

RESEARCH ARTICLE

A kinematics-based model for the settling of gravity-driven arbitrary-shaped particles on a surface

Mohsen Daghooghi^{1*}, Iman Borazjani²

1 College of Science and Engineering, University of Houston-Clear Lake, Houston, TX, United States of America, **2** J. Mike Walker '66 Department of Mechanical Engineering, Texas A&M University, College Station, TX, United States of America

* Daghooghi@uhcl.edu

Abstract

A discrete model is proposed for settling of an arbitrary-shaped particle onto a flat surface under the gravitational field. In this method, the particle dynamics is calculated such that (a) the particle does not create an overlap with the wall and (b) reaches a realistic equilibrium state, which are not guaranteed in the conventional discrete element methods that add a repulsive force (torque) based on the amount of overlap between the particle and the wall. Instead, upon the detection of collision, the particle's kinematics is modified depending on the type of contact, i.e., point, line, and surface types, by assuming the contact point/line as the instantaneous center/line of rotation for calculating the rigid body dynamics. Two different stability conditions are implemented by comparing the location of the projection of the center of mass on the wall along gravity direction against the contact points to identify the equilibrium (stable) state on the wall for particles with multiple contact points. A variety of simulations are presented, including smooth surface particles (ellipsoids), regular particles with sharp edges (cylinders and pyramids) and irregular-shaped particles, to show that the method can provide the analytically-known equilibrium state.

OPEN ACCESS

Citation: Daghooghi M, Borazjani I (2021) A kinematics-based model for the settling of gravity-driven arbitrary-shaped particles on a surface. PLoS ONE 16(2): e0243716. <https://doi.org/10.1371/journal.pone.0243716>

Editor: Mohammadreza Hadizadeh, Central State University & Ohio University, UNITED STATES

Received: November 11, 2019

Accepted: October 31, 2020

Published: February 9, 2021

Copyright: © 2021 Daghooghi, Borazjani. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All relevant data are within the manuscript.

Funding: No. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: NO authors have competing interests.

1 Introduction

Settling of particles on a surface is an important phenomenon in natural and industrial systems such as dam-break collapse [1], particles on a vibrating plate [2] and granular flow on an inclined surface [3], among others. Particles in such systems have various shapes and sizes [4], but it is estimated that at least 70% of the raw materials consist of non-spherical particles [5]. The shape of particles cannot be simply approximated by an equivalent sphere because the dynamics of a single particle [6] or an entire particle system [7] of non-spherical shapes is different than those of spherical ones. This is true for settling behavior as well because there are various scenarios for geometrical contact between non-spherical particles and walls [5], in contrast to spherical ones. In fact, finding the final orientation (resting condition) for spherical particles over a surface is straightforward (rotation does not change the orientation of a sphere) while for non-spherical particles is particularly challenging.

One of the most successful numerical techniques to model particulate system is the discrete element method (DEM) [8]. In discrete models, a finite number of discrete particles (depending on computational resource and technique) are considered [9]. These particles can interact by means of contact and non-contact forces, and every particle, which can move translationally and rotationally, is described by Newton's equations of motion and Euler's equations of rigid body rotation [10, 11]. The gravitational force acts as a field force on particles in all discrete models [12] and is one prime cause of motion in various applications such as packing [13], particle sedimentation [14] and fluidization [15], when particles settle inside a container [16] or a hopper [17]. The translational motion due to gravitational force continues until particles reach the bottom of a container and collision occurs between the particle and the container's surface. At this point, shape and surface characteristics of the particle and the wall determine the transient motion (usually of a rotation type) toward the final orientation of the particle with respect to the bottom surface, i.e., the equilibrium state. Collisions in discrete models are handled by calculating a contact force (moment) such that overlapping between particle (or between a particle and a wall) is avoided [12, 14]. However, the contact force (moment) is typically only applied when the particle and wall overlap (after actual collision) and it does not guarantee an equilibrium state which is physical. The method proposed in this work guarantees a physical equilibrium state for different non-spherical geometries.

A particle may rebound after collision with a wall if it has enough inertia, high coefficient of restitution, and high Stokes number [18] and the kinematics of particles can be obtained using impulse equations [19]. In such cases, auxiliary equations (coefficient of restitution and friction coefficient) along with some simplifying assumptions are needed to solve the problem [20–23]. Even if a rebound occurs, a fraction of the incoming kinetic energy of the particle is dissipated leading to a less energetic upcoming collision. This scenario could continue until the normal velocity (with respect to the plane surface) falls below a critical value for which rebound will not occur. This critical velocity depends upon the coefficient of restitution, adhesion energy, and geometry of the particle [18]. In other words, when a particle falls on a surface in a gravity-driven field, regardless of the number of rebounds (if any), there would be a final phase of the collision, in which no rebound would occur and the particle's kinematics reaches a stable (equilibrium) state. This equilibrium state of particles in the final phase of collision, which is usually neglected in DEM, is the main focus of this work. In fact, to focus the problem on the equilibrium state, only the final phase of the collision in which no rebound occurs is considered in this work. Nevertheless, this method can be added to any previously developed particle-wall collision model (e.g. [22]), which models rebound from the surface, to numerically simulate a complete collision phenomenon including rebound(s) from the surface and a stable orientation on the surface.

This paper is organized as follows: in *Materials and methods* section, numerical schemes for particle representation, collision detection, different types of contact between a particle and a wall, equations of motion upon the collision, and stability conditions to detect the equilibrium state are presented and explained. In *Results and discussion* section, the framework is tested for a triaxial ellipsoid (an example for a smooth surface particle), a cylinder, a tetrahedron, and an irregular highly jagged particle (non-smooth particles) and results are qualitatively validated and discussed. Finally, our findings of the present framework as well as plans for future works are summarized in *Conclusion* section.

2 Materials and methods

The simulations presented in this paper are based on tracking the particle's orientation during the wall collision. The fluid forces on the particle are neglected at this stage. Moreover, only

smooth flat walls are considered. The change of translational and angular velocities of the particle during the wall collision process is calculated based on the Newton's equations of motion and Euler's equations of rigid body rotation.

A brief overview of the framework can be seen in the flowchart of Fig 1 while details of each step will be explained in upcoming subsections. The particle translates downward under the effect of the gravitational field and its distance with respect to the wall surface is measured. The collision between the particle and the wall is detected when the minimum distance becomes less than a critical threshold and then all contact points are identified. Depending on the number and distribution of contact points, the type of contact is distinguished—see Fig 3. Then two possible scenarios are checked to determine whether or not the particle has reached its equilibrium state. If the particle is not stable, kinematics and orientation for the next time step is calculated and the particle is rotated to its new orientation.

2.1 Particle representation and collision detection

There are different methods to represent the particles, e.g., the surface discretization with different polygon/polyhedral elements [24], continuous functions to represent smooth non-spherical shapes (such as ellipses [25], super-quadratic curves [26], ellipsoids [27, 28], and 3D super-quadratic curves [29–31]), and discrete functions in which the shape is represented by an array of nodes with some specific associated information providing an ordered set of points [32, 33]. To handle arbitrary-shaped particles, the particles are discretized with triangular elements here. This enables us to represent different particles from smooth to irregular ones.

In our simulations, the particle moves under the gravitational field towards the wall. The first step in our methodology is to find when the collision occurs (Fig 1). The collision process starts when the minimum distance between the wall and the particle is smaller than a predefined threshold ϵ . This arbitrary threshold can be accounted for the surface roughness and in our simulations is set to be equal to $\epsilon = 0.05L$ where L is characteristic length of the particle (smaller values of ϵ have been successfully tested as well). Let the surface of the particle be discretized with an unstructured mesh consisting of a set of triangles and form a list of material points $k = 1, N$ where N is the total number of material points for the particle. The minimum distance between the wall and the particle, d is defined as the closest distance of any material point on the surface of the particle with respect to the wall. Assuming $z = 0$ is the surface wall equation and the particle is falling under the gravitational force along the z -direction (see Fig 2), the minimum distance is defined as:

$$d = \text{Min}\{\|z^k\|, k = 1 \dots N\} \quad (1)$$

where z^k is the z -component of the position vector of k -th material point on the surface of particle in the Cartesian coordinate system. The point or a set of points whose minimum distance relative to the wall is equal or less than the threshold value ($d \leq \epsilon$) is considered as the contact point(s) of collision.

