Precision and computational costs of Level Set-Discrete Element Method (LS-DEM) with respect to DEM

Jérôme Duriez, S. Bonelli

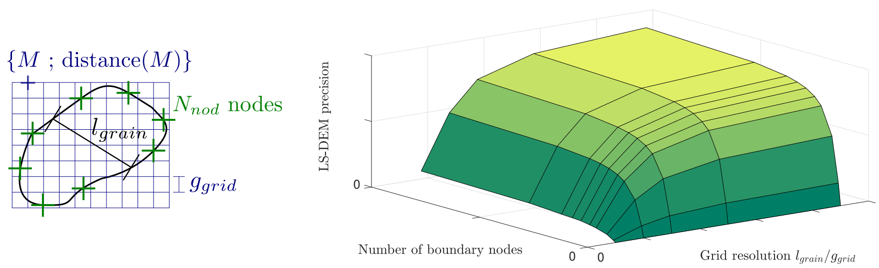
Precision and computational costs of Level Set-Discrete Element Method (LS-DEM) with respect to DEM

Jérôme Duriez, Stéphane Bonelli

Graphical Abstract

Precision and computational costs of Level Set-Discrete Element Method (LS-DEM) with respect to DEM

J. Duriez, S. Bonelli



Precision and computational costs of Level Set-Discrete Element Method (LS-DEM) with respect to DEM

J. Duriez[a,*], S. Bonelli[a]

[a]*INRAE, Aix Marseille Univ, RECOVER, Aix-en-Provence, France*

Abstract

The Level Set-Discrete Element Method (LS-DEM) extends DEM towards arbitrary grain shapes by storing distance-to-surface values on a grid for each Discrete Element (DE), together with considering boundary nodes located onto the DE's surface. Both these ingredients are shown to affect the precision and computational costs of LS-DEM, considering various numerical simulations at the contact- and packing-scales for ideal spherical and superellipsoid shapes. In the case of a triaxial compression for spherical particles, approaching with a reasonable precision the reference result obtained in classical DEM requires the grid spacing to be smaller than one tenth of particle size, as well as using a couple thousands of boundary nodes. Computational costs in terms of memory (RAM) or evaluation time then increase in LS-DEM by two or three orders of magnitude. Simple OpenMP parallel simulations nevertheless significantly reduce the increase in time cost, possibly dividing the latter by 20.

Keywords: computational cost, particle shape, Level Set-Discrete Element Method (LS-DEM)

1. Introduction

At the micro-scale considered by Discrete Element Methods (DEM), granular soils reveal diverse grain's shapes, that constitute one ingredient of their discrete nature. This shape enters soil classification and is directly used in geotechnical

*Corresponding author
Email address: jerome.duriez@inrae.fr (J. Duriez)

engineering for the ballast foundations of railtracks, which rely over angular, not spherical, particles. Outside of this pratical example, particle shape has been recognised as influencing the mechanical behavior of granular materials since several studies often adopting DEM approaches. In an early 2D study on rotating cylinders and heap configurations (Pöschel and Buchholtz, 1993), a non-spherical shape was shown to contribute even more to macro-behavior than contact friction in the sense non spherical particles in frictionless interaction revealed a higher slope stability than spherical particles in frictional interaction. For a given frictional interaction, a higher shear strength of non-spherical particles has also been found for biaxial configurations in other 2D studies (Szarf et al., 2009; Jerves et al., 2016), together with a shape influence onto the critical state line (Jerves et al., 2016).

Investigating the mechanical influence of shape in real 3D conditions remains however technically challenging. While experimental studies require a proper particle-scale characterization of the complex shapes exhibited in nature (Vlahinić et al., 2014; Wang et al., 2019), those same real shapes have to be correctly introduced in the numerical world for DEM approaches. This induces a much more complex contact treatment in the DEM workflow, as opposed to the use of spherical particles which entails straightforward definitions of contact normals and relative displacements from the branch vector and the radii of contacting spheres. These complex contact treatments may obey several strategies which are partially listed in the following. First, rigid clusters of spheres (Pöschel and Buchholtz, 1993; Szarf et al., 2009; Garcia et al., 2009) enable the DEM practitioner to get much closer to real shapes, making these rigid clusters probably the second most-commonly used shape for Discrete Elements, just after spheres. These clusters nevertheless inherently include some unrealistic local roundness that may affect the mechanical description (Cho et al., 2006). Convex polyedra (Eliáš, 2014; Gladkyy and Kuna, 2017) now constitute another quite classical shape enhancement since Cundall (1988), thanks to a variety of algorithms such as searching for surface points with a common normal and/or minimizing interparticle distance (Dubois, 2011). As described by Zhao and

Zhao (2019), some of those algorithms can also be adapted to superellipsoids and quite general convex shapes without any edges. A last DEM variant to be mentioned is the Level Set-DEM (LS-DEM) proposed in 3D by Kawamoto et al. (2016). LS-DEM appears as promising in terms of versatility, since it does not include any inherent requirement for convexity and may apply directly to X-ray tomography images of soil samples (Kawamoto et al., 2016). Level Set concepts were initially proposed to study time evolutions of surfaces (Sethian, 1999), and applied in this sense to geotechnics by Golay et al. (2010, 2011) for flow-induced interfacial soil erosion. In the sense of LS-DEM, those Level Set concepts are used for defining in space distance fields to particles' surfaces, that are at the heart of contact treatment.

One can finally think about introducing more complex contact laws as an indirect description of particle's shape (Wensrich and Katterfeld, 2012; Aboul Hosn et al., 2017). However, this strategy obviously induces additional model parameters and increased calibration efforts that diminish the appealing mechanical simplicity of DEM.

Advocating therefore for a direct description of particle's shape through e.g. LS-DEM, the present manuscript then aims to discuss associated technical aspects in terms of obtained precision and increased computational costs, in the case of an implementation based on the YADE code (Šmilauer et al., 2015). Detailed information in these technical aspects seem lacking until now, even though one can await significant costs from the mentions of gigabytes RAM footprint in (Kawamoto et al., 2016) or superprocessors with 480 cores in (Kawamoto et al., 2018).

Section 2 presents the YADE implementation of LS-DEM based on the principles given by Jerves et al. (2016); Kawamoto et al. (2016). Section 3 discusses the variable precision of LS-DEM in describing contact- or packing-scale configurations adopting spherical or superellipsoid shapes: ideal spherical shapes are in particular considered for the precision analysis to ground on reference results obtained using DEM. LS-DEM precision is then connected with computational costs in Section 4, before that parallel scalability is examined in Section 5 in order to alleviate time costs.

2. Outline of LS-DEM

2.1. Shape description

Describing shape, i.e. particle morphology, in LS-DEM relies on the signed distance function $\phi(\vec{x})$ that returns, for any point $\vec{x}$ in space, the shortest distance from $\vec{x}$ to the surface at hand, with the convention of negative distances when $\vec{x}$ lies inside the surface. The surface of a Discrete Element (DE) then corresponds to the zero level set of the function ϕ, while the exterior (resp. inner) to the surface obeys $\phi > 0$ (resp. $\phi < 0$).

In this sense, LS-DEM is similar to the potential particles approach proposed by Houlsby (2009); Boon et al. (2013) where the sign of a potential function f defines the position of any point with respect to particle's surface, with $f = 0$ along the surface. Potential particles however require convex shapes and polynomial equations for the potential f, unlike LS-DEM.

In LS-DEM, the signed distance function ϕ is actually defined in a discrete fashion, storing ϕ-values on a cartesian body-centered grid, for each DE (Figure 1). This minor requirement of a discrete distance field, instead of an analytical equation, confers LS-DEM a great versatility to mimic real shapes, as exemplified by Kawamoto et al. (2016, 2018).

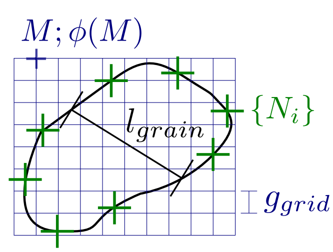


Figure 1: Plane view of the 3D regular grid at the roots of shape description in LS-DEM. Exact values of the signed distance function ϕ are known at each grid node M (the blue cross evidences just one of them). Boundary nodes N_i play a role in contact treatment as described in §2.2

From the knowledge of ϕ-values at each node of the grid, $\phi(\vec{x})$ is also defined for any point $\vec{x}$ within the grid extents from trilinear interpolation of ϕ-values at the eight surrounding grid nodes. In addition to defining particle's surface, and serving for contact treatment as described in the following section 2.2, this distance fied also enables one to define inertial quantities for DE summing mass and inertia contributions of all grid voxels that are considered inside a particle. Here, a grid voxel made of eight nodes $\{(i, j, k)\ ; i \in [i_0; i_0+1], j \in [j_0; j_0+1], k \in [k_0; k_0+1]\}$ is considered inside a particle depending on ϕ-value at the lowest node (i_0, j_0, k_0). A smoother description was proposed by Kawamoto et al. (2016) but is not considered here, having in mind quasi-static simulations with no influence from the inertial quantities onto the results.

