward normal unit vector corresponding to contact 1-2, n_i^{1-2} , is simply the normalized branch vector of that contact:

$$n_i^{1-2} = \frac{l_i^{1-2}}{|l_i^{1-2}|} \quad (194)$$

This vector was already given in equation (66) in section 3.2.1 and reads:

$$n_i^{1-2} = \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix} \quad (195)$$

Since the relative displacements of the spheres will be determined as the displacement of sphere 1 minus sphere 2, the branch vector is also defined

here as the difference of the initial positions of sphere 1 minus sphere 2, which explains the minus sign for n_1^{1-2} in equation (195). The dual branch vector h_i^{1-2} yields:

$$h_i^{1-2} = \frac{1}{2\sqrt{2}} l^2 \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -\frac{1}{2\sqrt{2}} \\ 0 \\ 0 \end{pmatrix} l^2 \quad (196)$$

To determine the other eleven dual branch vectors, similar calculations have to be performed. The normal vectors given in equation (66) can be used here, only their signs have to be flipped. The surface areas of the other eleven faces of the Dirichlet cell have not been calculated yet. However, due to the regularity of the polyhedron, there are only three different types of faces:

- The trapezoid face; this type occurs six times.
- The square face; this type occurs four times.
- The diamond face; this type occurs two times.

Only the areas of the square and diamond faces still have to be determined. One of such square faces is formed by corners **ACKM** (see figure 33). The length of all sides is $\sqrt{\frac{3}{8}}l^2$, so its surface area simply is $\frac{3}{8}l^2$. For the diamond face, it can be verified that its area is $\frac{1}{2\sqrt{2}}l^2$. The results for the dual brach vectors corresponding to the twelve contact types are summarized in table 22 below.

Volume of the contact region

The volume of the contact region was expressed with equation (51), which read:

$$V^{mn} = \frac{1}{3} l_i^{mn} h_i^{mn} .$$

The contact volume for the 1-2 contact is indicated in figure 38 by the blue-lined octahedron. For this contact, the volume becomes:

$$\begin{aligned} V^{1-2} &= \frac{1}{3} l_i^{1-2} h_i^{1-2} \\ &= \frac{1}{3} * \frac{1}{2\sqrt{2}} l^3 \\ &= \frac{1}{6\sqrt{2}} l^3 . \end{aligned} \quad (197)$$

Table 22: The dual branch vectors for the twelve contact types. The entries are divided by l^2 . The second column gives the shape of the face corresponding to the respective contact; ‘t’ = trapezoid, ‘s’ = square and ‘d’ = diamond.

vector	face	exact value			numerical value		
		h_1^{mn}	h_2^{mn}	h_3^{mn}	h_1^{mn}	h_2^{mn}	h_3^{mn}
h_i^{1-2}	t	$-\frac{1}{2\sqrt{2}}$	0	0	-0.3536	0	0
h_i^{1-3}	t	$-\frac{1}{4\sqrt{2}}$	$-\frac{\sqrt{3}}{4\sqrt{2}}$	0	-0.1768	-0.3062	0
h_i^{1-4}	t	$\frac{1}{4\sqrt{2}}$	$-\frac{\sqrt{3}}{4\sqrt{2}}$	0	0.1768	-0.3062	0
h_i^{1-8}	s	$-\frac{3}{16}$	$-\frac{\sqrt{3}}{16}$	$-\frac{\sqrt{3}}{4\sqrt{2}}$	-0.1875	-0.1083	-0.3062
h_i^{1-9}	s	$\frac{3}{16}$	$-\frac{\sqrt{3}}{16}$	$-\frac{\sqrt{3}}{4\sqrt{2}}$	0.1875	-0.1083	-0.3062
h_i^{1-10}	d	0	$\frac{\sqrt{1}}{2\sqrt{6}}$	$-\frac{1}{2\sqrt{3}}$	0	0.2041	-0.2887
h_i^{10-14}	t	$-\frac{1}{2\sqrt{2}}$	0	0	-0.3536	0	0
h_i^{8-10}	t	$\frac{1}{4\sqrt{2}}$	$\frac{\sqrt{3}}{4\sqrt{2}}$	0	0.1768	0.3062	0
h_i^{9-10}	t	$-\frac{1}{4\sqrt{2}}$	$\frac{\sqrt{3}}{4\sqrt{2}}$	0	-0.1768	0.3062	0
h_i^{1-11}	s	$-\frac{3}{16}$	$-\frac{\sqrt{3}}{16}$	$\frac{\sqrt{3}}{4\sqrt{2}}$	-0.1875	-0.1083	0.3062
h_i^{1-12}	s	$\frac{3}{16}$	$-\frac{\sqrt{3}}{16}$	$\frac{\sqrt{3}}{4\sqrt{2}}$	0.1875	-0.1083	0.3062
h_i^{1-13}	d	0	$\frac{\sqrt{1}}{2\sqrt{6}}$	$\frac{1}{2\sqrt{3}}$	0	0.2041	0.2887

The same contact volume is found for the other contacts, except for V^{1-8} , V^{1-9} , V^{1-11} and V^{1-12} , in these cases the volume is $\frac{1}{8}l^3$.

