Computational Mechanics (2022) 70:763–783
https://doi.org/10.1007/s00466-022-02220-8

ORIGINAL PAPER

Signed distance field framework for unified DEM modeling of granular media with arbitrary particle shapes

Zhengshou Lai[1,2] · Shiwei Zhao[2] · Jidong Zhao[2,3] · Linchong Huang[4]

Received: 5 May 2022 / Accepted: 27 July 2022 / Published online: 9 August 2022

Abstract

This paper presents a signed distance field (SDF) approach for unified discrete element method (DEM) modeling of granular media using arbitrarily shaped particles. The SDF approach employs a generic SDF-based interface defined by an SDF function and a surface projection function to rigorously model particle shapes and their ensuing complications on contact operations in DEM modeling. The signed distance is defined positive inside particles and negative when outside, and the zeroth isosurface of the SDF is conveniently used to represent the particle surface. The surface of a particle is discretized into a set of nodes. Node-to-surface algorithms are formulated to check the signs of the pertaining distance for contact detection. An energy-conserving contact theory is further employed to derive the contact interaction forces according to the contact potential defined on each intruding node. Based on the unified shape-contact description by SDF, specialised grain shape models are further developed to recover classical shape models as special cases, including poly-super-ellipsoid, poly-super-quadrics, spherical harmonics, polyhedron, and level set. A weighted spherical centroidal Voronoi tessellation-based numerical scheme is further developed for rigorous particle surface discretization and reconstruction. Demonstrative examples are presented to validate and showcase the capabilities of the proposed SDF approach for DEM modeling of granular media. The computational aspects, including the memory consumption and computational efficiency of the proposed approach for various particle models, are discussed.

Keywords Discrete element method · Arbitrary-shaped particle · Signed distance field · Contact potential · Spherical harmonics

1 Introduction

Granular materials are important to many engineering and industrial processes. To understand the complicated behavior of granular media, the discrete element method (DEM) [1] has been one of the most popular numerical approaches. A methodological focus of recent DEM developments has been placed on modeling the effect of irregular particle shape. Non-spherical particle models based on different geometries have been developed to better accommodate the shape effect in DEM modeling, including poly-ellipsoids [2], super-ellipsoid [3], poly-super-ellipsoids [4,5], super-quadrics [6–8], polyhedron [9–12], nonuniform rational basis splines [13–15], level set [16,17], and spherical harmonics [18,19]. Recent comprehensive reviews on the developments of irregular-shaped particle models in DEM and their engineering applications can be referred to Zhong et al. [20], Feng [21], He et al. [22]. The development of these particle models has significantly advanced the predictive capability of DEM in modeling realistic granular materials. Nonetheless, the application of these various particle models are largely limited by their specific purposes in modeling a certain, usually not general, class of shapes; and some particle models (e.g., level set or non-convex polyhedron) are too memory demanding and computationally expensive to be applicable to simulating large granular systems. Indeed, a versatile particle model that can accurately represent major shape features of particles in engineering and industrial processes meanwhile offering the best computational efficiency yet does

✉ Shiwei Zhao
ceswzhao@ust.hk

[1] School of Intelligent Systems Engineering, Sun Yat-sen University, Shenzhen 518107, China

[2] Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong SAR, China

[3] HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute, Futian, Shenzhen, China

[4] School of Aeronautics and Astronautics Engineering, Sun Yat-sen University, Shenzhen 518107, China

not exist. The overall performance, in terms of accuracy of shape representation, memory consumption, and computational efficiency, of a particle model is frequently problem dependent, and sometimes, there is a need to employ different models in one single simulation to pursue the optimal performance.

Core to DEM modeling is its contact detection and resolution algorithm. Contact detection refers to identifying inter-particle contacts, whereas contact resolution is to compute the contact geometric features such as contact overlap, contact normal and contact point, which serve as critical ingredients to evaluate contact forces via contact models. For generic convex particles, it is common to adopt the assumption of single inter-particle contact. The contact detection and resolution problem is hence a convex optimization problem that can be readily solved by using any numerical optimization method such as Newton's method [13,23,24], the Levenberg-Marquardt method [4], the Nelder-Mead simplex algorithm [25], and the Gilbert-Johnson-Keerthi algorithm [26,27]. By contrast, for non-convex particles, there might be multiple contacts so that the aforementioned optimization algorithms may fail to find all contacts correctly due to inherent limitations.

For non-convex particles, one popular approach dealing with contact is to decompose a non-convex particle into pieces of convex primitives, thereby converting the non-convex optimization problem into a set of convex optimization. One representative is the so-called clump technique, e.g., the multi-sphere clump approach [28–30] and the glued-convex-particles approach [27]. Although the clump technique has been rather popular in DEM simulations, there are two noteworthy pitfalls. 1) It often needs a large number of convex primitives to approximate a smooth and continuous irregular-shaped particle with high fidelity, thereby significantly increasing the overall computational cost, even though the computation of contact between an individual pair of convex primitives is efficient [31]; 2) It is difficult or even impossible to fully reproduce the true surface curvature by simply clumping convex primitives, making it difficult to properly implement the curvature-based contact model, e.g., the Hertz-Mindlin model [32–36]. Another popular alternative is to discretize particle surfaces into a grid of sub-domains, with the assumption that at most one solution of contact exists in a subdomain and thus the iterative-based optimization methods [37–39] can apply. In so doing, the contact solution is sensitive to the resolution (amount) of subdomains, incurring high computational cost if a high resolution is needed, especially for highly non-convex realistic particles.

Another branch of contact approach for generic irregular-shaped particles (non-convex in particular) is the node-to-surface method [40–42]. The basic idea is to discretize the surface of one particle into a set of nodes to turn the contact detection into an equivalent problem of determining if any surface node of the particle is intruding into another particle. Such kind of algorithm searches for contact solutions in a brute-force manner that is robust but not sufficiently efficient in general. The node-to-surface approach has been successfully adopted in the level-set based DEM [16,17] and spherical-harmonics based DEM [18,19]. Although the node-to-surface approach for contact detection is straightforward, the definition of contact geometric features in the contact resolution step remains some divergence and debates. For example, in the level set-DEM [16,17], the contact normal orientation is computed from the spacial derivatives of the level set function and the contact overlap is calculated as the distance from the intruding node to the intruded particle surface. However, in the spherical harmonics-DEM [19], the contact normal orientation is defined as the mean of the surface normal orientations, respectively, at the intruding node and the surface projection point. The contact force is evaluated based on the contact volume, which is approximated from the predefined prisms that surround the intruding nodes. As discussed in Feng [21], contacts between non-convex and irregular-shaped particles are much more complex than currently assumed. The definitions of contact geometric features in these aforementioned studies [16,19] are not in a consistent but an *ad-hoc* fashion, which may result in an energy increase for an elastic impact and therefore becomes a source for potential numerical instability.

It is also worth noting that for general (concave) polyhedra, besides the aforementioned clump approach and node-to-surface approach, some other algorithms for contact detection and resolution are available in the literature [10,43–47], such as the polygon-based contact approach [10], the mesh Boolean approach [46], node-to-node approach [45], and surface-to-surface contact approach [47]. Polyhedra are flexible to model arbitrary-shaped particles, while the computational efficiency degrades significantly with increasing shape complexity. These polyhedra-based contact approaches are developed particularly for polyhedra and thus may not be applicable to other types of geometries, such as quadrics and spherical harmonics.

In this work, we develop a signed distance field (SDF) approach for DEM modeling of generic irregular-shaped particles. This approach is built on the node-to-surface approach [40,48] for contact detection, but brings the following three new features: 1) particle models are proposed to inherit from a generic interface that provides an SDF function and a surface projection function. The contact detection and resolution algorithms are consistently developed and implemented based on this generic interface, providing a universal applicability to any classes of shapes that implement this interface. We will demonstrate that the implementation of this SDF-based generic interface renders it easy to generate classical geometries, including poly-super-ellipsoid, poly-super-quadrics, spherical harmonics and polyhedron and also facilitates the integration of existing level set particle models; 2) the contact behavior is derived from the theory of contact potential, therefore the total energy of two particles is conserved during the entire colliding process if there is no damping or friction. The energy-conserving contact theory is beneficial to the numerical stability and enables quantitative energy analysis of a DEM simulation; 3) we also offer quantitative analysis of the memory consumption and computational efficiency of the aforementioned particle models, aiming to provide some guidance on the selection of particle models for real applications.

The rest of the paper is organized as follows. Section 2 presents the methodology of the SDF-based contact approach, including the particle model interface, contact algorithms and contact theory. Section 3 provides the exemplified particle models, including poly-super-ellipsoid, poly-super-quadrics, spherical harmonics, polyhedron and level set. Section 4 describes a weighted spherical centroidal Voronoi tessellation (WSCVT)-based scheme for particle surface discretization and reconstruction. Section 5 shows several example DEM simulations to verify and demonstrate the capabilities of the SDF-based approach, and Sect. 6 further discusses about the computational aspects, e.g., memory consumption and computational efficiency. Finally, Sect. 7 summarizes the concluding remarks.

2 Unified SDF-DEM framework

2.1 SDF description of particles

An SDF is such a function $d = \Phi(\boldsymbol{x})$: $\mathcal{R}^3 \rightarrow \mathcal{R}$ that maps a point $\boldsymbol{x}$ to a signed value d. As illustrated in Fig. 1, in the SDF approach, a particle model inherits from an SDF-based generic interface that provides two basic functions:

- Signed distance function: it returns the sign distance at a query point. Without loss of generality, the signed distance is taken as positive if the point is inside the particle surface and negative otherwise. The zeroth isosurface of the SDF is taken to conveniently represent the particle surface.
- Surface projection function: it defines the projection of a given point onto the particle surface. For example, point P is projected onto the surface of Particle A at point Q as shown in Fig. 1. The surface projection function is used to determine contact point.

Conventionally, as in the case of level set-DEM [16], a signed distance is defined as the shortest distance from query point to particle surface, and the surface projection point is the point that provides the shortest distance. In this work, we adopt a broad-sense definition for signed distance which can be either the shortest distance from the query point to the particle surface, the radial distance from the query point to the particle surface, or any other distance that satisfy the condition that the SDF isosurfaces are non-self-intersection. The extended SDF may lose some properties (e.g., the gradient is of norm length) possessed by the conventional SDF defined upon Euclidean distances, while rendering it capable to accommodate various types of particle models and typically efficient for discrete element modeling. With particle shape described by SDF, in this work, the particle properties, such as mass and moment of inertia, are calculated from the triangle mesh reconstructed for the particle surface (i.e., the zeroth isosurface of the SDF). Exemplified particle models with concrete signed distance function and surface projection function will be presented in Sect. 3. The approach of particle surface discretization and reconstruction will be presented in Sect. 4.

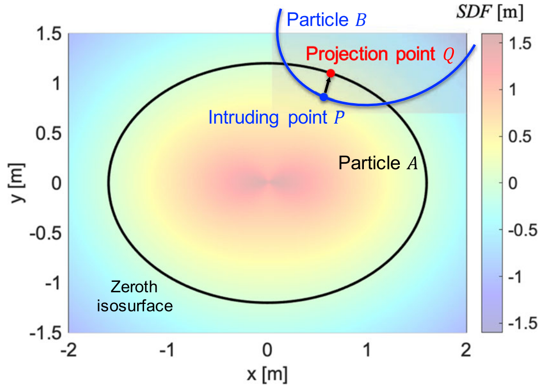


Fig. 1 Illustration of the SDF model for DEM. (Color figure online)

2.2 Contact detection

The node-to-surface approach [16,19,41,42] is adopted to detect the contacts between SDF-based particles. As shown in Fig. 2, for two particles in query, Particle A is described by an SDF, whereas the surface of Particle B is discretizied with a set of nodes. The contact detection problem becomes equivalent to determining if there exists a surface node on Particle B that intrudes into Particle A, which can be easily accomplished by checking the distance signs of the surface nodes.

2.3 Contact resolution

In this work, the contact normal descriptions, including the contact geometric features and contact normal forces, are derived from the distance-potential-based energy-conserving contact theory. For contact shear and frictional forces, the linear-spring model with Coulomb's law is adopted.

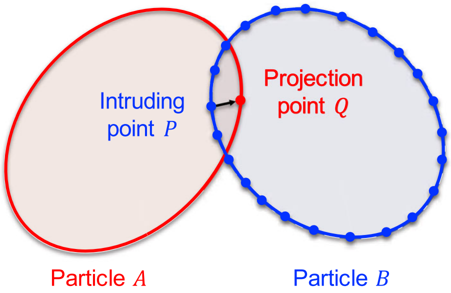


Fig. 2 Illustration of the node-to-surface approach for contact detection. (Color figure online)

2.3.1 Normal force

For two particles in contact, it is assumed that Particle A is in its reference configuration and is located at the origin, and the configuration of Particle B is characterized by its position $\boldsymbol{x}$ and orientation $\boldsymbol{\theta}$ with respect to the body coordinates of Particle A. In reference to the energy-conserving contact theory proposed by Feng [21,46], the inter-particle contact (normal) force $\boldsymbol{F}_n$ and moment $\boldsymbol{M}_n$ can be characterized as

$$\boldsymbol{F}_n = -\frac{\partial w(\boldsymbol{x},\boldsymbol{\theta})}{\partial \boldsymbol{x}} \tag{1}$$

$$\boldsymbol{M}_n = -\frac{\partial w(\boldsymbol{x},\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \tag{2}$$

where $w(\boldsymbol{x},\boldsymbol{\theta})$ denotes the contact potential. As the moment can be represented by a force pair, it is convenient to leave out the contact moment term by translating the contact force from the origin to the so-called contact point $\boldsymbol{x}_c$, given by

$$\boldsymbol{x}_c = \frac{\boldsymbol{n}\times\boldsymbol{M}_n}{F_n} + \lambda\boldsymbol{n} \tag{3}$$

where F_n is the magnitude of contact normal force, $\boldsymbol{n} = \boldsymbol{F}_n/F_n$ is the contact normal orientation, and λ is a free parameter. It is also worth noting that the contact normal is not defined as a prior but is calculated from the direction of contact normal force, and thus it may not be orthogonal to the surface of any particle. The integration of energy-conserving contact theory with the node-to-surface contact algorithm, including the choice of contact potential and parameter λ for contact point, is elaborated as following.