The distance measurement described above is fairly straightforward to implement but it is quite costly with a computational cost that is $O(N)$. To reduce the computational costs a bounding box strategy is used. This bounding box strategy is similar to the “Bounding box” to identify immersed nodes in CURVIB method [34]. For a particle, the bounding box is defined as a Cartesian box that contains all vertices describing the solid object's surface, i.e., the size of the bounding box is defined as:

$$[x_{\min} - \epsilon, x_{\max} + \epsilon] \times [y_{\min} - \epsilon, y_{\max} + \epsilon] \times [z_{\min} - \epsilon, z_{\max} + \epsilon] \quad (2)$$

where $x_{\min} = \text{Min}\{x^k, k = 1 \dots N\}$, $x_{\max} = \text{Max}\{x^k, k = 1 \dots N\}$, etc. If the bounding box of the

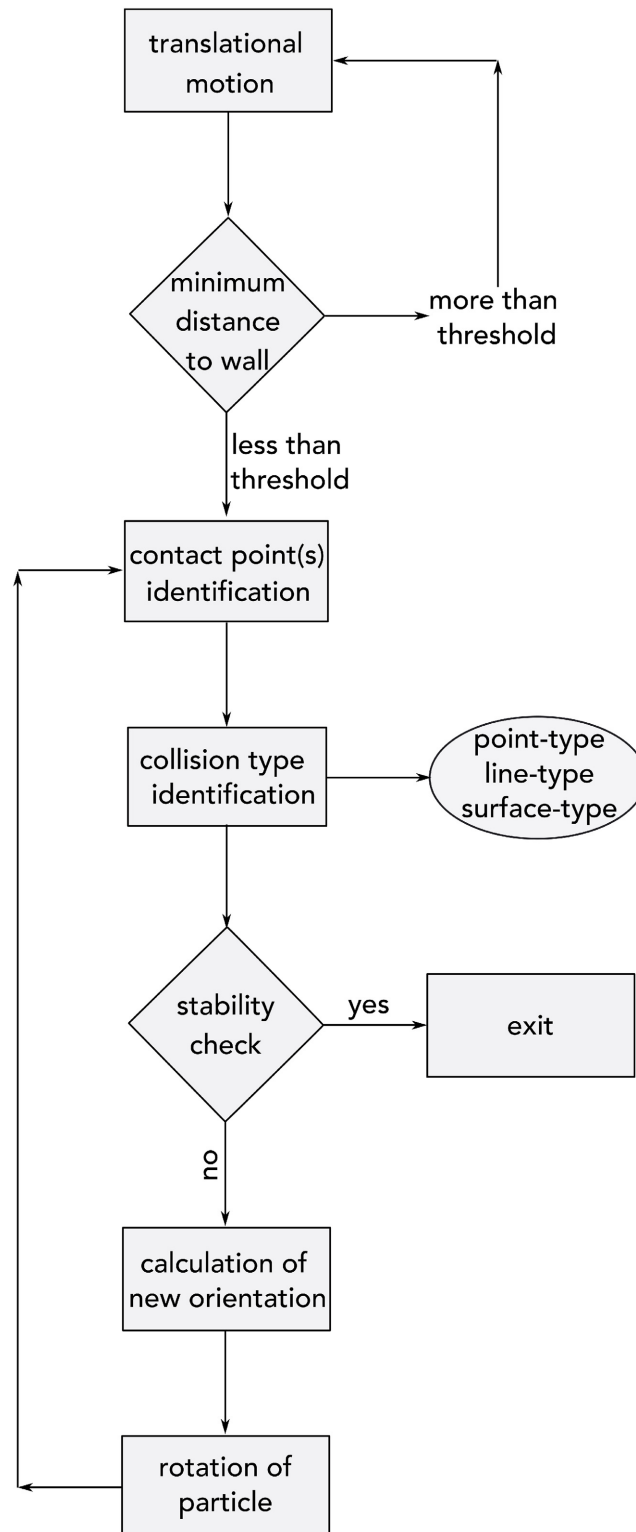


Fig 1. The flowchart shows essential steps in the wall collision treatment. The particle translates downward under the effect of the gravitational field and its distance with respect to the wall surface is measured. The collision between the particle and the wall is detected when the minimum distance becomes less than a critical threshold. All contact points are identified and depending on the number and distribution of contact points, the type of contact is distinguished. Two possible scenarios are checked to determine whether or not the particle has reached its equilibrium

state. If the particle is not stable kinematics and orientation for the next time step is calculated and the particle is rotated according to its new orientation.

<https://doi.org/10.1371/journal.pone.0243716.g001>

particles does not overlap with the wall surface, the particle is not about to collide and consequently, we do not need to measure the exact relative distance. This step allows us to run the minimum distance algorithm only when the particle is close to the wall.

2.2 Identification of contact type

The ultimate position of an object when falls and collides with a stationary surface is a stable one, but the kinematics during the collision depends on its shape. Shapes of particles that affect the collision kinematics are categorized into two groups in this work (Fig 2). The first group belongs to objects which have surfaces with a strictly convex shape (curvature strictly larger than 0), e.g. an ellipsoid (Fig 2a). Any other shape with a concave (curvature < 0) or discontinuous curvature such as objects with edges or flat faces (curvature = 0), e.g., a cube, a pyramid, or C shaped, falls into the second group (Fig 2b). Reaching the equilibrium (stable) position for the first group (hereinafter referred to as smooth particles) is achieved smoothly through a continuous oscillatory rotation, whereas for the second group (hereinafter referred to as non-smooth particles) stability occurs at a sudden moment when a face (sometimes some edges or head points) of the object totally lays on the wall, i.e, discontinuous rotation.

For smooth particles, a single point of contact is present at every time instant but the location of this point (referred to as the instantaneous center of rotation) can change as the object rotates (rolls). In this type of rotation, center of mass (COM) oscillates around the center of rotation until the object lays down on the wall such that COM has the minimum vertical distance with respect to the wall regardless of the initial position of the particle when collision occurs, i.e, the potential energy reaches its minimum value. The number of oscillations and the overall collision time depends on the surface roughness and object's material (damping coefficient) and will be discussed in upcoming sections.

For non-smooth particles, however, the initial contact point does not usually change over time and the object just rotates about this point until it lays on a contact face (in some cases edges or heads). Various scenarios can happen when a non-smooth particle hits the wall. The initial collision can be either a point-type contact, a line-type contact, or a surface-type contact depending on the particle's orientation and its shape (Fig 3).

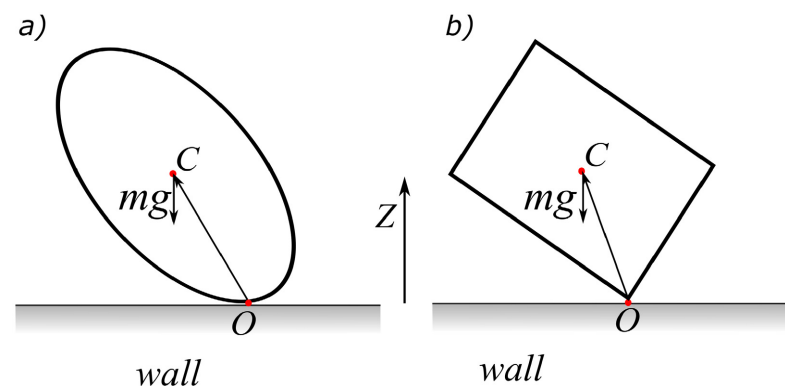


Fig 2. Kinematics during the collision depends on the shape of a particle. Shapes of particles categorized into two groups (a) strictly convex objects (curvature > 0) which smoothly oscillate to reach the stable position, and (b) objects with concave (curvature < 0) or non-smooth/flat (curvature = 0) surfaces which suddenly reach the stable position.

<https://doi.org/10.1371/journal.pone.0243716.g002>

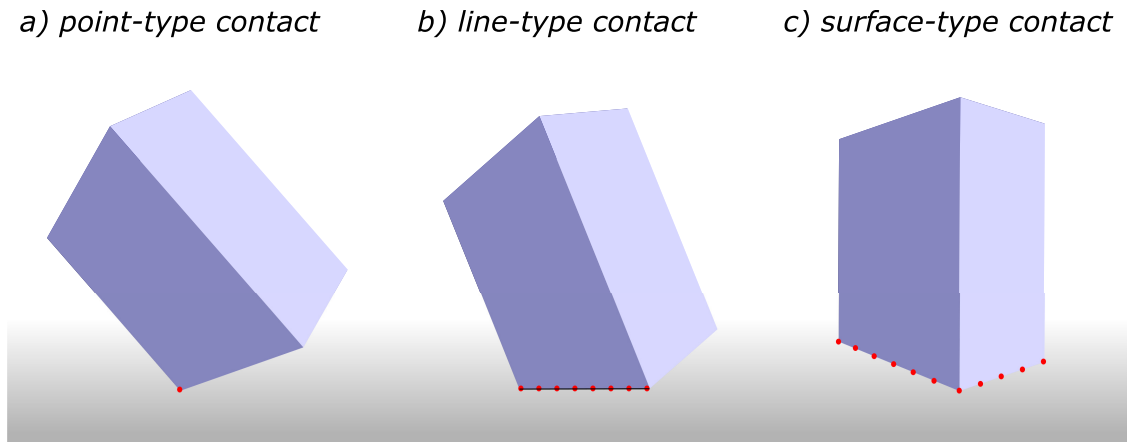


Fig 3. Different types of wall-particle contact. Various scenarios can happen when a particle collides with the wall: (collision points are shown as red dots) (a) point-type contact, (b) line-type contact, and (c) surface-type contact.

<https://doi.org/10.1371/journal.pone.0243716.g003>

2.2.1 Point-type contact. When a sharp head of the particle's surface collides with the wall, the initial contact point usually does not change and particle rotates about this point as this is an unstable position—see Fig 3a. In this type of contact, the rotation has three degrees of freedom (see section 2.3 for equations) and it continues until other points of particle's surface reach the wall surface (within the threshold) and point-type contact turns into either a line-type contact or a surface-type contact. In the case of a very jagged surface, the particle could stay stable on a few contact points as will be discussed in the section 2.4.