As will be discussed in more detail in section 3, the grid spacing g_{grid}, compared with particle's characteristic length l_{grain} obviously affects the precision of the interpolated distance field, and that of LS-DEM.

Moreover such a distance field, the contact algorithm precised below in § 2.2 introduces a second key ingredient for the method, since a LS-DEM shape also involves a set of so-called boundary nodes, being exactly located on the surface (Figure 1). These are obtained through ray tracing (e.g. Lin and Ching, 1996): starting from the center of mass of a DE, as determined from the inside voxels, a half-line ray defined by its direction $\vec{v}$ is followed until crossing the DE's surface. Rays $\vec{v}$ could be chosen adopting various partitions of the (θ, φ) space, with $\theta \in [0; \pi]$ and $\varphi \in [0; 2\pi]$ being the two spherical angles. Here, boundary nodes follow a spiral path in the spirit of (Rakhmanov et al., 1994), where a total number N_{nod} of boundary nodes is located along the following spherical coordinates (θ_k, φ_k), $k \in [0; N_{nod} - 1]$:

$$\theta_k = \arccos\left(-1 + \frac{1+2k}{N_{nod}}\right) \tag{1}$$

$$\varphi_k = \pi(3 - \sqrt{5})k \tag{2}$$

For spheres at least, such a spiral path seeds boundary nodes more uniformly over the particle's surface, when compared with a rectangular partition of the (θ, φ) space. As a matter of fact, it avoids an overdiscretization of the poles ($\theta = 0\ [\pi]$) thanks to the non-constant step in θ. For each ray direction $\vec{v}$, and due to the trilinear description of distance within each grid voxel, the ray-surface intersection can be obtained solving the roots of a cubic polynom, giving the position of boundary nodes.

As it will be detailed in the following paragraph, no real update of the boundary nodes, nor of the distance field is needed during LS-DEM simulations: considering rigid particles with constant shapes, both are determined once for all at the beginning of a simulation, in reference configurations of the DE.

The present shape description appears as very general and distance fields for non-convex shapes could be readily obtained through Level Set algorithms (Sethian, 1999) that also apply to such cases. Ray traced boundary nodes may also follow non-convex shapes, with the only limitation being that ray tracing leads to a maximum of one boundary node per grid cell, along a given ray, due to the trilinear description of the distance field.

2.2. Kinematics of contact from Level Set shape and boundary nodes

Contact detection between two Level Set-shaped DEs first implies an approximate neighbour search that is common to all YADE simulations, following a so-called sweep and prune algorithm working on bodies' axis-aligned bounding boxes (Dubois, 2011; Šmilauer et al., 2015). This leads to a reduced list of potential contacts between bodies pairs.

Exact determination of contact between two bodies in this list then relies on a master-slave algorithm whereby the exact determination of interparticle distance both relies on the distance field ϕ_B to the biggest (in volume) particle, and on the boundary nodes $\overrightarrow{ON_i}$ (with O the origin) of the smallest particle (Figure 2). For convenience, labels 1,2 will replace in the following the mention of small or big particles, with $\phi_2 = \phi_B$. Contact is then obtained for at least one boundary node $\overrightarrow{ON_i}$ showing $\phi_2(\overrightarrow{ON_i}) \leqslant 0$. Boundary nodes logically need to be numerous enough to avoid bias in the LS-DEM results through missing contacts if $\phi_2(\overrightarrow{ON_i}) > 0 \ \forall N_i$, as it will be investigated in the following sections.

After detecting at least one boundary node of 1 touching 2, the interaction description is based on the node N_c showing the greatest penetration, leading

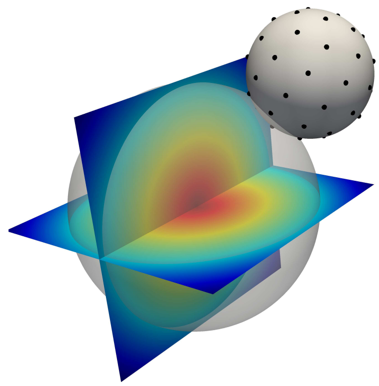


Figure 2: Distance field (colored map) and boundary nodes (black points) serving for the LS-DEM contact algorithm, illustrated for spherical particles

to the following interparticle overlap u_n:

$$u_n = -\min(\phi_2(\overrightarrow{ON_i}),\ \overrightarrow{ON_i} \in \mathcal{S}_1) = -\phi_2(\overrightarrow{ON_c}) \geqslant 0 \tag{3}$$

The current "greatest penetration" choice follows classical contact laws in DEM and corresponds to another recent LS-DEM study (Li et al., 2019). On the other hand, LS-DEM was initially proposed by Jerves et al. (2016); Kawamoto et al. (2016) with a mechanical interaction at each contacting node, which used to make the model behavior directly dependent on the number of boundary nodes, in addition to the k_n and k_t stiffnesses discussed below. That other choice would still enable to address non-convex shapes, which is not done here.

While the overlap u_n serves as the normal relative displacement, the present contact treatment does not resort to any total tangential displacement but just to an incremental one at the subsequent stage of applying the contact law, see the next § 2.3. The normal and tangential contact directions actually refer to the normal to $\mathcal{S}_1$ at N_c, chosen as the contact normal:

$$\vec{n} = \vec{\nabla}\phi_1(\overrightarrow{ON_c}) \tag{4}$$

For simplicity, special shapes showing pathological definitions of the normal, with tips or edges, are not considered here.

For e.g. the purpose of subsequent torque computations, a contact point $\vec{x}_c$

is defined in the middle of the overlap between 1 and 2:

$$\vec{x}_c = \overrightarrow{ON_c} - \frac{u_n}{2}\vec{n} \tag{5}$$

Considering the rigid bodies transformations of 1 and 2, the current contact algorithm easily makes use of the initial distance field and boundary nodes, as defined in the previous § 2.1 in reference configurations.

In line with its master-slave nature, such a contact treatment is not symmetric and this could be seen as a possible source of inaccuracy in the contact model in the sense different results could have been obtained adopting other choices, using e.g. ϕ_2 instead of ϕ_1 in Eq. (4). It is however reasonably believed that a sufficient discretization of particle's surfaces with many boundary nodes would cancel this possible bias. One should also note that the present choice of the smallest particle for carrying the boundary nodes allows to explore distance fields (whose precision depends upon grid resolution only) with the greatest surface density in nodes.

2.3. Mechanics of contact

Once a contact is detected and kinematically described as presented in the above, classical elastic (resp. elastic-plastic) contact laws apply in the normal (resp. tangential) directions, with k_n and k_t the normal and tangential stiffnesses and μ the contact friction coefficient.

The repulsive normal force $\vec{F}_n$ is first given by:

$$\vec{F}_n = k_n\, u_n\, \vec{n} \tag{6}$$

In the tangent plane, the frictional tangential force is incrementally computed from $\vec{0}$, one time step after another as per the following equation:

$$d\vec{F}_t = d\left(||\vec{F}_t||\frac{\vec{F}_t}{||\vec{F}_t||}\right) = ||\vec{F}_t||d\left(\frac{\vec{F}_t}{||\vec{F}_t||}\right) + d(||\vec{F}_t||)\frac{\vec{F}_t}{||\vec{F}_t||} \tag{7}$$

In the rhs of Eq. (7), the first term just accounts for a possible change in the tangential force direction (its unit vector $\vec{F}_t/||\vec{F}_t||$) while the interacting pair would move as a rigid body with possible variations in the orientation of the tangent plane. This first term is computed from the previous and current normal directions and from the angular velocities of each DE (Šmilauer et al., 2015). On the contrary, the last term in Eq. (7) accounts for the force variation due to a incremental tangential relative displacement, $d\vec{u}_t$, as computed at the contact point between the two DEs. A classical elastic-plastic force-displacement relationship here applies:

$$d(||\vec{F}_t||)\frac{\vec{F}_t}{||\vec{F}_t||} = k_t\, d\vec{u}_t \quad \text{enforcing } ||\vec{F}_t|| \leqslant \mu||\vec{F}_n|| \tag{8}$$

The interaction force being determined, an associated torque is also imposed with a possible contribution of the normal force for arbitrary shapes, unlike spheres.