The relative displacement of contacting spheres

The results from the same simulation as with the previous two methods were used to obtain values for the relative displacements of the contacting spheres. The characteristics of the simulation are shown in table 15. The same spheres as with Bagi’s method were used to determine the relative displacements of the contacting spheres. These results are shown in table 19. Only the relative displacements u_i^{1-14} and u_i^{1-15} are not used in this case.

The strain tensor

The strain tensor related to one contact pair can be calculated with equation (54):

$$\epsilon_c = \frac{1}{3V_c} \mathbf{h}_c \mathbf{u}_c .$$

The twelve edges of the Delaunay network used in this section form a representative elementary unit. This unit can be copied in the x -, y - and z -direction to create a complete and ‘perfect’ HCP system. Also here it holds that the redundant edges at one side of the boundary compensate for the missing edges at the opposite side. The overall strain tensor for a granular assembly, as suggested by Satake, is:

$$\epsilon = \frac{1}{V} \sum \mathbf{h}_c \mathbf{u}_c = \frac{1}{V} \sum 3V_c \epsilon_c .$$

For an HCP, the sum runs over the twelve representative edges of the Delaunay network and the volume V is the sum of the twelve contact volumes connected to every contact, which is:

$$V = \left(8 * \frac{1}{6\sqrt{2}} + 4 * \frac{1}{8} \right) * 0.002139^3 = 1.412 * 10^{-8} . \quad (198)$$

Using the dual branch vectors from table 22 and the corresponding relative displacement vectors from table 19, the strain tensor for the granular assembly becomes, using equation (56):

$$\epsilon_{ij} = \begin{pmatrix} -4.652 * 10^{-4} & 0.000 & 0.000 \\ 4.659 * 10^{-16} & 2.323 * 10^{-6} & 4.012 * 10^{-14} \\ 0.000 & 0.000 & -2.920 * 10^{-12} \end{pmatrix} . \quad (199)$$

Satake did not take the symmetric part of this tensor to define the strain tensor; instead he called this instant result the overall strain tensor for the granular assembly. Multiplying ϵ_{11} with the length of the packing in the x -direction, 0.02139, gives:

$$-4.652 * 10^{-4} * 0.02139 = -9.950 * 10^{-6} ,$$

which is within 0.5% agreement with the actual displacement of the corresponding wall, which is 10^{-5} . The ϵ_{22} entry is a factor 200 smaller than ϵ_{11} , while the entry should be zero. All the non-diagonal elements are (close to) zero, so it this result gives a fairly accurate approximation of the actual macroscopic strain in the system.

5.3.4 Overview of results

The strain tensor was determined for two packings for both wall and periodic boundary systems, using all three methods described in this chapter. All packings were compressed in all the x -, y - and z -directions in separate simulations, so only one wall was moved at a time. The results are summarized in table 23 below. The first two packings are in periodic boundary systems and have 1000 ($n = 10, p = 5, q = 5$) and 2744 ($n = 14, p = 7, q = 7$) spheres, respectively. The last two packings are in wall boundary systems and have 900 ($n = 10, p = 5, q = 5$) and 3840 ($n = 16, p = 8, q = 8$) spheres, respectively. The deviations refer to the deviations between the values of the diagonal elements and the ‘real’ strain in that direction. This table only shows these deviations for the diagonal entries of the corresponding directions of compression; if the compression was in the y -direction, the deviation of the ϵ_{22} -entry of the strain tensor is shown. In all packings, the strain is determined from a group of spheres located in the centers of the packings.

Table 23: Overview of the performance of the three strain definitions for 3D granular assemblies. The description in the left column means: boundary condition (P =periodic, W =Wall) - number of spheres - direction of compression. The deviations refer to the deviations between the values of the diagonal elements of the strain tensor and the ‘real’ strain in the direction indicated in the first column.

description HCP	Error in strain estimation		
	Bagi	Best-fit	Satake
P_1000_x	1.04%	-2.21%	0.50%
P_1000_y	3.12%	-0.11%	0.99%
P_1000_z	0.00%	0.00%	-1.98%
P_2744_x	1.04%	0.00%	-0.50%
P_2744_y	3.13%	-0.11%	0.99%
P_2744_z	0.00%	0.00%	-1.98%
W_900_x	35.48%	34.57%	35.63%
W_900_y	41.96%	41.47%	40.38%
W_900_z	19.09%	19.22%	17.39%
W_3840_x	23.04%	22.23%	22.68%
W_3840_y	31.72%	26.05%	30.20%
W_3840_z	13.77%	13.77%	12.08%

This table shows that the strain for an HCP in a periodic boundary system is fairly accurately predicted by all three strain definitions. The size of the packing does not seem to have a significant influence on the results. The errors made in the prediction of the strain of packings in the wall boundary systems are very large, up to 42%. The results clearly show that the predictions are better for the packing of 3840 spheres. This can be explained

by the fact that wall-effects become less significant as the size of the packing increases. However, the large errors for the wall boundary systems are not solely due to the size of the packing. Like was done for the periodic boundary packings, only the relative displacements for a couple of spheres were determined and the relative displacements between similarly oriented contacting spheres were assumed equal. This may be true in the case of periodic boundary systems, but for wall boundary systems the validity of this assumption is doubtful. However, determining the orientations, and relative displacements of all contacts in the packing is far too time-consuming to do manually, so up to this point such an assumption is necessary to predict the strain. It is expected that the accuracy of the predictions of the strain tensor will improve significantly when the strain definitions are implemented in the processing tools of the DEM simulation, because the relative displacements can then be determined for all contacting pairs of spheres individually.