2.2.2 Line-type contact. This situation happens when a particle hits the wall on its edge—see Fig 3b. In this occasion, there are more than one contact point and all contact points forms a line, i.e. contact line. The particle rotates about this line until the situation would turn into a surface-type contact. In order to obtain this line numerically, first, all contact points are found and then the best fit (linear regression) for these points is obtained. Knowing the equation of the contact line, kinematics of rotation along this line are calculated. Note that in this situation, the rotation has only one degree of freedom.

2.2.3 Surface-type contact. This situation happens when a particle hits the wall on its face—see Fig 3c. In this occasion, there are more than one contact point but these contact points are not located along a line and form a contact surface instead. This situation could be either stable or unstable depending on the particle's orientation. The necessary condition for the stability of a particle is that the projection of COM on the wall lay inside the contact surface, as will be further discussed in the section 2.4. If the initial contact surface is not stable, particle rotates on one of its edges (surface-type contact turn into a line-type contact) and eventually lays on a stable face. Since in the surface-type contact there is no updated axis of rotation, the particle is rotated based on the axis of rotation from the previous time step. If the surface-type contact occurs at the beginning of rotation, the axis of rotation cannot be determined from the previous time step. In this case, the line of best fit (linear regression) is used as the axis of rotation.

The algorithm for detecting the contact type is provided in algorithm 1. The contact type for smooth particles is always point-type. For non-smooth particles if only one contact point is found, the collision is of point-type. Else, if there are more than one contact point and all contact points are aligned on a line the collision is line-type. Finally, if contact points don't represent a geometric line, the collision is of surface-type. The line and surface types in the framework are distinguished by the magnitude of the residual sum of squares (RSS) of the

linear regression technique used to find the best straight line that passes through contact points. If the RSS of the best line is less than a small threshold (0.1 in our work), contact points represent a line of contact, otherwise the collision is considered as a surface-type. Equations of motion for different types of contact are explained in the upcoming subsection.

2.3 Equations of motion after collision

To simulate the final phase of a collision in which no rebound occurs, it is assumed that the translational motion stops and upon the collision the particle rotates about either a point or a line of contact. For smooth particles the rotation is always about a contact point, whereas for non-smooth particles both scenarios could occur, i.e., for point-type contact rotation is about a single point (similar to a smooth particle), and for line and surface types the particle is rotated about an axis of rotation. It should be noted that no sliding is considered in this work regardless of the type of contact.

Algorithm 1: Identifying the type of contact between a particle and a wall

```
All contact points are found;
count = number of contact points;
if count = 1 then
  contact is point-type;
else
  linear regression is used to find the best fit line through contact
  points;
  RSS = residual sum of squares;
  if RSS < 0.1 then
    contact is line-type;
  else
    contact is surface-type;
  end
end
```

The shape of the rigid body is defined in terms of a reference coordinate system attached to the body (shown as *CXYZ* in Fig 4) at the COM (choice of this point is arbitrary). This coordinate system is referred to as *body space*. As it is depicted in the figure, this coordinate system moves and rotates with the same translational and angular velocity of the rigid body. Therefore, the position vector of any material point p on the body's surface remains unchanged over-time in this reference frame. Having the geometric description of the rigid body in the body space, we use rotation matrix $R(t)$ to transform the body-space description into the inertial space *xyz* description and find the instantaneous position vector of point p in the inertial coordinate system as:

$$\mathbf{r}^p(t) = R(t)\mathfrak{R}^p + \mathbf{r}_c(t) \quad (3)$$

where $\mathbf{r}_c(t)$ is the position vector of the particle's COM and \mathfrak{R}^p is the position vector of point p in the body space. Therefore, to fully define the orientation of the body, the rotation matrix and the vector position of COM need to be calculated at each specific time.

2.3.1 Rotation about a contact point. When the contact between the particle and the wall is of point-type (either smooth or non-smooth surfaces), the particle rotates about the contact point (denoted by O in Fig 2) until it reaches to its final position. The total external torque with respect to the contact point O consists of the moment of the object's weight with respect to point O and a damping (friction) torque as:

$$\sum \mathbf{M}_O(t) = \mathbf{r}_{Oc} \times m\mathbf{g} - C\boldsymbol{\omega} \quad (4)$$

where \mathbf{r}_{Oc} is the position vector of COM with respect to the collision point, m the mass, \mathbf{g} the

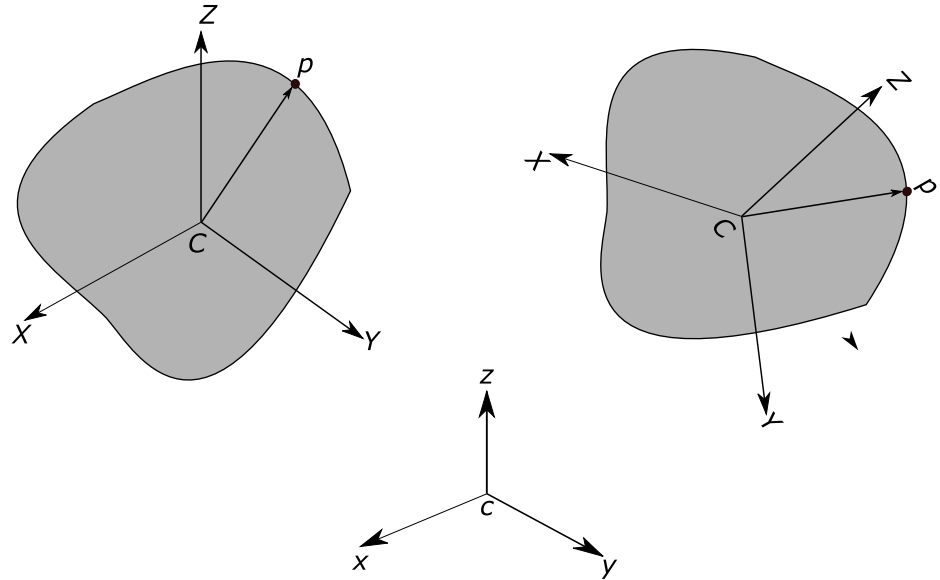


Fig 4. Orientation of the body with respect to the body space and the inertial coordinate systems. Body space coordinate system (CXY Z) moves and rotates with the same translational and angular velocity of the rigid body.

<https://doi.org/10.1371/journal.pone.0243716.g004>

gravitational acceleration vector, C the damping coefficient, and ω the angular velocity of the object. Knowing the total external torque exerted on the body we can compute angular momentum of the body by integrating the following equation:

$$\frac{dL_o(t)}{dt} = M_o(t) \tag{5}$$

where L_o is angular momentum of the body with respect to the contact point defined as:

$$L_o(t) = I_o(t)\omega(t) \tag{6}$$

where $I_o(t)$ is the instantaneous moment of inertia and can be calculated in terms of the constant-value moment of inertia in the body reference frame I^B , the rotation matrix, and position vector of COM with respect to the collision point as:

$$I_o(t) = R(t)I^B R^T(t) + m[\|\mathbf{r}_{oc}\|^2 E_3 - \mathbf{r}_{oc}\mathbf{r}_{oc}^T] \tag{7}$$

where $\|\cdot\|$ and \cdot^T denote L^2 norm and transpose operations, respectively and E_3 is the 3×3 identity matrix.

The last equation to close the system of differential equation and fully determine the dynamic of rigid body rotation is time evolution of the rotation matrix due to the angular velocity of the body. However, in any numerical scheme for time integration of the rotational matrix some computational error is unavoidable. This error will be accumulated over time resulting in a skewing effect on the shape of the object [35]. In order to circumvent this problem and following our previous method [36], a quaternion vector is used instead of a rotational matrix to represent the orientation of the body in 3-D space [37]. Quaternions use four parameters to describe the three degrees of freedom of any arbitrary rotation. The advantage of implementing the quaternion method is that by re-normalizing the value of the quaternion to unit length, the numerical drift that is caused during the numerical integration can be eliminated [35]. Because of this property, it is desirable to represent the orientation of a body directly as a unit quaternion $q(t)$. Since we still need the rotation matrix to calculate the

moment of inertia (Eq 7) and position vector (Eq 3), $R(t)$ is computed as an auxiliary variable from the quaternion vector $\mathbf{q}(t) = [q_1 \ q_2 \ q_3 \ q_4]^T$ as follows:

$$R = \begin{bmatrix} 1 - 2q_2^2 - 2q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & 1 - 2q_1^2 - 2q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 1 - 2q_1^2 - 2q_2^2 \end{bmatrix} \tag{8}$$

Finally, the time evolution of the quaternion vector is used to update the particle’s orientation during the simulation according to:

$$\frac{d\mathbf{q}(t)}{dt} = \frac{1}{2}\mathbf{q}(t) \times \boldsymbol{\omega} = \frac{1}{2} \begin{bmatrix} 0 & -\omega_1 & -\omega_2 & -\omega_3 \\ \omega_1 & 0 & -\omega_3 & \omega_2 \\ \omega_2 & \omega_3 & 0 & -\omega_1 \\ \omega_3 & -\omega_2 & \omega_1 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{9}$$

Eqs 5 to 9 are discretized in time using explicit integration scheme and solved iteratively for all the unknown variables of the particle’s kinematics ($\boldsymbol{\omega}$, R , \mathbf{q}) as follows:

1. The total torque exerted on the particle for the current time step $n + 1$ is calculated explicitly based on the known angular velocity and vector position of COM from the previous time step n (see Eq 4).