2.4. Equations of motion

Sustaining resultant forces and torques, each DE is classically characterized in space using $\vec{x}(t)$, the current position of its center of mass P, as well as a rotation matrix $\boldsymbol{R}(t)$ that describes its current orientation, i.e. the orientation of the local frame of eigenvectors for the inertia matrix, $(\vec{e_i}), i \in [1;3]$, as seen in the global frame. The rotation matrix $\boldsymbol{R}$ actually transforms each vector $\vec{u}_L$ of the local frame in its current counterpart in the global frame $\vec{u}_G$ through classical change of basis relation $\vec{u}_G = \boldsymbol{R}\,\vec{u}_L$. Newton-Euler equations for the motion of rigid bodies then rule the evolutions of $\vec{v}$, the velocity of point P and of $\vec{\omega}$, the angular velocity of the body:

$$m\frac{d\vec{v}}{dt} = \vec{f} \tag{9}$$

$$\boldsymbol{I}\frac{d\vec{\omega}}{dt} + \vec{\omega} \wedge \boldsymbol{I}\vec{\omega} = \vec{\Gamma} \tag{10}$$

, with $\vec{f}$ the resultant force on the DE and $\vec{\Gamma}$ the resultant torque computed at the center of mass P. For the purposes of deriving Eq. (10) $\vec{\Gamma}$ and $\vec{\omega}$ are expressed in the local frame $(\vec{e_i})$, where $\boldsymbol{I}$ components are constant. We recall that Eq. (10) would simplify to $\boldsymbol{I}d\vec{\omega}/dt = \vec{\Gamma}$ for simple, isotropic, shapes with a spherical inertia matrix $\boldsymbol{I} = k\boldsymbol{\delta}$ (with $\boldsymbol{\delta}$ the identity matrix), such as spheres or cubes.

Global damping is classically considered, modifying the resultant forces and torques in Eqs. (9)-(10) in dynamic cases where those are non-zero. A damping coefficient D, taken here equal to 0.2, enters the equations such that the right hand sides of Eqs. (9)-(10) actually are $(1 \pm D)\vec{f}$ or $(1 \pm D)\vec{\Gamma}$, depending on the power of resultant forces or torques. Accelerating cases with a positive power are hindered, considering $(1 - D)$, while decelerating conditions with a negative power are amplified through the use of $(1 + D)$.

Time variations of position and orientation finally follow from the above Newton-Euler equations as per:

$$\frac{d\vec{x}}{dt} = \vec{v} \tag{11}$$

$$\frac{d\boldsymbol{R}}{dt} = \boldsymbol{R}\,\boldsymbol{\Omega} \tag{12}$$

, with $\boldsymbol{\Omega}$ in Eq. (12) being the antisymmetric matrix such that $\boldsymbol{\Omega}\,\vec{x} = \vec{\omega} \wedge \vec{x}$, $\forall \vec{x}$. Integrating these Eqs. (9) to (12) is achieved in YADE from appropriate explicit numerical schemes and using a quaternion equivalent for the rotation matrix $\boldsymbol{R}$ (Šmilauer et al., 2015).

3. Precision of LS-DEM

3.1. Materials and methods

The precision of LS-DEM in connection with boundary nodes and grid spacing is now investigated for different kinds of numerical simulation, comparing when possible LS-DEM with classical DEM serving as a numerical reference. For comparison purposes, ideal spherical shapes are then often adopted, since they enable one to obtain such a DEM reference result. The distance fields necessary to LS-DEM are straightforward to define for spheres of given radii.

Extending towards arbitrary shapes, superellipsoids, also known as superquadrics (Barr, 1995), are also considered. Generalizing ellipsoids, they constitute a convenient choice for exploring non-spherical shapes, e.g. (Wang et al., 2019), since they offer an analytical description through three radii r_x, r_y, r_z distorting length along the three axes, combined with two additional exponents

Shape index	Half-extents (length unit)			Curvature exponents (-)	
	r_x	r_y	r_z	ϵ_e	ϵ_n
0	0.4	1	0.8	0.4	1.6
1	0.42	=	0.83	0.1	1
2	=	=	=	1	0.5
3	0.5	0.7	1	1.4	1.2

Table 1: Shape parameters of the four superellipsoids shown in Figure 3

ϵ_e, ϵ_n that modify the surface curvature. In local axes, their surface equation namely reads:

$$f(x,y,z) = \left(\left| \frac{x}{r_x} \right|^{\frac{2}{\epsilon_e}} + \left| \frac{y}{r_y} \right|^{\frac{2}{\epsilon_e}} \right)^{\frac{\epsilon_e}{\epsilon_n}} + \left| \frac{z}{r_z} \right|^{\frac{2}{\epsilon_n}} - 1 = 0 \tag{13}$$

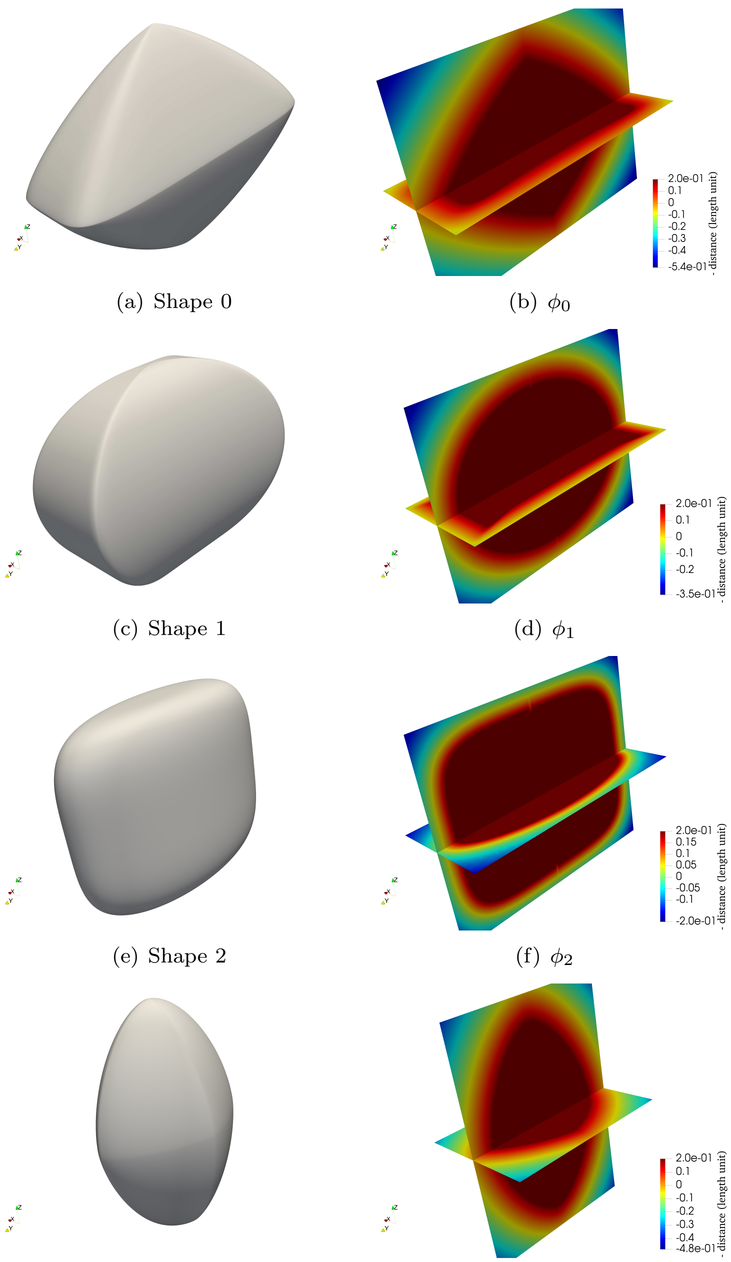
While such an analytical description is not required in LS-DEM, it aptly provides a first order approximation for the signed distance function to a superellipsoid, which is herein simply proposed as:

$$\phi \approx \frac{f}{||\vec{\nabla} f||} \tag{14}$$

Eq. (14) obviously describes a zero distance, $\phi = 0$, along the surface. It is furthermore easily verified that the Eikonal equation defining distances, $||\vec{\nabla}\phi|| = 1$ (Sethian, 1999), is by construction verified at the first order close to the surface. This approximation, illustrated in Figure 3, is sufficient for typical LS-DEM simulations with negligible overlaps since an accurate distance field is then necessary close to the surface only.

The Table 1 lists a chosen set of 4 shape parameters, with the corresponding 4 different superellipsoids being depicted in Figure 3. The radii r_x, r_y, r_z shown therein will be scaled to appropriate lengths in the following.

