Accepted Manuscript

Advances in particle packing algorithms for generating the medium in the Discrete Element Method

Carlos Recarey, Irvin Pérez, Roberto Roselló, Márcio Muniz,
Elizabeth Hernández, Robinson Giraldo, Eugenio Oñate

PII: S0045-7825(18)30566-8
DOI: <https://doi.org/10.1016/j.cma.2018.11.011>
Reference: CMA 12163

To appear in: *Comput. Methods Appl. Mech. Engrg.*

Received date : 9 August 2018
Revised date : 5 November 2018
Accepted date : 5 November 2018

Please cite this article as: C. Recarey, I. Pérez, R. Roselló et al., Advances in particle packing algorithms for generating the medium in the Discrete Element Method, *Computer Methods in Applied Mechanics and Engineering* (2018), <https://doi.org/10.1016/j.cma.2018.11.011>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Advances in particle packing algorithms for generating the medium in the Discrete Element Method

Authors

Carlos Recarey^{1, 2, 3} (recarey@uclv.edu.cu and recarey@cimne.upc.edu), Irvin Pérez (corresponding author)¹ (hotei.mat@gmail.com), Roberto Roselló¹ (rosello@uclv.edu.cu), Márcio Muniz² (mmuniz94@gmail.com), Elizabeth Hernández² (elihz.br@gmail.com), Robinson Giraldo² (roangizu@gmail.com), Eugenio Oñate^{3, 4} (onate@cimne.upc.edu)

¹ - Center of Computational Mechanics and Numerical Methods in Engineering, UCLV-CIMNE Classroom, Central University Marta Abreu of Las Villas, Santa Clara, Cuba

² - InfraLab, University of Brasilia, Faculty of Technology, Brasilia, Brazil

³ - Centre Internacional de Mètodes Numèrics en Enginyeria, Barcelona Spain

⁴ - Universitat Politècnica de Catalunya

Abstract

Several theoretical contributions in order to establish a particle packing methodology are presented. In this respect, a generic formulation of a new method for packing particles based on a constructive advancing front method, which uses Monte Carlo techniques for the generation of particle dimensions, is also shown. The method can be used to obtain virtual dense packings of particles with several geometrical shapes. It employs continuous, discrete and empirical statistical distributions in order to generate the dimensions of particles. The packing algorithm is very flexible and allows alternatives for: 1- The direction of the advancing front (inwards or outwards), 2- The selection of the local advancing front, 3- The method for placing a mobile particle in contact with others and 4- The overlap checks. The algorithm also allows obtaining highly porous media when it is slightly modified. Practical applications of the formulations are presented in the end of this paper.

Keywords: Particle packing, Discrete Element Method, advancing front, particle in contact

1.0 - Introduction

Modeling with particle methods is an alternative to the traditional continuum approach, which is under fast development lately in the context of Computational Mechanics. The Discrete Element Method (DEM) is a sound technology among other particle methods. In a similar way to the Finite Element Method (FEM), where the media has to be discretized into finite elements, it is necessary to have an initial set of particles or “packing” in DEM. Given that obtaining such initial set is not a trivial task, it is necessary to develop formulations of particle packing algorithms, which are essential to carry out DEM applications. Despite the diversity of particle shapes used in DEM, spheres are most commonly employed due to their simplicity. The large variety of packing techniques can be divided in two large groups: dynamic techniques [1-4] and constructive techniques [5-7]. Dynamic techniques are characterized by a previous DEM simulation to produce the initial packing. On the other hand, Constructive techniques generate the packing with pure geometrical calculations. Dynamic techniques include: 1- *Isotropic compression*, 2- *Particle growth*, 3- *Gravitational deposition* and 4- *Multi-layer compaction*.

The following constructive techniques can be found: 1- *Regular arrangements*, 2- *Sequential inhibition*, 3- *Sedimentation techniques* and 4- *Advancing front*, with its variants of *inwards packing* and *outwards packing*.

The most significant difference of dynamic techniques versus constructive techniques is their computational cost. However, they generally allow obtaining packings with a higher volume fraction (VF), which is defined as the ratio of solid volume to total volume (an exploration in this topic can be seen in [8]). The current trend in the development of packing techniques consists of using a combined approach of both dynamic and constructive methods, taking advantage of each method while minimizing its disadvantages. In order to obtain a packing of high volume fraction, an assembly of particles may be initially generated with a constructive method and then it may be further densified with some dynamic technique [7].

Perhaps the easiest to code and the most ineffective constructive packing technique is the sequential inhibition or Monte Carlo placement [9], where the positions and dimensions of particles are obtained with random number generators. With this method by itself, any prescribed distribution for particles dimensions can be obtained, but the density of the packings will be low and the coordination number will always be zero.

Constructive packing techniques have been developed for several types of particles. An algorithm for packing disks that can be relatively easily extended to other shapes in 2D such as polygons and ellipses can be seen in [6]. Such extension only needs to declare how to check the intersection between any two given particles, as well as how to translate a particle in such a way that it is in outer contact with other two given particles. An algorithm for packing ellipsoids (in 3D) can be seen in [10]. Such algorithm is much less efficient and direct, and is based on a kinematic model that does not allow the intersection of solid bodies. It is in fact a dynamic packing technique.

An advancing front algorithm for generating packings of disks and spheres can be seen in [11]. It is based on a straightforward procedure for selecting the advancing front. The algorithm also includes a process to eliminate the remaining heterogeneities of empty spaces. Another constructive algorithm for packing spheres, which is not of advancing front type, is shown in [12]. It is a purely geometrical multilayer compaction method, much more efficient than previous analogous packing methods such as for example [13], which also uses a multilayer compaction, but it is based on the Conjugate Gradient optimization method. Another formulation of the sphere packing problem using optimization is shown in [14]. In this case, it consists of a quadratic non-convex optimization problem with restrictions, and a linear objective function. It has the disadvantage of a high computational cost, and hence this model is not yet appropriate for a large number of spheres.

There are other constructive packing methods based on a triangulation of the space. One example is present in [15], which has the advantage of low computational cost, but has the limitation of yielding a low volume fraction and a low coordination number. This drawback has been improved in [16, 17], but without having full control of the statistical distribution of particle sizes. A multilayer method similar to that described in [13] has been proposed in [18]. It allows controlling the coordination number and the volume fraction, but has a high computational cost.

Other constructive advancing front packing methods are based on the generation of point clouds with a prescribed mean point distance [19-21], using the points as the centers of particles. This approach is ideal for many applications where the particle size distribution is not essential, but is not straightforward in other cases. However, several ideas of these point cloud generation algorithms can be used even in advancing front algorithms that respect any prescribed particle size distribution. That is the case of the deposition patterns, which include layered growth or seed-point (radius) growing [20], and have been used in advancing front algorithms using prescribed particle size distributions such as [22, 23].

The analysis of the state-of-the-art particle packing generation for DEM makes clear the necessity to continue its development. As general guidelines, the packing techniques should be independent with respect to the particle shapes. They should have a low computational cost and meet the requirements of engineering applications.

2.0 – General methodology for generating particle packings

Simulations with the Discrete Element Method (DEM) need an initial packing of particles that may have diverse shapes. In the present research work, a general methodology (Figure 1) for generating particle packings for DEM is presented. The first stage of such methodology consists in generating a packing using only (advancing front) geometrical methods, and optionally compacting such packing with dynamic methods. The second stage consists in the evaluation and characterization of the obtained packings. The evaluation techniques are grouped according to their function, and they allow to determine the similitude of the packings with respect to the studied materials. The generated packings can be used to calculate Laguerre diagrams in order to model the microstructure of polycrystalline materials.

Figure 1. General methodology for modeling the microstructure of materials

3.0 – Generic formulation of particle packing algorithm

Particle packing algorithms based on advancing front consist of several components. Some of them are the following: 1- type of particle; 2- initial advancing front, which determines if the

packing will be generated outwards or inwards; 3- definition of the local advancing front comprised by the already generated particles that will be in contact with the new particle later added to the packing; 4- neighbor search (space query); 5- intersection check between pairs of particles and 6- particle construction, which is a key element in achieving a high volume fraction. Some authors [7] state that each new particle in 2D packings must be built in contact with other two in order to get a higher local density. In the analogous 3D case, each new particle must be built in contact with other three for the same purpose (like in [22], for example). If it is necessary to increase the volume fraction of a packing obtained with an advancing front algorithm, then some dynamic technique such as isotropic compression can be used.