$$\mathbf{M}^{n+1} = \mathbf{M}(\boldsymbol{\omega}^n, \mathbf{q}^n) \tag{10}$$

2. Eq 5 is integrated explicitly (trapezoidal rule) to obtain angular momentum for time step $n + 1$.

$$\mathbf{L}^{n+1} = \mathbf{L}^n + \Delta t \mathbf{M}^{n+1} \tag{11}$$

3. To obtain orientation and angular velocity of the body for the next time step we have to solve coupled nonlinear Eqs 6 to 9 iterating over pseudo-time step l as:

$$\frac{\mathbf{q}^{l+1} - \mathbf{q}^l}{\Delta\tau} + \frac{\mathbf{q}^l - \mathbf{q}^n}{\Delta t} = \frac{1}{2} \left(\frac{1}{2}\mathbf{q}^l \times \boldsymbol{\omega}^l + \frac{1}{2}\mathbf{q}^n \times \boldsymbol{\omega}^n \right) \tag{12}$$

$$R^{l+1} = R(\mathbf{q}^{l+1}) \tag{13}$$

$$\mathbb{I}_O^{l+1}(t) = R^{l+1} \mathbb{I}_b R^{l+1T} + m \left[\|\mathbf{r}_{OC}^n\|^2 \mathbf{E}_3 - \mathbf{r}_{OC}^n \mathbf{r}_{OC}^n T \right] \tag{14}$$

$$\boldsymbol{\omega}^{l+1} = (\mathbb{I}^{l+1})^{-1} \mathbf{L}^{n+1} \tag{15}$$

4. Check for convergence of the quaternion solution:

$$\|\mathbf{q}^{l+1} - \mathbf{q}^l\| < \epsilon$$

where ϵ is a preset convergence ratio set equal to $\epsilon = 10^{-8}$.

5. At the convergence we set:

$$\mathbf{q}^{n+1} = \mathbf{q}^{l+1} \quad (16)$$

$$\mathbf{R}^{n+1} = \mathbf{R}^{l+1}$$

$$\boldsymbol{\omega}^{n+1} = \boldsymbol{\omega}^{l+1}$$

Upon the calculation of kinematics for the current time step $n + 1$, the orientation of particle is updated (according to Eq 3) by rotating it about the collision point.

2.3.2 Rotation about a line of contact. Equations of rotational motion explained earlier are used for the rotation about a point for both smooth and non-smooth particles. For rotation about an axis of rotation which might occur during the rotation of non-smooth particles for line and surface types of contact, the unit vector of rotation axis \mathbf{e}_r is first calculated using linear regression technique through all contact points. Then the external torque is projected on this vector as:

$$\sum \mathbf{M}_r(t) = (\mathbf{r}_{OC} \times m\mathbf{g} - \mathcal{C}\boldsymbol{\omega}) \cdot \mathbf{e}_r \quad (17)$$

Eqs 5 to 9 are integrated in time as explained earlier (for a rotation about the collision point) but this time the updated angular velocity is projected on the axis of rotation as:

$$\boldsymbol{\omega}_r(t) = (\boldsymbol{\omega} \cdot \mathbf{e}_r) \mathbf{e}_r \quad (18)$$

Consequently, the particle would be constrained to rotate about the collision line.

2.4 Stability condition

As mentioned earlier, in the final phase of collision a smooth particle goes through a continuous oscillatory rotation until it reaches its final position (minimum gravitational potential energy). For a non-smooth particle, there is a different scenario. During the rotation, there will be a unique orientation in which the particle is stable and the rotation suddenly stops. If this stable orientation is not detected, the particle would oscillate on a flat/concave surface, which is non-physical. Therefore, checking the stability of a non-smooth particle during the rotation is essential for realistic simulations.

Two situations for the stability of a particle are monitored when it collides with a surface. In the first stability condition, which occurs more frequently, a contact surface consisting of all contact points is detected. Contact points on the contact surface belong to a face of the particle and, consequently, these material points belong to the triangular surface mesh on the particle surface (see Figs 6–8). If all vertices of a triangular mesh are within the collision threshold, i.e., all vertices are contact points, we call the mesh element a contact element (green triangle in Fig 5a). After all contact elements are detected, a ray-casting algorithm adopted from [38] is used to see if the projection of the particle's COM intersects with any of contact elements. If a half ray shoots from the COM along the vertical direction intersects with a contact element that means the COM projection on the wall lays inside the contact surface and the particle is

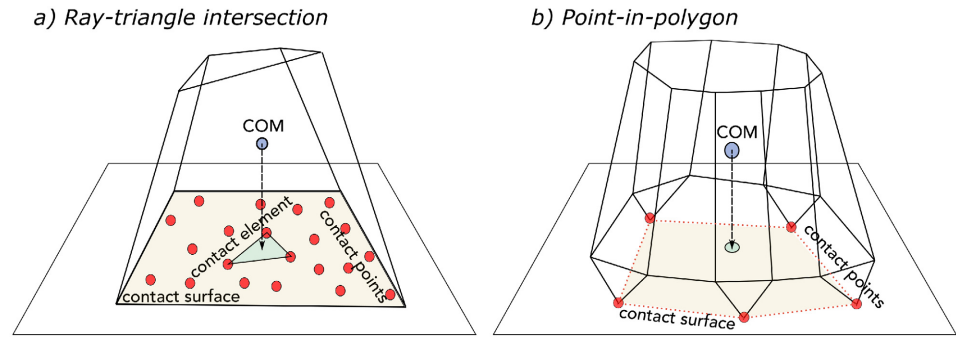


Fig 5. Sketch showing the ray-casting method. (a) In the ray-triangle scheme, a half ray shoots from the COM along the vertical direction. If this ray intersects with a contact element it means the COM projection on the wall lay inside the contact surface and the particle is stable. (b) In a point-in-polygon scheme, a polygon consisting of all contact points is considered. If the projection of particle's COM on the wall is inside this polygon, the particle is considered stable.

<https://doi.org/10.1371/journal.pone.0243716.g005>

stable (see Fig 5a). The basic idea of this ray-triangle intersection method is that for a triangle defined by its three vertices $(v_1^i, v_2^i, v_3^i)(i = 1, 2, 3)$, where v_j^i is the i^{th} Cartesian coordinate of the j^{th} vertex, any point $T = (\tau_1, \tau_2, \tau_3)$, where τ_i is the i^{th} Cartesian component of point T , within the triangle can be expressed as follows

$$\tau_i(a, b) = (1 - a - b)v_i^1 + av_i^2 + bv_i^3 \tag{19}$$

with $a \geq 0, b \geq 0$ and $a + b \leq 1$. The ray shoots from COM and pointing downward can be expressed as $\Gamma = (C_1, C_2, C_3 - \delta)$, where C_i is the i^{th} Cartesian component of COM and $\delta > 0$. Finding the intersection between the ray and the triangle is equivalent to finding the solution to the following equation:

$$\Gamma_i(\delta) = \tau_i(a, b) \tag{20}$$

If the solution to Eq 20 satisfies $\delta > 0, a \geq 0, b \geq 0$ and $a + b \leq 1$, then the ray intersects with a triangle.

In the second stability condition, which only happens for highly jagged particle surfaces, very few collision points are found (Fig 5b). These points, contrary to the previous scenario, do not necessarily belong to a collision element, i.e., they may belong to surface elements that are not entirely lay on the wall surface. The number of these collision points for this type of surface are significantly less than the previous situation and are quite limited. After identifying all collision points, a polygon consisting all these points is considered and then we check to see if the projection of particle's COM on the wall (shown as green dot in Fig 5b) is inside this polygon, which indicates the stability of the particle. This test is the classical problem of point-in-polygon problem in computational geometry. This problem is solved by the so-called ray-casting method [39], which requires the location of the polygon vertices (contact points) and the point (projection of COM on the wall) as inputs. Starting from the projection of particle's COM on the wall (green dot in Fig 5b), a random half-infinite ray is casted and the number of intersections between the half ray and polygon edges (shown as dotted lines in Fig 5b) is counted. If the number of intersections is odd then the point is located inside the polygon, otherwise it is located outside. Note that, we automatically check both scenarios for the stability of a particle and only one is a sufficient condition to consider the particle stable.

3 Results and discussion

In this section, the method is validated and verified against analytical and benchmark solutions. It is shown that the method can simulate the final phase of a collision (without rebound) until the particle reaches a physical and realistic equilibrium (stable) position. Results for one smooth and three non-smooth particles are presented and discussed.

As it was mentioned in *Materials and methods*, for smooth particles the final stable state is reached smoothly through a continuous process in terms of kinematics (e.g. angular velocity) and there is always a single collision point. For a non-smooth particle, on the other hand, the final stable state is reached abruptly and the type of contact could change from a point to a line and finally to a surface type. A major difference in terms of implementing the governing equation for both surface types is the value of the damping coefficient in Eq 4. The value of the damping coefficient C depends on many factors such as surface roughness and object's material. For particles with a continuous curvature (smooth) a non-zero positive value can be considered $C > 0$, whereas for non-smooth particles, it is assumed that the particle does not oscillate around the contact point but simply rotates until it lays over one of its surfaces or some single contact points abruptly. This assumption is physically acceptable for objects whose density is significantly higher than air. To implement this scenario the damping coefficient for a non-smooth particle is set to be zero ($C = 0$).