Regardless of the shape or the modelling approach (DEM or LS-DEM) chosen thereafter, the same contact parameters and particle size distribution are used, see Table 2. The distribution of particle's diameter D is uniform in number between extreme D_{min} and D_{max}, whose values do not necessarily correspond to any physical entity. Numerical samples made of superellipsoids include in equal proportion the 4 shapes presented in the above and conform that same particle



(a) Shape 0 (b) ϕ_0

(c) Shape 1 (d) ϕ_1

(e) Shape 2 (f) ϕ_2

(g) Shape 3 (h) ϕ_3

Figure 3: The four superellipsoids (left) defined in Table 1, illustrated together with their distance fields (right). Image scales are constant for each shape (on each row), and the positive range of color maps (shape's interior) is capped to 0.2 length units for convenience

size distribution. Doing so, a sieve diameter is chosen for each superellipsoid as the diameter of its circumscribed sphere, i.e. twice the greatest center-boundary node distance.

Table 2: DEM and LS-DEM mechanical parameters

k_n (N/m)	k_t/k_n (-)	μ (-)	D_{min} (cm)	D_{max}/D_{min} (-)
6×10^5	0.3	0.577	6.1	3

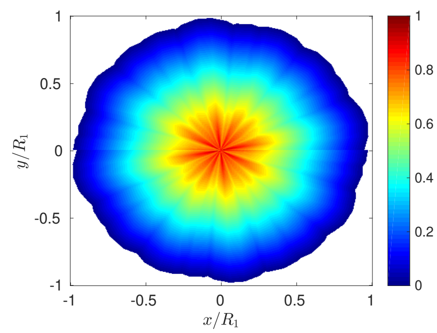
3.2. Single contact description

The precision of LS-DEM is first analyzed for the simple case of a single contact between two spherical particles, with a possible discrepancy in size (Figure 2). While the precision of each particle's distance field is fully defined by the resolution D/g_{grid} of its underlying grid, the ability of the LS-DEM contact algorithm to capture the distance field furthermore depends upon boundary nodes, in the number of N_{nod}, and on the diameter ratio $D_2/D_1 \geqslant 1$. The Figure 4 illustrates how these three parameters affect the LS-DEM measure of an overlap between the two spherical particles.

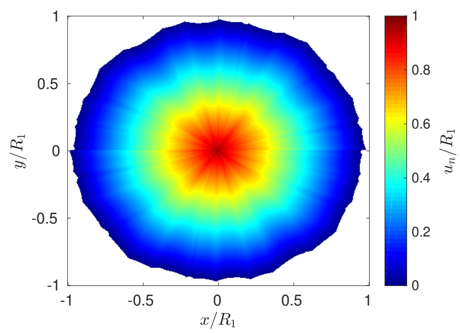
It is for instance observed in Figure 4(a) that using just 100 boundary nodes (in 3D space) leads to miss interactions close to the unit circle of the map, and to an approximation between the detected overlap and the true distance to a sphere. On the other hand, the Figure 4(d) confirms the true distance field can be re-obtained with a very good precision, i.e. $u_n = -\phi$, using $D/g_{grid} = 50$ and $N_{nod} = 1600$, with $D_2/D_1 = 1$. Thanks to the present choice of locating boundary nodes on the smallest sphere, cases with $D_2/D_1 > 1$ are described with a greater precision, see Figure 4(b) vs 4(a).

3.3. Isotropic reconstruction

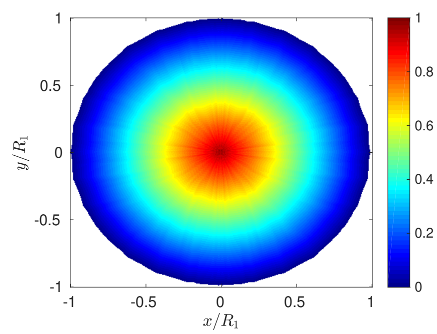
A second examples devotes to the LS-DEM reconstruction of a dense packing of 8000 spherical particles. While the current reconstruction procedure is essentially similar to the definition a LS-DEM sample from an experimental one, e.g. through computed tomography (Kawamoto et al., 2016, 2018), it actually here applies to DEM data describing the isotropic state of a numerical



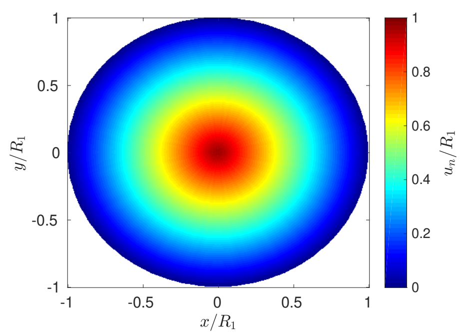
(a) $D/g_{grid} = 10; N_{nod} = 100; D_2/D_1 = 1$



(b) $D/g_{grid} = 10; N_{nod} = 100; D_2/D_1 = 3$



(c) $D/g_{grid} = 10; N_{nod} = 1600; D_2/D_1 = 1$



(d) $D/g_{grid} = 50; N_{nod} = 1600; D_2/D_1 = 1$

Figure 4: Precision of the LS-DEM contact algorithm in capturing a sphere's distance field. Color maps show the overlap $u_n(x, y)$ of a LS-DEM interaction between a sphere 1 centered at (x_c, y_c, z_c) and a bigger sphere 2 centered at $(x_c + (r + R_2)\cos(\theta), y_c + (r + R_2)\sin(\theta), z_c)$ with (r, θ) the polar counterparts to the cartesian (x, y). The origin of the map, $x = y = 0$, for instance corresponds to the center of 1 belonging the surface of 2, and to an expected overlap value equal to R_1. White region correspond to the absence of an interaction. Each map is constructed using 401^2 colored pixels and as many relative configurations of the two spheres

sample, showing a $n_{ref} \approx 0.372$ porosity while subjected to an hydrostatic pressure $p_{ref} = 16.5$ kPa. This pressure value corresponds to a stiffness ratio $\kappa = k_n/(pD_{50}) \approx 300$ which is an intermediate value among DEM studies. One can for instance mention κ-values in the order of several hundreds up to one thousand in qualitative (Duriez et al., 2018) as well as quantitative (Aboul Hosn et al., 2017) studies.

As such, a first DEM simulation, whose parameters were presented in Table 2, is run to reach that mechanical state. After exporting from the DEM model the positions and diameters D of all spherical particles, a LS-DEM reconstruction is attempted using at the particle scale different numbers of boundary nodes $N_{nod} \in \{0;100;400;900;1200;1600;2000;2500;4000;9000\}$ and grid resolution $D/g_{grid} \in \{10;20;30;50;90\}$. LS-DEM spheres being so defined from known positions and radii, reconstructed porosity n can be measured and one LS-DEM iteration is finally performed in order to also reconstruct normal contact forces being responsible for the sample's mean stress p, while preventing any movements of the DE. The obtained precision in terms of porosity or mean stress can be quantified through the n/n_{ref} or p/p_{ref} ratios, where a value of 1 or 100% indicates a perfect LS-DEM reconstruction of the reference case.

Porosity precision is actually independent of the boundary nodes and can be seen as geometric in nature since voxellised particles volumes are fully determined from the grid resolution. As such, the Figure 5 disregards boundary nodes number N_{nod} and evidences how spherical morphologies can be satisfactorily described with tens of grid voxels per diameter, with the error on porosity i.e. solid volumes reducing below 4% for $D/g_{grid} \geqslant 20$.