An advancing front is a group of particles in the surroundings of the evolving system of particles under generation. A group of previously placed particles lie inside the advancing front, while new particles are placed in contact with the outer particles of the front. The packing usually starts with a set of two or three particles at any given position or one or two particles in contact with the walls defining the domain (walls are also considered particles in this context). These particles comprise the initial advancing front. Then, a new particle is generated or chosen from a repertoire of particles to be added to the packing. Next, the new particle is placed at a position that just touches other particles in the advancing front. Then, the advancing front is updated and the process continues. Pseudocode 1 summarizes the basic steps of a generic advancing front algorithm.

1. *Initialize the packing (usually with two or three particles).*
2. *Generate or select the particle to be packed.*
3. *Select an active front and determine the position at which the particle just touches the particles in the front.*
4. *Check if the particle at this position overlaps with any existing particles.*
5. *If no overlap occurs, accept the new particle and return to step 2 for the next particle. Otherwise go to step 6.*
6. *Reject the position and repeat step 3 for another active front.*

Pseudocode 1. General scheme common to all advancing front packing algorithms.

The packing algorithm we propose in this paper (Pseudocode 2) already existed for the 2D case [7], but its extension to 3D is not straightforward due to the fact that there are no gaps in a string of tangent particles in 2D, but there will always be gaps in a layer of a finite number of tangent bodies in 3D. In some algorithms this is partly solved, but limiting the number of possible contacts of new particles (see [22] for more details).

The packing algorithm of Pseudocode 2 has some essential steps. The first one, initialization, consists of creating the initial advancing front. The second one, which is successively repeated until the end, consists of the construction of a new particle in contact with one already in the advancing front and other two not necessarily in the front. In this step, the advancing front is constantly updated. There are also two optional steps to increase or decrease the volume fraction, and another additional step to analyze the quality of the obtained packing.

The selection of particle type of the packing is implicit in Pseudocode 2, and the shape and dimensions of such particles can be generated by means of Monte Carlo techniques [24]. The initialization of the packing, in which the initial advancing front is defined, consists in generating

a few particles that will determine the growing direction of the packing process, which can be either outwards or inwards. The active front in 3D consists of a particle of the advancing front, and two other particles not necessarily in the advancing front. There are currently three variants implemented for the selection of these three particles from the lists that contain them. The variants to select those elements from their containing lists are: 1- following the order of insertion; 2- following the inverse order of insertion and 3- random selection. Each new particle added to the packing is built in outer contact with the ones in the selected local advancing front. For accepting a new particle p in the packing, it must be verified that it does not overlap with any other particle in the packing. This is accomplished by first finding a list V of neighboring particles of p , which could be overlapping with p . And then, if p does not overlap with any element of V , then it is accepted in the packing.

The steps for computational implementation of the advancing front packing algorithm proposed in this paper are listed in Pseudocode 2. Assuming that: 1) the lower and upper bounds of particle sizes have been previously set and 2) the domain geometry G is bounded, it can be proven that the execution of the particle packing algorithm always ends in a finite number of steps. In fact, the two previous conditions imply that only a finite number of particles can be placed inside G , given that particles may not overlap. When the maximum number of particles that can be contained in G is reached, the expression $C_{ext} := C_{ext} - \{p_0\}$ of step 2.3 will eliminate particles from C_{ext} until $C_{ext} = \emptyset$, which causes the termination of step 2 in a finite time. Steps 3 to 5 also end in a finite time. The algorithmic complexity has been proven to be of order $O(N)$, where N is the total number of particles [8].

Step 1 of the algorithm creates the initial advancing front. The way this front is created allows two growing directions: 1) *Inwards generation*, when the initial front is created in the interior of the bounding geometry (Figure 3); and 2) *Outwards generation*, when the initial front is created in outer contact with the border of the bounding geometry (Figure 4). After that, beginning with the initial advancing front, step 2 is cyclically repeated. In this step, several variants for the selection of the local advancing front can be applied. It also requires the construction of a particle in contact with other three particles, as well as the overlap check between particles, which employs neighbor search algorithms [12, 25].

Step 2 of Pseudocode 2 is the most complex one and will be explained in more detail. For the 2D case of this pseudocode, Figure 2 shows an example of two iterations of Step 2. The elements of C_{ext} are in dark. In Figure 2 (a) the dimension and rotation of the next particle p_{new} is generated. It is represented in the lower right corner of the square. In Figure 2 (b), an element $p_0 \in C_{ext}$ is selected (pointed by the arrow). Since it is not possible to place the new particle p_{new} in contact with p_0 and one of its neighbors (marked with stars), p_0 is removed from C_{ext} and a new element $p'_0 \in C_{ext}$ is selected (pointed by the arrow), as seen in Figure 2 (c). Since in this case it is possible to place p_{new} in contact with $p'_0 \in C_{ext}$, the particle p_{new} in its current position is added to C_{ext} (it is marked in black at Figure 2 (d)). The parameters $\tilde{r}$ in this step 2 are generated using Monte Carlo techniques.

Function W of step 2.2.1 must also be defined. Let p_1, p_2 and p_3 be any three particles of $\mathbb{R}^3$, $G \subseteq \mathbb{R}^3$ the bounding geometry that contains all the particles, and V a set of particles. Then $W(p_1, p_2, p_3, G, V, \tilde{r})$ is defined as the set of particles (or the empty set) such that they are in

outer contact with p_1 , p_2 and p_3 simultaneously, are fully contained in G and do not overlap with any element of V .

Input: containing geometry G (of finite volume), distribution of particles' dimensions Γ (including lower and upper bounds for each dimension)
Output: set E of particles
Step 1) (Initialization)
{
Create k initial particles $p_1, \dots, p_k \subseteq G$:
 $E := \{p_1, \dots, p_k\}$
 $C_{ext} := \{p_1, \dots, p_k\}$
}
Step 2)
 $\tilde{r} :=$ set of parameters that define the next particle except for its center
While $C_{ext} \neq \emptyset$
{
Step 2.1) (Definition of local front)
{
 $p_0 :=$ select particle of C_{ext}
 $V :=$ all neighbor particles of p_0
}
Step 2.2)
For all $p'_1, p'_2 \in V$
Step 2.2.1)
If $W(p_0, p'_1, p'_2, G, V, \tilde{r}) \neq \emptyset$
{
 $p_{new} :=$ any element of $W(p_0, p'_1, p'_2, G, V, \tilde{r})$
 $C_{ext} := C_{ext} \cup p_{new}$
 $E := E \cup p_{new}$
 $\tilde{r} :=$ set of parameters that define the next particle except for its center
Go to step 2.1
}
Step 2.3)
{
 $C_{ext} := C_{ext} - \{p_0\}$
}
}
Step 3) (Optional) Application of a compaction technique such as gravity compression to increase the VF.
Step 4) (Optional) Elimination of particles or reduction of their sizes to decrease the VF.
Step 5) (Optional) Application of packing quality analysis techniques.
Step 6) End

Pseudocode 2 Advancing front particle packing algorithm in 3D.

Figure 2. Example of step 2 of Pseudocode 2 for 2D

Step 3 can be used to eliminate the physical instabilities and increase the VF. A collective rearrangement technique, as recommended in [1, 3, 4, 7, 11, 20, 26, 27]), can be used for that purpose. For example, a gravity field can be applied [1, 3, 4], or else an isotropic compression [7], which consists in moving the walls that define the bounding domain so as to reduce the volume that contains the particles.

The output of Pseudocode 2 is the set E , where all obtained particles are stored. The particles comprising the advancing front are stored at C_{ext} . Such particles are used to define the local advancing front, which consists of an element of C_{ext} and other two elements of E not necessarily members of C_{ext} . Each time it is not possible to build a particle in contact with an element $p_0 \in C_{ext}$, such element is removed from C_{ext} (step 2.3). And each time a new particle p_{new} is built in contact with an element $p_0 \in C_{ext}$, such particle is added to C_{ext} and to E . To add a new particle to E , it has to not overlap with any of its elements.

One important thing of steps 1 and 2 of Pseudocode 2 is their independence with respect to the type of particle, the neighbor search method and the method to check the particles overlap. There is some flexibility in the definition of the initial advancing front, which allows to generate particles outwards or inwards, and also flexibility to build the local advancing fronts. Particles' dimensions can follow any given distribution and are generated with Monte Carlo techniques. Another flexibility of the algorithm is the possibility to choose among the available methods to build a particle in contact with others. The integration of the previously mentioned features in an advancing front algorithm, makes possible to have a systemic formulation of packing techniques for the DEM.

Steps 3 and 4 include other techniques in accordance with the systematic approach of the formulation. They modify the particles so as, for example, to increase the physical stability or to modify the volume fraction, among other aspects, in order to improve the quality of the packing.