3.1 Smooth particles

The settling of a gravity-driven triaxial ellipsoid on a plane surface is simulated for two different values of damping coefficients (Fig 6)—see also the [S1 Video](#). The triaxial ellipsoid has major diameters of (1.25, 1, 0.57)cm and density of $1.2\text{gr}/\text{cm}^3$ and it is initially rotated such that its orientation with respect to the inertial frame of reference xyz (see Fig 4) agrees with what is shown in Fig 6a and 6b. The damping coefficient for two tested cases are chosen as $C = (0.75, 1.5)\text{gr}\cdot\text{cm}^2/\text{sec}$.

At the beginning of the collision procedure ($t = 0$ in Fig 6c and 6e), the particle's COM has the highest elevation with respect to the surface and it is located at the right-hand side of the collision point. Therefore, the particle starts to rotate clockwise. This angular motion for the particle with the lower value of damping coefficient $C = 0.75$ (shown in Fig 6c and 6d) is faster than the one with the higher value of $C = 1.5$ (shown in Fig 6e and 6f). This can be easily verified by comparing amplitude of components of angular velocity ω_x and ω_y , in Fig 6d and 6f. For instance, for the particle with a lower damping coefficient it takes $t = 0.035\text{sec}$ to sweep $\theta = 0.65\text{rad}$ along x -axis, while for the particle with a higher damping coefficient the required time for the same swept angle is $t = 0.039\text{sec}$. The clockwise rotation continues until the particle's COM reaches its second elevation peak. The elevation of this peak is inversely proportional to the damping coefficient. In other words, the particle with damping coefficient $C = 0.75$ sweeps a maximum angle of $\theta_{max} = 1.79\text{rad}$ during the first oscillation, which takes $t = 0.064\text{sec}$, whereas for the particle with damping coefficient $C = 1.5$ the required time for the maximum swept angle of $\theta_{max} = 1.55\text{rad}$ is $t = 0.070\text{sec}$. At this time instant, the particle's COM is located at the left-hand side of the collision point and consequently, the particle starts to rotate counterclockwise. The counterclockwise rotation continues until the particle's COM reaches to its third peak, which is lower than previous ones. This cycle continues and particle oscillates around the collision point until it reaches the final steady state position (Fig 6c and 6e) in which the particle's COM has the minimum elevation, i.e., minimum gravitational potential energy. Fig 6d and 6f clearly show that the higher the value of damping coefficient, the sooner the particle reaches steady state with fewer oscillations around the contact point.

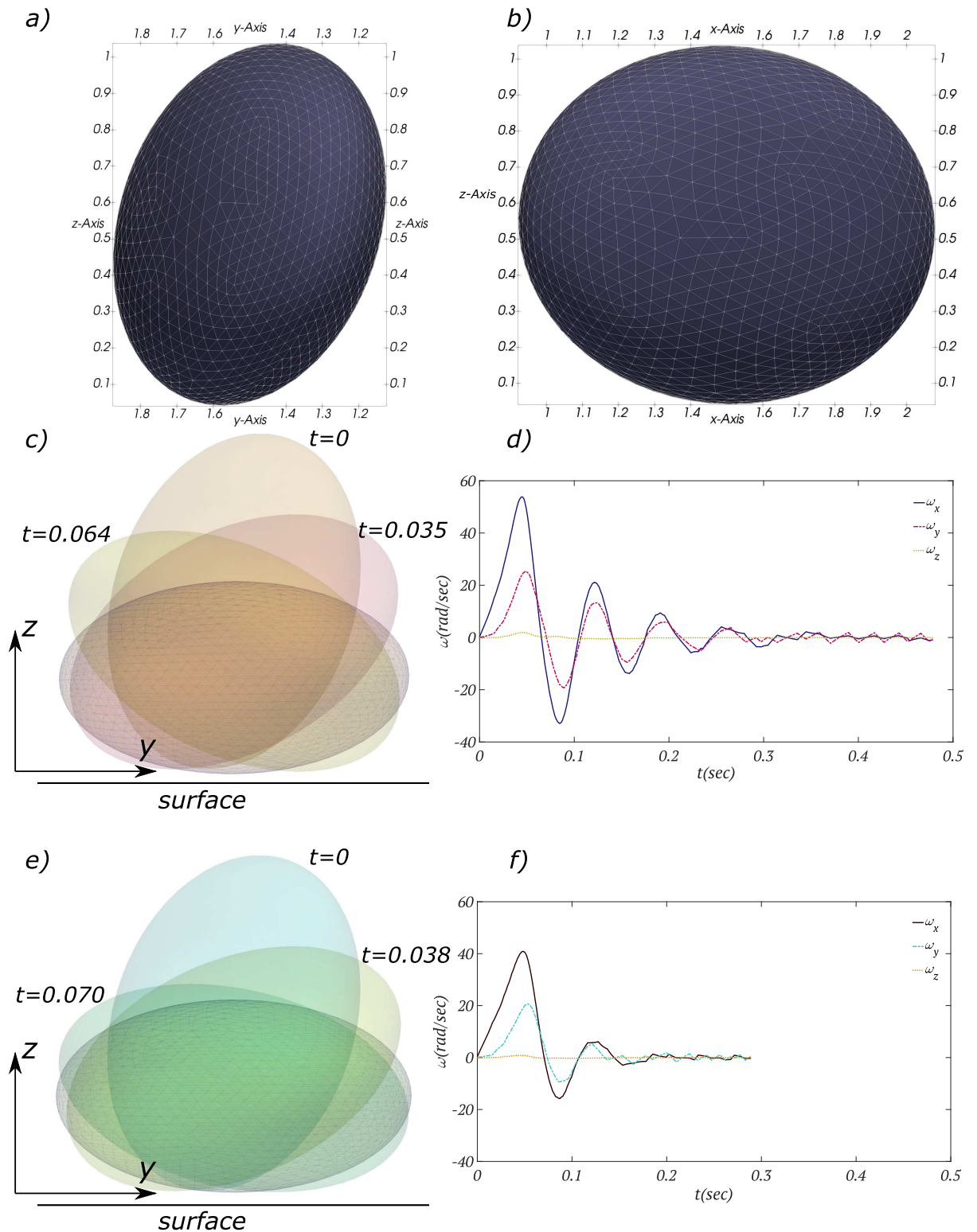


Fig 6. Settling of a triaxial ellipsoid on a flat surface. a) and b) Projections of the particle on Cartesian coordinate planes with correlated scales are shown for the initial orientation. c) and d) Angular velocity and time series of the particle with damping coefficient $C = 0.75$ are shown. e) and f) Angular velocity and time series of the particle with damping coefficient $C = 1.5$ are shown.

<https://doi.org/10.1371/journal.pone.0243716.g006>

This type of rotation with alternative signs and a damping amplitude qualitatively agrees with numerical results of Mohaghegh and Udaykumar [23] for a 2-D ellipse in a viscous fluid.

3.2 Non-smooth particles

Contrary to a smooth particle, the final equilibrium orientation for a non-smooth particle during the settling could depend on its initial orientation. For instance, Fig 7a and 7b show a cylindrical particle of density 0.96gr/cm^3 , height of $h = 0.67\text{cm}$ and radius of $r = 0.67\text{cm}$ with an axis oriented at angle θ with respect to z -direction. It can be analytically verified that the gravitational moment about the contact point (the first term on the right-hand-side of Eq 4) changes its sign at a critical angle of $\theta_c = \pi/2 - \arctan(h/2r)$. In other words, for the cylinder shown in Fig 7, the gravitational moment about the x -axis is negative for $\theta < 1.107\text{rad}$ and positive for $\theta > 1.107\text{rad}$, and $\theta = 1.107\text{rad}$ is the unsteady equilibrium position. Since the damping coefficient is set to be zero for non-smooth particles, the sign of angular velocity and, consequently, the angle of rotation agrees with the sign of the gravitational moment. In order to numerically verify this result, we tested the cylindrical particle with two initial orientations: $\theta_0 = 1.106\text{rad}$ (shown in Fig 7c) and $\theta_0 = 1.108\text{rad}$ (shown in Fig 7d).

For both initial orientations the rotation starts from the vicinity of the unsteady equilibrium position $\theta_0 = 1.107 \pm 0.001\text{rad}$ and in both cases, collision starts with a handful of adjacent contact points. The axis of rotation is consequently obtained by finding the best line that passes through collision points during the rotation as explained in section 2.2 (see *line-type contact*). As can be seen in Fig 7c and 7d, the axis of rotation remains along x -axis in both cases, where the cylinder initially starts from $\theta_0 = 1.106$ rotates clockwise ($\omega_x < 0$), and the cylinder initially starts from $\theta_0 = 1.108$ rotates counterclockwise ($\omega_x > 0$).

The absolute value of angular velocity for both cases are shown and compared in Fig 7e. Since the initial conditions in both cases are symmetric about the equilibrium position, the rotations are expected to be symmetric as well. This analytical answer is verified by our numerical results and as it can be clearly seen in the plot, the magnitude of angular velocities are equal (with opposite signs) at any given time instant. The rotation for the cylinder initially started from $\theta_0 = 1.108$ takes $t = 0.976\text{sec}$ sweeping an angle of $\Delta\theta = 0.463\text{rad}$, whereas the other cylinder rotates for a longer time $t = 1.105\text{sec}$ to sweep $\Delta\theta = 1.106\text{rad}$.