On the other hand, in terms of mean stress p/p_{ref} data (Figure 6) illustrate how grid resolution and boundary nodes both contribute to the mechanical precision of LS-DEM. Starting from an absence of contacts and stress in the extreme case of $N_{nod} = 0$, boundary nodes obviously have to be numerous enough for all contacts to be detected. For a given number of boundary nodes, grid resolution still improves precision since it contributes to more exact locations of these boundary nodes, closer to the true surface, as well as to a better overlap

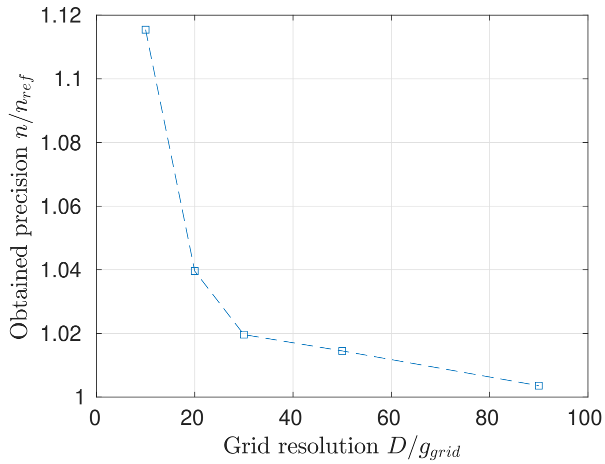


Figure 5: Geometric precision of LS-DEM in terms of porosity n after reconstructing a fully determined spherical packing in isotropic state

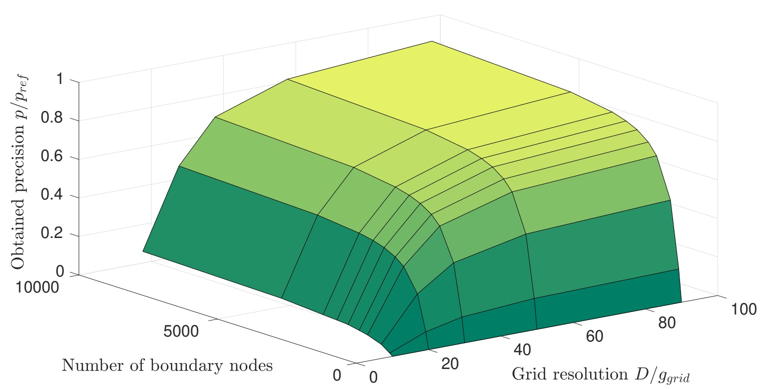


Figure 6: Mechanical precision of LS-DEM in terms of mean stress p after reconstructing a fully determined spherical packing in isotropic state. Each vertex of the depicted surface corresponds to one LS-DEM reconstruction

estimation. As a matter of fact, a 80% precision can here be obtained choosing $\{N_{nod}; D/g_{grid}\}$ either as {2500;50} or {1600;90}. Among the cases tested, a maximum precision of 94 % is reached for 9000 boundary nodes and a grid resolution of 90, which is another step towards validating the present LS-DEM implementation with respect to DEM and investigating the role of its technical ingredients $\{N_{nod}; D/g_{grid}\}$. This is pushed further in the following section.

3.4. Triaxial compression

Another comparison between DEM and LS-DEM for spherical shapes eventually considers the triaxial compression of that same dense sample, under the confining stress $\sigma_2 = \sigma_3 = 16.5$ kPa and until an axial strain $\varepsilon_1 = 5$ %. This axial strain value is posterior to the peak in deviatoric stress $q = \sigma_1 - \sigma_3$ that is observed in DEM.

Again, several LS-DEM simulations are carried on, for $N_{nod} \in$ {100;400;1600; 2500;4000} and $D/g_{grid} \in$ {10;20;50}. Any LS-DEM simulation starts with the same sample definition than before, defining appropriate Level Set shaped bodies from the DEM data that describe the isotropic stress $p_{ref} = 16.5$ kPa. Because the same mechanical state is not directly captured within LS-DEM, confining phase is pursued further, with a servo-control of boundary walls until that reference isotropic stress p_{ref} is re-obtained. Then, both DEM and LS-DEM simulations apply triaxial shear loading with a constant axial strain rate $\dot{\varepsilon}_1$ that corresponds to an inertial number $I = \dot{\varepsilon}_1 D_{50}\sqrt{\rho/\sigma_3} \approx 10^{-4}$ low enough for its influence and the one of global damping to vanish. It is actually verified in DEM and LS-DEM that stresses measured along the boundary walls equal homogenized Love-Weber stresses (Love, 1892; Weber, 1966; Drescher and de Josselin de Jong, 1972) for static equilibrium conditions. Table 3 details relevant parameters, with a fictitious $\rho = 1000$ kg/m^3 density being herein adopted. The latter could be replaced by another value provided that time step and loading rate are also modified in order to avoid divergence of the explicit scheme and maintain the same inertial number. Such changes would keep constant the total number of DEM iterations required for simulating triaxial shear until $\varepsilon_1 = 5$ %.

Table 3: DEM and LS-DEM numerical parameters for the triaxial compressions

Density ρ (kg/m^3)	Timestep Δt (s)		Damping coefficient D (-)	Loading rate $\dot{\varepsilon}_1$ (s^{-1})
	Spheres	Superquadrics		
1000	3.4×10^{-4}	1.7×10^{-4}	0.2	2.5×10^{-3}

On that second example, the LS-DEM precision is quantified comparing the deviator peak q^{max} of each LS-DEM simulation with the reference DEM value $q^{max}_{ref} \approx 33$ kPa, through a q^{max}/q^{max}_{ref} ratio that is illustrated in the Figure 7.

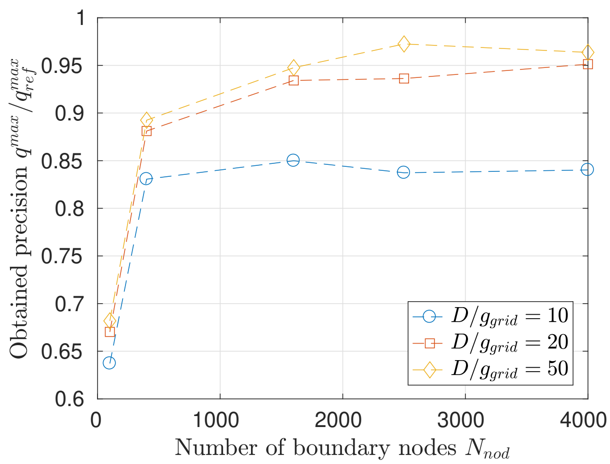


Figure 7: Precision of LS-DEM in terms of peak strength during the triaxial loading of spherical grains

Similar trends in precision are observed on this third example, with a joint influence of the grid resolution and the number of boundary nodes. This being said, the present DEM *vs* LS-DEM comparison with non-fixed DEs under deviatoric loading is more favorable than the isotropic reconstruction. Indeed, using 4000 boundary nodes and a grid resolution of 50 now enables one to reach an excellent 96% overall precision, whereas it previously led to just 85% for the isotropic example. This 85% precision would here be exceeded choosing $\{N_{nod} = 400; D/g_{grid} = 20\}$ only. The particular case of $\{N_{nod} \geqslant 1600; D/g_{grid} = 10\}$ illustrates the marginal possibility for a non-monotonous increase in precision with respect to N_{nod}. One may think for instance to the very specific case of two spheres in contact that could be perfectly described with just one boundary node located along their branch vector.

In addition to the only consideration of peak deviatoric stress, the Figure 8 illustrates the effects of $\{N_{nod}; D/g_{grid}\}$ choices onto the evolutions of other average quantities according to axial strain. LS-DEM is therein also compared with DEM for what concerns the volumetric strain ε_V, the anisotropy a_c of the contact network, and the average contact number z_c. As for the contact anisotropy a_c, the latter is expressed as the difference between the axial and the lateral components of the fabric tensor $\boldsymbol{F}$ whose expression is represented in the following Eq. (15).

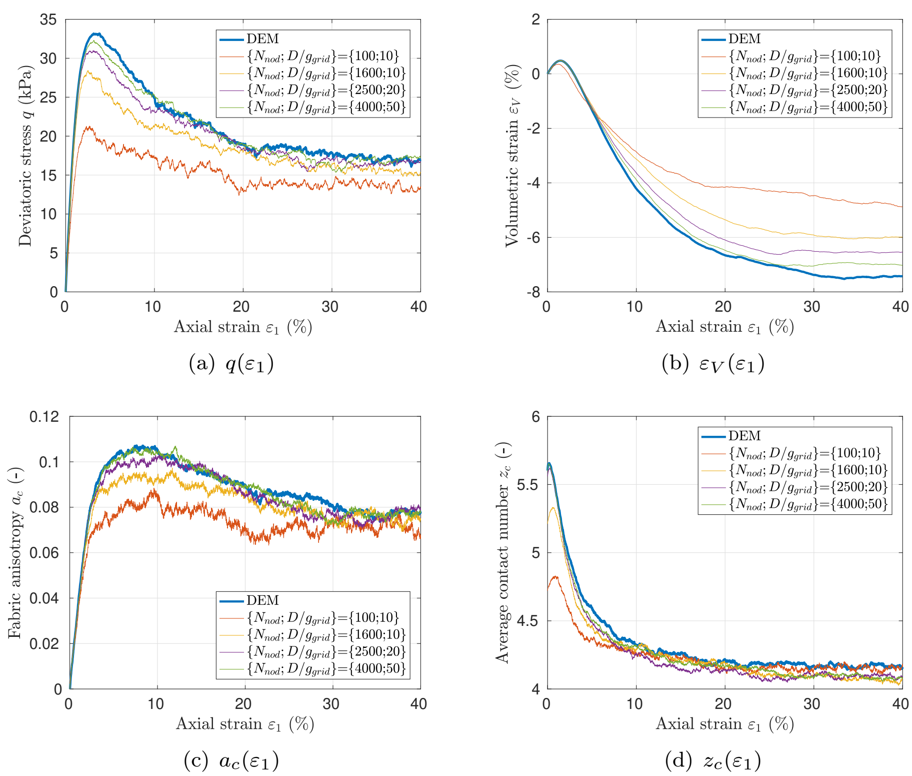
$$\boldsymbol{F} = \frac{1}{N_c} \sum_c \vec{n} \otimes \vec{n} \tag{15}$$

For the purpose of computing $\boldsymbol{F}$ in LS-DEM, it is recalled contact normals are computed in this case from the distance gradient as per the previous Eq. (4). The precision in evaluating this distance gradient again depends on grid resolution.