In this work, we define the contact potential $w(\boldsymbol{x},\boldsymbol{\theta})$ as a function of the signed distances d_i, such that

$$w(\boldsymbol{x},\boldsymbol{\theta}) = \sum_{P_i}^{P_i\in B\rightarrow A} k_i\mathcal{F}(d_i) \tag{4}$$

where k_i is a stiffness-dependent parameter, $\mathcal{F}(d_i)$ is a function that satisfies $\mathcal{F}(0)=0$, and $P_i \in B \rightarrow A$ represents the surface nodes of Particle B that intrude into Particle A. In particular, a three-halves-power formulation of contact potential can be written as

$$\mathcal{F}(d_i) = \frac{2}{3}\beta d_i^{3/2} \tag{5}$$

which resembles a Hertzian-like nonlinear contact force-displacement formulation and is adopted in this work for the example DEM simulations. Parameter β is a coefficient to account for the effect of surface nodes distribution; it can be taken as 1 for evenly distributed surface nodes or as a function of the squared node spacing (i.e., distance between adjacent surface nodes) for an uneven distribution. Since the contact potential is defined upon intruding nodes, it is convenient to set the reference point of the contact potential exactly at the intruding nodes. Benefiting from the fact that the signed distance d_i is rotation-invariant with respect to the intruding node P_i, the contact moment with respect to this choice of reference point becomes zero. Therefore, the contact normal behavior can be simplified as

$$\boldsymbol{F}_n = \sum_{P_i}^{P_i\in B\rightarrow A} \boldsymbol{F}_{n,i} \tag{6}$$

$$\boldsymbol{F}_{n,i} = -k_i\mathcal{F}'(d_i)\frac{\partial d_i}{\partial \boldsymbol{x}} \tag{7}$$

$$\boldsymbol{n}_i = \boldsymbol{F}_{n,i}/F_{n,i} \tag{8}$$

$$\boldsymbol{x}_{c,i} = \boldsymbol{x}_{P_i} + \lambda\boldsymbol{n}_i \tag{9}$$

where the subscript i indicates the node index. Note that Eq. (9) defines the contact point $\boldsymbol{x}_{c,i}$ only on the line that passes the reference point P_i and points to direction $\boldsymbol{n}_i$. In this work, the exact contact point and thus the parameter λ are obtained as follows. As shown in Fig. 3, for an intruding node P_i, the middle of the node and its corresponding surface projection point Q_i is first obtained as point M_i. The contact point is then defined as the projection (i.e., E_i in Fig. 3) of this middle point onto the line defined by Eq. (8).

In implementation, the derivative term in Eq. (7) is calculated using the finite difference approach by translating Particle B with a tiny displacement, respectively, in each dimension. The sampling interval (i.e., the tiny displacement) is taken as the average node spacing multiplied by a user-define parameter (e.g., 1.0e-6 used in this work). Such an adoption of sampling interval makes it relatively independent from particle size. In addition, the effect of sampling interval on the differential would become minor if the sampling interval gets reasonably small, making the results relatively independent from the exact value of sampling interval (i.e., the user-define parameter). Another factor that might affect the robustness of the numerical differentiation is the complexity (e.g., the smoothness of the isosurfaces) of particle

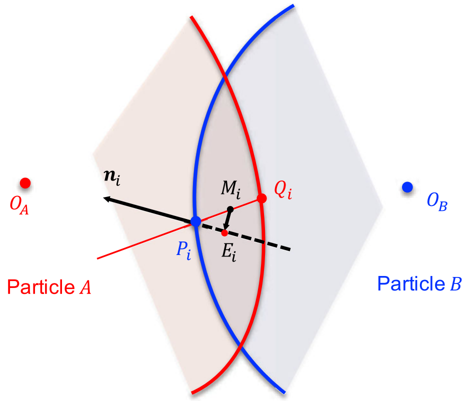


Fig. 3 Illustration of the definition of contact point. (Color figure online)

SDF, where the effect can be mitigated by taking small sampling interval. It is worth noting, however, that the use of too small sampling interval may lead to truncation errors due to limited machine precision and thus may bring a potential of numerical instability.

2.3.2 Tangential force

In the node-based linear-spring model for tangential contact force, each intruding node is assumed to carry a tangential force, which is updated incrementally according to

$$F_s = F_s^0 - k_t\delta_s \tag{10}$$

$$F_t = F_t^0 - k_t\delta_t \tag{11}$$

where F_s and F_t are the tangential contact forces in the tangential directions $\boldsymbol{s}$ and $\boldsymbol{t}$, respectively; the superscript 0 indicates the tangential forces at the previous timestep; k_t is the contact tangential stiffness; and δ_s and δ_t, respectively, are the relative displacements in the contact tangential directions. The tangential contact force is tracked in two tangential directions to facilitate the transformation of contact frictional force due to contact plane rotation [49]. Specifically, when a contact is first activated, the tangential direction $\boldsymbol{s}$ is initialized as the projection of the global x or y direction (whichever is not parallel with the contact normal) onto the contact plane, and the tangential direction t is initialized as $\boldsymbol{t} = \boldsymbol{s} \times \boldsymbol{n}$. Then, for each time step, due to rigid particle motion and thus the rotation of contact plane, tangential direction $\boldsymbol{s}$ is updated accordingly with two rotations, given by

$$\boldsymbol{s}_1 = \boldsymbol{s}^* - \boldsymbol{s}^* \times (\boldsymbol{n}^* \times \boldsymbol{n}) \tag{12a}$$

$$\boldsymbol{s}_2 = \boldsymbol{s}_1 - \boldsymbol{s}_1 \times \bar{\boldsymbol{\omega}}\Delta t \tag{12b}$$

$$\boldsymbol{s} = \boldsymbol{s}_2/||\boldsymbol{s}_2|| \tag{12c}$$

where the first rotation, which gives $\boldsymbol{s}_1$, is about the line common to the previous and present contact planes; the second rotation, which gives $\boldsymbol{s}_2$, is about the present contact normal; * denotes the values of previous time step; $\bar{\boldsymbol{\omega}}$ is the projection of the average spin velocity of the two particles in contact onto the contact normal; and Δt is the timestep. Lastly, the tangential direction $\boldsymbol{t}$ is obtained from $\boldsymbol{t} = \boldsymbol{s} \times \boldsymbol{n}$. Note that each intruding node P_i will have a tangential contact force, whereas the subscript i is left out in the equations for simplicity. The relative displacements δ_s and δ_t are calculated as

$$\delta_s = \boldsymbol{v}_{rel} \cdot \boldsymbol{s}\Delta t \tag{13}$$

$$\delta_t = \boldsymbol{v}_{rel} \cdot \boldsymbol{t}\Delta t \tag{14}$$

$$\boldsymbol{v}_{rel} = \boldsymbol{v}_{c_A} - \boldsymbol{v}_{c_B} + \boldsymbol{\omega}_A \times \boldsymbol{b}_A - \boldsymbol{\omega}_B \times \boldsymbol{b}_B \tag{15}$$

where $\boldsymbol{v}_{rel}$ is the relative velocity at a contact point; $\boldsymbol{v}_{c_A}$ and $\boldsymbol{v}_{c_B}$ are the velocities of the centroid of particle A and B, respectively; $\boldsymbol{\omega}_A$ and $\boldsymbol{\omega}_B$ are the spin velocities of particle A and B, respectively; $\boldsymbol{b}_A$ and $\boldsymbol{b}_B$ are the corresponding branch vectors, i.e., the vector starting from particle centroid and directed towards contact point; and symbol $\cdot$ indicates the dot production. In implementation, at each time step, the contact normal and contact point will be calculated first via Eqs. (8) and (9), respectively. Then, the tangential directions will be updated accordingly via Eq. (12) to account for the rotation of contact interface due to rigid particle motion. Next, the relative velocity of the particle surfaces at the contact point is calculated via Eq. (15). Finally, the relative tangential displacements are computed from Eqs. (13) and (14), respectively.

To further incorporate the Coulomb's law of friction [34, 50], the tangential contact forces are constrained as

$$F'_{st,i} = \min\left(\sqrt{F_{s,i}^2 + F_{t,i}^2}, \mu F_{n,i}\right) \tag{16}$$

$$F'_{s,i} = F'_{st,i}\frac{F_{s,i}}{\sqrt{F_{s,i}^2 + F_{t,i}^2}} \tag{17}$$

$$F'_{t,i} = F'_{st,i}\frac{F_{t,i}}{\sqrt{F_{s,i}^2 + F_{t,i}^2}} \tag{18}$$

where the symbol prime $'$ is used to indicate the updated tangential contact force based on the Coulomb's law of friction, the subscript i indicates the node index, F_{st} is the total tangential contact force, and μ is the friction coefficient. It is worth noting that the present Coulomb-based tangential contact force model does not fully consider the complete force history and the partial slip effect when slide occurs, which will not make much difference in the cases of continuous loading but will in the cases of small-strain cyclic loading (e.g., ratcheting). This is a difficult problem in DEM which is often overlooked, and currently there is not perfect solution for it.

3 Exemplified particle models

The proposed unified SDF-based framework is applicable to recovering a broad range of particle models as special cases, such as poly-super-ellipsoid, poly-super-quadrics, spherical harmonics, polyhedron, and level set, that have been prevailingly employed to model non-spherical particles in the DEM community. Thanks to the unified framework, it is allowable to use all of these particle models together in one DEM simulation for a better performance in terms of shape representation accuracy, memory consumption and computational efficiency.

3.1 Poly-super-ellipsoid

The surface of a poly-super-ellipsoid particle can be formulated as [4]

$$f(\boldsymbol{x}) = \left(\left|\frac{x}{r_x}\right|^{\frac{2}{\epsilon_1}} + \left|\frac{y}{r_y}\right|^{\frac{2}{\epsilon_1}}\right)^{\frac{\epsilon_1}{\epsilon_2}} + \left|\frac{z}{r_z}\right|^{\frac{2}{\epsilon_2}} - 1 = 0 \tag{19}$$

where r_x, r_y, and r_z are the semi-major axis lengths in the direction of x, y, and z, respectively; and ϵ_1 and ϵ_2 are the shape parameters determining the blockiness of the particle. A poly-super-ellipsoid particle is convex if ϵ_1 and ϵ_2 are within range (0, 2), and is non-convex otherwise. The semi-major axis length r_x is taken as r_{x^-} for negative x and r_{x^+} for positive x (same for r_y and r_z). Figure 4 shows an example of poly-super-ellipsoid particle. The poly-super-ellipsoid degrades to a superellipsoid if r_{x^-} and r_{x^+}, r_{y^-} and r_{y^+}, r_{z^-} and r_{z^+} are, respectively, equal.

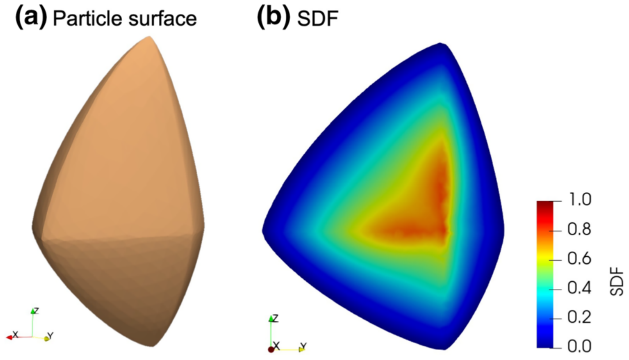


Fig. 4 An exemplified poly-super-ellipsoid particle ($r_{x^-} = 0.5$, $r_{x^+} = 1.0$, $r_{y^-} = 1.5$, $r_{y^+} = 0.5$, $r_{z^-} = 1.0$, $r_{z^+} = 1.5$, $\epsilon_1 = \epsilon_2 = 1.5$): **a** particle surface, **b** SDF at cross-section y-o-z. (Color figure online)

Given a query node P, the surface projection function is defined, in a radial distance fashion, as

$$\boldsymbol{x}_Q = (x_Q, y_Q, z_Q) = (cx_P, cy_P, cz_P) \tag{20}$$

where x, y, and z are the coordinates, respectively, in each dimension; subscripts P and Q indicate points P and Q, respectively; and c is a scaling coefficient that makes point Q satisfy Eq. (19). By casting point Q into Eq. (19), the coefficient c can be solved as

$$c = \left(\frac{1}{f(\boldsymbol{x}_P) + 1}\right)^{\frac{\epsilon_2}{2}} \tag{21}$$

The SDF is then defined as the distance between points P and Q that

$$\Phi(\boldsymbol{x}_P) = (c - 1)||\boldsymbol{x}_P|| \tag{22}$$

where $||\boldsymbol{x}_P||$ represents the Euclidean norm length of $\boldsymbol{x}_P$.