Upon reaching the equilibrium state, particle's motion terminates abruptly (surface-type collision) where a line-type collision is followed by a surface-type collision—see also the S2 Video. One might argue that for the cylinder initially started from $\theta_0 = 1.108$, the final collision state should be of a line-type mathematically, but in real physical and computational situations, there would be a narrow strip contact surface when the cylinder become steady as shown in Fig 7d.

Another example that can show the transition from a point-type contact to a line-type contact before a final steady surface-type collision is shown in Fig 8. The regular tetrahedron in this simulation has a density of 1.2gr/cm^3 and all four faces are equilateral triangles with edge length of 1cm . When the particle reaches the surface for the first time, it collides with the wall with its head, i.e., a point-type collision (Fig 8b). In this collision type, there is one constraint in the rotation as the contact point does not change during the rotation. The axis of rotation passes through this contact point and is perpendicular to the line that connects the contact point and the projection of COM on the wall surface. The triangular pyramid rotates due to its gravitational torque about this axis until other points located on one of its edges collide with the wall and a contact line is formed (Fig 8c). This time instant can be tracked on the plot of Fig 8e, which shows components of angular velocities, as the first discontinuity. The first phase of rotation from the appearance of contact point till the formation of contact line takes $\Delta t = 0.0053\text{sec}$ for this simulation.

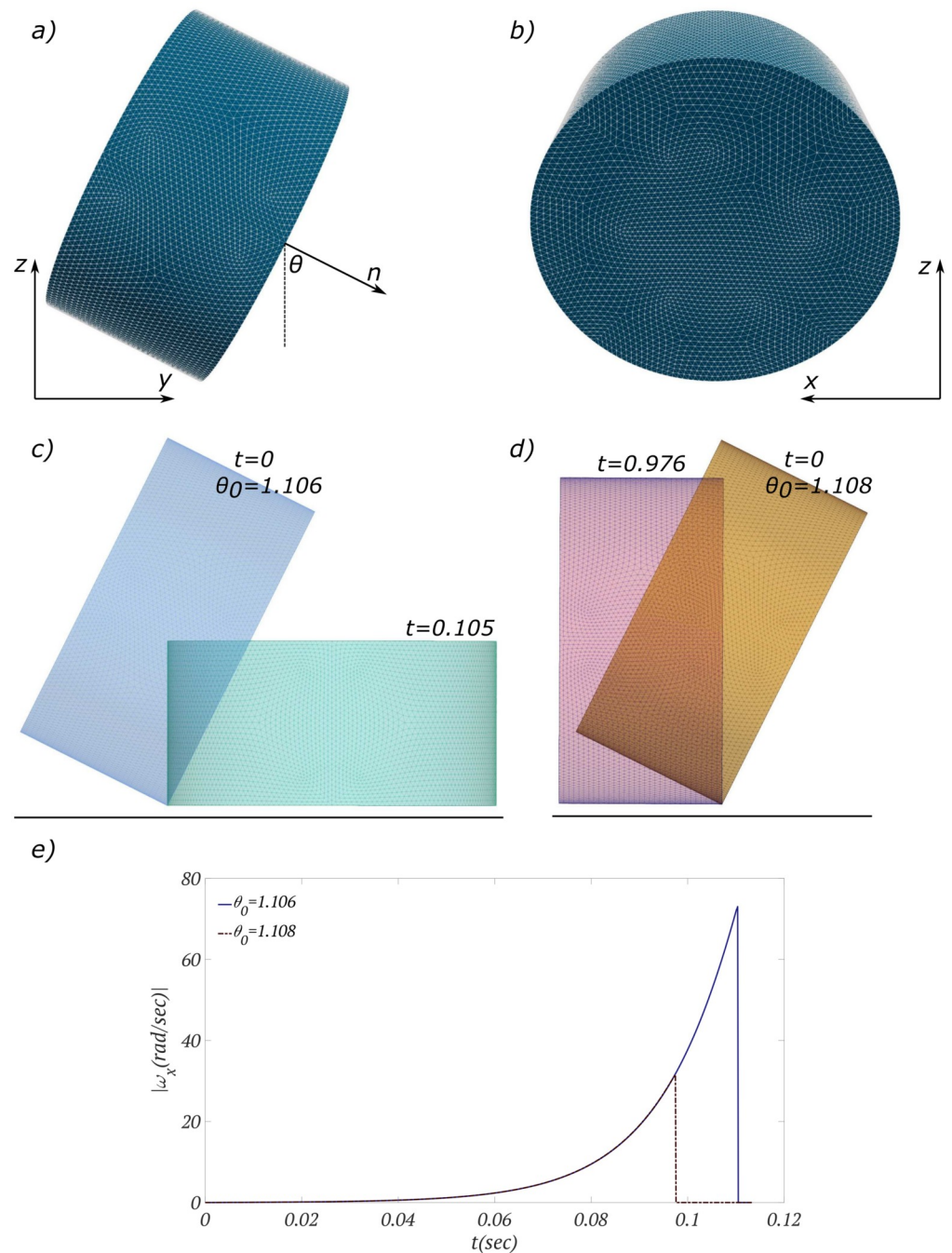


Fig 7. Settling of a cylinder on a flat surface. a) and b) Initial orientation of the particle with respect to an inertial frame of reference xyz is shown. c) Axis of the cylinder is initially at $\theta_0 = 1.106\text{rad}$ and it finally lays on its base surface. d) Axis of the cylinder is initially at $\theta_0 = 1.108\text{rad}$ and it finally lays on its side surface. e) Angular velocity of both cases during the settling on the surface are compared.

<https://doi.org/10.1371/journal.pone.0243716.g007>

When the contact line forms, the particle rotates about this line, i.e., contact line acts as the new axis of rotation (Fig 8c). The sudden change in the axis of rotation could result in a abrupt change in components of angular velocities. For instance, upon the transition from contact point to contact line (phase one to phase two of rotation), ω_x changes its sign from positive to

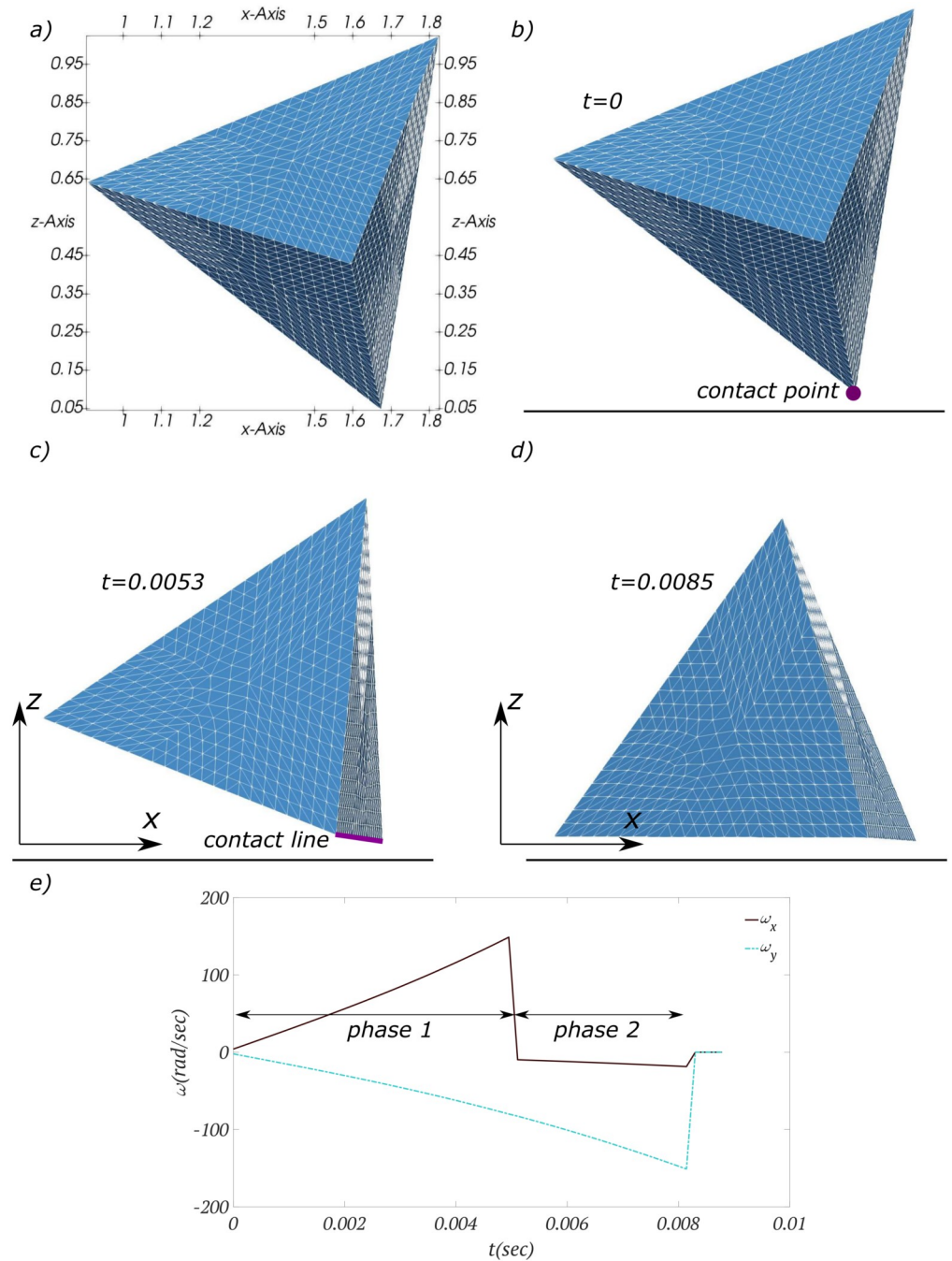


Fig 8. Settling of a regular tetrahedron on a flat surface. a) Projections of the particle on Cartesian coordinate planes with correlated scales are shown for the initial orientation. b) The particle collides with the wall with its head and a point-type collision occurs. c) Multiple contact points form a contact line. d) The pyramid lays on one of its flat surfaces at equilibrium. e) Two components of angular velocity are shown. Sudden change in angular velocity is the result of the change in axis of rotation, i.e. transition from phase 1 to phase 2.