The Figure 8 confirms that the LS-DEM evaluation of any quantity of interest tends to its DEM counterpart for $\{N_{nod}; D/g_{grid}\}$ reaching the order of {4000;50}. It furthemore illustrates how the dense-like behavior traits, with softening and dilation, of the present numerical sample appear as diminished when using an insufficient LS-DEM discretization in terms of boundary nodes and grid resolution. One can lastly note that LS-DEM curves are generally speaking somewhat more noisy than DEM counterparts, due to the surface discretization in boundary nodes. Such a surface discretization, when poor in particular, may indeed enhance the discontinuous i.e. sudden changes in overlap and contact forces already present in DEM due to the time discretization, possibly affecting the curves at the macro-scale.

3.5. Triaxial compression of superellipsoids

A last example devotes to a packing of 8000 superquadrics, as defined in the above § 3.1, under the same triaxial loading than the one imposed on spherical particles. After reaching the isotropic state (Figure 9) $p = 16.5$ kPa and $n \approx 0.32$ through compressing an initial cloud of superellipsoids, in a similar manner than



(a) $q(\varepsilon_1)$

(b) $\varepsilon_V(\varepsilon_1)$

(c) $a_c(\varepsilon_1)$

(d) $z_c(\varepsilon_1)$

Figure 8: DEM vs LS-DEM comparisons during a triaxial loading of spherical grains: effects of LS-DEM discretization onto averaged quantities

for spheres, triaxial shear is again pursued until an axial strain $\varepsilon_1 = 5$ % being posterior to the deviator's peak. Among the simulation parameters, being listed in Tables 2 and 3, time step is modified from the spherical case because of a possibly lower volume, hence mass, of a superellipsoid when compared to a sphere having the same circumscribed diameter.

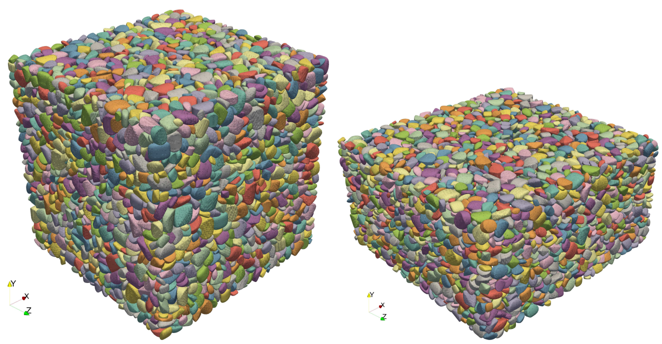


Figure 9: Initial (left) and sheared (right, for $\varepsilon_1 = 40$ %) configurations of the superellipsoids packing under triaxial loading

Such a LS-DEM simulation is carried on for different choices of $N_{nod} \in \{400; 1600; 2500; 4000\}$ and $2\min(r_x, r_y, r_z)/g_{grid} \in \{10;20;50\}$, disregarding here the less precise case $N_{nod} = 100$. Looking at the obtained peak in q, the data illustrated in the Figure 10 once again show how both the grid resolution and the boundary nodes number affect the LS-DEM results. With respect to the ideal spherical shapes considered in the above, the results also suggest that capturing more complex shapes might be more demanding in terms e.g. of boundary nodes number N_{nod}. While using $N_{nod} \geqslant 1600$ induced fairly constant LS-DEM results for spheres (within a 2-3 % variation, see Figure 7), the present results on superellipsoids still vary by nearly 10 % in that range, without a clear plateau.

As for the deviator strength itself, one can also note from the most precise LS-DEM simulations that the superquadrics packing exhibits a deviator strength $q^{max} \approx 48$ kPa, which is approximately 45 % higher than the ones for spheres (where $q^{max}_{ref} \approx 33$ kPa) and combined with differences in initial porosity or coordination number. A greater ultimate triaxial strength at critical state is also obtained, with $M = q/p \approx 0.76$ for spheres, versus $M \approx 1.13$ for superel-

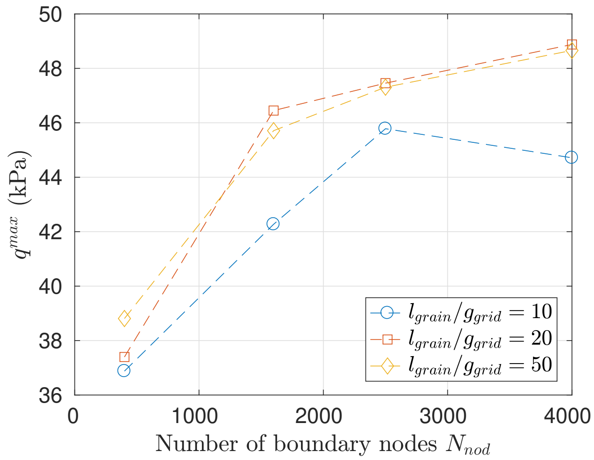


Figure 10: LS-DEM description of the peak strength for a triaxial loading imposed on superellipsoids, choosing $l_{grain} = 2\min(r_x, r_y, r_z)$.

lipsoids using $N_{nod} = 2500$ and $2\min(r_x, r_y, r_z)/g_{grid} = 20$ until $\varepsilon_1 = 40$ %. While further discussion is left for future work, these results confirm the shape influence upon the mechanical properties.

3.6. Discussion

From the comparisons shown in the above, and with a greater focus on the more meaningful triaxial simulation with moving DEs, one could advice to use a grid resolution (l_{grain}/g_{grid}) in the order of few tenths, and a couple of thousands boundary nodes at least. Even though previous LS-DEM studies (Jerves et al., 2016; Kawamoto et al., 2016, 2018) did not explicitly provide such technical details, similar order of magnitudes can be inferred as follows.

Regarding the boundary nodes, the key references (Jerves et al., 2016; Kawamoto et al., 2016) formulated the same guideline in terms of node-to-node spacing, proposing therein that restricting these distances to one tenth of particle diameter would avoid bias in the results. In addition to distance considerations, a proper set of boundary nodes should obviously cover the whole direction space $\theta \times \varphi = [0; \pi] \times [0; 2\pi]$. Assuming this was done in (Kawamoto et al., 2016) with a rectangular partition, and considering that $R\sqrt{\Delta\theta^2 + \Delta\varphi^2}$, with $\Delta\theta$, $\Delta\varphi$ the increments in the spherical angles θ, φ between two adjacent nodes, is an upper bound to that node-to-node distance, one can connect node-to-node spacing to the increments $\Delta\theta$, $\Delta\varphi$, then to the total number of nodes N_{nod}. As such, the above distance guideline quoted by Jerves et al. (2016); Kawamoto et al. (2016) can eventually be related to a total number of nodes N_{nod} being in the order of 1200. The present comparisons rather confirm this order of magnitude of thousand of boundary nodes as a minimum, and they furthermore illustrate how the grid resolution articulates with N_{nod} for what concerns the precision of the method.

As for the grid resolution itself, no exact mention of the latter seems to be found in (Jerves et al., 2016; Kawamoto et al., 2016, 2018). One can nevertheless speculate from Kawamoto et al. (2016) that a resolution l_{grain}/g_{grid} in the order of 30 or 40 was adopted therein, which also appears to be the required order of magnitude.