3.1 – Type of particle in the packing

There are several particle shapes that can be used in DEM. For using a specific shape in advancing front packing algorithms, it is only required to know how to detect intersections for such shape, and how to place a particle in contact with other two (2D case) or three (3D case). In the present work, results have been obtained with circles and spheres, ellipses and ellipsoids, polygons and polyhedra, clusters of circles or spheres, superquadrics and spherocylinders, and methods for using more general shapes are explained.

3.2 – Initial advancing front

Step 1 of Pseudocode 2, which is used to create the initial advancing front, is described in this section. The way such initial front is defined determines the future growing direction of the advancing front, denoted by variable C_{ext} in Pseudocode 2. The growing direction can be outwards (Figure 3) [6, 28] or inwards (Figure 4) [11]. An initial advancing front that determines an outwards packing is given by the following steps:

Step 1) of Pseudocode 2 for an initial front that yields an outwards packing

Step 1) Find a point $\mathbf{c}_0 \in G$ such that $\inf\{d(\mathbf{c}, \mathbf{c}_0): \mathbf{c} \in G^c\} \geq \inf\{d(\mathbf{c}, \mathbf{q}): \mathbf{c} \in G^c\} \quad \forall \mathbf{q} \in G$, where d is the usual distance metric in $\mathbb{R}^n$ and G^c denotes the complement set of G . In other words, $\mathbf{c}_0$ is the innermost point in G .

Step 2) Build a set of four particles e_1, e_2, e_3 and e_4 in close contact in such a way that the arithmetic mean of their centers is $\mathbf{c}_0$.

Step 3) Make $E := C_{ext} := \{e_1, e_2, e_3, e_4\}$.

Figure 3. Example of outwards packing. Particles comprising the advancing front are in red. The first square to the left represents the initial advancing front

On the other hand, an initial advancing front that determines an inwards packing is given by the following steps:

Step 1) of Pseudocode 2 for an initial front that yields an inwards packing

Step 1) Build a set of particles $\{e_1, \dots, e_{N_0}\}$ as close to each other as possible, and in contact with the boundary of G . The initial set may comprise a single particle in contact with the corner walls (which may also be considered as particles)

Step 2) Make $C_{ext} := \{e_i, \dots\}$ $E := C_{ext} := \{e_1, \dots, e_{N_0}\}$

Step 3) Apply a modification of step 2 of Pseudocode 2 until the whole boundary of G has been covered by particles, where the modification is as follows: instead of placing each new particle in contact with other three, place it in contact with other two and with the boundary of G

Step 4) Assign to E and C_{ext} all the particles obtained so far

Figure 4. Example of inwards packing. Particles comprising the advancing front are in red. The first square to the left represents the initial advancing front

In the 2D inwards packing it is necessary to be able to build each new particle in contact with other two or with the domain boundary and another particle. The 3D inwards packing is similar, but each new particle is in contact with one more particle than the analogous 2D case. These requirements increase the complexity of the formulation but have the advantage that the packing fits exactly into G . If the domain boundary is also considered as a particle in this context, then each new particle will always be in contact with other two (three) in the 2D (3D) case. The pore left in the center of the packing is usually smaller than the average particle size, and is negligible if there are enough particles in the packing.

In the 2D (3D) outwards packing, particles are only built in contact with other two (three). In this case the packing does not fit exactly into G , and there remain large gaps between the particles and the boundary of G in the case where all particles are completely contained in G .

3.3 – Alternatives for the local advancing front

In Pseudocode 2, each new particle added to the packing will be in contact with three particles defining a local advancing front. Such local advancing front is formed by p_0 , selected from C_{ext} at step 2.1, and by particles p'_1 and p'_2 , selected from V in the *For* loop at step 2.2. Particles can be selected from C_{ext} and from V in several ways. In the present research the following schemes have been used: random selection, first come-first out selection, and last come-first out selection. Those three selection possibilities make a total of nine alternatives for forming the local advancing front in Pseudocode 2. The best results with respect to high volume fraction have been obtained with random selection in both lists C_{ext} and V .

3.4 – Construction of a particle in outer contact with others

To achieve a higher local density in advancing front packing algorithms, each particle should be placed in contact with other two (2D case) or three (3D case) particles [7]. Another essential aspect of Pseudocode 2 is its step 2.2.1, at which a particle is constructed in outer contact with other particles. Several methods for constructing a particle in outer contact with other two (in 2D) or three (in 3D) particles are presented in this section, as well as their applications to some geometrical shapes.

Let $p[\mathbf{x}] \subseteq \mathbb{R}^n$ be a particle whose center of mass is point $\mathbf{x} \in \mathbb{R}^n$. A rigorous formulation of the problem to which this section is dedicated is the following:

Problem 1

Let $p_1, \dots, p_n$ be n fixed particles in $\mathbb{R}^n$ ($n \in \{2,3\}$), and let $p_{mob}[\mathbf{x}]$ be another particle whose center of mass is the point $\mathbf{x} \in \mathbb{R}^n$. Translate $\mathbf{x}$ without rotating $p_{mob}[\mathbf{x}]$ in such a way that $p_{mob}[\mathbf{x}]$ is in outer contact with $p_1, \dots, p_n$.

It has been verified in practice that in the general case, *Problem 1* has at most two solutions when particles $p_1, \dots, p_n$ and p_{mob} are convex and are close enough to each other (Figure 5 (a)). This can degenerate to only one solution when the mobile particle fits exactly in the gap between the fixed particles (Figure 5 (b)). Obviously, there is no solution when $p_1, \dots, p_n$ are apart from each other by a distance greater than the larger dimension of the particle to be placed (Figure 5 (c)). From now on, the hypothesis that the number of solutions always behaves according to the last three cases for convex particles, will be called *Conjecture 1*.

Figure 5. Number of solutions for the problem of placing a particle in contact with others

Several techniques for solving *Problem 1* have been developed [29, 30]. Such techniques can be classified into exact methods and approximate methods. With the exact methods, the mobile particle is constructed in outer contact with all of the fixed particles, while with the approximate methods the mobile particle is constructed in outer contact with at least one of the fixed particles, and close to the other fixed particles. There is also the option of approximating particles by using more convenient representations, before applying the methods for solving *Problem 1*.

Approximate methods produce packings with a smaller volume fraction, but also have a lower computational cost. Exact methods include *wrappers intersection* and *potential minimization*, while approximate methods include *direct approach* and *binary approach*. The *wrappers intersection* method already existed for some shapes such as disks [7], spheres [31], ellipses and polygons [6]. Apparently it did not exist for polyhedra, and it is presented in this section. The

authors of this paper also applied this method to clusters of disks or spheres [29]. The *potential minimization method* apparently did not exist for any type of particle. This new method is formulated in this section and its application to spherocylinders is shown. The potential minimization method is completely general and its application only requires knowing the corresponding potential function.

Also, the methods of *direct approach* and *binary approach* apparently did not exist either. They do not depend on the type of particle, and may be a valid alternative when the exact methods demand a high computational cost. The authors have applied the *binary approach* to polyhedra, superquadrics and spherocylinders, and the *direct approach* to clusters of disks or spheres [32]. To apply the *direct approach*, it is necessary to know how to calculate a directional distance between particles, and to apply the *binary approach* it is enough to have a function for detecting intersections between particles.

3.4.1 – Wrappers intersection method

Let p_{fix} be a fixed particle and $p_{mob}[\mathbf{c}]$ be a mobile particle of center $\mathbf{c}$. The locus defined by all points $\mathbf{c}$ such that p_{fix} and $p_{mob}[\mathbf{c}]$ are in outer contact, will be called *wrapper*.

In two dimensions, if the fixed and mobile particles are circles with radii equal to r_{fix} and r_{mob} respectively, then the corresponding wrapper is obviously a circle with radius $r_{fix} + r_{mob}$. When the two particles are described by polygons, the wrapper is a polygon with twice the number of sides of the fixed one. In the case of mixed particles, a circle and a polygon, the resulting wrapper is a polygon with rounded corners. Similar geometries are generated in the 3D case. Other authors have only applied the method of wrappers intersection to circles, polygons, ellipses and spheres [6]. However, we have shown that it can also be seen applied to other shapes including clusters of circles or spheres, polyhedra and mixes of circles and polygons, as can be seen in [29] in more detail.

The method of wrappers intersection in $\mathbb{R}^n$, for translating a mobile particle p_{mob} in such a way that it is in outer contact with other fixed particles $p_1, \dots, p_n$, without overlapping, consists of finding the loci described by the center of mass of p_{mob} when sliding this particle around each of the fixed particles, then finding the intersection of these loci, and finally centering p_{mob} in these intersections. For convex particles in 2D, the number of these intersections should be, according to *Conjecture 1*, equal to two in the general case, equal to one in the degenerate case, or equal to zero in the case of no solution.