3.2 Poly-super-quadrics

The surface of a poly-super-quadrics particle is formulated as

$$f(\boldsymbol{x}) = \left|\frac{x}{r_x}\right|^{\frac{2}{\epsilon_x}} + \left|\frac{y}{r_y}\right|^{\frac{2}{\epsilon_y}} + \left|\frac{z}{r_z}\right|^{\frac{2}{\epsilon_z}} - 1 = 0 \tag{23}$$

where r_x, r_y, and r_z are the semi-major axis lengths in the direction of x, y, and z, respectively; and ϵ_x, ϵ_y and ϵ_z are the shape parameters determining the blockiness of the particle. Similar to the poly-super-ellipsoid case, a poly-super-quadrics particle is convex if ϵ_x, ϵ_y and ϵ_z are within range (0, 2), and is non-convex otherwise. The semi-major axis length r_x is taken as r_{x^-} for negative x and r_{x^+} for positive x (same for r_y, r_z, ϵ_x, ϵ_y and ϵ_z). Figure 5 shows an example of poly-super-quadrics particle.

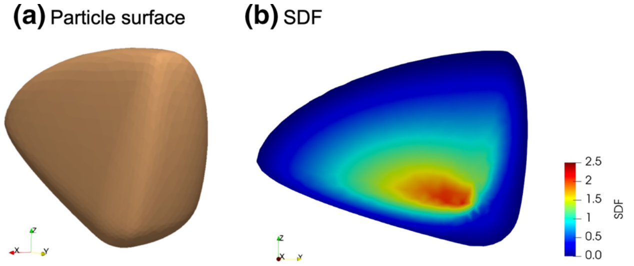


Fig. 5 An exemplified poly-super-quadrics particle ($r_{x^-} = 0.5$, $r_{x^+} = 1.0$, $r_{y^-} = 2.5$, $r_{y^+} = 0.5$, $r_{z^-} = 1.7$, $r_{z^+} = 0.5$, $\epsilon_{x^-} = 0.5$, $\epsilon_{x^+} = 1.0$, $\epsilon_{y^-} = 1.5$, $\epsilon_{y^+} = 1.0$, $\epsilon_{z^-} = 1.2$, $\epsilon_{z^+} = 0.8$): **a** particle surface, **b** SDF at cross-section y-o-z. (Color figure online)

For the poly-super-quadrics particle model, the surface projection function is written as

$$\boldsymbol{x}_Q = (x_Q, y_Q, z_Q) = (c_x x_P, c_y y_P, c_z z_P) \tag{24}$$

where c_x, c_y, and c_z are the scaling coefficients that make point Q satisfy Eq. (23), and are calculated as

$$c_i = \left(\frac{1}{f(\boldsymbol{x}_P) + 1}\right)^{\frac{\epsilon_i}{2}} \tag{25}$$

where $i = x, y, z$, indicating the dimensions. And, the SDF is calculated as

$$\Phi(\boldsymbol{x}_P) = \mathrm{sgn}(-f(\boldsymbol{x}_P))||\boldsymbol{x}_P - \boldsymbol{x}_Q|| \tag{26}$$

where $\mathrm{sgn}(x)$ represents the sign of variable x.

3.3 Spherical harmonics

Spherical harmonics, as a three-dimensional variant of the two-dimensional Fourier series, has been popularly adopted to characterize particle morphology and generate virtual particles [51–54]. Given a star-shaped particle, its surface points can be expressed in spherical coordinates (r, θ, ϕ) and characterized by a function $r(\theta, \phi)$, where r is the radial distance, and θ and ϕ are the polar and azimuth angles, respectively. In general, a radial distance function can be represented by spherical harmonics, that

$$r(\theta, \phi) = \sum_{n=0}^{N} \sum_{m=-n}^{n} a_{n,m} Y_n^m(\theta, \varphi), \ \theta \in [0, \pi], \ \phi \in [0, 2\pi) \tag{27}$$

with

$$Y_n^m(\theta, \varphi) = \begin{cases} (-1)^m \sqrt{2}\sqrt{\frac{(2n+1)(n+m)!}{4\pi(n-m)!}} P_n^{-m}(\cos\theta)\sin(-m\varphi) & \text{if } m < 0 \\ \sqrt{\frac{2n+1}{4\pi}} P_l^m(\cos\theta) & \text{if } m = 0 \\ (-1)^m \sqrt{2}\sqrt{\frac{(2n+1)(n-m)!}{4\pi(n+m)!}} P_n^{m}(\cos\theta)\cos(m\varphi) & \text{if } m > 0 \end{cases} \tag{28}$$

where N is the spherical harmonics order; n and m are the spherical harmonics basis indexes; $a_{n,m}$ are the spherical harmonics coefficients, and $P_n^m(\cos\theta)$ is the associated Legendre polynomial function of variable $\cos\theta$. Figure 6 shows an example of spherical harmonics particle.

For the SDF-based spherical harmonics particle model, the surface project function is formulated as

$$\boldsymbol{x}_Q = (x_Q, y_Q, z_Q) = (c x_P, c y_P, c z_P) \tag{29}$$

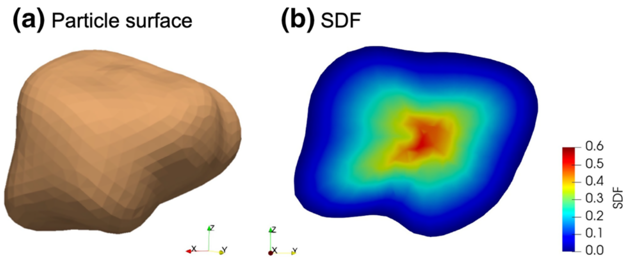


Fig. 6 An exemplified spherical harmonics particle: **a** particle surface, **b** SDF at cross-section y-o-z. (Color figure online)

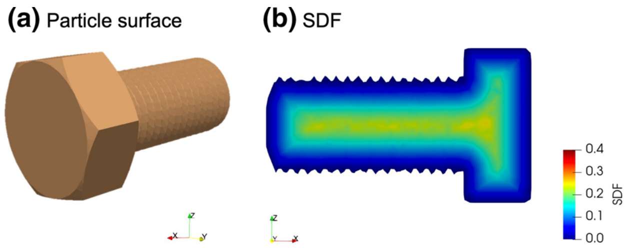


Fig. 7 An exemplified polyhedron particle: **a** particle surface, **b** SDF at cross-section y-o-z. (Color figure online)

with the scaling coefficient c calculated as

$$c = \frac{r(\theta_P, \phi_P)}{||\boldsymbol{x}_P||} \tag{30}$$

The SDF is calculated as

$$\Phi(\boldsymbol{x}_P) = r(\theta_P, \varphi_P) - ||\boldsymbol{x}_P|| \tag{31}$$

3.4 Polyhedron

Polyhedron has been popularly adopted in DEM for modeling irregular-shaped particles [26,55]. The surface of a polyhedron can be described by a set of triangle mesh. Thus, the SDF of a polyhedron can be defined as the shortest distance from the query point to the triangle mesh, whereas the point providing the shortest distance is taken as the surface projection. The SDF of a polyhedron has been prevailingly employed in computer graphics with popularly implemented libraries [56,57]. An open-source code, *libigl* (A simple C++ geometry processing library) [58], is adopted in this work to obtain the SDF of polyhedral particles. The SDF function in *libigl* implements the axis-aligned bounding box hierarchy algorithm, which offers an efficient and robust way to query the signed distance and surface projection. Figure 7 shows an example of polyhedron particle.

3.5 Level set

Level set is a prevalent method to capture the motion of interfaces in many areas of computational physics [59–61]. It has been previously adopted by Kawamoto et al. [16,17] to develop the level set-based particle model in DEM for modeling realistic particles. In this work, we show that the level set particle model can also be integrated into the proposed unified SDF-based framework via the generic SDF-based particle interface. For the sake of completeness, the following provides a brief introduction to the level set model.

Considering a cuboid domain with dimensions $L_x \times L_y \times L_z$, the domain is discretized into a uniform grid of $N_x \times N_y \times N_z$ locations with spacing intervals l_x, l_y and l_z, respectively, in each dimension. The matrix that stores the distance values at the grid of locations is denoted as Φ_{ijk}, where i, j and k indicate the indices of the grid nodes. Given a query point P, the indices of the grid cell that encloses this point are first obtained by

$$ID_x = \min(N_x - 2, \max(0, \text{floor}((x_P - x_C)/l_x))) \quad (32)$$

$$ID_y = \min(N_y - 2, \max(0, \text{floor}((y_P - y_C)/l_y))) \quad (33)$$

$$ID_z = \min(N_z - 2, \max(0, \text{floor}((z_P - z_C)/l_z))) \quad (34)$$

where ID_x, ID_y, and ID_z are the indices; x_C, y_C, and z_C are the coordinate of the lowest corner of the cuboid domain; and floor(x) indicate the largest integer smaller than x. Note that the indices are assumed to start from 0, and they are constrained between 0 and $N_x - 2$ to prevent the error of out of array range. The signed distance of point P can then be calculated as

$$\Phi(\boldsymbol{x}_P) = \sum_{a=0}^{1}\sum_{b=0}^{1}\sum_{c=0}^{1}[(1-a)(1-x)+ax] [(1-b)(1-y)+by][(1-c)(1-z)+cz] \Phi_{[ID_x+a][ID_y+b][ID_z+c]} \quad (35)$$

where x, y, z represent the local coordinates of point P with respect to the grid cell, and are calculated as

$$x = \min(1, \max(0, (x_P - x_C)/l_x - ID_x) \quad (36)$$

$$y = \min(1, \max(0, (y_P - y_C)/l_y - ID_y) \quad (37)$$

$$z = \min(1, \max(0, (z_P - z_C)/l_z - ID_z) \quad (38)$$

The surface projection point Q is calculated as

$$\boldsymbol{x}_Q = (x_Q, y_Q, z_Q) = (x_P, y_P, z_P) - \Phi(\boldsymbol{x}_p)\frac{\nabla\Phi(\boldsymbol{x}_p)}{||\nabla\Phi(\boldsymbol{x}_p)||} \quad (39)$$

where $\nabla\Phi(\boldsymbol{x}_p) = (\nabla_x\Phi, \nabla_y\Phi, \nabla_z\Phi)$ represent the gradients of $\Phi(\boldsymbol{x}_p)$ with respect to dimensions x, y and z, respectively, and are calculated as

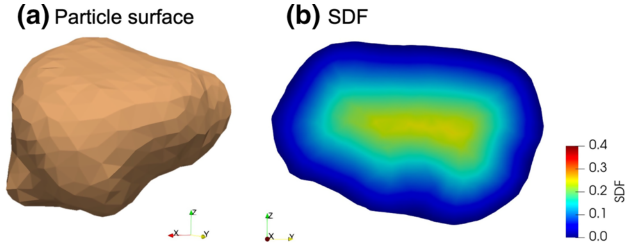


Fig. 8 An exemplified level set particle: **a** particle surface reconstructed using marching cubes algorithm based on the discrete level set values, **b** SDF at cross-section y-o-z. (Color figure online)

$$\nabla_x\Phi(\boldsymbol{x}_p) = \sum_{a=0}^{1}\sum_{b=0}^{1}\sum_{c=0}^{1}[2a-1][(1-b)(1-y)+by] [(1-c)(1-z)+cz]\Phi_{[ID_x+a][ID_y+b][ID_z+c]} \quad (40)$$

$$\nabla_y\Phi(\boldsymbol{x}_p) = \sum_{a=0}^{1}\sum_{b=0}^{1}\sum_{c=0}^{1}[(1-a)(1-x)+ax][2b-1] [(1-c)(1-z)+cz]\Phi_{[ID_x+a][ID_y+b][ID_z+c]} \quad (41)$$

$$\nabla_z\Phi(\boldsymbol{x}_p) = \sum_{a=0}^{1}\sum_{b=0}^{1}\sum_{c=0}^{1}[(1-a)(1-x)+ax] [(1-b)(1-y)+by][2c-1] \Phi_{[ID_x+a][ID_y+b][ID_z+c]} \quad (42)$$

An example of level set particle is shown in Fig. 8.

4 Particle surface discretization and reconstruction

For continuous-function particle models such as poly-super-ellipsoid, poly-super-quadrics and spherical harmonics, the particle surface needs to be discretized into query nodes for contact detection. Given a particle surface parameterized with spherical coordinates (θ, ϕ), it is ready to discretize the surface by drawing samples in the parametric space, e.g., the equal angle grid approach, the icosahedral subdivision approach and the Fibonacci or golden spiral lattice approach [12]. Nonetheless, these conventional sampling approaches have some limitations. For example, equal angle grid approach may result in severely dense points in the vicinity of poles, whereas the other two could provide a perfectly even point distribution (in the unit sphere parametric space) but may still suffer from the non-distance-preserving issue [12]. To approach more uniform sampling, a novel sampling scheme is proposed here based on weighted spherical centroidal Voronoi tessellation (WSCVT) to offer two extra advantages: (1) it can be used to obtain an arbitrary number of surface points, and (2) it offers a novel way to control the local density of the points.