6 Conclusions and discussion

The first aim of this thesis was to judge the accuracy of the estimations of properties of 3D granular assemblies that were determined from DEM simulations. The effect of the walls on these properties was evaluated and methods to diminish these effects were tested. The second aim was to determine the strain tensor for 3D granular packings that were uniaxially compressed in DEM simulations, using three different strain definitions. To accomplish these goals, a structured packing was created for the DEM simulations; the Hexagonal Close Packing. The properties of a structured packing are known and can ‘easily’ be derived theoretically. The obtained properties from the simulations can then be compared to these theoretically derived properties. For two of the three strain definitions (Bagi’s and Satake’s definitions), the knowledge of the structure of the granular assembly forms the basis of the derivation of the respective strain tensors, so also here, the use of a structured packing is helpful.

In the following, the most important conclusions that can be drawn from the obtained results in this research will be presented and discussed. These conclusions and discussions will then lead to the recommendations given in the next chapter.

6.1 The properties of the HCP

6.1.1 The volume fraction

First, the volume fraction of a ‘perfect’ HCP was theoretically derived with the aid of a unit cell. It was shown that the volume fraction is 0.7405 and independent of the size of the spheres in the packing. For the HCP in the periodic boundary packing a density of 0.7451 was found for all packing sizes. This value is slightly higher due to the small overlap allowed between the spheres ($\delta/a \approx 10^{-3}$).

In wall boundary systems, the volume fraction of the HCP is strongly related to the size of the packing. As the size increases, the volume fraction comes closer to the ‘perfect’ density of $\eta_{HCP} = 0.7405$. By linearization of the graph relating the volume fraction against the number of spheres, it was predicted that the volume fraction of an HCP in a wall boundary system will approach 0.712, but this prediction is doubtful, even though the R^2 of the trend line was 1. A second approach, which used the equations that determine the wall positions and the number of spheres, indeed showed that the volume fraction of an infinitely large HCP in a wall boundary system approaches the volume fraction of a ‘perfect’ HCP.

The moving-average method is a suitable method to cancel out the wall-effects, such that the volume fraction of an ideal HCP is approximated. In order to get a random structure inside the averaging volume, a smaller sub-cell was moved diagonally through the packing in eleven steps, while the

volume fraction within that cell was estimated at every step. This method was tested on a packing that originally had a volume fraction that deviated 11.3% from the ideal density. After averaging the eleven volume fractions, this deviation was reduced to 0.39%.

6.1.2 The contact network

With the theoretical derivation of the contact network of the spheres in an HCP it was shown that this configuration alternates every layer. The differences only showed up in the sign of the y -components of the branch vectors (and the corresponding normal unit vectors) of the contacting spheres that were positioned in neighbouring layers. This result was also found in both the wall boundary and periodic boundary systems.

Furthermore, it can be concluded that the relaxation of an HCP in a wall boundary system does not lead to a significant change in the contact network, but the overlap between spheres are greatly affected. An initially isotropic overlap distribution changes into an anisotropic overlap distribution and the average overlap length is significantly reduced. These effects can be diminished by increasing the size of the system, but in order to be able to neglect these effects, the packing should be much larger than the maximum system size that the DEM simulation can handle.

6.1.3 Kinetic and potential energy

First, the potential energy density of an HCP was theoretically derived, relating it to the stiffness constant, overlap length and the branch length of the contacts. For this derivation, the same unit cell was used as for the derivation of the volume fraction. In the DEM simulation with an HCP in a periodic boundary system, the same potential energy density was found as was theoretically predicted. For the wall boundary system the potential energy density could not be compared with a theoretical prediction, due to the anisotropic and inhomogeneous overlap distribution after relaxation.

6.1.4 The elastic modulus tensor

The theoretically derived elastic modulus tensor for the HCP showed that there is an anisotropy in the elastic behaviour with respect to direction of compression; the C_{3333} component is larger than the C_{1111} and C_{2222} components. Moreover it can be concluded that in a ‘perfect’ packing the contribution of every sphere to the EMT is the same, regardless of the layer that the sphere is in.

With the DEM simulations the EMT for both the wall and periodic boundary systems was determined in two different ways. With the first method, the information obtained from a static snapshot was used to construct the EMT. In the case of the periodic packing, the results agreed within

an accuracy of 0.8% with the theoretically derived entries of the EMT. This deviation is probably caused by the slightly higher volume fraction of the system in the simulation, compared to the theoretical volume fraction. For the wall boundary systems, these deviations are larger due to non-crystal structure near the walls. For an HCP consisting of 900 spheres, the maximum deviation of the entries obtained from a static full-system snapshot was 15.5%. After correcting for the deviation between the volume fraction of the entire packing in the wall boundary system and the density of a perfect packing, the maximum deviation could be reduced to 6.5%. Furthermore, it can be concluded that estimating the EMT from sub-cells of a packing does not lead to better results. The best way to estimate the EMT from a static snapshot of a granular packing in a wall boundary system is to determine the EMT for the entire packing and correct for it by a factor that is obtained by dividing the overall density of the packing by the average density in the center of the packing that it determined with the moving-average method.