<https://doi.org/10.1371/journal.pone.0243716.g008>

negative in Fig 8e. The rotation in this phase continues until other points on the bottom face of the particle reach the threshold distance for the collision with the wall, i.e., a surface-type collision is detected. This time instant can be seen on the plot of Fig 8e as the second discontinuity. The second phase of rotation from the appearance of contact line until the formation of contact surface takes $\Delta t = 0.0032\text{sec}$ for this simulation.

The surface-type collision is immediately followed by the stable condition when the COM projection intersects with a contact element, i.e., Ray-triangle intersection stability condition (Fig 5a). In this final state, the pyramid is on one of its flat surfaces (Fig 8d). Three types of contact and transition from one to another can be better seen in the S3 Video.

As one might expect and previously mentioned in section 2.4, some particles have highly jagged surfaces and do not have flat or smooth surfaces similar to what are shown in Figs 6–8. In order to show the capability of our framework to handle collision and stability of any arbitrary irregular-shaped particle, a highly skewed and jagged-shape particle is tested and the result is shown in Fig 9—see also the S4 Video. For this type of particle shape as shown in the figure, only a few contact points (indicated as color circles) can be expected depending on the initial orientation. Similar to previous tests, the particle first hits the wall with a single sharp head and starts to rotate around this point (red circle in Fig 9a). The rotation continues until the second contact point (orange circle in Fig 9b) is found. From this time, the particle rotates along an axis, which passes through both contact points. If there are more than two points, the best fit line technique would be used (see section 2.2). Similar to the rotation of a tetrahedron in Fig 8, for this particle, the axis of rotation can change as the surface is jagged and bumpy. This can be seen in Fig 9c where a new contact point (green circle in Fig 9c) is found after a bit of particle rotation about the first axis and replaces the previous contact point. Orientation of the axis of rotation changes at this moment, consequently. The shift of the rotation axis from the first one to the second one can be better seen in the S4 Video. This procedure is repeated until the particle rests stably on several collision points (Fig 9d) such that the projection of COM on the wall locates inside a polygon that passes through collision points (see Point-in-polyhedron stability condition and Fig 5b).

4 Conclusion

A discrete element algorithm is developed to resolve the settling of an arbitrary-shaped particle on a flat surface. In this method, the particle's orientation is solved such that the particle reaches a realistic and physical equilibrium (stable) state, which may not be guaranteed by traditional DEM that add a contact (repulsive) force (torque) based on the amount of overlap. This is achieved by identifying the type of contact (section 2.2), i.e., point, line, surface; solving the rigid body dynamics based on the type of contact (section 2.3); and identifying the equilibrium (stable) state (section 2.4), i.e., stable on a surface or on a few contact points. The method has been validated for different geometries and shown to model the analytical stable state.

It should be noted that the current method does not consider rebound when particles fall on a surface. This is an acceptable physical assumption for several gravity-driven objects including particles with small normal velocity (low incoming kinetic energy) or those whose inertia is at the order of the viscous force exerted from the ambient fluid especially for non-smooth surfaces. Even in cases that a particle (usually smooth particles) bounces back from the surface, it loses some of its momentum (energy) during each collision and eventually goes through a collision without bouncing, i.e., final phase of collision. Therefore, this method can be integrated with other collision treatments, in which the rebound from the surface is also a part of a wall-particle interaction, e.g., non-zero coefficient of restitution and friction [22]. This will be carried out as part of a future work. In fact, this method can be implemented in

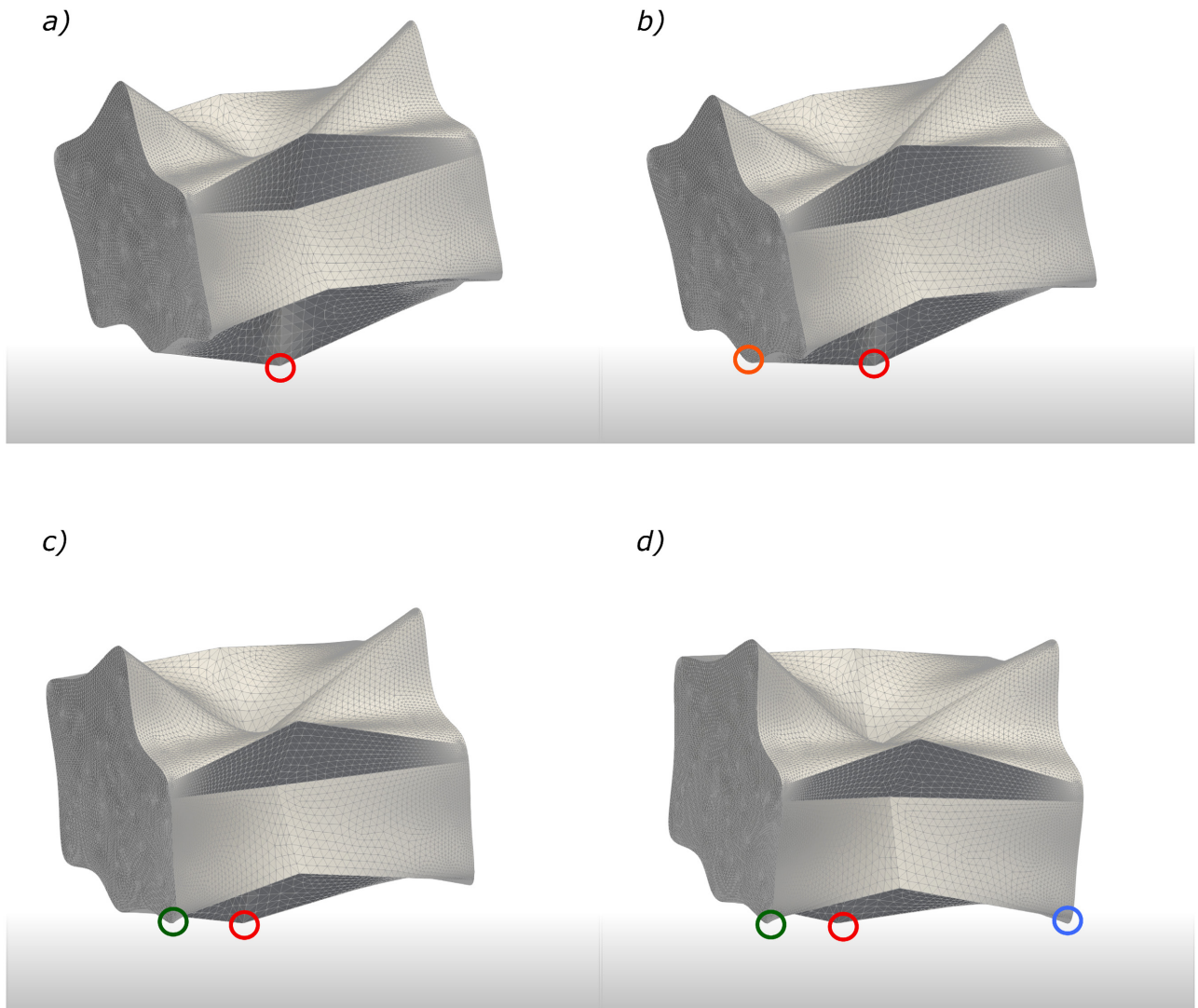


Fig 9. Time series of the collision procedure and resting of an irregular particle on a rigid wall is shown. For this type of particle shape in the shown orientation only a few collision points (indicated as color circles) are expected. a) The particle first hits the wall with a single sharp head and starts to rotate around this point. b) The rotation continues until the second collision point is found. From this time, the particle rotates along an axis, which passes through both collision points. c) After a bit rotation along the first axis, another collision point is found and replaces the second one. d) Finally, particle is rested on three collision points such that the projection of COM on the wall locates inside these points.

<https://doi.org/10.1371/journal.pone.0243716.g009>

any discrete element model to obtain a more accurate and realistic final equilibrium state of a particle on the wall. In addition, this numerical scheme will be coupled to our immersed boundary method [6, 36] to simulate sedimentation of rigid particles in a viscous fluid in the future.

Supporting information

S1 Video. Settling of a triaxial ellipsoid on a plane surface.

(MOV)

S2 Video. Settling of a cylinder on a plane surface.

(AVI)

S3 Video. Settling of a tetrahedral pyramid on a plane surface.