4.1 Methodology of WSCVT

In general, a Voronoi tessellation (VT) is a partition of a domain into regions based on the distances to points in a specific subset of the domain. The points are often called Voronoi seeds, and the regions are called Voronoi cells. Each Voronoi seed has a corresponding Voronoi cell, which consists of all points closer to that seed than to any other. Centroidal Voronoi tessellation (CVT) is a typical tessellation in which the seed of a Voronoi cell coincides with its centroid (i.e., the arithmetic mean or center of Voronoi cell). More specially, a weighted CVT is the tessellation in which each centroid is weighted according to a certain function [62]. Different from conventional VT that is defined in Cartesian space, spherical VT is a variant of VT in which the Voronoi seeds, separation distance and the Voronoi cells are defined on a unit sphere. Figure 9 shows an example of a spherical VT. In the spherical VT, the distance between two points are measured as the great-arc distance, i.e., the length of the minor arc of the great circle on the unit sphere. Each Voronoi cell then consists of a set of great arcs. On the basis of spherical VT, the spherical CVT and weighted spherical CVT could be defined accordingly. Interested readers are referred to Larrea et al. [63] for more detail.

In the WSCVT-based sampling scheme, each Voronoi seed is selected as a sample of parametric coordinate $(\theta, \ \phi)$. Surface points are then calculated from these parametric coordinates based on the specific spherical parametrization. Figure 10 shows two examples of the particle surface reconstructed based on WSCVT. The first example is weighted by constant. In this case, the acquired parametric coordinates are evenly distributed in the unit sphere space. The result of constant WSCVT is similar to that of the golden spiral lattice approach [21]. The second example is weighted by the radial distance of particle surface points. In this case, the Voronoi seeds at the area with large radial distance (i.e., the area in blue) are specified with relatively small weights, whereas the seeds at the area with small radial distance (i.e., the area in red) are specified with larger weights. Consequently, the area with large radial distance is sampled with more points and vice versa, and the resultant point samples on particle surface would be more evenly distributed after scaled by the radial distance. The radial distance-based seed weights could mitigate the non-distance-preserving issue of the golden spiral lattice sampling scheme.

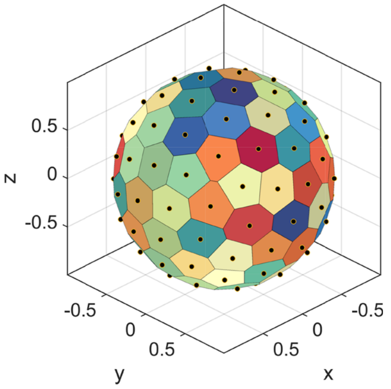


Fig. 9 Example of spherical CVT. The black dots represent Voronoi seeds, and the spherical polygons represent the Voronoi cells. The color is randomly selected for visualizing different Voronoi cells. (Color figure online)

4.2 Construction algorithm

The weighted centroid of triangle and centroid of spherical polygon are critical ingredients to construct WSCVT and are introduced as follows. Given a triangle with vertices denoted as A, B and C, the weighted centroid is the location O that satisfies

$$d_w(\boldsymbol{x}_O, \boldsymbol{x}_A, w_A) = d_w(\boldsymbol{x}_O, \boldsymbol{x}_B, w_B) = d_w(\boldsymbol{x}_O, \boldsymbol{x}_C, w_C) \tag{43}$$

where $\boldsymbol{x}_A$, $\boldsymbol{x}_B$, $\boldsymbol{x}_C$ and $\boldsymbol{x}_O$ are the coordinates of points A, B, C and O, respectively; w is the corresponding weights at the vertices; and d_w is the weighted distance function, given by

$$d_w(\boldsymbol{x}_O, \boldsymbol{x}_A, w_A) = ||\boldsymbol{x}_A - \boldsymbol{x}_O||^2 - w_A \tag{44}$$

where the symbol $||\boldsymbol{x}||$ indicates the Euclidean distance.

The weighted centroid of a triangle may not always exist for a certain set of weights. To overcome this issue, an approximation of the weighted centroid of a triangle is introduced as a workaround. As shown in Fig. 11(a), the weighted middles of each edge of the triangle are first obtained as O_{AB}, O_{BC} and O_{CA}, where the weighted middle O_{AB} of edge AB is defined as the point that satisfies $d_w(\boldsymbol{x}_{O_{AB}}, \boldsymbol{x}_A, w_A) = d_w(\boldsymbol{x}_{O_{AB}}, \boldsymbol{x}_B, w_B)$ on the edge. One can then draw three lines through the weighted middle of each edge and perpendicular to the corresponding edge to obtain the intersection of each pair (see e.g., I_x, I_y and I_z). Finally, the algorithmic mean of these three intersections is taken as the weighted centroid of the triangle. In addition, since WSCVT is defined on a unit sphere, the weighted centroid of the triangle is normalized by its Euclidean distance to snap it onto the unit sphere.

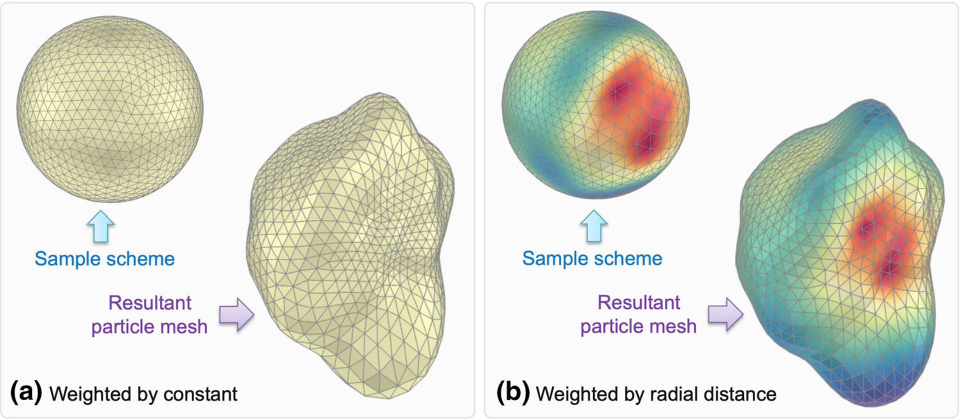
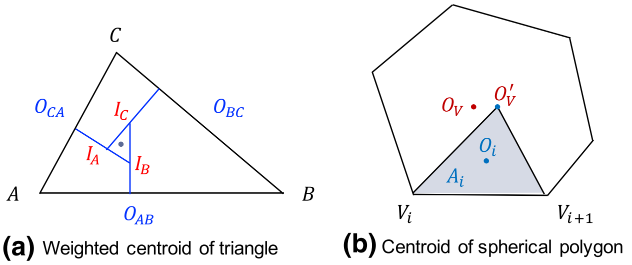


Fig. 10 Illustrations of surface point sampling based on the WSCVT: **a** weighted by constant, and **b** weighted by radial distance. The vertices on the unit sphere represent the sampled parametric coordinates, and the irregular-shaped triangular mesh represents the reconstructed particle surface. (Color figure online)



(a) Weighted centroid of triangle **(b)** Centroid of spherical polygon

Fig. 11 Illustration of the calculation of **a** weighted centroid of triangle and **b** centroid of spherical polygon for WSCVT. (Color figure online)

As shown in Fig. 11(b), the centroid of spherical polygon is calculated as

$$\boldsymbol{x}_{O_V} = \frac{\Sigma_i^N A_i \boldsymbol{x}_{O_i}}{\Sigma_i^N A_i} \tag{45}$$

where A_i is the planar (Euclidean) area of triangle $V_i V_{i_1} O'_V$ with O'_V being an initial guess of the centroid which can be taken as the algorithmic mean of all vertices of the spherical polygon; $\boldsymbol{x}_{O_i}$ is the centroid of triangle $V_i V_{i_1} O'_V$; and the subscript i indicate the index and N is the total number of vertex. Also, $\boldsymbol{x}_{O_V}$ will be normalized by its Euclidean distance to snap to the unit sphere. It should be noted that, for simplicity and efficiency, the calculation of weighted centroid of triangle and centroid of spherical polygon has been simplified by using Euclidean distance and planar area instead of the exact great-circle distance and spherical area, which, however, may introduce negligible errors for adequate Voronoi seeds.

With the definition of weighted centroid of triangle and centroid of spherical polygon, a modified algorithm from Larrea et al. [63], Yang et al. [64] is proposed to construct WSCVT as described in Algorithm 1. Based on the radial distance of surface points, the weight reads

$$w = \frac{S}{\pi N r^2} \tag{46}$$

where S is the surface area of particle; N is the number of Voronoi seeds; and r is the radial distance of the particle surface point in the radial direction of corresponding Voronoi seed. Other formulations of weights for surface point sampling with desired patterns (e.g., sharp corner preserving) merit further exploration but is beyond the scope of this work.

Algorithm 1 Weighted spherical centroidal Voronoi tessellation (WSCVT)

```
Input: Weight function that defines on unit sphere.
Output: Voronoi seeds and cells.
1: Initialize a random set of Voronoi seeds.
2: Update the weights of the Voronoi seeds based on their current positions.
3: Compute the Delaunay triangulation of the Voronoi seeds using the convex hull algorithm.
4: Compute the weighted centroid of each Delaunay triangle.
5: Obtain the Voronoi cell that corresponds to a Voronoi seed, by collecting all the weighted centroids of Delaunay triangles containing the Voronoi seed and sorting the centroids in counter-clockwise order. Repeat this step for all Voronoi seeds.
6: Move the Voronoi seeds to the centroids of the corresponding Voronoi cells.
7: Repeat steps 2-6 until difference between the Voronoi seeds and Voronoi cell centroidss are within a preset tolerance or a maximum number of iterations is reached.
```

5 Simulation demos

To validate and demonstrate the capabilities of the unified SDF-based DEM framework, we provide a series of representative simulations, including (1) random packing, (2) column collapse, (3) nut spinning into bolt, and (4) triaxial compression. They represent some of classical problems ranging from dynamic, quasi-static to multi-contacts in computational particle mechanics.

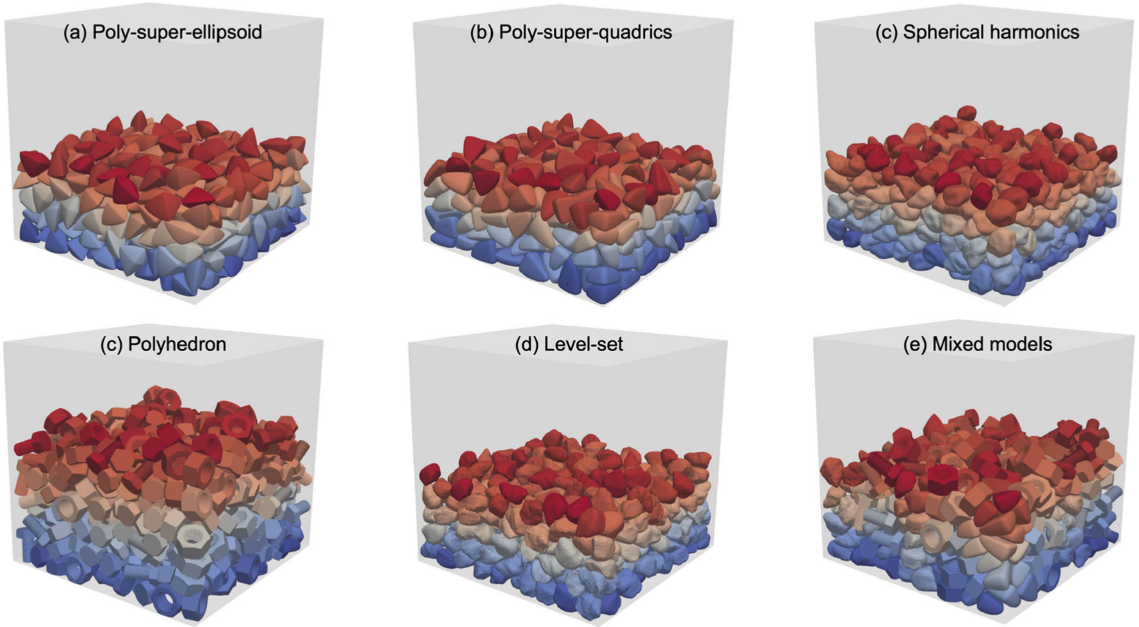


Fig. 12 Snapshots of granular packing for different particle models. The packing consists of 500 particles, which are colored by their indicies for visualization. (animation in supplementary). (Color figure online)

5.1 Random packing

In the random packing example, particles are iteratively inserted into the top part of a box container with a 1.0-by-1.0 m base. Meanwhile, the particles are allowed to settle down under gravity and reach equilibrium. Cases of different particle models, i.e., the particle models described in Sect. 3, are considered. In all cases, the particles are assumed to have an equivalent size (the diameter of equal volume sphere) of 0.1 m and density 2,650 kg/m^3. The normal stiffness parameter k_i is 2 KN/m$^{1/2}$, shear stiffness k_t is 10 KN/m, and the contact frictions μ is 0.3. Parameter β (i.e., in Eq. (5)) is taken as 1 for simplicity. It should be noted that in the context of energy-conserving contact theory, the unit and magnitude of the contact normal stiffness parameter k_i are different from those of the conventional-sense stiffness of linear spring contact model. In practice, the contact stiffness needs to be determined through a calibration process. The classical velocity-based damping force and moment [13,14] are considered with the damping coefficient being 3.0. The poly-super-ellipsoid, poly-super-quadrics and spherical harmonics particles are discretized with 1,000 surface nodes, whereas the polyhedron and level set particles take the vertices of the (reconstructed) surface triangle mesh as surface nodes. Figure 12 shows the snapshots of the particle packing for different particles models, where the packing consists of 500 particles. In all cases, the particles eventually form a stable configuration without observable unrealistic overlaps, indicating the good capability and stability of the SDF-based DEM approach for modeling irregular-shaped particles and accommodating various particle models.