3.4.2 – Potential minimization method

The potential minimization method [30] uses an optimization approach to solve the problem of placing a particle in contact with others. As far as we know, this method has never been applied by other authors to obtain packings of particles. In some cases, it can be easier to apply than wrappers intersection because it only calls for knowing a continuous function $\omega(p_1, p_2)$ defined over pairs of particles (p_1, p_2) such that:

$$\omega(p_1, p_2) \begin{cases} \leq 0 & \text{if } p_1 \cap p_2 \neq \emptyset \\ > 0 & \text{otherwise} \end{cases} \quad (1)$$

Function $\omega(p_1, p_2)$ is a measure of the gap between the surfaces of two particles. Condition (1) implies that p_1 and p_2 are in outer contact without overlapping if and only if $\omega(p_1, p_2) = 0$. Function ω is usually not unique.

Once the gap function ω has been chosen, the solution to the problem of placing a particle in contact with others can be obtained by solving the following optimization problem:

$$\begin{aligned} & \text{minimize} \quad |\omega(p_1, p_{mob}[\mathbf{x}])| + \dots + |\omega(p_n, p_{mob}[\mathbf{x}])| \\ & \text{subject to} \quad \mathbf{x} \in \mathbb{R}^n \end{aligned} \quad (2)$$

3.4.3 – Binary approach and direct approach methods

The binary approach and the direct approach are approximate methods to build a particle in outer contact with others. They can be used when the other methods available are very costly in terms of computational time. However, their use implies a decrease in the volume fraction of the packings obtained. As far as we know, from these two approach methods, only the direct approach for the case of spheres has been used in some way [12], obtaining high density packings. Some definitions are necessary as follows:

$\mathbf{c}_1, \dots$, centers of particles $p_1, \dots, p_n$ respectively, defined at *Problem 1*.

$\mathbf{c}_n$:

$r_1, \dots, r_n$: circumscribed radii of $p_1, \dots, p_n$ respectively.

r_{mob} : circumscribed radii of particle p_{mob} .

$\mathbf{c}_{med} = (\mathbf{c}_1 + \dots + \mathbf{c}_n)/n$: arithmetic mean of $\mathbf{c}_1, \dots, \mathbf{c}_n$.

r : line that passes by $\mathbf{c}_{med}$ and is perpendicular to segment $\overline{\mathbf{c}_1\mathbf{c}_2}$ (triangle $\mathbf{c}_1\mathbf{c}_2\mathbf{c}_3$) for $n = 2$ ($n = 3$), being $\mathbf{n}$ the space dimension

$\mathbf{n}_0$: versor parallel to r .

$r_{max} = \max\{r_{mob}, r_1, \dots, r_n\}$: maximum circumscribed radii of $p_{mob}, p_1, \dots, p_n$.

$\mathbf{c}_0 = \mathbf{c}_{med} + 2r_{max}\mathbf{n}_0$: point lying on r , whose definition implies that $p_{mob}[\mathbf{c}_0] \cap p_i = \emptyset, i = \overline{1,3}$.

Let also f be a function defined as:

$$\begin{aligned} & f: [0, 1] \rightarrow [-1, 1] \\ & \lambda \mapsto \begin{cases} -1, & \text{if } p_{mob}[\mathbf{c}_0 + \lambda(\mathbf{c}_{med} - \mathbf{c}_0)] \cap (p_1 \cup \dots \cup p_n) = \emptyset \\ 1, & \text{in another case} \end{cases} \end{aligned} \quad (3)$$

which is an indicator of the intersection of $p_{mob}[\mathbf{c}_0 + \lambda(\mathbf{c}_{med} - \mathbf{c}_0)]$ with the fixed particles $p_1, \dots, p_n$. For the previous definition of function f , the standard notation $f: A \rightarrow B$, meaning that f is defined in A and taking values on B , was used.

The *approach method* in order to build a mobile particle p_{mob} in contact with n fixed particles $p_1, \dots, p_n$, consists of finding a value $\lambda_0 = \sup\{\lambda \in [0, 1]: f(\lambda) = -1\}$, where f is given by (3), and then to center particle p_{mob} at point $\mathbf{c}_4 = \mathbf{c}_0 + \lambda_0(\mathbf{c}_{med} - \mathbf{c}_0)$. In this method, the obtained

particle $p_{mob}[c_4]$ will be in contact with some of the p_i , $i = \overline{1, n}$, and will not overlap with any of them, but will be close to all of the p_i . The number of solutions will be two in the general case, because there are two possibilities for defining n_0 . From a geometrical point of view, it can be seen that $p_{mob}[c_4]$ is obtained by moving $p_{mob}[c_0]$ along the line r until it makes contact with some of the p_i , and that is why the method is called *particle approach*. To illustrate the idea graphically, it is convenient to observe the 2D case (Figure 6). In this approach method in 2D, particle p_{mob} is first centered at c_0 , and then moved along line r until its center coincides with point c_4 . At this position in the example of Figure 6, particle p_{mob} is in contact with p_2 and does not overlap with p_1 .

One way to calculate λ_0 is through a bisection process, completely analogous to the case of finding a root of a function in an interval by making successive bisections [33]). In practice, good results are obtained by assuming that f is a monotonic increasing function (it is always monotonic increasing at interval $[0, \lambda_0]$). When λ_0 is found this way, the *binary approach method* is being used [28].

Figure 6. Particle approach method in 2D

Given its generality, the *binary approach method* can be applied to any type of particles for which there is a given function to detect overlaps. Another particular case of the *particle approach method* is the *direct approach method*, at which it is possible to place p_{mob} in its final position with only one iteration, without having to make a bisection process. In order to apply the latter method, it is sufficient to have a function $g(p_{mob}, p_{fix}, \mathbf{d})$ that is called *directional distance function* between two particles p_{mob} and p_{fix} , mobile and fixed respectively, with respect to a vector $\mathbf{d} \in \mathbb{R}^n$, assuming that $p_{mob}[c_0] \cap p_{fix} = \emptyset$. This function can be defined by the expression:

$$g(p_{mob}, p_{fix}, \mathbf{d}) = \sup\{\lambda \geq 0 : p_{mob}[c_0 + \lambda \mathbf{d}] \cap p_{fix} = \emptyset\} \quad (4)$$

which is easy to calculate for circles, spheres or clusters of them. Here will only be given details for circles and spheres, because the extension to clusters does not require many modifications. A formulation of directional distance for convex polyhedra is also available [34]. A way to find the directional distance for circles and spheres is explained in the remainder of this section. Despite all the explanations are given for circles, they also hold for spheres without making any major

changes. Examples of packings generated with the directional distance approach can be seen in Figure 7.

Figure 7. a) Packing of polyhedral particles, b) Packing of real grains from Daytona beach

Let p_{mob} and p_{fix} two circles of centers $\mathbf{c}_0$ and $\mathbf{c}_{fix}$, and radii r_{mob} and r_{fix} respectively, and $\mathbf{d} \in \mathbb{R}^2$. Let r be the ray with origin in $\mathbf{c}_0$ and the same direction as $\mathbf{d}$. Then, in this case one has

$$\lambda_0 = g(p_{mob}, p_{fix}, \mathbf{d}) = \arg \min_{\lambda \geq 0} \left| \|\mathbf{c}_0 + \lambda \mathbf{d} - \mathbf{c}_{fix}\| - (r_{mob} + r_{fix}) \right| \quad (5)$$

From all circles centered at r , the one with center equal to $\mathbf{c}_4 = \mathbf{c}_0 + \lambda_0 \mathbf{d}$ is the closest to p_{fix} without overlapping with it (Figure 8). The value of λ_0 can be calculated by the following expression:

$$\lambda_0 = \begin{cases} \max \left\{ 0, \frac{-b - \sqrt{D}}{2a} \right\}, & \text{if } D > 0 \\ \max \left\{ 0, -\frac{b}{2a} \right\}, & \text{otherwise} \end{cases} \quad (6)$$

where a , b , c and D are given by the following expressions:

$$a = \|\mathbf{d}\|^2 \quad (7)$$

$$b = 2\mathbf{d} \bullet (\mathbf{c}_0 - \mathbf{c}_{fix}) \quad (8)$$

$$c = \|\mathbf{c}_0 - \mathbf{c}_{fix}\|^2 - (r_{mob} + r_{fix})^2 \quad (9)$$

$$D = b^2 - 4ac \quad (10)$$

The value of λ_0 in formula (5) is also valid when p_{mob} and p_{fix} are spheres. It is equal to the value $\lambda_0 = \frac{-b - \sqrt{D}}{2a}$ for the case of top left Figure 8, equal to the value $\lambda_0 = -\frac{b}{2a}$ for the case of top right Figure 8, and equal to $\lambda_0 = 0$ for both cases of lower Figure 8.