With the second method, the EMT was obtained from several snapshots of the simulation. For a periodic boundary system, the entries of the EMT were estimated within an accuracy of 10.5%, with respect to the theoretically derived values. In the case of packings in wall boundary systems, the predictions are very inaccurate. Therefore it is not recommended to apply this method to such systems. The deviations get smaller as the size of the packing increases, but even simulations with the largest packings that can be handled will probably still not lead to reliable predictions of the EMT.

6.2 The strain tensor for an HCP

Based on the literature survey described in chapter 2, three strain definitions were tested for their ability to describe the macroscopic strain tensor that relates the compression of the packing to its initial length. These strain definitions use the microscopic properties of 3D (or 2D) granular packings and the (relative) displacements of individual spheres to obtain the macroscopic strain tensor.

The first strain definition that was tested in this thesis was Bagi's strain definition based on equivalent continua. This method is rather complicated and time-consuming. It involves the construction of material and space cells, the assignment of complementary area vectors, the relative translations must be determined and finally the strain tensor must be calculated.

The best-fit strain based on particle translations, on the other hand, is a very straightforward and fast method. The initial and final positions of the spheres already provide enough information to calculate the strain tensor.

The last strain definition that was tested, the Satake strain, is also a very time-consuming method. Like Bagi's method, it involves the construction of sub-spaces, the assignment of vectors (in this case the dual branch vector and the relative displacement vectors) and finally the calculation of the strain

tensor.

Comparing the results presented in table 23 shows that the differences in the obtained macroscopic strain tensors are rather small. For the periodic boundary systems, all three methods give reliable predictions; within a maximum deviation of 3.2% from the actual strain. For the wall boundary systems, all three strain definitions result in large deviations, but they are all within the same range. The most important reason for these deviations is that only information from a couple of spheres was used to calculate the strain tensors and in the case of wall boundary systems, this information is not representative for the behaviour of the entire packing. It is expected that the results will improve when information of much more spheres is used in the calculations, but this is too time-consuming to do by hand. The implementation of these strain definitions in the DEM simulation processing software will make it possible to use information from all the spheres in the packing, which will presumably lead to better results.

7 Recommendations

Based on the research that was done for this thesis, some recommendations will be given for possible future research.

In the DEM simulations performed during this project a structured regular mono-dispersed frictionless packing was used, in order to be able to compare its properties with theoretically derived properties. Since good agreement was obtained between the properties of this packing in periodic boundary systems and theory, these properties can be also be determined for more complex packings in periodic boundary systems with DEM simulations. Possible adjustments are:

- Randomization of the packing.
- Use different spring stiffnesses for differently oriented contacts.
- Creating a poly-dispersed (random) packing.

The above mentioned adjustments can also be introduced for wall boundary systems, but due to the wall-effects additional efforts are needed to retrieve useful information:

- For determining the volume fraction inside the packing, it is recommended to use the moving-average method, by (diagonally) shifting a sub-cell through the packing. With the implementation of this method in the DEM program, the number of shifts can easily be increased so more accurate results can be obtained.
- For retrieving the elastic modulus tensor, it is recommended to determine it from a single snapshot for the entire system first (so not for a sub-volume) and multiply this result with a correction factor. This correction factor is the volume fraction in the center of the packing divided by the volume fraction of the complete packing. The overall density can directly be retrieved from the output files and the density inside the packing can be estimated with the moving-average method. The EMT obtained from the slopes of the stress-strain graphs is not reliable enough, since the system size is too small. When processor technology significantly improves in the future it may be possible to do DEM simulations with sufficiently large packings such that the EMT can also determined with this method. However, in ‘real’ experiments the EMT is obtained from the slope of the stress-strain graph, so in that sense it is still valuable to use this method in the simulations as well.

The long-term goal is to be able to determine the local average of the properties of the packing in a rather small volume.

The elaboration of the strain definitions for 3D granular assemblies applied to the HCP showed that all three methods performed approximately equally well under the applied conditions in the DEM simulations. However, the best-fit method is by far the fastest and simplest method, so its implementation in the DEM program will be much easier than Bagi's and Satake's strain definitions. However, these definitions were only tested on a structured frictionless packing and more research must be done to test the reliability of their predictions of the strain tensor using simulations with more complex packings.