For quantitative verification, the packing density is calculated and compared with the results of existing numerical simulations and physical experiments. This time, we consider a special case of poly-super-ellipsoid with $\eta r_{x^\pm} = \eta r_{y^\pm} = r_{z^\pm}$, and $\epsilon_1 = \epsilon_2 = \zeta$, where η and ζ are the aspect ratio and blockiness indices, respectively. As depicted in Fig. 13, η less than 1 gives oblate particles and grater than 1 gives prolate particles. The packing density of this type of super-ellipsoid has been previously studied in Donev et al. [65], Jiao et al. [66], Delaney and Cleary [67], Zhou et al. [68], Zhao et al. [69], the results of which will be adopted as the benchmarks of the present SDF-based DEM. Following the packing procedure, packings of particles with different aspect ratios and blockiness are generated. Specially, 8,000 frictionless particles of size 0.05 m are used in this case, while the other model parameters are kept the same. Figure 14 plots the relationship between the packing solid volume fraction, aspect ratios and blockiness. The evolution of solid volume fraction with aspect ratio increasing from 0.4 to 2.5 presents an “M”-like profile, and it presents a “V”-like profile for the solid volume fraction and blockiness (within range 0.4∼1.4) relationship. The results of the SDF-based DEM matches reasonably well with those in the literature. The shown dependency in the results are mainly due to the inconsistent contact theories and contact parameters used in these works. For example, the common normal contact approach with single maximum penetration is adopted in Zhao et al. [69], whereas this work employs the SDF-based contact approach.

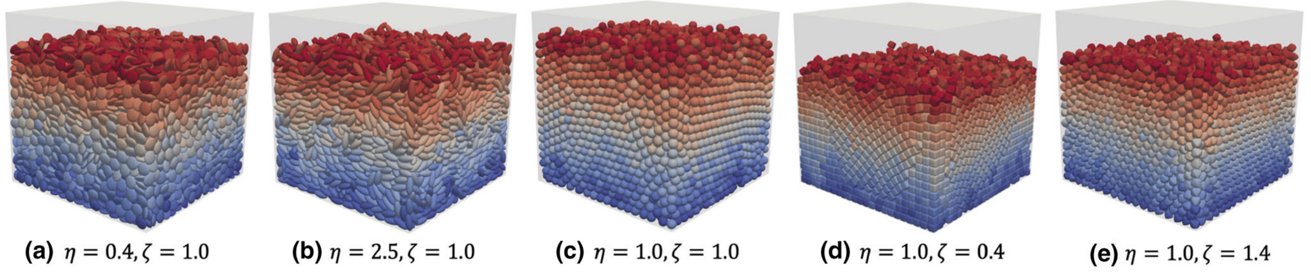

(a) $\eta = 0.4, \zeta = 1.0$
(b) $\eta = 2.5, \zeta = 1.0$
(c) $\eta = 1.0, \zeta = 1.0$
(d) $\eta = 1.0, \zeta = 0.4$
(e) $\eta = 1.0, \zeta = 1.4$

Fig. 13 Snapshots of granular packing of 8,000 super-ellipsoid particles with different aspect ratios and blockiness. (Color figure online)

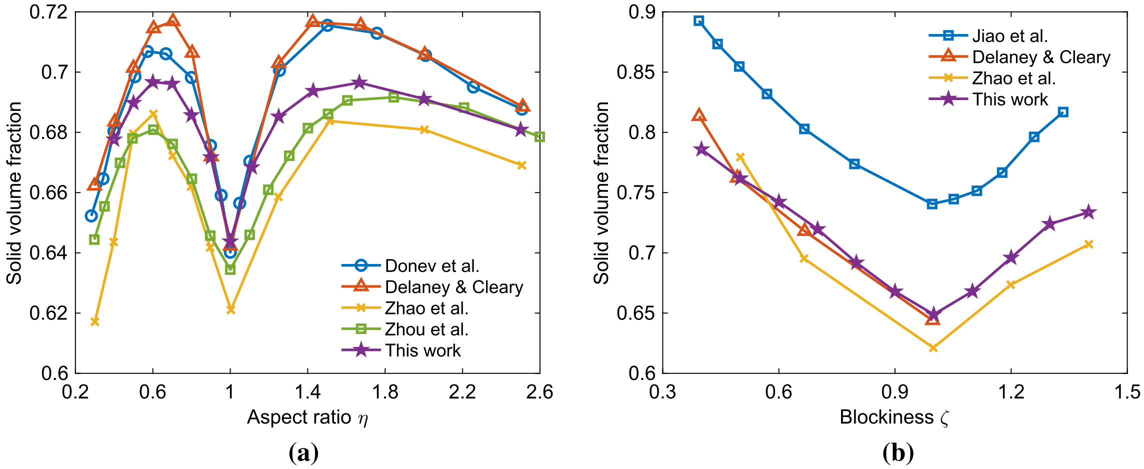


Fig. 14 Packing solid volume fraction of super-ellipsoid particles with different **a** aspect ratios and **b** blockiness. (Color figure online)

5.2 Column collapse

The column collapse test involves 1, 250 spherical harmonics particles of (equivalent) size 0.005 m. The particles are packed into a box container with a 0.04-by-0.04 m base using the rain-falling approach, and the height of the packing is approximately 0.08 m after equilibrium. After packing and equilibrium, the collapsing process is triggered by removing one of the lateral boundary of the box container. The simulation lasts for 10 s, which is deemed to be sufficient for the particles to reach equilibrium by observing the particle velocity and trajectory. In this example, the following model parameters are used: normal stiffness parameter k_i is 100 N/m$^{1/2}$, shear stiffness k_t is 10 KN/m, parameter β is 1. Two contact frictions μ, i.e. 0.1 and 0.3, are considered, to reproduce the effect of contact friction on the repose angle. The particle density is 265, 000 kg/m^3, which is artificially enlarged by two orders from the conventional density of sand particles to accommodate a timestep of 1.0×10^{-4} s. Similar to the random packing example, the particles are discretized with 1000 surface nodes and the damping coefficient is 3.0.

Figure 15 shows several snapshots of particle configurations during the reposing process. The particle packing of contact friction 0.3 is observed to be slightly higher than that of contact friction 0.1. This phenomenon is reasonable as contact friction enhances the packing porosity. With time elapsing, the particles rearrange steadily during the very beginning 0.3 s. After that, the particles slow down and gradually reach equilibrium. Figure 16 shows the zoom-in snapshots of the particle configurations at the end of reposing and the corresponding repose angle. The particles with contact friction 0.3 exhibits a repose angle higher (25° v.s. 11°) than the particles with contact friction 0.1. The effect of contact friction on the repose angle of a granular material can be well captured and reproduced by the SDF-based DEM.

5.3 Nut spinning into bolt

In the example of nut spinning into bolt, the bolt is fixed whereas the nut can move and spin along the track of the bolt, as shown in Fig. 17. The nut and bolt are modeled with polyhedrons, which consist of 5,196 and 8,170 facets, respectively. The bolt has a slightly smaller average node spacing and thus is taken as the particle contains the intruding nodes, whereas the nut is intruded. Nonetheless, reversing choice of the intruding and intruded particles would not make a difference to the results as the nut and bolt are discretized with high accuracy and their average node spacings are close to each other. In this typical example, the motion of nut is highly sensitive to the accuracy and stability of contact

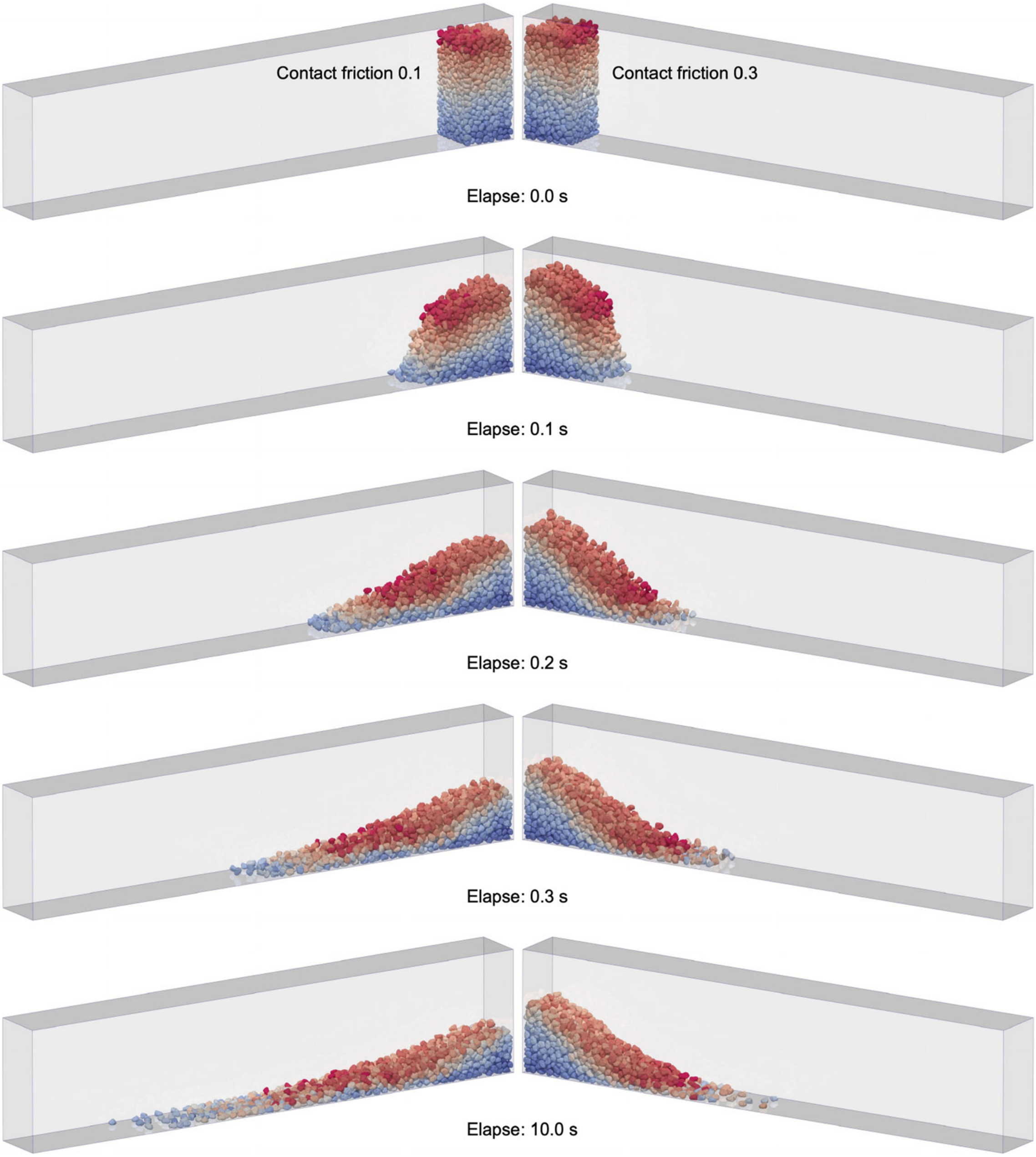


Fig. 15 Snapshots of particle configurations during the collapsing process. The example involves 1,250 particles. The left column represents the case of contact friction 0.1, and right column represents the case of contact friction 0.3. (animation in supplementary). (Color figure online)

detection and resolution, due to the complex multi-contacts between the nut and bolt. If the surfaces of nut and bolt are discretized inaccurately or the contact behavior is evaluated inappropriately, it would result in numerical instability, such as nut vibrating or being trapped within the bolt tracks. For the case without damping and friction, the velocity of the nut can be solved from the energy conserving equation $mg\Delta h + 0.5 I_z \omega_z^2 + 0.5 m v_z^2 = 0$, where Δh represents the vertical translation of the nut and can be related to the nut velocity through $\partial \Delta h / \partial t = v_z$. In addition, the spin velocity and translational velocity of the nut are correlated through $r\omega_z / v_z = 2\pi r / h$, where r represent the inside radius of the nut ring and h represents the span of one resolution of the bolt track. Thereby, the analytic solution of the nut velocity during the spinning process can be obtained by solving these equations. The normal stiffness parameter is 2 KN/m$^{1/2}$ and the shear stiffness is 10 KN/m. Parameter β is taken as 1 for simplicity, and no damping or friction is introduced. The particle density is 7, 850 kg/m^3 and the timestep is fixed at 1.0×10^{-4} s. Figure 18 plots the spin velocity of the nut based