Figure 8. Directional distance between two circles or spheres

4.0 –Algorithm for modeling polycrystal structures

Polycrystal media can be modeled with Laguerre diagrams, which are a modification of Voronoi diagrams. In this case, the Voronoi and Laguerre diagrams are obtained from sphere packings, which can be generated with the methods explained in previous sections of this paper. The process for modeling polycrystals is described in the following Pseudocode 3 and in Figure 9:

- Step 1)** Acquisition of microscopical images
Step 2) Image segmentation for detecting grains
Step 3) Definition of circles or spheres inscribed in the grains
Step 4) Estimation of the statistical distribution D of the radii of the circles or spheres in the previous step
Step 5) Generation of a packing of circles or spheres whose radii follow the distribution D
Step 6) Generation of a Laguerre diagram corresponding to the packing of the previous step, whose centers are the centers of the circles or spheres and whose weights are the radii of the circles or spheres

Pseudocode 3. Virtual modeling of polycrystal structures

In Pseudocode 3, steps 1 to 3 are carried out with image processing techniques. For step 4, it is sufficient with the histogram of the radii. Step 5 can be carried out with the packing generation methods previously explained in this paper, and step 6 consists of first calculating a Voronoi diagram whose centers are the centers of the spheres of the packing obtained in step 5, and then modifying such diagram by relocating its vertices according to the spheres' radii, in such a way that the cells of the diagram contain the spheres. This only requires solving small systems of linear equations. More formal details of step 6 are given in the following paragraph.

Figure 9. Example of steps 5 and 6 of Pseudocode 3 for the virtual generation of a polycrystal structure

Let $E = (X, d)$ be a metric space defined by set X and the metric d ; and set $P = \{p_1, p_2, \dots, p_N\} \subseteq X$. The Voronoi cell corresponding to point p_i is defined as $C_i = \{x \in X: d(x, p_i) < d(x, p_j) \forall j \neq i\}$, with $i = \overline{1, N}$. The set of cells C_i , $i = \overline{1, N}$, is called *Voronoi Diagram* associated to set P . In case $X = \mathbb{R}^2(\mathbb{R}^3)$ and d is the usual metric, the cells are polygons (polyhedra), and their vertexes are called *Voronoi vertexes*. As previously mentioned, a Voronoi diagram is obtained from spheres' centers, and such a diagram must be modified so that the cells contain the spheres. To see how this modification can be done in the $\mathbb{R}^3$ case, let $\mathbf{c}$ be a vertex of the Voronoi diagram to be modified, and let $\mathbf{c}_i$, $i = \overline{1, 4}$, be the centers of the four Voronoi cells having the common vertex $\mathbf{c}$, which are also sphere's centers of the sphere packing obtained with Pseudocode 2. Let r_i be the radius of the sphere centered at $\mathbf{c}_i$, $i = \overline{1, 4}$. Then the position of $\mathbf{c}$ can be recalculated in such a way that it simultaneously satisfies the following equalities [11]:

$$d(\mathbf{c}, \mathbf{c}_1)^2 - r_1^2 = d(\mathbf{c}, \mathbf{c}_2)^2 - r_2^2 = d(\mathbf{c}, \mathbf{c}_3)^2 - r_3^2 = d(\mathbf{c}, \mathbf{c}_4)^2 - r_4^2 \quad (11)$$

which can be reduced to a system of three linear equations with three unknowns. An example of the application of the equalities (11), for the analogous case of circles, can be seen in Figure 9 (b) and (c).

5.0 – Applications of the particle packing algorithm

In this section, some practical applications of the packing algorithm are described. The most basic applications include simple geometries with large numbers of particles, showing the temporal performance of the algorithm with one of them. Other basic applications obtained with the optional step 4 of Pseudocode 2, are the generation of cylindrical geometries with a controlled (high) porosity, and the mix of two very different sized distribution of spheres representing particles embedded into a resin. The reproduction of media consisting of glass spheres or sand particles are also presented. Some more complicated applications such as the

modeling of the granular microstructure of steels and rocks are also given. Finally, the use of the packing algorithm in several different fields such as bioengineering, mechanical and civil engineering, and the pharmaceutical and meals industry, is presented in the last subsection, in which complex meshes filled with large numbers of particles are shown.

5.1 – Basic applications of the packing algorithm

For the basic applications, a cylindrical geometry was used to contain the spheres of the packing, whose radii follow a normal distribution (Figure 10). The data of the bounding geometry and the particles radii distribution can be seen in Table 1, and the main descriptors of the packing can be seen in Table 2.

Bounding geometry	Particle radii		Volume
	Distribution	Interval	
<i>Cylinder</i>	$N(\mu = 0.3, \sigma^2 = 0.03)$	$[0.2, 0.50]$	$5301.19 u^3$

Table 1. Data used for the packing generation.

No. of particles	Volume fraction	No. of coordination	Generation time
25400	0.5439	5	44.293 s

Table 2. Results of the packing process.

Figure 10. Bounding geometry (left) and particles of the packing (right)

The volume fraction and the coordination number in the packing obtained are high, and the time consumed to generate the packing is relatively short. Another example was generated to observe the temporal performance of the algorithm with a higher number of particles, a few million in this case. The spheres' radii now follow a uniform distribution. The data of the bounding geometry and the particle radii distribution can be seen in Table 3.

Bounding geometry	Particle radii		Volume
	Distribution	Interval	
<i>Cube</i>	$U[0.8, 1.2]$	$[0.8, 1.2]$	$12.5 * 10^7 u^3$

Table 3. Data used for the packing generation

The number of particles in this other packing was about 16 million, obtained in about two and a half hours on a conventional desktop PC. The main results are in Table 4. A curve which shows the time elapsed until the placement of each particle can be seen in Figure 11. The curve in this figure is apparently linear, in agreement theoretical results [8].

No. of particles	Volume fraction	Coordination number	Generation time
15 864 150	0.5503	5	8067

Table 4. Results of the generation of 16×10^6 particles

Figure 11. Number of particles vs. time during the execution of the algorithm, with linear regression model

5.2 – Basic applications to cases of high porosity (low volume fraction)

In the generation of packings with a high porosity, or equivalently, a low volume fraction, the optional step 4 of Pseudocode is applied. In the present example, the target is to obtain a packing with a volume fraction lower than 6%. It was achieved by first obtaining a packing with a high volume fraction, and then reducing the particle's radii or eliminating spheres when necessary. Some examples can be seen in Figure 12.

Figure 12. Generation of packings with low volume fraction (5%, 2.3% and 1.2 % from left to right respectively)

5.3 – Basic application from real grain size distribution curves

Pseudocode 2 is independent with respect to the particle size distribution. In this section, two examples of packings whose particles size distribution come from a grain size distribution curve, are presented. The first example is the filling of a silo with spheres (Figure 14) whose radii come from the grain size distribution curve of Figure 13. The input data is listed in Table 5, and the summary of results is shown in Table 6.

Figure 13. Grain size distribution curve used to generate the spheres' radii

Geometry	Particles radii		Volume
	Distribution	Interval	
<i>Silo</i>	grain size distribution curve	[31.5,100]	1.87844 u^3

Table 5. Data used for the packing generation.

No. of particles	Volume fraction	Coordination number	Generation time
26343	0.442	5	180.907s

Table 6. Results of the packing process.

Figure 14. Bounding geometry (left) and packing (right)

The second example of this section describes the generation of a packing consisting of a mix of particles that come from two different grain size distribution curves. The packing was used to model an adhesive foil formed by spherical particles that were in suspension in a resin (Figure 15 (a)). The foil has dimensions 1.5 x 1.0 x 0.1 cm. The proportion of the two sets of spheres in the foil is 84% for the first type of spheres and 16% for the second type (particle size distributions can be seen in Figure 15 (b)). The proportions obtained in the packing can be seen in Table 7. The two examples of this section show that, with Pseudocode 2, it is possible to generate packings of particles whose sizes are defined by different grain size distribution curves.

Set of particles	No. of particles	Proportion
Set of particles 1	76016	16.3 %
Set of particles 2	389279	83.7 %
Total	465295	100 %

Table 7. Proportions of the mix obtained with the packing algorithm.

Figure 15. Result of virtual generation. a) Geometrical generation of adhesive material comprised by two sets of particles. Left: mix of the two sets of particles. Right: second set of particles. b) Original radii size distribution (upper right) and virtual (upper left) radii size distribution of particle set 1 (16.3% of the total number of particles); and original radii size distribution (lower right) and virtual (lower left) radii size distribution of particle set 2 (83.7%)

5.4 – Reproduction of artificial and natural materials

The reproduction of sets of glass spheres and sands is presented in this section. In some of the cases, the input were images obtained from Computerized Axial Tomographies (CAT).