References

- [1] Bagi, K. (1993), *On the definition of stress and strain in granular assemblies through the relation between micro- and macro-level characteristics*, Powders and Grains 93, ed. C. Thornton, Balkema, 117-121.
- [2] Bagi, K. (1996), *Stress and strain in granular assemblies*, Mechanics of Materials, **22**, 165-177.
- [3] Cambou, B., Chaze, M., Dedecker, F. (2000), *Change of scale in granular materials*, European Journal of Mechanics - A/Solids **19**, 999-1014.
- [4] Cundall, P.A., Strack, O.D.L. (1979), *A discrete numerical model for granular assemblies*, Géotechnique, **29**(1), 47-65.
- [5] Dedecker, F., Chaze, M., Dubujet, Ph., Cambou, B. (2000), *Specific features of strain in granular materials*, Mechanics of Cohesive-Frictional Materials **5**, 173-193.
- [6] Hertz, H. (1882), *Über die Berührung fester elastischer Körper*, J. für die reine u. angew. Math., **92**, 136.
- [7] Itasca (1999) *Particle flow code in 2 dimensions. Users manual, part 'Theory and background'*, pp. 3.11-3.13 (Itasca Consulting Group, 708 South Third Street, Suite 310, Minneapolis MN 55415)
- [8] Kruyt, N.P., Rothenburg, L. (1996), *Micromechanical definition of the stress tensor for granular materials*, Journal of Applied Mechanics (Transactions of the ASME) **63**, 706-711.
- [9] Kruyt, N.P. (2003) *Statics and kinematics of discrete Cosserat-type granular materials*, International Journal of Solids and Structures **40**, 511-534.
- [10] Kuhn, M. (1997), *Deformation measures for granular materials*, In: Chang, C.S., Misra, A., Liang, R.Y., Babic, M. (Eds.), Mechanics of Deformations and Flow of Particulate Materials. ASCE, New York, pp. 91-104.
- [11] Kuhn, M. (1999), *Structured deformation in granular materials*, Mechanics of Materials **31**, 407-429.
- [12] Liao, C.L., Chang, T.P., Young, D.H. (1997) *Stress-strain relationship for granular materials based on the hypothesis of best fit*, International Journal of Solids and Structures **34**, 4087-4100.
- [13] Luding, S. (2004), *Micro-macro transition for an-isotropic, frictional granular packings*, International Journal of Solids and Structures, **41**, 5821-5836.

- [14] Rothenburg, L. (1980), *Micromechanics of idealized granular Materials*, PhD Dissertation, Carleton University, Canada.
- [15] Satake, M. (2002) *Micro-mechanical definition of strain tensor for granular assemblies*, 15th ASCE Engineering Mechanics Conference.
- [16] Satake, M. (2004) *Tensorial form definitions of discrete mechanical quantities for granular assemblies*, International Journal of Solids and Structures **41**, 5775-5791.

Appendices

A Derivation of u for one contact

In this part, the derivation of the energy density correlation will be worked out. The energy density u will be expressed as a function of the contact volume V_c , the spring stiffness in normal, k and the overlap in normal direction δ . The starting point of this derivation is equation (200):

$$\frac{1}{2} (\boldsymbol{\sigma}^T : \boldsymbol{\epsilon}^{sym}) = u , \quad (200)$$

where $\boldsymbol{\sigma}^T$ is the transposed stress tensor and $\boldsymbol{\epsilon}^{sym}$ is the symmetric part of the strain tensor. The double dot product of two tensors is equal to the first invariant (i.e. the trace) of their dot product and is denoted by the colon ($:$) between the two tensor symbols:

$$\frac{1}{2} (\boldsymbol{\sigma}^T : \boldsymbol{\epsilon}^{sym}) = \frac{1}{2} (tr(\boldsymbol{\sigma}^T \cdot \boldsymbol{\epsilon}^{sym})) . \quad (201)$$

The transpose of a matrix of any dimension is obtained by putting the initial row elements in columns and vice versa. The initial stress tensor and its transposed form from equation (201) are given in equations (202a) and (202b) below:

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} \quad (202a)$$

$$\boldsymbol{\sigma}^T = \begin{bmatrix} \sigma_{11} & \sigma_{21} \\ \sigma_{12} & \sigma_{22} \end{bmatrix} . \quad (202b)$$

The symmetric strain tensor is

$$\boldsymbol{\epsilon}^{sym} = \begin{bmatrix} \epsilon_{11}^{sym} & \epsilon_{12}^{sym} \\ \epsilon_{21}^{sym} & \epsilon_{22}^{sym} \end{bmatrix} . \quad (203)$$

The dot product of the transposed stress and the strain tensor becomes

$$\boldsymbol{\sigma}^T \cdot \boldsymbol{\epsilon}^{sym} = \begin{bmatrix} \sigma_{11} & \sigma_{21} \\ \sigma_{12} & \sigma_{22} \end{bmatrix} \cdot \begin{bmatrix} \epsilon_{11}^{sym} & \epsilon_{12}^{sym} \\ \epsilon_{21}^{sym} & \epsilon_{22}^{sym} \end{bmatrix} \quad (204a)$$

$$\boldsymbol{\sigma}^T \cdot \boldsymbol{\epsilon}^{sym} = \begin{bmatrix} \sigma_{11}\epsilon_{11}^{sym} + \sigma_{21}\epsilon_{21}^{sym} & \sigma_{11}\epsilon_{12}^{sym} + \sigma_{21}\epsilon_{22}^{sym} \\ \sigma_{12}\epsilon_{11}^{sym} + \sigma_{22}\epsilon_{21}^{sym} & \sigma_{12}\epsilon_{12}^{sym} + \sigma_{22}\epsilon_{22}^{sym} \end{bmatrix} . \quad (204b)$$

The trace of the product ($\boldsymbol{\sigma}^T \cdot \boldsymbol{\epsilon}^{sym}$) is obtained by the summation of the diagonal elements of the product tensor:

$$\frac{1}{2} (\text{tr}(\boldsymbol{\sigma}^T \cdot \boldsymbol{\epsilon}^{sym})) = \frac{1}{2} (\sigma_{11}\epsilon_{11}^{sym} + \sigma_{21}\epsilon_{21}^{sym} + \sigma_{12}\epsilon_{12}^{sym} + \sigma_{22}\epsilon_{22}^{sym}) = u . \quad (205)$$