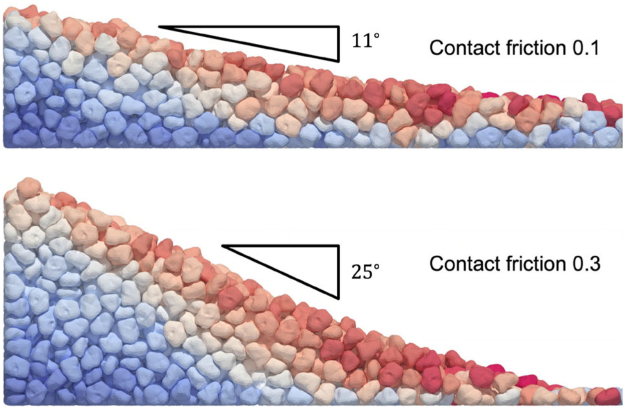


Fig. 16 Zoom-in snapshots of the particle configurations at the end of collapsing and the corresponding repose angle. (Color figure online)

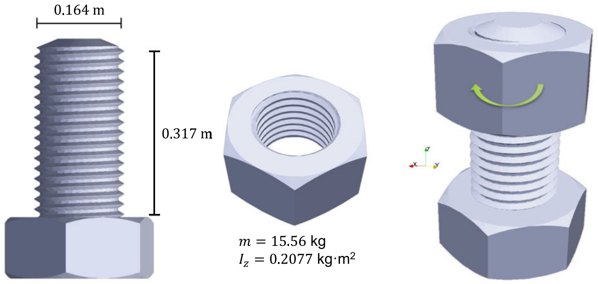


Fig. 17 Sketch of nut spinning into bolt under gravity and without damping or friction. (Color figure online)

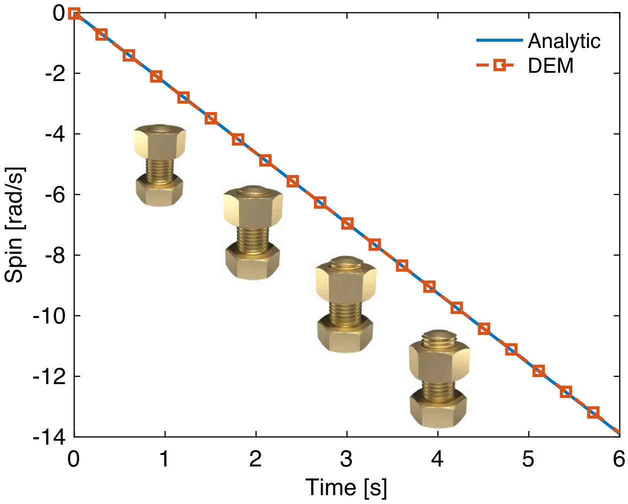


Fig. 18 Spin velocity of the nut along the vertical direction. (animation in supplementary). (Color figure online)

on analytic solution and DEM simulation, which agree well with each other. The results indicate the good performance, in terms of accuracy and stability, of the proposed SDF-based DEM for simulating particles of complex shapes and with multiple inter-particle contacts.

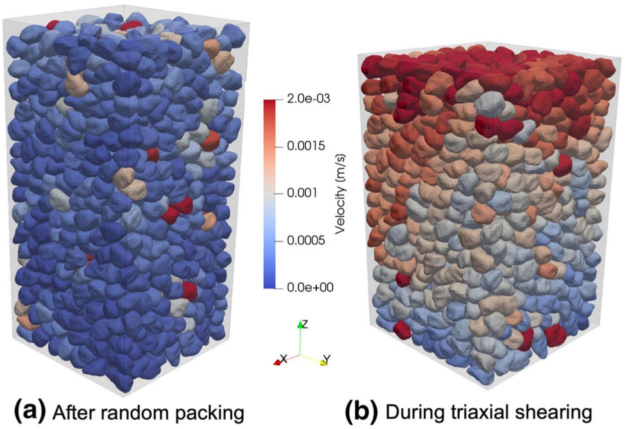


Fig. 19 Snapshots of the particle configurations **a** after random packing and **b** during triaxial shearing. The specimen consists of 1,250 particles. (animation in supplementary). (Color figure online)

5.4 Triaxial compression

Similar to the column collapse example, the triaxial compression test also involves 1, 250 spherical harmonics particles of size 0.005 m. After obtaining the packing, the particles are first compressed with a given isotropic confining pressure, and are then sheared by moving the top wall downwards at a speed of 0.002 m/s meanwhile maintaining the lateral confining pressure constant via servo control [70,71]. Figure 19 shows the snapshots of the particle configurations after equilibrium and during shearing. The shearing process lasts for 10 s, which accounts for a final axial strain of about 25%. In this example, the model parameters are kept the same as those in the column collapse example. Three levels of confining pressure, namely 100 kPa, 200 kPa and 300 kPa, are considered.

Figure 20 shows the evolution of deviatoric stress ratio and volumetric strain during the shearing process. Herein, the deviatoric stress is given by

$$p = \frac{1}{3}\sigma_{ii}, \quad q = \sqrt{\frac{3}{2}\sigma'_{ij}\sigma'_{ij}}, \quad \sigma'_{ij} = \sigma_{ij} - p\delta_{ij} \tag{47}$$

where σ'_{ij} is the deviatoric part of stress tensor σ_{ij} with p as the mean stress and δ_{ij} as Kronecker delta. The stress tensor σ_{ij} is calculated from the contact forces of each of the confining walls divided by the wall areas, respectively. The axial strain is defined as $\epsilon_z = \ln(H_0/H)$ and the volumetric strain is defined as $\epsilon_v = \ln(V_0/V)$, where H and V, respectively, are the height and volume of the packing, with the subscript 0 indicating their initial values and a negative value of volumetric strain indicating dilatation of the packing. For all three levels of confining pressure, the deviatoric stress first increases rapidly before dropping gradually, whereas the volumetric strain first exhibits a slight contraction followed

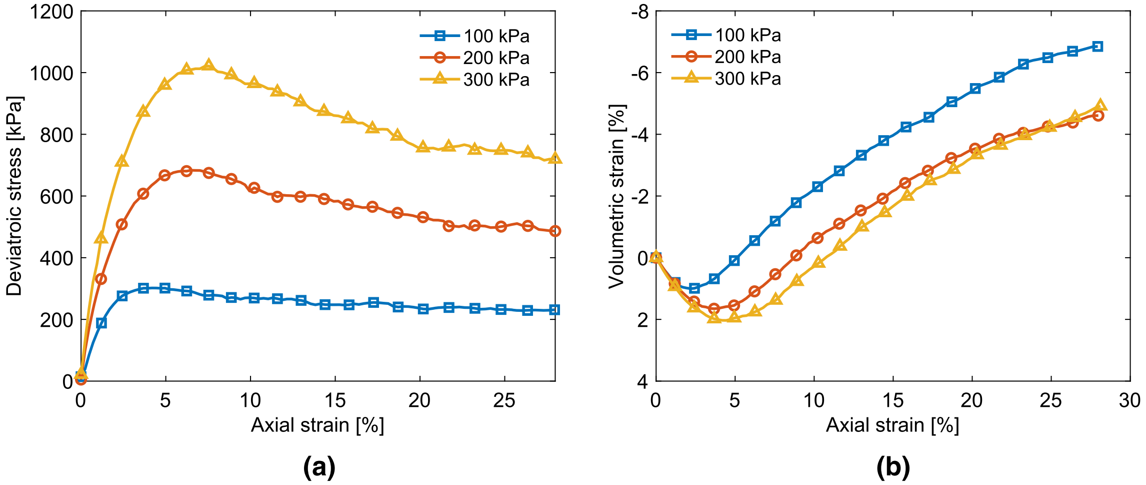


Fig. 20 Evolution of **a** deviatoric stress ratio and **b** volumetric strain with increasing axial strain during shear. (Color figure online)

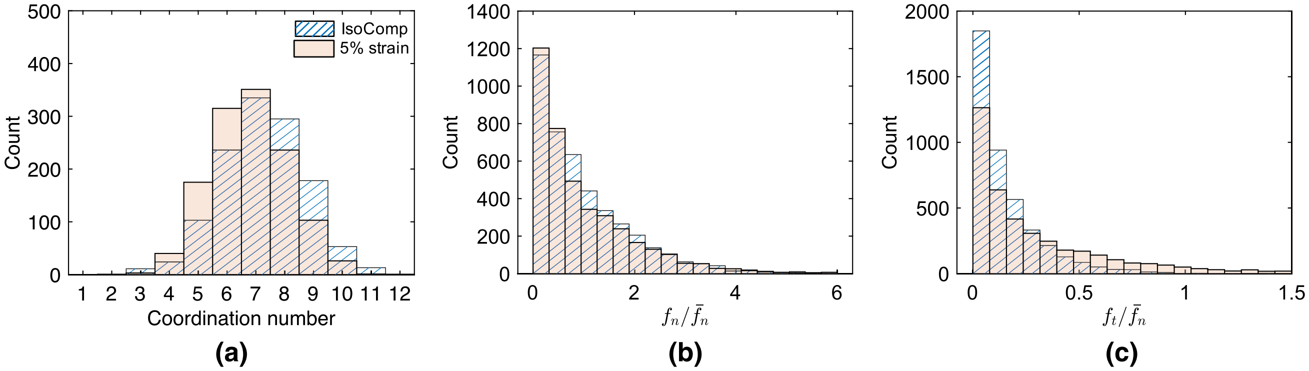


Fig. 21 Microscopic responses of the irregular-shaped particles after isotropic compression and at 5% axial strain during shearing: **a** coordination number, **b** contact normal force distribution, and **c** contact shear force distribution. (Color figure online)

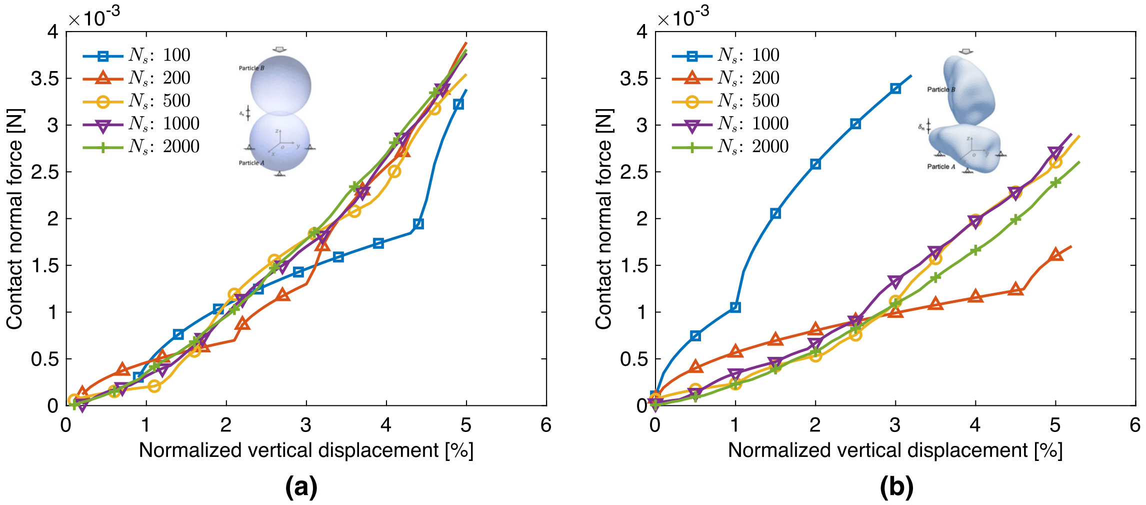


Fig. 22 The force-displacement profiles with different numbers of surface nodes for **a** spherical particles and **b** irregular-shaped particles, respectively. (Color figure online)

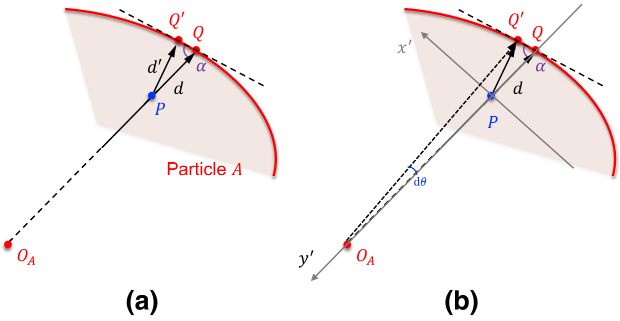


Fig. 23 Illustration of **a** relationship between the radial distance and shortest distance and **b** its effect on the contact normal. (Color figure online)

with pronounced dilation. The response of the irregular-shaped particles is similar to the behavior of medium dense sands well observed in laboratory tests. Figure 21 presents the microscopic responses, including coordination number, contact normal force distribution, and contact shear force distribution, of the irregular-shaped particles after isotropic compression and at 5% axial strain during shearing. A notable decrease in the coordination number is observed, which is due to the dilation behavior of the particles. The contact shear force increases significantly due to the shearing, indicating a good number of contact friction mobilization. The results indicate the good capability of the SDF-based DEM for simulating quasi-static geomechanical problems.