The study of the behavior of a material consisting of micro glass spheres is presented in [35]. Figure 16 shows images of the micro spheres, obtained with resolutions of 200X and 400X. The equal size of the spheres and their shape can be seen in such figure.

Figure 16. Micro glass spheres [35]

The samples were prepared by first pouring the spheres into a hollow cylinder, and then compacting them with vibration and a weight above them [35]. The samples of AA type microspheres, which have a mean diameter of 0.45 mm, had a high density. On the other hand, for the samples of AH type microspheres, which have a mean diameter of 0.063 mm, it was not possible to obtain high densities [35]. Table 8 shows the internal variable *density* (ρ_0) in the initial conditions of the sample.

Both sample types AA and AH were virtually reproduced with 30 sphere packings each, and the virtual packings were compared with the experimental results, in order to validate the packing algorithms. For both AA and AH sphere types, the descriptive statistics for initial pore index (e_0) are shown in Table 9.

Statistical parameters	Microspheres AA		Microspheres AH	
	e_0	ρ_0	e_0	ρ_0
Arithmetic mean	0,672	1,256	0,797	0,591
Typical error	0,018	0,019	0,021	0,020
Median	0,660	1,270	0,790	0,600
Standard deviation	0,040	0,043	0,055	0,052
Variance	0,002	0,002	0,003	0,003
95,0% confidence level (95%)	$\pm 0,049$	$\pm 0,053$	$\pm 0,051$	$\pm 0,048$

Table 8. Descriptive statistics of the samples' initial parameters

Figure 17. Cumulative percentage vs size curves for the AA and AH type microspheres, including real and virtual cases. Horizontal axis has logarithmic units and vertical axis is in percentage.

Statistical parameters	Microspheres AA	Microspheres AH
Arithmetic mean	0.652	0.784
Typical error	0.002	0.001
Median	0.648	0.783
Standard deviation	0.012	0.006
Variance	0.00014	0.00003
95,0% confidence level (95%)	± 0.004	± 0.002

Table 9. Descriptive statistics of the samples' initial pore index e_0 .

Parametric and non-parametric statistical tests [36] showed that there are no significant differences of pore index values between the experimental samples and the virtual packings. This is an evidence that the packing algorithms presented in this paper are useful for virtually reproducing physical media. The distributions of particle sizes in the virtual packings also match the analogous real distributions, as can be seen in Figure 17 and Tables 10 and 11. The errors appearing in those tables are the Mean Absolute Error (MAE), the Mean Squared Error (MSE), Relative Mean Absolute Error (RMAE) and the Mean Relative Error (MRE).

Point number	Sphere diameter	Experimental cumulative percentage	Virtual cumulative percentage	Error	Relative error (%)	Quadratic error	Absolute error	Absolute relative error (%)
1	0.14	0.3	0.305	-0.005	-1.67	0.000	0.005	1.67
2	0.176	1.94	1.86877	0.071	3.67	0.005	0.071	3.67
3	0.208	2.26	2.33738	-0.077	-3.42	0.006	0.077	3.42
4	0.25	3.01	4.40837	-0.498	-12.75	0.248	0.498	12.75
5	0.29	10.94	11.7401	-0.800	-7.31	0.640	0.800	7.31
6	0.4224	39.7	40.9511	-1.251	-3.15	1.565	1.251	3.15
7	0.591	96.13	96.0188	0.111	0.12	0.012	0.111	0.12
8	0.717	99.83	99.8178	0.012	0.01	0.000	0.012	0.01
9	0.857	100	100	0.000	0.00	0.000	0.000	0.00
				SUM	-24.50	2.477	2.827	32.10

	MAE	0.40	MSE	0.35	RMAE	4.59%	MRE	-3.50%

Table 10. Errors between the experimental and virtual size curves for the AA microspheres

Porous media can be studied with CAT images, and two more materials were studied with this technique: another artificial material consisting of micro glass spheres and the natural material which is pure sand [37]. The plane images were obtained with a fourth generation X-ray scanner, model Skyscan 1172 (Table 12). The configuration used for scanning the images can be seen in Table 13. Such images are binary in black and white, where the white pixels represent the particles and the black ones represent the empty space. For each image, the Binary Occupation Ratio (BOR) is calculated, which is defined as the ratio of black pixels to the total number of pixels as in the following formula:

$$BOR = \frac{p}{p + b} \quad (12),$$

where p and b represent the number of black and white pixels respectively.

Point number	Sphere diameter	Experimental cumulative percentage	Virtual cumulative percentage	Error	Relative error (%)	Quadratic error	Absolute error	Absolute relative error (%)
1	0.0094	0.980	0.980272	0.000272	-0.03	0.000	0.000	0.03
2	0.013	2.580	2.591000	0.011000	-0.45	0.000	0.012	0.45
3	0.018	2.740	2.89303	0.15303	-5.59	0.023	0.153	5.59
4	0.0256	3.870	4.24357	0.37357	-9.70	0.141	0.376	9.70
5	0.0294	5.160	5.34449	0.18449	-3.58	0.034	0.184	3.58
6	0.036	6.450	6.64500	0.19500	-3.04	0.038	0.196	3.04
7	0.042	7.740	7.91426	0.17426	-2.25	0.030	0.174	2.25
8	0.05	20.520	22.41333	1.89333	-10.30	4.382	2.093	10.30
9	0.056	30.390	35.6489	5.2589	-6.77	5.103	2.259	6.77
10	0.064	55.720	54.28313	-1.43687	-1.05	0.317	0.563	1.05
11	0.085	71.720	72.06208	0.34208	-0.48	0.117	0.342	0.48
12	0.104	83.760	83.8994	0.1394	-0.14	0.014	0.119	0.14
13	0.140	100.000	100	0.000	0.00	0.000	0.000	0.00
				SUM	-41.70%	9.752	5.447	41.70
	MAE	0.78	MSE	1.39	RMAE	5.96%	MRE	-5.96%

Table 11. Errors between the experimental and virtual size curves for the AH microspheres

Many images were obtained from the generated samples, but only 5 images per sample were selected for the present study (Figure 18). The virtual reproduction of the samples was carried out once the experimental study in [37] was finished. For such virtual reproduction, the packing

algorithm described in section 3 was used. The algorithm generated spheres for the case of micro glass spheres, and for the case of sand particles it generated irregular particles calculated with shape descriptors based on Fourier series [38].

Feature	Description
Detail detectability	1 μ m
Pixel size (maximum magnification)	<0.8 μ m
X ray source	10000 h lifetime, 20-100 kV, 0-250 μ A (10 W max.)
X ray Detector	11Mp X-ray camera (4000 $\times$ 2300) with 12 Bits CCD
Maximum object size	68 mm diameter (with offset camera)
Reconstruction algorithm	Conic – volumetric (Fidkamp)
Radiation level	< 1 μ Sv/h at any point of the surface

Table 12. Skyscan 1172 technical specifications

Technical data	Scanning process				
	0.5 mm glass spheres	1 mm glass spheres	Sand		
Distance from source to object	57.20	57.20	186.71	57.20	57.20
Distance from camera to source	217.87	217.87	282.88	217.87	217.87
Camera type	2x2	2x2	2x2	2x2	2x2
Pixel size	5.9400	5.9400	14.9300	5.9400	5.9400
Acquisition time	05:03:20	05:01:34	07:09:25	05:07:10	02:43:54
	Reconstruction process				
Slices	771	771	811	851	801
Image width	2000	2000	2000	2000	2000
Image height	2000	2000	2000	2000	2000
Pixel size	5.9402	5.9402	14.9323	5.9402	5.9402
Reconstruction time	01:05:51	01:03:49	01:11:02	01:08:36	01:08:22

Table 13. Technical data used in the acquisition and reconstruction of the scanned images

Figure 18. Position of images analyzed in each sample

Like in the case of microspheres AA and AH, the packing algorithm output was compared with the real samples using the same statistical tests. Such tests suggest that there is no significant difference between the *BOR* of the generated packings and the one of the real samples of [37], except for the case of the 1mm glass spheres, since in that case, the *BOR* value of slice C5 (value with asterisk in Table 14) is an outlier, according to Dixon's Q test [36].