The symmetric strain tensor is the sum of the original strain tensor and its transpose, divided by two:

$$\boldsymbol{\epsilon}^{sym} = \frac{1}{2}(\boldsymbol{\epsilon} + \boldsymbol{\epsilon}^T) \quad (206a)$$

$$\begin{bmatrix} \epsilon_{11}^{sym} & \epsilon_{12}^{sym} \\ \epsilon_{21}^{sym} & \epsilon_{22}^{sym} \end{bmatrix} = \frac{1}{2} \left(\begin{bmatrix} \epsilon_{11} & \epsilon_{12} \\ \epsilon_{21} & \epsilon_{22} \end{bmatrix} + \begin{bmatrix} \epsilon_{11} & \epsilon_{21} \\ \epsilon_{12} & \epsilon_{22} \end{bmatrix} \right) \quad (206b)$$

$$\begin{bmatrix} \epsilon_{11}^{sym} & \epsilon_{12}^{sym} \\ \epsilon_{21}^{sym} & \epsilon_{22}^{sym} \end{bmatrix} = \begin{bmatrix} \epsilon_{11} & \frac{1}{2}(\epsilon_{12} + \epsilon_{21}) \\ \frac{1}{2}(\epsilon_{21} + \epsilon_{12}) & \epsilon_{22} \end{bmatrix} . \quad (206c)$$

From equation (206c), the following equalities can be deduced:

$$\epsilon_{11}^{sym} = \epsilon_{11} \quad (206d)$$

$$\epsilon_{12}^{sym} = \frac{1}{2}(\epsilon_{12} + \epsilon_{21}) \quad (206e)$$

$$\epsilon_{21}^{sym} = \frac{1}{2}(\epsilon_{21} + \epsilon_{12}) \quad (206f)$$

$$\epsilon_{22}^{sym} = \epsilon_{22} . \quad (206g)$$

Not surprisingly, the elements ϵ_{12}^{sym} and ϵ_{21}^{sym} are equal, as follows from equations (206e) and (206f):

$$\epsilon_{12}^{sym} = \epsilon_{21}^{sym} . \quad (206h)$$

Substitution of the equations (206d) to (206g) into equation (205) gives:

$$u = \frac{1}{2} \left(\sigma_{11}\epsilon_{11} + \frac{1}{2}\sigma_{21}(\epsilon_{21} + \epsilon_{12}) + \frac{1}{2}\sigma_{12}(\epsilon_{12} + \epsilon_{21}) + \sigma_{22}\epsilon_{22} \right) . \quad (207)$$

When only deformations in the normal direction are considered, the strain tensor can be written as

$$\boldsymbol{\epsilon} = \frac{\delta}{l} \hat{\mathbf{n}} \hat{\mathbf{n}} \quad (208a)$$

$$\begin{bmatrix} \epsilon_{11} & \epsilon_{12} \\ \epsilon_{21} & \epsilon_{22} \end{bmatrix} = \frac{\delta}{l} \begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix} \begin{bmatrix} \hat{n}_1 & \hat{n}_2 \end{bmatrix} , \quad (208b)$$

where l is the distance between the centre points of the two spheres and $\hat{\mathbf{n}}$ is the unit vector in the normal direction. The dyadic product between the two unit vectors in normal direction results in a tensor:

$$\hat{\mathbf{n}}\hat{\mathbf{n}} = \begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix} \begin{bmatrix} \hat{n}_1 & \hat{n}_2 \end{bmatrix} = \begin{bmatrix} \hat{n}_1\hat{n}_1 & \hat{n}_1\hat{n}_2 \\ \hat{n}_2\hat{n}_1 & \hat{n}_2\hat{n}_2 \end{bmatrix}. \quad (208c)$$

The entries of the strain tensor can therefore be written as

$$\epsilon_{11} = \frac{\delta}{l} \hat{n}_1 \hat{n}_1 \quad (209a)$$

$$\epsilon_{12} = \frac{\delta}{l} \hat{n}_1 \hat{n}_2 \quad (209b)$$

$$\epsilon_{21} = \frac{\delta}{l} \hat{n}_2 \hat{n}_1 \quad (209c)$$

$$\epsilon_{22} = \frac{\delta}{l} \hat{n}_2 \hat{n}_2. \quad (209d)$$

The overlap in normal direction is defined as

$$\delta = \mathbf{l} - 2a\hat{\mathbf{n}} \quad (210a)$$

$$\begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} = \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} - 2a \begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix}, \quad (210b)$$

where a is the radius of the sphere. The unit vector in normal direction is defined as the ratio between the branch vector and its magnitude. The branch vector is defined as

$$\mathbf{l} = \mathbf{r}_1 - \mathbf{r}_2 \quad (211a)$$

$$\begin{bmatrix} l_1 \\ l_2 \end{bmatrix} = \begin{bmatrix} (r_1)_1 \\ (r_1)_2 \end{bmatrix} - \begin{bmatrix} (r_2)_1 \\ (r_2)_2 \end{bmatrix}. \quad (211b)$$

The magnitude of the branch vector is

$$l = |\mathbf{l}| = \sqrt{l_1^2 + l_2^2}. \quad (212)$$

The unit vector in normal direction is

$$\hat{\mathbf{n}} = \frac{\mathbf{l}}{l} \quad (213a)$$

$$\begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix} = \frac{1}{l} \begin{bmatrix} l_1 \\ l_2 \end{bmatrix}. \quad (213b)$$