6 Discussions

6.1 Sensitivity to surface discretization

Within the adopted energy-conserving contact theory, contact forces are calculated from the derivatives of contact potential and are carried by multiple intruding nodes; thus, the contact stiffness parameter k_i in Eq. (7) and equivalent overall contact stiffness between two particles are different from the convectional-sense contact stiffness. In addition, the contact behavior between two particles may also be affected by the number and layout of surface nodes. To gain insight into the force-displacement profile and its sensitivity to surface discretization, we perform a single contact test by gradually pushing Particle B towards Particle A, as illustrated in the embedded images in Figs. 22a and 22b for spherical and general irregular-shaped particles, respectively. The irregular-shaped particles are simulated by spherical harmonics particle model. Particle A is fixed at the origin, whereas Particle B is placed on the top of Particle A with their centroids align with the z axis. Particle B is then pushing downwards into Particle A, to cause a collision with increasing penetration. Both particles have a unit (equivalent) size, and the particles surfaces are discretized into different numbers of evenly distributed nodes using WSCVT described in Sect. 4. For example, in the spherical case, the surface discretization of 1000 points corresponds to an angular distance of about 3.5° between two adjacent surface nodes. The contact parameter k_i is set to 1 $\text{N/m}^{1/2}$, and β is taken as $1/N_s$ with N_s being the number of surface nodes to account for the change of node spacing.

Figure 22 shows the force-displacement results for the cases of spherical and irregular-shaped particles with different numbers of surface nodes. In the cases of 100 and 200 surface nodes, the force-displacement profile exhibits notable variations and fluctuations. The force-displacement profile gradually converges to a smooth and power-law-like curve with the number of surface nodes increased to 500 and more. Hence, particles are discretized with 1000 surface nodes in the example simulations to weigh a balance between efficiency and accuracy. With sufficient surface nodes, the force-displacement profile can be well fitted by a power function with an order of about 1.5, indicating that the adopted three-halves-power contact potential can well reproduce the characteristic feature of the conventional Hertzian contact model. Nonetheless, in practice the contact stiffness parameter needs to be determined through a calibration process and we are preparing a separate future work trying to establish a connection between the contact stiffness parameter of the potential-based contact models and that of classical contact models, including linear spring contact model and non-linear Hertzian contact model. More choices on the formulations of contact potential and the corresponding overall inter-particle contact behavior will be also explored in the future.

6.2 Radial distance v.s. shortest distance

In the poly-super-ellipsoid and spherical harmonics particle models, we have been utilizing the radial distance to defined the SDF, while the radial distance may not necessarily be the shortest distance from the point to the surface. For example, Fig. 23 shows a comparison between the radial distance and the shortest distance field. For a query point P, the surface projection Q is defined as the intersection between the particle surface and the radial line that passes the particle centroid O_A and query point P. The radial distance-based SDF is defined as the distance between point P and Q, whereas the shortest distance is the length PQ'. Figure 24 shows an example of the radial distance-based SDF and the shortest distance-based SDF. In comparison with the shortest distance-based SDF, the contours of the radial distance-SDF are slightly skewed towards the origin, presenting a maximum of signed distance at the vicinity of the origin. Nonetheless, the difference between the radial distance-SDF and the shortest distance-based SDF gradually vanishes with points moving from the origin to the surface. For points close

Table 1 Memory consumption of basic DEM objects and different particle models

DEM objects	Properties to store	Memory consumption
Particle	Mass, moment of inertia, position, rotation quaternion, velocity, spin, etc.	∼1,000 bytes
Contact	Contact point, normal, tangential, forces, moments, particle indices, etc.	∼500 bytes
Poly-super-ellipsoid	6 axis lengths, 2 shape parameters	64 bytes
Poly-super-quadrics	6 axis lengths, 6 shape parameters	96 bytes
Spherical harmonics	81 spherical harmonics coefficients for an oder of 8	648 bytes
Polyhedron	(2×1,000-4)×3 facet connectivity, an axis-aligned bounding box hierarchy tree	∼48,168 bytes
Level set	21×21×21 SDF values for a grid of same size, 3 coordinate values, 3 grid spacing values	74,136 bytes
Surface nodes	1,000×3 coordinate values for nodal spacing of ∼1/20 particle size	24,000 bytes

All values are stored in double precision (i.e., 8 bytes). It is presumed that the particle surface is discretized with 1,000 nodes

to the particle surface, the radial distance-based SDF deviates from the shortest distance-based SDF by a factor of $\cos\alpha$, where α is the angle from the radial direction to tangential plane of the surface at the projection point Q. As a consequence, the resultant contact forces may deviate by a factor of $\cos\alpha$, which may not become an issue as in DEM the contact stiffness of particles is usually determined through a calibration process.

To gain insight into the difference in the contact normal of using radial distance and shortest distance, one can create a local coordinate system with the origin being intruding node P, y' directed to the particle centroid, and x' perpendicular to y', as illustrated in Fig. 23(b). It can be derived that $\frac{\partial d}{\partial y'_P} = 1$ and $\frac{\partial d}{\partial x'_P} \approx \frac{r\mathrm{d}\theta\cot\alpha}{(r-d)\mathrm{d}\theta} \approx \cot\alpha$, where d is the SDF of node P, r is the radial distance of point Q and $d << r$ for infinitesimal intruding depth. Thereby, considering only the two-dimensional cut-plane, the contact normal is close to $(\cot\alpha, 1)$, i.e., $(\cos\alpha, \sin\alpha)$, which is the direction from the intruding point to the surface point giving the shortest distance. In this regard, the adopted energy-conserving contact theory provides similar results to the conventional-sense definition of contact normal, while the former offers a robust and flexible way to reproduce complex contact behavior by modifying the contact potential.

6.3 Memory consumption

Comparing with the conventional DEM, the SDF-based approach requires extra memory to store surface nodes and node-based multiple contacts. The memory consumption of basic DEM objects and different particle models are summarized in Table 1. The results indicate that the classical polyhedron and level set particle models may require significant memory for storing shape properties, whereas the poly-super-ellipsoid, poly-super-quadrics and spherical harmonics particle models are rather memory friendly. The surface nodes of the particles and node-based multiple contacts would significantly increase the memory consumption of an SDF-based DEM simulation as compared to conventional DEM, which reduces the affordable number of particles for a given workstation. For instance, for a particle shape with spherical harmonics order 8 and 1,000 surface nodes, it requires 24,648 bytes of memory. Thereby, along with other particle properties (e.g., mass, moment of inertia, position, rotation quaternion, velocity, etc.), a particle would consume roughly 25 kilobytes of memory. With respect to the contact information, in our present implementation, a contact would consume roughly 0.5 kilobytes of memory, to store the information about the contact geometric features and contact forces. Given a node-based contact with ten intruding nodes, the memory usage would be about 5 kilobytes, which is tenfold that of the conventional single-contact case per particle pair. In addition, considering a particulate system with a mean coordination number of 8, the memory usage for the contacts of each particle would be about 5×8/2=20 kilobytes. Putting all together, for the distance-potential-based framework, each particle would consume about 45 extra kilobytes of memory comparing with conventional DEM. According to our simulations, the memory consumption of the DEM environment (e.g., functions and global variables) was around 10 megabytes, which means that the particle shape and node-based contacts would dominate the memory usage. One gigabytes of memory would be able to accommodate approximately 22,000 particles. The affordable number of particles would reduce to one fourth for the case of level set particles (with SDF grid of dimension 21×21×21).

6.4 Computational efficiency

To investigate the computational efficiency of the SDF-based DEM, the time cost of signed distance query for different particle models is measured and plotted in Fig. 25. The results indicate that the level set particle model presents the best performance in terms of computational efficiency, whereas the polyhedron particle model is the worst. Comparing with the level set particle model, the poly-super-ellipsoid and

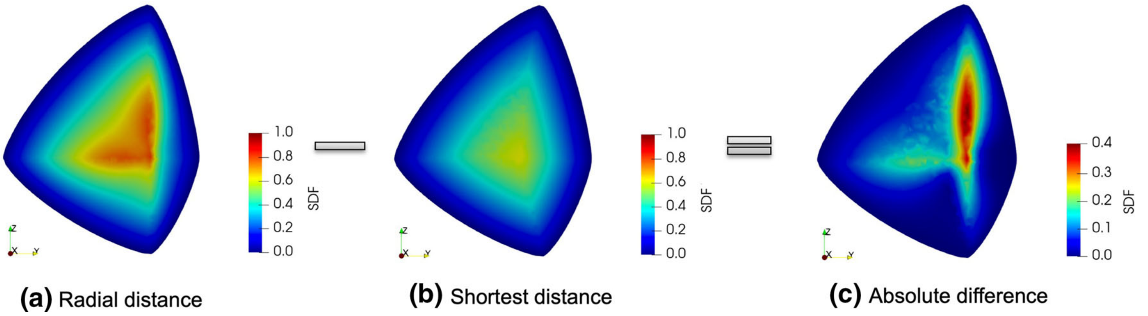


Fig. 24 Illustration of the difference between the radial distance-based SDF and shortest distance-based SDF for poly-super-ellipsoid particle: **a** radial distance-based SDF, **b** shortest distance-based SDF, and **c** the absolute difference between the two SDFs, at cross-section y-o-z. (Color figure online)

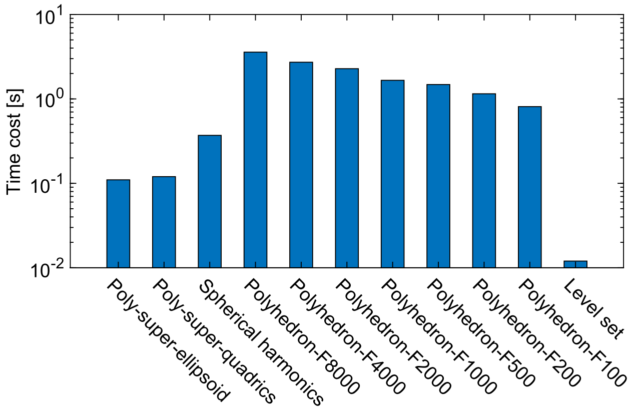


Fig. 25 Time cost per million trials of signed distance query for different particle models (on one Intel i7-10700 CPU with one core). The label Polyhedron-F8000 indicates that the polyhedron has 8000 facets. (Color figure online)

poly-super-quadrics particle models are approximately 10 times more computationally expensive, and 30 times for the case of spherical harmonics particle model. In addition, it is noted that the computational efficiency of the poly-super-ellipsoid, poly-super-quadrics, spherical harmonics and level set particle models are generally independent from the shape complexity of a particle, whereas the computational efficiency of the polyhedron particle model are heavily affected by the shape complexity, in terms of the number of vertices and facets of the polyhedron. The time cost of signed distance query of polyhedron particle model is (in log-log scale) linearly proportional to the number of facets.

Computational time costs of the example DEM simulations presented in the previous section are summarized in Table 2. All simulations have been run on one Intel i7-10700 CPU with 16 logic cores. In the present implementation, the DEM code is paralleled with OpenMP. The results of the same simulation albeit with just one core show that the performance gain with 16 cores is about tenfold.

7 Summary

In this paper, we proposed a unified SDF-based framework for contact detection and resolution among arbitrary particle shapes in DEM. The framework describes particle shapes using a generic SDF-based interface, which features two basic functions, namely a signed distance function and a surface projection function, for querying a signed distance and mapping its corresponding surface point for a given a query point, respectively. To facilitate particle surface discretization and reconstruction, we proposed a WSCVT-based point sampling scheme to handle an arbitrary number of surface points efficiently and to offer a flexible way to control the local density of the points. With the SDF-based description of particle shapes, contact detection can be conveniently implemented by the node-to-surface algorithm, i.e., checking the distance signs of the surface nodes of one particle with respect to another. The energy-conserving contact theory is adopted to derive the geometric features and interaction forces at contact. It is noteworthy that a novel contact potential carried on each intruding node can be introduced in terms of signed distance. The three-halves-power contact potential is proposed to approximate the Hertzian contact behavior in a more efficient manner.

To demonstrate the robustness and capability of the proposed framework, illustrative particle models have been developed using classical geometries, including poly-super-ellipsoid, poly-super-quadrics, spherical harmonics and polyhedron. The level set particle model is also described and integrated into this framework. A series of representative simulations, including random packing, column collapse, nut spinning into bot and triaxial compression tests, are exemplified to verify and validate the framework. The memory consumption and computational efficiency of these particle models have been also discussed. It is illustrated that polyhedron particle model is flexible in reproducing complex shapes, whereas the level set particle model has the best computational efficiency but may consume extreme amount

Table 2 Specifications and computational costs of the DEM simulations (on one Intel i7-10700 CPU with 16 logic cores)

Example case	Number of particles+walls	Average contact checks per particle and cycle	Number of DEM cycles	Computational cost [hours]
Packing: poly-super-ellipsoid	500+6	22	50,000	0.23
Packing: poly-super-quadrics	500+6	26	50,000	0.30
Packing: spherical harmonics	500+6	16	50,000	0.38
Packing: polyhedron	500+6	15	50,000	9.83
Packing: level set	500+6	18	50,000	0.12
Packing: mixed particle models	500+6	18	50,000	5.95
Packing: poly-super-ellipsoid, $\eta = 0.4$, $\zeta = 1.0$	8,000+6	36	50,000	3.26
Packing: poly-super-ellipsoid, $\eta = 2.5$, $\zeta = 1.0$	8,000+6	15	50,000	1.30
Packing: poly-super-ellipsoid, $\eta = 1.0$, $\zeta = 0.4$	8,000+6	11	50,000	0.97
Packing: poly-super-ellipsoid, $\eta = 1.0$, $\zeta = 1.4$	8,000+6	10	50,000	0.98
Column collapse: spherical harmonics	1,250+5	12	115,000	2.3
Nut sliding into bolt: polyhedron	1+1	1	30,000	0.13
Triaxial compression: spherical harmonics	1,250+6	14	118,000	2.5

In the polyhedron cases, the nut and bolt consist of 5,196 and 8,170 facets, respectively of memory. The poly-super-ellipsoid, poly-super-quadrics and spherical harmonics particle models are rather memory friendly and present the medium computational performance. As demonstrated, the SDF-based framework is a promising general-purpose alternative to model arbitrary-shaped particles (non-convex particles in particular) accurately and efficiently in DEM, which is flexible for users to pursue desired optimal performance.