Sample	Slice	<i>BOR</i> (image)	Porosity (packing)
Glass 0.5 mm. Figure 19 (a)	C1	0.455	0.454
	C2	0.458	0.461
	C3	0.457	0.456
	C4	0.446	0.460
	C5	0.461	0.461
	Mean	0.455	0.458
Glass 1 mm. Figure 19 (b)	C1	0.453	0.489
	C2	0.444	0.473
	C3	0.457	0.501
	C4	0.442	0.473
	C5	0.295*	0.490
	Mean	0.434	0.485
Sand. Figure 19 (c)	C1	0.461	0.454
	C2	0.439	0.466
	C3	0.435	0.464
	C4	0.429	0.458
	C5	0.495	0.460
	Mean	0.454	0.460

Table 14. Values of *BOR* and porosity. The *BOR* value with asterisk is an outlier

Figure 19. Planar sections. (a) 0.5 mm glass spheres (b) 1 mm glass spheres (c) sand

5.5 – Reproduction of polycrystal structure of real materials

Two different metals and a geomaterial are studied in this section. For the case of metals, Table 15 contains the distribution of the microstructural grain size of the 304L stainless steel and the 30XTC steel, as well as the size of their Representative Volume Element (RVE). With this input data, 30 packings and their corresponding Laguerre diagrams were generated for each steel. Figures 19 and 20 (left) show only the first of the 30 packings and diagrams, since the other 29 are very similar. The right part of such figures shows the cell volume of all the Laguerre diagrams.

Material	Particle size		RVE size
	Distribution	Radii interval	
304L stainless steel	$N(\mu=0.115, \sigma^2=0.039)$	[0.02 μm , 0.21 μm]	9 μm
30XTC steel	$N(\mu=47.50, \sigma^2=4.850)$	[35 μm , 60 μm]	1000 μm

Table 15. Input data for the generation of polycrystal structure of steels

Figure 20. RVE of 304L stainless steel. Left: packing of spheres and corresponding Laguerre diagram of the first of the 30 packings. Right: mean cell volume of the Laguerre cells for the 30 packings.

The advantage of generating Laguerre (i.e. modified Voronoi) cells based on a set of spheres instead of the standard procedure based on a set of randomly generated points, relies on the fact that the cell sizes are strongly correlated to the spheres' radii, which can follow any prescribed distribution, which can be for example continuous (i.e. normal, uniform, etc.) or discrete (i.e. constant, bimodal, multimodal, etc.).

Another of the studied materials is a sedimentary rock called sandstone. The available data are planar images obtained with a high resolution microscope. From the segmentation of such images, a histogram for the microstructural grain sizes was obtained. Table 16 shows that from the considered distributions, the Weibull distribution is the one that best fits such histogram.

After knowing the statistical distribution that follow the grains of the sandstone's microstructure, it is possible to virtually generate an RVE that models such microstructure. Such generation was carried out with the packing algorithm of section 3 and the algorithm of section 4. The results for the 2D (3D) case can be seen in Figure 21 (Figure 22).

Probability distribution	p-value
Cauchy	0.0045
Chi-cuadrado	0.0
Exponencial	4.31E-7
Gamma	0.02131
Normal	0.0043973
Uniforme	0.0
Weibull	0.26007

Table 16. Kolmogorov-Smirnov goodness of fit test results for selecting a continuous distribution that fits the grain size histogram

Figure 21. Planar sections of sandstone. Left: real microstructure. Right: virtual microstructure

Figure 22. Virtual model of sandstone's RVE

5.6 – Practical engineering applications

The packing algorithm of section 3 has also been validated with other engineering applications having complicated meshes and large numbers of particles. The first application is in biomechanical engineering, with the filling of meshes of a human skull and a human hip (Figure 23). Those packings have 1101921 and 1349748 spheres respectively.

Figure 23. Biomechanical applications of the packing algorithms. Top left: skull mesh. Top right: skull sphere packing. Bottom left: hip mesh. Bottom right: hip sphere packing

For the case of structural engineering, the packing generation of a building with spheres is presented (Figure 24 upper center). Such type of packings can be used not only for structural calculations, but also for studying the formation and propagation of fractures, which is a phenomenon very difficult to model with traditional numerical methods such as the Finite Element Method. The packing of the building has a volume fraction of 42%, and is composed by 1383973 spheres whose radii are normally distributed in the interval $[0.2, 0.4]$. A packing filling the same building mesh, but whose particles are clusters of spheres, was also obtained (Figure 24 upper right).

Another application in structural engineering is the packing of a bridge with spheres (Figure 24 center). In this case, the input for the algorithm was a normal distribution for the spheres' radii, bounded in the interval $[0.2, 0.4]$. The packing comprised 6399691 spheres and had a density of 53%. The same bridge mesh was also filled with clusters of spheres (Figure 24 lower right).

Figure 24. Civil Engineering applications. Top: mesh and sphere packing of building. Bottom: mesh and sphere packing of bridge.

An example application in Mechanical Engineering is also presented. It consists in a mechanical tool filled with 3778952 spheres (Figure 25 center). The set of spheres have a volume fraction of 54%, and their radii are uniformly distributed in the continuous interval $[0.2, 0.4]$. An analogous packing with clusters of spheres was also obtained for this case (Figure 25 right).

Figure 25. Mechanical tool. Left: surface mesh. Center: sphere packing. Right: packing of clusters of two spheres.

In the previous examples of this section, the generations with clusters of two spheres were obtained with the following data of Table 17:

Geometry	Number of particles	Volume fraction	Generation time
Mechanical tool	2 511 551	0.44	2919.9 sec. (49 min)
Building	4 001 337	0.42	9941.66 sec. (2h 45 min)

Bridge	9 771 672	0.43	41633.6 sec. (11h 33 min)
--------	-----------	------	---------------------------

Table 17. Data of the generations with clusters of two spheres

Two final applications of the particle packing generation algorithm presented here are shown below, being the first one the modeling of medical spherocylindrical tablets within a container (Figure 26).

Figure 26. Container filled with pharmaceutical spherocylindrical tablets

The last engineering application is shown in Figure 27, with two different grain configurations of seeds: soy and corn, modeled with clusters of spheres, stored inside silos. Despite the importance of this type of storage for preserving grains from plagues, the granular flow in silos is a phenomenon not completely understood yet.

Figure 27. Silos filled with corn (a) and soy (b) grains represented by clusters of spheres

6 - Analysis of particle system fabric

The topology of a system of particles and contacts is highly complex, but its analysis is important in order to extract valuable information from a packing or a DEM simulation. The

fabric tensor is a descriptor that can be useful in such analysis. This and other particle system fabric analysis techniques are explored in depth in [9].

In tensorial notation the second-order fabric tensor (for contact orientation) is given as

$$\Phi_{ij} = \frac{1}{N_c} \sum_{k=1}^{N_c} n_i^k n_j^k \quad (13),$$

where n_i^k is the unit vector describing the contact normal orientation and the summation takes place over the N_c contacts in the system. For non-spherical particles, the normals n_i^k can be taken as the particles' orientations. The fabric tensor of a totally isotropic packing must have its eigenvalues equal to $1/3$, and the normals of the packing must be uniformly distributed over the unit sphere.

The contact or orientation normals were plotted in 3D histograms (Figure 28) for some of the packings presented in previous sections. The eigenvalues of the corresponding fabric tensors calculated with formula (13) can be found in Table 13, as well as the standard deviations of such eigenvalues. One has that the standard deviations are about 30 times bigger for the cases of the Daytona sand, the cylinder and the building (in bold face in Table 18), and this matches the fact that at least for the Daytona sand and the building, their normals' histograms are graphically more irregular than the others, as can be seen in Figure 28. The relatively flat shape of the histogram of Figure 28 (f) could be explained by the fact that there are many spheres that do not have contact normals close to the vertical direction, because the building has a relatively large flat surface.

Figure 28. 3D histograms of contact normals (b-f) or orientation normals (a) for some of the packings of previous sections. (a) Histogram corresponding to packing of grains of Daytona beach sand shown in Figure 7 (b) Histogram corresponding to packing of cylinder shown in Figure 10 (c) Histogram corresponding to packing of silo shown in Figure 14 (d) Histogram corresponding to packing of 304L steel shown in Figure 20 (e) Histogram corresponding to

packing of sandstone shown in Figure 22 (f) Histogram corresponding to packing of building shown in Figure 24

Packing	Fabric tensor eigenvalues	Standard Deviation
Daytona beach sand	0.354, 0.347, 0.299	0.0299866
Cylinder	0.355, 0.351, 0.294	0.0345451
Silo	0.340, 0.337, 0.323	0.00935274
304L steel	0.342, 0.333, 0.325	0.00828153
Sandstone	0.338, 0.333, 0.329	0.0046029
Building	0.359, 0.348, 0.293	0.0374539

Table 18. Normals' eigenvalues and their standard deviation for several packings

7 - Conclusions

A general algorithm for generating packings of particles is presented. In the algorithm, the geometrical part is emphasized, but optional dynamic steps for improving the final packing are also present. The algorithm is independent of the particles shapes and sizes, which can be generated with any method such as for example Monte Carlo. It is also independent with respect to the definition of the initial advancing front and the directions the advancing front grows too. The algorithm does not even depend on other features such as the method for constructing the new particles in contact with others, the neighbor search method and the pairwise intersection checks. Moreover, the non-optional steps included in the algorithm have a linear time with respect to the number of particles. This was proven analytically and validated with practical applications too, with packings of even the order of millions of particles.