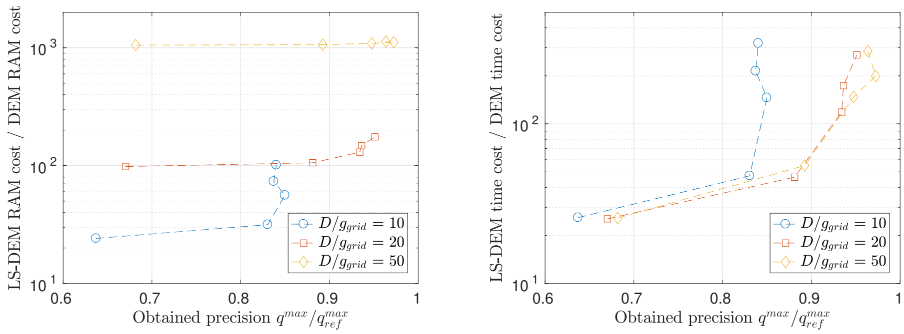
To conclude, LS-DEM practice certainly requires to consider grid resolution and boundary nodes as similar technical ingredients than meshes for Finite Element Methods, and eventually to check their (non-)influence onto the results.

4. Computational costs

The greater flexibility of LS-DEM logically comes along greater computational costs, be in terms of memory (RAM) footprint or evaluation time. These are now carefully investigated for the triaxial compression of spherical particles until $\varepsilon_1 = 5\%$ that was considered in the previous section 3.4, with the same choices of grid resolution D/g_{grid} and N_{nod} boundary nodes than before. The consideration of spheres allows once again direct comparisons with the classical DEM, but it is an interesting LS-DEM feature that computational costs are naturally insensible to the shapes being described, since they depend only upon grid resolution and boundary nodes number.

First of all, the RAM costs associated with the definition of DEs in LS-DEM are quantified and compared with the corresponding RAM cost in DEM. While the introduction of classical spheres here requires 10 megabytes of RAM for a DEM simulation, LS-DEM requires 100 or 1000 times more, i.e. gigabytes (Figure 11(a)). An important RAM consumption obviously arises due to the distance grid and its distance values counting in the order of r^3 for a grid resolution $r = D/g_{grid}$, per particle. Boundary nodes also contribute to RAM footprint since $3\,N_{nod}$ coordinate values have to be stored for each particle with N_{nod} boundary nodes. Several cases considered in previous sections 3.3 and 3.4 make these two quantities comparable. The Figure 11(a) illustrates how RAM footprint is affected by boundary nodes number (then precision) for low grid resolution: $D/g_{grid} = 10$ or 20, while being fairly constant for the finest grid with $D/g_{grid} = 50$. For such a fine grid, most storage requirements indeed concern the distance values, with, in proportion, little extra-requirements coming from the boundary nodes.

Second, evaluation costs are measured as the average wall clock duration of one iteration during the triaxial shearing. All LS-DEM simulations as well as the reference DEM simulations run sequentially as one thread executed on the same server machine. The server includes two Intel Xeon Platinum 8270, 2.7 GHz, processors with 26 cores and 36 MB of cache memory each. It thus offers a total of 52 cores and 104 threads, together with 1.5 TB 2.9 GHz RAM. On that machine, LS-DEM execution takes approximately 25 to 300 times longer than classical DEM, depending on LS-DEM parameters such as N_{nod}. From a quantitative point of view, these observations should be cautiously interpreted since they suffer from a non-exactly reproducible nature of evaluation times, in connection e.g. with temperature changes. They furthermore certainly depend on the hardware and simulation at hand, and on the present implementation into the YADE code. The comparison nevertheless provides useful orders of magnitude for (LS-)DEM practitioners. From a qualitative point of view, the Figure 11(b) illustrates how the present time cost is primarily affected by the number of boundary nodes, with an increasing N_{nod} leading to longer loops for contact treatment, in the same time it globally improves precision. For a given N_{nod}, slight variations in time cost are observed depending on the grid resolution D/g_{grid}, which just come from the previously mentioned non-reproducible nature of evaluation times.



(a) Memory (RAM) cost

(b) Time cost

Figure 11: LS-DEM computational costs according to precision for the triaxial shear on spheres, relative to the costs of DEM. Each datapoint corresponds to the use of different numbers of boundary nodes N_{nod}, among {100;400;1600;2500;4000}, resulting into different costs and precision for a given grid resolution D/g_{grid}

Finally, the present cost analysis also recalls the combined influence of both boundary nodes and grid resolution onto the results. It actually illustrates the possibility for different strategies of ressource managements, when seeking a given precision. Aiming to limit RAM consumption, a 95% precision could be here obtained choosing $D/g_{grid} = 20$ and 4000 boundary nodes. On the other hand, choosing $D/g_{grid} = 50$ and 1600 boundary nodes would show higher memory requirements, but would lead to the same precision after faster simulations.

5. OpenMP scalability for parallel simulations

Parallel computing is an obvious strategy to alleviate the high time costs of LS-DEM, and is available in YADE e.g. in a OpenMP shared memory framework (Šmilauer, 2010). The OpenMP framework distributes the treatment of DEM variables among parallel threads that will collectively move forward the simulation. Typical examples include integrating motion for different DEs with different threads, or the parallel computing of interaction forces for different interactions. However, the shared memory paradim inherently requires costly safeguards to avoid conflicts between possible operations from different threads onto the same DEM variable. One can think for instance to the resultant force of one given DE contributing to different interactions, which could be modified by different threads after parallel computations of interaction forces. After performing extra-operations to avoid such pitfalls, OpenMP speedups in YADE usually do not reach the optimal value of threads number (Šmilauer, 2010), with possible peaks in speedup around 8 threads for spherical particles (Zhao and Zhao, 2019).

As for LS-DEM, parallel speedups are investigated hereafter for the same triaxial shear on spheres and until $\varepsilon_1 = 5$ % than considered in the previous sections 3.4 and 4, using 1600 boundary nodes and a grid resolution of 20 which confered LS-DEM a sufficient precison (93%). Allocating a variable number of threads, the LS-DEM simulation is executed on the server machine mentioned in the above section 4, as well as on a workstation with one 4 cores (8 threads) Intel i7-7700, 3.60GHz processor with 8 MB of cache memory, as well as 64 GB of 2.4 GHz RAM.

Allocated threads go from 1 to 8 for the workstation, and from 1 to 100 for the server. For each thread number j (including the sequential case $j = 1$), simulation time t is measured repeating 3 times the simulation to account for the possible variations in time cost. Then, 9 parallel speedups can be measured for a given j, through the 9 ratios $t(j)/t(j = 1)$.

After averaging among these 9 measurements and quantifying error as one standard deviation, the data (Figure 12) show LS-DEM parallel simulations follow a linear speedup until 22 threads approximately. Under those conditions the workstation shows a fairly optimal speedup, while a 0.6 speedup coefficient, 40% smaller than the optimal one, is obtained on the server. Using even more threads, simulations then continue to speed up, at a lower rate, until 50 threads approximately. For that number of threads, parallel execution is more than 20 times faster than the sequential one. The simulation speed afterwards starts to decrease with the number of threads, whereby allocating more ressources eventually just increases evaluation time.

Even though the OpenMP scalability is not necessarily optimal, significant

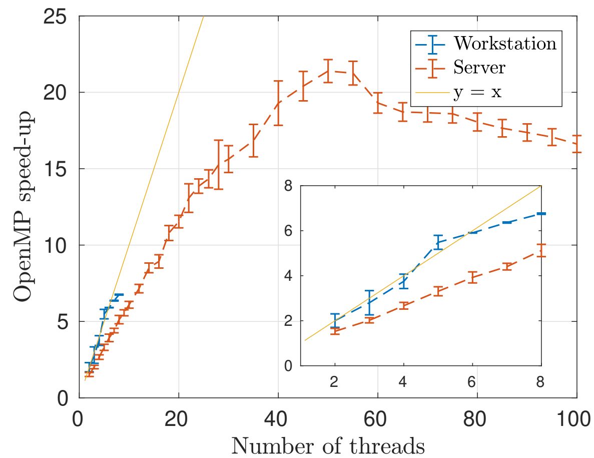


Figure 12: OpenMP speed up for the LS-DEM triaxial compression using spherical grains

time can then be saved in a LS-DEM simulation using an appropriate number of threads between 20 and 50. Time gains are even greater in proportion than one could get for classical DEM simulations. Indeed, the maximum parallel speed-up for the DEM simulation approximates 3.5 only, which is obtained for 10 threads approximately (Figure 13). Such a scalability corresponds to the one observed for spheres by Zhao and Zhao (2019). Allocating more threads to the DEM simulation does not bring any benefit and can even be detrimental since parallel simulations using more 60 threads are eventually slower than the sequential one. This enhanced scalability of LS-DEM versus DEM relates with the former's specificity that more than 99% of a sequential simulation is spent in contact treatment, with costly loops over boundary nodes.