Equation (213b) can be rewritten as

$$\begin{bmatrix} l_1 \\ l_2 \end{bmatrix} = l \begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix}. \quad (213c)$$

Substitution of equation (213c) into equation (210b) results in

$$\begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} = l \begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix} - 2a \begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix} = (l - 2a) \begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix}. \quad (214a)$$

Rewriting equation (214a) gives

$$\begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix} = \frac{1}{l - 2a} \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix}. \quad (214b)$$

The term $(l - 2a)$ corresponds to the magnitude of $\boldsymbol{\delta}$. Replacing the term for δ gives

$$\begin{bmatrix} \hat{n}_1 \\ \hat{n}_2 \end{bmatrix} = \frac{1}{\delta} \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix}. \quad (214c)$$

Therefore, the entries of $\hat{\mathbf{n}}$ become

$$\hat{n}_1 = \frac{\delta_1}{\delta} \quad (214d)$$

$$\hat{n}_2 = \frac{\delta_2}{\delta}. \quad (214e)$$

Substitution of equations (214d) and (214e) into equations (209a–209d) results in

$$\epsilon_{11} = \frac{\delta}{l} \frac{\delta_1}{\delta} \frac{\delta_1}{\delta} = \frac{\delta_1^2}{l\delta} \quad (215a)$$

$$\epsilon_{12} = \frac{\delta}{l} \frac{\delta_1}{\delta} \frac{\delta_2}{\delta} = \frac{\delta_1 \delta_2}{l\delta} \quad (215b)$$

$$\epsilon_{21} = \frac{\delta}{l} \frac{\delta_2}{\delta} \frac{\delta_1}{\delta} = \frac{\delta_2 \delta_1}{l\delta} \quad (215c)$$

$$\epsilon_{22} = \frac{\delta}{l} \frac{\delta_2}{\delta} \frac{\delta_2}{\delta} = \frac{\delta_2^2}{l\delta}. \quad (215d)$$

The stress tensor is, when only the contributions on normal direction are considered

$$\boldsymbol{\sigma} = \frac{kl\delta}{V_c} \hat{\mathbf{n}}\hat{\mathbf{n}} \quad (216a)$$

$$\begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} = \frac{kl\delta}{V_c} \begin{bmatrix} \hat{n}_1\hat{n}_1 & \hat{n}_1\hat{n}_2 \\ \hat{n}_2\hat{n}_1 & \hat{n}_2\hat{n}_2 \end{bmatrix}. \quad (216b)$$

Again using equations (214d) and (214e), the entries of the stress tensor can be written as

$$\sigma_{11} = \frac{kl\delta}{V_c} \hat{n}_1 \hat{n}_1 = \frac{kl\delta}{V_c} \frac{\delta_1}{\delta} \frac{\delta_1}{\delta} = \frac{kl\delta_1^2}{V_c\delta} \quad (217a)$$

$$\sigma_{12} = \frac{kl\delta}{V_c} \hat{n}_1 \hat{n}_2 = \frac{kl\delta}{V_c} \frac{\delta_1}{\delta} \frac{\delta_2}{\delta} = \frac{kl\delta_1\delta_2}{V_c\delta} \quad (217b)$$

$$\sigma_{21} = \frac{kl\delta}{V_c} \hat{n}_2 \hat{n}_1 = \frac{kl\delta}{V_c} \frac{\delta_2}{\delta} \frac{\delta_1}{\delta} = \frac{kl\delta_2\delta_1}{V_c\delta} \quad (217c)$$

$$\sigma_{22} = \frac{kl\delta}{V_c} \hat{n}_2 \hat{n}_2 = \frac{kl\delta}{V_c} \frac{\delta_2}{\delta} \frac{\delta_2}{\delta} = \frac{kl\delta_2^2}{V_c\delta} . \quad (217d)$$

Substitution of equations (217a) and (215a) into the term $(\sigma_{11}\epsilon_{11})$ of equation (207) gives

$$\sigma_{11}\epsilon_{11} = \frac{kl\delta_1^2}{V_c\delta} \frac{\delta_1^2}{l\delta} = \frac{k\delta_1^4}{V_c\delta^2} . \quad (218a)$$

Substitution of equations (217c), (215b) and (215c) into the term $\frac{1}{2}\sigma_{21}(\epsilon_{21} + \epsilon_{12})$ of equation (207) gives

$$\frac{1}{2}\sigma_{21}(\epsilon_{21} + \epsilon_{12}) = \frac{1}{2} \frac{kl\delta_1\delta_2}{V_c\delta} \left(\frac{\delta_1\delta_2}{l\delta} + \frac{\delta_1\delta_2}{l\delta} \right)$$

$$\frac{1}{2}\sigma_{21}(\epsilon_{21} + \epsilon_{12}) = \frac{1}{2} \frac{kl\delta_1\delta_2}{V_c\delta} 2 \frac{\delta_1\delta_2}{l\delta} = \frac{k\delta_1^2\delta_2^2}{V_c\delta} . \quad (218b)$$