Three solution methods are given for the problem of constructing the new particle in contact with others, which are wrappers intersection, potential minimization and direct or binary approach. The latter one is very general, since it allows using almost any type of particle for generating the packings.

The combination of packing algorithms with Laguerre diagrams allows to model the polycrystal structure of several materials. In this sense, contributions to the geometrical modeling of real materials are given, with applications to rocks and steels. Other real materials that were modeled with the packing algorithm alone are sets of micro glass spheres and also sands, being CAT images the starting point of the modeling process in this case.

Several more packings were generated in order to show the applicability of the packing algorithm in different fields of engineering. After obtaining some relatively simple packings, some highly porous media were reproduced, being one of them the combination of particles having two very different sized distributions. Moreover, examples of applicability in problems of biomedical, civil and mechanical engineering were presented.

8 - Acknowledgements

The authors are deeply grateful to the valuable funding, resources and support of the following institutions:

- Coordination for the Improvement of Higher Education Personnel - CAPES. Brazilian Ministry of Education. CAPES-MES Project No. 208/13.
- Brazilian National Research Council, Process No. 163622/2013-2.
- InfraLab. Faculty of Technology. University of Brasilia, Brasil.
- International Center for Numerical Methods in Engineering. CIMNE. Barcelona, Spain.

9 - References

1. Cundall, P.A., *Computer simulation of dense sphere assemblies*, in *Micro-mechanics of Granular Materials*, M. Satake and J.T. Jenkins, Editors. 1988, Elsevier. p. 113-123.
2. Cundall, P.A. and O.D.L. Strack, *A discrete numerical model for granular assemblies*. Geotechnique, 1979. **29**(1): p. 47-65.
3. Kuhn, M.R., *A smooth convex three-dimensional particle for the Discrete Element Method*. J. Eng. Mech., 2003. **129**(5): p. 539-547.
4. Ting, J.M., et al., *An ellipse-based discrete element model for granular materials*. International Journal for Numerical and Analytical Methods in Geomechanics, 1993. **17**: p. 603- 623.
5. Bagi, K., *An algorithm to generate random dense arrangements for discrete element simulations of granular assemblies*. Granular Matter, 2005. **7**: p. 31-43.
6. Feng, Y.T., K. Han, and D.R.J. Owen, *An advancing front packing of polygons, ellipses and spheres*, in *Discrete Element Methods. Numerical Modeling of Discontinua*. 2002, American Society of Civil Engineers. p. 23-90.
7. Feng, Y.T., K. Han, and D.R.J. Owen, *Filling domains with disks: an advancing front approach*. International Journal for Numerical Methods in Engineering, 2003. **56**: p. 699-713.
8. Roselló Valera, R., et al., *Modified algorithm for generating high volume fraction sphere packings*. 2015. **2**(2): p. 161-172.
9. O'Sullivan, C., *Particulate Discrete Element Modelling: A Geomechanics Perspective*. Applied Geotechnics. Vol. 4. 2011, London and New York: Spon Press. 574.
10. Wang, C.-Y., C.-F. Wang, and J. Sheng, *A packing generation scheme for the granular assemblies with 3d ellipsoidal particles*. International Journal for Numerical and Analytical Methods in Geomechanics, 1999. **23**: p. 815-828.
11. Benabbou, A., et al., *Numerical modeling of nanostructured materials*. Finite Elements in Analysis and Design. 2010. **46**(1-2): p. 165-180.
12. Han, K., Y.T. Feng, and D.R.J. Owen, *Sphere packing with a geometric based compression algorithm*. Powder Technology, 2005. **155**: p. 33-41.
13. Nandakumar, K., Y. Shu, and K.T. Chuang, *Predicting Geometrical Properties of Random Packed Beds from Computer Simulation*. AIChE Journal, 1999. **45**(11).
14. Sutou, A. and Y. Dai, *Global optimization approach to unequal sphere packing problems in 3D*. Journal of Optimization Theory and Applications, 2002. **114**(3): p. 671 - 694.
15. Cui, L. and C. O'Sullivan, *Analysis of a triangulation based approach for specimen generation for discrete element simulations*. Granular Matter, 2003. **5**: p. 135-145.
16. De la Cruz, F.V., *Discrete Element Group for Hazard Mitigation. Annual Report*. 2008: Grenoble.

17. Jerier, J.-F., et al., *Packing spherical discrete elements for large scale simulations*. Computer Methods in Applied Mechanics and Engineering, 2010. **192**(25-28): p. 1668-1676.
18. Georgalli, G.A. and M.A. Reuter, *A particle packing algorithm for packed beds with size distribution*. Granular Matter, 2008. **10**: p. 257-262.
19. Löhner, R. and E. Oñate, *An advancing front point generation technique*. Communications in Numerical Methods in Engineering, 1998. **14**(12): p. 1097-1108.
20. Löhner, R. and E. Oñate, *A general advancing front technique for filling space with arbitrary objects*. International Journal for Numerical Methods in Engineering, 2004. **61**: p. 1977-1991.
21. Löhner, R. and E. Oñate, *Advancing front techniques for filling space with arbitrary separated objects*. Finite Elements in Analysis and Design, 2010. **46**(1-2): p. 140-151.
22. Benabbou, A., et al., *Geometrical modeling of granular structures in two and three dimensions. Application to nanostructures*. International Journal for Numerical Methods in Engineering, 2009. **80**(4): p. 425-454.
23. Valera, R.R., et al., *Modified algorithm for generating high volume fraction sphere packings*. Computational Particle Mechanics, 2015. **2**(2): p. 161-172.
24. Saucier, R., *Computer Generation of Statistical Distributions*. 2000, ARMY RESEARCH LABORATORY.
25. Munjiza, A. and K.R.F. Andrews, *NBS contact detection algorithm for bodies of similar size*. International Journal for Numerical Methods in Engineering, 1998. **43**: p. 131-49.
26. Labra, C. and E. Oñate, *High-density sphere packing for discrete element method simulations*. COMMUNICATIONS IN NUMERICAL METHODS IN ENGINEERING, 2009.
27. Ferrez, J.-A., *Dynamic triangulations for efficient 3d simulation of granular materials*, in *Département de Mathématiques*. 2001, École Polytechnique Fédérale de Lausanne: Lausanne.
28. Pérez Morales, I., et al., *Procedimiento de empaquetamiento de partículas genéricas para el Método de Elementos Discretos*. Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería, 2009. **25**(2): p. 95-110.
29. Morales, I.P., et al., *Contributions to the generalization of advancing front particle packing algorithms*. International Journal for Numerical Methods in Engineering, 2016. **107**(12): p. 993-1006.
30. Pérez Morales, I., et al., *Dense packing of general-shaped particles using a minimization technique*. Computational Particle Mechanics, 2017. **4**(2): p. 165-179.
31. Hernández Ortega, J.A., *Simulación numérica de procesos de llenado mediante elementos discretos*. 2003.
32. Morfa, C.A.R., et al., *General advancing front packing algorithm for the discrete element method*. Computational Particle Mechanics.
33. Zarowski, C.M., *An Introduction to Numerical Analysis for Electrical and Computer Engineers*. 2004: Wiley and Sons.
34. Choi, Y.K., et al., *Computing the Minimum Directional Distance Between Two Convex Polyhedra*. 2006, Hong Kong University.
35. Zuluaga, R.G., *Relação entre características microestruturais e o comportamento macroscópico de solos granulares*. 2016, University of Brasilia.

36. Everitt, B.S. and A. Skrondal, *The Cambridge dictionary of statistics*. Fourth edition ed. 2010: Cambridge University Press.
37. Zubeldia, E.H., *Uso dos autômatos celulares bidimensionais e imagens tomográficas na geração de meios porosos artificiais*. 2013, University of Brasilia.
38. Garboczi, E.J., *Three-dimensional mathematical analysis of particle shape using X-ray tomography and spherical harmonics: Application to aggregates used in concrete*. Cement and Concrete Research, 2002. **32**(10): p. 1621-1638.