6. Conclusions and perspectives

LS-DEM offers promising capabilities for arbitrary shape description in DEM with e.g. no inherent convexity requirements. Such a versatility requires a very significant amount of data per DE to be stored and numerically estimated during the DEM workflow, with three-dimensional tables of distance values on a grid, together with a set of boundary nodes for the purpose of master-slave contact

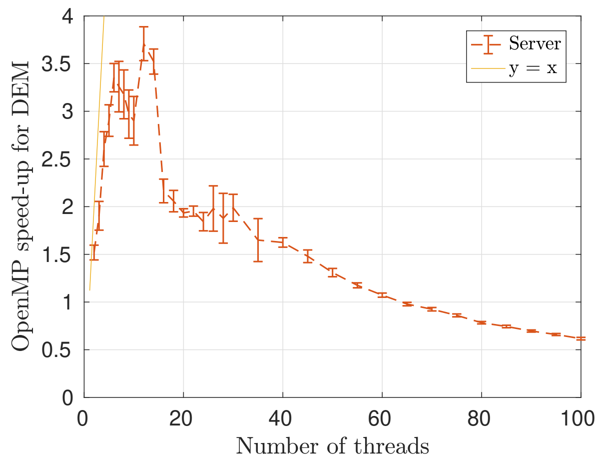


Figure 13: OpenMP speed up for the DEM triaxial compression on spheres

algorithms. By investigating simple configurations at the contact- and packing-scales for ideal spherical shapes with DEM serving as a reference, as well as superellipsoid ones, the precision of LS-DEM is shown to depend both on grid resolution and boundary nodes. On the present comparisons, reaching a good precision requires few tenths of grid spacings per particle size, as well as a couple of thousands boundary nodes.

Such choices dramatically increase computational costs of the simulations, be it in terms of memory (RAM) requirements or evaluation time. While sequential 3D DEM simulations at the sample scale usually weigh hours and megabytes, LS-DEM requires days and gigabytes, after an implementation based onto the YADE code. Time costs nevertheless can be significantly decreased through parallel computing with few tenths of threads, whereby a simple OpenMP framework decrease time costs by more than an order of magnitude.

Other parallel paradigms such as MPI, distributing memory instead of sharing it, may be even more useful and have yet to be investigated. Together with possible code and algorithmic (Duriez and Galusinski, 2020) improvements, they will hopefully make geotechnical simulations with real particle's shape even more affordable.

Acknowledgements

Financial support from the French Sud region to the LS-ENROC project is warmly acknowledged, as well as fruitful discussions with Cédric Galusinski and Frédéric Golay (Université de Toulon, IMATH). Technical assistance was furthermore provided by Jérémy Verrier (INRAE) for the server acquisition.

References

Aboul Hosn, R., Sibille, L., Benahmed, N., Chareyre, B., 2017. Discrete numerical modeling of loose soil with spherical particles and interparticle rolling friction. Granular Matter 19 (4), 11–12.

Barr, A. H., 1995. Rigid physically based superquadrics. In: Kirk, D. (Ed.), Graphics Gems III. Academic Press, pp. 137–159.

Boon, C., Houlsby, G., Utili, S., 2013. A new contact detection algorithm for three-dimensional non-spherical particles. Powder Technology 248, 94 – 102.

Cho, G.-C., Dodds, J., Santamarina, J. C., 2006. Particle shape effects on packing density, stiffness, and strength: Natural and crushed sands. Journal of Geotechnical and Geoenvironmental Engineering 132 (5), 591–602.

Cundall, P., 1988. Formulation of a three-dimensional distinct element model–Part I. A scheme to detect and represent contacts in a system composed of many polyhedral blocks. International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts 25 (3), 107 – 116.

Drescher, A., de Josselin de Jong, G., 1972. Photoelastic verification of a mechanical model for the flow of a granular material. Journal of the Mechanics and Physics of Solids 20 (5), 337 – 340.

Dubois, F., 2011. Numerical modeling of granular media composed of polyhedral particles. In: Radjai, F., Dubois, F. (Eds.), Discrete-element Modeling of Granular Materials. ISTE-Wiley, pp. 233–262.

Duriez, J., Galusinski, C., 2020. Level set representation on octree for granular material with arbitrary grain shape. In: Šimurda, D., Bodnár, T. (Eds.), Proceedings Topical Problems of Fluid Mechanics 2020. Prague, pp. 64–71.

Duriez, J., Wan, R., Pouragha, M., Darve, F., 2018. Revisiting the existence of an effective stress for wet granular soils with micromechanics. International Journal for Numerical and Analytical Methods in Geomechanics 42 (8), 959–978.

Eliáš, J., 2014. Simulation of railway ballast using crushable polyhedral particles. Powder Technology 264, 458 – 465.

Garcia, X., Latham, J.-P., Xiang, J., Harrison, J., 2009. A clustered overlapping sphere algorithm to represent real particles in discrete element modelling. Géotechnique 59 (9), 779–784.

Gladkyy, A., Kuna, M., 2017. DEM simulation of polyhedral particle cracking using a combined Mohr–Coulomb–Weibull failure criterion. Granular Matter 19 (3), 41.

Golay, F., Lachouette, D., Bonelli, S., Seppecher, P., 2010. Interfacial erosion: A three-dimensional numerical model. Comptes Rendus Mécanique 338 (6), 333 – 337.

Golay, F., Lachouette, D., Bonelli, S., Seppecher, P., 2011. Numerical modelling of interfacial soil erosion with viscous incompressible flows. Computer Methods in Applied Mechanics and Engineering 200 (1), 383 – 391.

Houlsby, G., 2009. Potential particles: a method for modelling non-circular particles in DEM. Computers and Geotechnics 36 (6), 953 – 959.

Jerves, A. X., Kawamoto, R. Y., Andrade, J. E., 2016. Effects of grain morphology on critical state: a computational analysis. Acta Geotechnica 11 (3), 493–503.

Kawamoto, R., Andò, E., Viggiani, G., Andrade, J. E., 2016. Level set discrete element method for three-dimensional computations with triaxial case study. Journal of the Mechanics and Physics of Solids 91, 1–13.

Kawamoto, R., Andò, E., Viggiani, G., Andrade, J. E., 2018. All you need is shape: Predicting shear banding in sand with LS-DEM. Journal of the Mechanics and Physics of Solids 111, 375–392.

Li, L., Marteau, E., Andrade, J. E., 2019. Capturing the inter-particle force distribution in granular material using LS-DEM. Granular Matter 21 (3), 43.

Lin, C.-C., Ching, Y.-T., 1996. An efficient volume-rendering algorithm with an analytic approach. The Visual Computer 12 (10), 515–526.

Love, A., 1892. A treatise on the mathematical theory of elasticity. Cambridge: At the University Press.

Pöschel, T., Buchholtz, V., 1993. Static friction phenomena in granular materials: Coulomb law versus particle geometry. Phys. Rev. Lett. 71, 3963–3966.

Rakhmanov, E. A., Saff, E. B., Zhou, Y. M., 1994. Minimal discrete energy on the sphere. Mathematical Research Letters 1, 647–662.

Sethian, J., 1999. Level set methods and fast marching methods. Cambridge University Press.

Szarf, K., Combe, G., Villard, P., 2009. Influence of the grains shape on the mechanical behavior of granular materials. AIP Conference Proceedings 1145 (1), 357–360, Powders and grains 2009: Proceedings of the 6th international conference on micromechanics of granular media.

Vlahinić, I., Andò, E., Viggiani, G., Andrade, J. E., 2014. Towards a more accurate characterization of granular media: extracting quantitative descriptors from tomographic images. Granular Matter 16 (1), 9–21.

Šmilauer, V., 2010. Cohesive particle model using the Discrete Element Method on the Yade platform. Ph.D. thesis, Czech Technical University in Prague, Faculty of Civil Engineering & Université Grenoble I – Joseph Fourier.

Šmilauer, V., et al., 2015. Yade Documentation 2nd ed. The Yade Project, http://yade-dem.org/doc/.

Wang, X., Tian, K., Su, D., Zhao, J., 2019. Superellipsoid-based study on reproducing 3D particle geometry from 2D projections. Computers and Geotechnics 114, 103131.

Weber, J., 1966. Recherches concernant les contraintes intergranulaires dans les milieux pulvérulents. Bulletin de liaison des Ponts et Chaussées 20, 1–20.

Wensrich, C., Katterfeld, A., 2012. Rolling friction as a technique for modelling particle shape in DEM. Powder Technology 217, 409 – 417.

Zhao, S., Zhao, J., 2019. A poly-superellipsoid-based approach on particle morphology for DEM modeling of granular media. International Journal for Numerical and Analytical Methods in Geomechanics 43 (13), 2147–2169.