Substitution of equations (217b), (215b) and (215c) into the term $\frac{1}{2}\sigma_{12}(\epsilon_{12} + \epsilon_{21})$ of equation (207) gives

$$\frac{1}{2}\sigma_{12}(\epsilon_{12} + \epsilon_{21}) = \frac{1}{2} \frac{kl\delta_1\delta_2}{V_c\delta} \left(\frac{\delta_1\delta_2}{l\delta} + \frac{\delta_1\delta_2}{l\delta} \right)$$

$$\frac{1}{2}\sigma_{12}(\epsilon_{12} + \epsilon_{21}) = \frac{1}{2} \frac{kl\delta_1\delta_2}{V_c\delta} 2 \frac{\delta_1\delta_2}{l\delta} = \frac{k\delta_1^2\delta_2^2}{V_c\delta} . \quad (218c)$$

Finally, substituting equations (217d) and (215d) into the term $(\sigma_{22}\epsilon_{22})$ of equation (207) gives:

$$\sigma_{22}\epsilon_{22} = \frac{kl\delta_2^2}{V_c\delta} \frac{\delta_2^2}{l\delta} = \frac{k\delta_2^4}{V_c\delta^2} . \quad (218d)$$

Replacing all the terms of equation (207) by equations (218a–218d) gives for the energy density

$$u = \frac{1}{2} \left(\frac{k\delta_1^4}{V_c\delta^2} + \frac{k\delta_1^2\delta_2^2}{V_c\delta^2} + \frac{k\delta_1^2\delta_2^2}{V_c\delta^2} + \frac{k\delta_2^4}{V_c\delta^2} \right) . \quad (219a)$$

This equation can be rewritten as

$$u = \frac{k}{2V_c\delta^2} (\delta_1^4 + 2\delta_1^2\delta_2^2 + \delta_2^4) . \quad (219b)$$

The term between the brackets can be rewritten to give

$$u = \frac{k}{2V_c\delta^2} (\delta_1^2 + \delta_2^2)^2 . \quad (219c)$$

By looking at the physical interpretation of the symbols δ_1 and δ_2 it can be deduced that the term $(\delta_1^2 + \delta_2^2)$ in equation (219c) can be replaced by δ^2 , by applying pythagoras' theorem: $\delta^2 = \delta_1^2 + \delta_2^2$, see also figure (40).

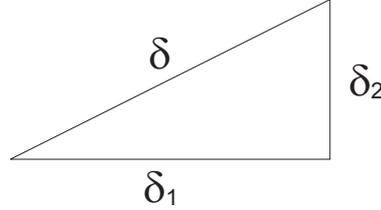


Figure 40: The overlap in the normal and tangential direction, δ_1 and δ_2 , respectively, define the magnitude of the overlap δ by applying Pythagoras' theorem.

With this simplification, the energy density becomes

$$u = \frac{k}{2V_c\delta^2} (\delta^2)^2 . \quad (220)$$

Finally, equation (220) can be simplified to give the desired energy density relation:

$$\boxed{u = \frac{k\delta^2}{2V_c}} \quad (221)$$

B Algorithms for defining an HCP in Fortran

B.1 Periodic boundaries

The algorithm for creating the Hexagonal Close Packing (HCP) with periodic boundaries with Fortran Code.

```
1-  do    k = 1, q
2-      do    j = 1, p
3-          do    i = 1, n
4-              place a sphere in an odd row of an 'A' layer
5-          enddo return to line 3 until i = n
6-          do    i = 1, n
7-              place a sphere in an even row of an 'A' layer
8-          enddo return to line 6 until i = n
9-      enddo return to line 2 until j = p
10-     do    j = 1, p
11-         do    i = 1, n
12-             place a sphere in an odd row of a 'B' layer
13-         enddo return to line 11 until i = n
14-         do    i = 1, n
15-             place a sphere in an even row of a 'B' layer
16-         enddo return to line 14 until i = n
17-     enddo return to line 10 until j = p
18- enddo return to line 1 until k = q
```

n = number of spheres in one row (in the x -direction)

p = number of odd/even row pairs

q = number of A/B layer pairs

i = the i th sphere in a row of n spheres

j = the j th odd/even row pair of p row pairs

k = the k th A/B layer pair of q layer pairs

B.2 Wall boundaries

The algorithm for creating the Hexagonal Close Packing (HCP) with wall boundaries with Fortran Code.

```
1-  do    k = 1, q
2-      do    j = 1, p
3-          do    i = 1, n
4-              place a sphere in an odd row of an ‘A’ layer
5-          enddo return to line 3 until i = n
6-          do    i = 1, n - 1
7-              place a sphere in an even row of an ‘A’ layer
8-          enddo return to line 6 until i = n - 1
9-      enddo return to line 2 until j = p
10-     do    j = 1, p - 1
11-         do    i = 1, n - 1
12-             place a sphere in an odd row of a ‘B’ layer
13-         enddo return to line 11 until i = n - 1
14-         do    i = 1, n
15-             place a sphere in an even row of a ‘B’ layer
16-         enddo return to line 14 until i = n
17-     enddo return to line 10 until j = p - 1
18-     do    i = 1, n - 1
19-         place a sphere in the final row of a ‘B’ layer
20-     enddo return to line 18 until i = n - 1
21- enddo return to line 1 until k = q
```

n = number of spheres in one row (in the x -direction)

p = number of odd/even row pairs

q = number of A/B layer pairs

i = the i th sphere in a row of n spheres

j = the j th odd/even row pair of p row pairs

k = the k th A/B layer pair of q layer pairs