Acknowledgements This work was financially supported by the Hong Kong Scholars Program (2020), the Research Grants Council of Hong Kong by GRF Project No. 16208720, the National Natural Science Foundation of China (51909289, 51978677, 11972030, 51909095, 5201101539, 52111530089), the Guangdong Basic and Applied Basic Research Foundation (2022A1515010848), the Shenzhen Science and Technology Project for Sustainable Development (KCXFZ202002011008532), the Shenzhen Natural Science Foundation (JCYJ20190807162401662), and the Project of Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone (HZQB-KCZYB-2020083).

References

1. Cundall PA, Strack ODL (1979) A discrete numerical model for granular assemblies. Géotechnique 29(1):47–65
2. Munjiza A, Peters JF, Hopkins MA, Kala R, Wahl RE, A poly-ellipsoid particle for non-spherical discrete element method. Engineering Computations
3. Wellmann C, Lillie C, Wriggers P, A contact detection algorithm for superellipsoids based on the common-normal concept. Engineering Computations
4. Zhao S, Zhao J (2019) A poly-superellipsoid-based approach on particle morphology for DEM modeling of granular media. Int J for Numer and Anal Methods in Geom 43(13):2147–2169
5. Wang S, Ji S (2021) Poly-superquadric model for DEM simulations of asymmetrically shaped particles. Computational Particle Mechanics 1–15
6. Cleary PW, Sawley ML (2002) DEM modelling of industrial granular flows: 3D case studies and the effect of particle shape on hopper discharge. Appl Math Modelling 26(2):89–111
7. Podlozhnyuk A, Pirker S, Kloss C (2017) Efficient implementation of superquadric particles in Discrete Element Method within an open-source framework. Comput Particle Mech 4(1):101–118
8. Kildashti K, Dong K, Samali B (2020) An accurate geometric contact force model for super-quadric particles. Comput Methods in Appl Mech and Eng 360:112774
9. Smeets B, Odenthal T, Keresztes J, Vanmaercke S, Van Liedekerke P, Tijskens E, Saeys W, Van Oosterwyck H, Ramon H (2014) Modeling contact interactions between triangulated rounded bodies for the discrete element method. Comput Methods in Appl Mech and Eng 277:219–238
10. Smeets B, Odenthal T, Vanmaercke S, Ramon H (2015) Polygon-based contact description for modeling arbitrary polyhedra in the Discrete Element Method. Comput Methods in Appl Mech and Eng 290:277–289
11. Zhao S, Zhou X, Liu W, Lai C (2015) Random packing of tetrahedral particles using the polyhedral discrete element method. Part 23:109–117
12. Feng YT (2021) An effective energy-conserving contact modelling strategy for spherical harmonic particles represented by surface triangular meshes with automatic simplification. Comput Methods in Appl Mech and Eng 379:113750
13. Andrade JE, Lim KW, Avila CF, Vlahinić I (2012) Granular element method for computational particle mechanics. Comput Methods in Appl Mech and Eng 241:262–274
14. Lim K-W, Andrade JE (2014) Granular element method for three-dimensional discrete element calculations. Int J for Numer and Anal Methods in Geom 38(2):167–188
15. Craveiro MV, Neto AG, Wriggers P (2021) Contact between rigid convex NURBS particles based on computer graphics concepts. Comput Methods in Appl Mech and Eng 386:114097
16. Kawamoto R, Andò E, Viggiani G, Andrade JE (2016) Level set discrete element method for three-dimensional computations with triaxial case study. J of the Mech and Phys of Solids 91:1–13
17. Kawamoto R, Andò E, Viggiani G, Andrade JE (2018) All you need is shape: Predicting shear banding in sand with LS-DEM. J of the Mech and Phys of Solids 111:375–392
18. Capozza R, Hanley KJ (2021) A hierarchical, spherical harmonic-based approach to simulate abradable, irregularly shaped particles in DEM. Powder Technol 378:528–537
19. Wang X, Yin Z, Xiong H, Dong S, Feng YT, A spherical-harmonic-based approach to discrete element modeling of 3D irregular particles. International Journal for Numerical Methods in Engineering
20. Zhong W, Yu A, Liu X, Tong Z, Zhang H (2016) DEM/CFD-DEM modelling of non-spherical particulate systems: theoretical developments and applications. Powder Technol 302:108–152
21. Feng YT (2021) An energy-conserving contact theory for discrete element modelling of arbitrarily shaped particles: Basic framework and general contact model. Comput Methods in Appl Mech and Eng 373:113454
22. He H, Zheng J, Schaefer VR (2021) Simulating shearing behavior of realistic granular soils using physics engine. Granular Matter 23(3):1–20
23. Lai Z, Chen Q, Huang L (2020) Fourier series-based discrete element method for computational mechanics of irregular-shaped particles. Comput Methods in Appl Mech and Eng 362:112873
24. Zhang P, Dong Y, Galindo-Torres SA, Scheuermann A, Li L (2021) Metaball based discrete element method for general shaped particles with round features. Comput Mech 67(4):1243–1254
25. Zhao S, Zhou X (2017) Effects of particle asphericity on the macro- and micro-mechanical behaviors of granular assemblies. Granular Matter 19(2):38
26. Wachs A, Girolami L, Vinay G, Ferrer G (2012) Grains3D, a flexible DEM approach for particles of arbitrary convex shape - Part I: Numerical model and validations. Powder Technol 224:374–389
27. Rakotonirina AD, Delenne J-Y, Radjai F, Wachs A (2019) Grains3D, a flexible DEM approach for particles of arbitrary convex shape - Part III: extension to non-convex particles modelled as glued convex particles. Comput Particle Mech 6(1):55–84
28. Shi C, Li D, Xu W, Wang R (2015) Discrete element cluster modeling of complex mesoscopic particles for use with the particle flow code method. Granular Matter 17(3):377–387
29. Zheng J, Hryciw RD (2017) An image based clump library for DEM simulations. Granular Matter 19(2):26
30. Shen Z, Wang G, Huang D, Jin F (2022) A resolved CFD-DEM coupling model for modeling two-phase fluids interaction with irregularly shaped particles. J of Comput Phys 448:110695
31. Markauskas D, Kačianauskas R, Džiugys A, Navakas R (2010) Investigation of adequacy of multi-sphere approximation of elliptical particles for DEM simulations. Granular Matter 12(1):107–123
32. Hertz H (1882) Ueber die Beruhrung fester elastischer Korper (On the contact of elastic solids). J für die reine und angewandte Mathematik 92:156–171
33. Mindlin RD (1949) Compliance of elastic bodies in contact. J Appl Mech ASME 16:259–268
34. Johnson KL (1987) Contact mechanics. Cambridge University Press, Cambridge, England

35. Zhao S, Evans TM, Zhou X (2018) Effects of curvature-related DEM contact model on the macro-and micro-mechanical behaviours of granular soils. Géotechnique 68(12):1085–1098
36. Lai Z, Chen Q, Huang L (2021) A semianalytical Hertzian frictional contact model in 2D. Appl Math Model 92:546–564
37. Su D, Wang X (2021) Fourier series-based discrete element method for two-dimensional concave irregular particles. Comput and Geotech 132:103991
38. Fan H, Huang D, Wang G, Wang J (2020) Discontinuous deformation analysis for SH-body. Comput and Geotech 117:103234
39. Wang X, Yin Z, Su D, Xiong H, Feng YT (2021) A novel Arcs-based discrete element modeling of arbitrary convex and concave 2D particles. Comput Methods in Appl Mech and Eng 386:114071
40. Wriggers P, Zavarise G Computational contact mechanics. Encyclopedia of computational mechanics
41. Khoei AR, Biabanaki SOR, Parvaneh SM (2013) 3D dynamic modeling of powder forming processes via a simple and efficient node-to-surface contact algorithm. Appl Math Model 37(1–2):443–462
42. Paggi M, Wriggers P (2016) Node-to-segment and node-to-surface interface finite elements for fracture mechanics. Comput Methods in Appl Mech and Eng 300:540–560
43. Jiménez JJ, Segura RJ (2008) Collision detection between complex polyhedra. Comput & Graphics 32(4):402–411
44. Sheng B, Liu B, Li P, Fu H, Ma L, Wu E (2018) Accelerated robust Boolean operations based on hybrid representations. Comput Aided Geom Design 62:133–153
45. Zhan L, Peng C, Zhang B, Wu W (2021) A surface mesh represented discrete element method (SMR-DEM) for particles of arbitrary shape. Powder Technol 377:760–779
46. Feng YT (2021) An energy-conserving contact theory for discrete element modelling of arbitrarily shaped particles: Contact volume based model and computational issues. Comput Methods in Appl Mech and Eng 373:113493
47. Neto AG, Wriggers P (2022) Discrete element model for general polyhedra. Comput Particle Mech 9(2):353–380
48. Wolff S, Bucher C (2013) Distance fields on unstructured grids: Stable interpolation, assumed gradients, collision detection and gap function. Comput methods in appl mech and eng 259:77–92
49. PFC documentation (2021). http://docs.itascacg.com/pfc700/contents.html, accessed: 2021-06-30
50. Balevičius R, Mróz Z (2013) A finite sliding model of two identical spheres under displacement and force control - part I: static analysis. Acta Mechanica 224(8):1659–1684
51. Mollon G, Zhao J (2012) Fourier-Voronoi-based generation of realistic samples for discrete modelling of granular materials. Granular matter 14(5):621–638
52. Mollon G, Zhao J (2014) 3D generation of realistic granular samples based on random fields theory and Fourier shape descriptors. Comput Methods in Appl Mech and Eng 279:46–65
53. Zhou B, Wang J, Zhao B (2015) Micromorphology characterization and reconstruction of sand particles using micro X-ray tomography and spherical harmonics. Eng Geol 184:126–137
54. Wei D, Wang J, Zhao B (2018) A simple method for particle shape generation with spherical harmonics. Powder technol 330:284–291
55. Spellings M, Marson RL, Anderson JA, Glotzer SC (2017) GPU accelerated Discrete Element Method (DEM) molecular dynamics for conservative, faceted particle simulations. J of Comp Phys 334:460–467
56. Park JJ, Florence P, Straub J, Newcombe R, Lovegrove S (2019) Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition pp 165–174
57. Macklin M, Erleben K, Müller M, Chentanez N, Jeschke S, Corse Z (2020) Local optimization for robust signed distance field collision, Proceedings of the ACM on Computer Graphics and Interactive. Techniques 3(1):1–17
58. Jacobson A, Panozzo D et al. (2018) libigl: A simple C++ geometry processing library. https://libigl.github.io/
59. Hartmann D, Meinke M, Schröder W (2010) The constrained reinitialization equation for level set methods. J of Comput Phys 229(5):1514–1535
60. Harmon JM, Arthur D, Andrade JE (2020) Level set splitting in DEM for modeling breakage mechanics. Comput Methods in Appl Mech and Eng 365:112961
61. Liu C, Sun W (2020) ILS-MPM: An implicit level-set-based material point method for frictional particulate contact mechanics of deformable particles. Comput Methods in Appl Mech and Eng 369:113168
62. Du Q, Gunzburger M, Ju L (2010) Advances in studies and applications of centroidal Voronoi tessellations. Numer Math: Theory, Methods and Appl 3(2):119–142
63. Larrea M, Urribarri D, Martig S, Castro S Spherical layout implementation using centroidal voronoi tessellations. arXiv preprint arXiv:0912.3974
64. Yang H, Gunzburger M, Ju L (2018) Fast spherical centroidal Voronoi mesh generation: A Lloyd-preconditioned LBFGS method in parallel. J of Comput Phys 367:235–252
65. Donev A, Cisse I, Sachs D, Variano EA, Stillinger FH, Connelly R, Torquato S, Chaikin PM (2004) Improving the density of jammed disordered packings using ellipsoids. Sci 303(5660):990–993
66. Jiao Y, Stillinger FH, Torquato S (2009) Optimal packings of superballs. Phys Review E 79(4):041309
67. Delaney GW, Cleary PW (2010) The packing properties of superellipsoids. EPL (Europhysics Letters) 89(3):34002
68. Zhou Z, Zou R, Pinson D, Yu A (2011) Dynamic simulation of the packing of ellipsoidal particles. Industrial & eng chem res 50(16):9787–9798
69. Zhao S, Zhang N, Zhou X, Zhang L (2017) Particle shape effects on fabric of granular random packing. Powder Technol 310:175–186
70. Abbas A, Masa E, Papagiannakis T, Shenoy A (2005) Modelling asphalt mastic stiffness using discrete element analysis and micromechanics-based models. Int J of Pavement Eng 6(2):137–146
71. Faramarzi L., Kheradmandian A, Azhari A (2020) Evaluation and Optimization of the Effective Parameters on the Shield TBM Performance: Torque and Thrust – Using Discrete Element Method (DEM). Geotechnical and Geological Engineering 1–15

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.