(AVI)

S4 Video. Settling of an irregular-shaped particle on a plane surface.

(AVI)

Author Contributions**Supervision:** Iman Borazjani.**Writing – original draft:** Mohsen Daghooghi.**Writing – review & editing:** Mohsen Daghooghi, Iman Borazjani.**References**

1. Girolami L, Hergault V, Vinay G, Wachs A. A three-dimensional discrete-grain model for the simulation of dam-break rectangular collapses: comparison between numerical results and experiments. *Granular Matter*. 2012; 14(3):381–392. <https://doi.org/10.1007/s10035-012-0342-3>
2. Cleary PW. Industrial particle flow modelling using discrete element method. *Engineering Computations*. 2009; 26(6):698–743. <https://doi.org/10.1108/02644400910975487>
3. Teufelsbauer H, Wang Y, Chiou MC, Wu W. Flow–obstacle interaction in rapid granular avalanches: DEM simulation and comparison with experiment. *Granular Matter*. 2009; 11(4):209–220. <https://doi.org/10.1007/s10035-009-0142-6>
4. Mandø M, Rosendahl L. On the motion of non-spherical particles at high Reynolds number. *Powder Technology*. 2010; 202(1-3):1–13. <https://doi.org/10.1016/j.powtec.2010.05.001>
5. Zhong W, Yu A, Liu X, Tong Z, Zhang H. DEM/CFD-DEM modelling of non-spherical particulate systems: theoretical developments and applications. *Powder Technology*. 2016; 302:108–152. <https://doi.org/10.1016/j.powtec.2016.07.010>
6. Daghooghi M, Borazjani I. The effects of irregular shape on the particle stress of dilute suspensions. *Journal of Fluid Mechanics*. 2018; 839:663–692. <https://doi.org/10.1017/jfm.2018.65>
7. Jin F, Xin H, Zhang C, Sun Q. Probability-based contact algorithm for non-spherical particles in DEM. *Powder technology*. 2011; 212(1):134–144. <https://doi.org/10.1016/j.powtec.2011.05.003>
8. Cundall PA, Strack OD. A discrete numerical model for granular assemblies. *geotechnique*. 1979; 29(1):47–65. <https://doi.org/10.1680/geot.1979.29.1.47>
9. Wachs A. PeliGRUFF, a parallel DEM-DLM/FD direct numerical simulation tool for 3D particulate flows. *Journal of Engineering Mathematics*. 2011; 71(1):131–155. <https://doi.org/10.1007/s10665-010-9436-2>
10. Zhu H, Zhou Z, Yang R, Yu A. Discrete particle simulation of particulate systems: a review of major applications and findings. *Chemical Engineering Science*. 2008; 63(23):5728–5770. <https://doi.org/10.1016/j.ces.2008.08.006>
11. Moreno-Atanasio R, Williams RA, Jia X. Combining X-ray microtomography with computer simulation for analysis of granular and porous materials. *Particology*. 2010; 8(2):81–99. <https://doi.org/10.1016/j.partic.2010.01.001>
12. Zhu H, Zhou Z, Yang R, Yu A. Discrete particle simulation of particulate systems: theoretical developments. *Chemical Engineering Science*. 2007; 62(13):3378–3396. <https://doi.org/10.1016/j.ces.2006.12.089>
13. Dong K, Wang C, Yu A. A novel method based on orientation discretization for discrete element modeling of non-spherical particles. *Chemical Engineering Science*. 2015; 126:500–516. <https://doi.org/10.1016/j.ces.2014.12.059>
14. Wachs A. A DEM-DLM/FD method for direct numerical simulation of particulate flows: Sedimentation of polygonal isometric particles in a Newtonian fluid with collisions. *Computers & Fluids*. 2009; 38(8):1608–1628. <https://doi.org/10.1016/j.compfluid.2009.01.005>
15. Kafui K, Thornton C, Adams M. Discrete particle-continuum fluid modelling of gas–solid fluidised beds. *Chemical Engineering Science*. 2002; 57(13):2395–2410. [https://doi.org/10.1016/S0009-2509\(02\)00140-9](https://doi.org/10.1016/S0009-2509(02)00140-9)
16. Zhou ZY, Zou RP, Pinson D, Yu AB. Dynamic simulation of the packing of ellipsoidal particles. *Industrial & Engineering Chemistry Research*. 2011; 50(16):9787–9798. <https://doi.org/10.1021/ie200862n>

17. Liu S, Zhou Z, Zou R, Pinson D, Yu A. Flow characteristics and discharge rate of ellipsoidal particles in a flat bottom hopper. *Powder Technology*. 2014; 253:70–79. <https://doi.org/10.1016/j.powtec.2013.11.001>
18. Davis RH, Serayssol JM, Hinch E. The elastohydrodynamic collision of two spheres. *Journal of Fluid Mechanics*. 1986; 163:479–497. <https://doi.org/10.1017/S0022112086002392>
19. Schwarzkopf JD, Sommerfeld M, Crowe CT, Tsuji Y. *Multiphase flows with droplets and particles*. CRC press; 2011.
20. Tsuji Y, Shen N, Morikawa Y. Lagrangian simulation of dilute gas-solid flows in a horizontal pipe. *Advanced Powder Technology*. 1991; 2(1):63–81. [https://doi.org/10.1016/S0921-8831\(08\)60721-9](https://doi.org/10.1016/S0921-8831(08)60721-9)
21. Wynn E. Simulations of rebound of an elastic ellipsoid colliding with a plane. *Powder Technology*. 2009; 196(1):62–73. <https://doi.org/10.1016/j.powtec.2009.07.004>
22. Gui N, Yang X, Tu J, Jiang S. A generalized particle-to-wall collision model for non-spherical rigid particles. *Advanced Powder Technology*. 2016; 27(1):154–163. <https://doi.org/10.1016/j.apt.2015.12.002>
23. Mohaghegh F, Udaykumar H. Modeling collisions of arbitrary-shaped particles in simulations of particulate flows. *Powder technology*. 2019; 344:756–772. <https://doi.org/10.1016/j.powtec.2018.12.062>
24. Cundall PA. In: *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*. vol. 25. Elsevier; 1988. p. 107–116.
25. Ting JM, Khwaja M, Meachum LR, Rowell JD. An ellipse-based discrete element model for granular materials. *International Journal for Numerical and Analytical Methods in Geomechanics*. 1993; 17(9):603–623. <https://doi.org/10.1002/nag.1610170902>
26. Williams JR, Pentland AP. Superquadrics and modal dynamics for discrete elements in interactive design. *Engineering Computations*. 1992; 9(2):115–127. <https://doi.org/10.1108/eb023852>
27. Vu-Quoc L, Zhang X, Walton O. A 3-D discrete-element method for dry granular flows of ellipsoidal particles. *Computer methods in applied mechanics and engineering*. 2000; 187(3-4):483–528. [https://doi.org/10.1016/S0045-7825\(99\)00337-0](https://doi.org/10.1016/S0045-7825(99)00337-0)
28. Lin X, Ng TT. A three-dimensional discrete element model using arrays of ellipsoids. *Geotechnique*. 1997; 47(2):319–329. <https://doi.org/10.1680/geot.1997.47.2.319>
29. Barr AH. Superquadrics and angle-preserving transformations. *IEEE Computer graphics and Applications*. 1981; 1(1):11–23. <https://doi.org/10.1109/MCG.1981.1673799>
30. Podlozhnyuk A, Pirker S, Kloss C. Efficient implementation of superquadric particles in Discrete Element Method within an open-source framework. *Computational Particle Mechanics*. 2017; 4(1):101–118. <https://doi.org/10.1007/s40571-016-0131-6>
31. Lu G, Third J, Müller C. Critical assessment of two approaches for evaluating contacts between superquadric shaped particles in DEM simulations. *Chemical engineering science*. 2012; 78:226–235. <https://doi.org/10.1016/j.ces.2012.05.041>
32. Hogue C. Shape representation and contact detection for discrete element simulations of arbitrary geometries. *Engineering Computations*. 1998; 15(3):374–390. <https://doi.org/10.1108/02644409810208525>
33. Williams JR, O'Connor R. A linear complexity intersection algorithm for discrete element simulation of arbitrary geometries. *Engineering computations*. 1995;. <https://doi.org/10.1108/02644409510799550>
34. Borazjani I, Ge L, Sotiropoulos F. Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3D rigid bodies. *Journal of Computational physics*. 2008; 227(16):7587–7620. <https://doi.org/10.1016/j.jcp.2008.04.028>
35. Baraff D. An introduction to physically based modeling: rigid body simulation I—unconstrained rigid body dynamics. *SIGGRAPH course notes*. 1997; 82.
36. Daghooghi M, Borazjani I. The influence of inertia on the rheology of a periodic suspension of neutrally buoyant rigid ellipsoids. *Journal of Fluid Mechanics*. 2015; 781:506–549. <https://doi.org/10.1017/jfm.2015.504>
37. Kuipers JB, et al. *Quaternions and rotation sequences*. vol. 66. Princeton university press Princeton; 1999.
38. Möller T, Trumbore B. Fast, minimum storage ray-triangle intersection. *Journal of graphics tools*. 1997; 2(1):21–28. <https://doi.org/10.1080/10867651.1997.10487468>
39. Haines E. Point in polygon strategies. *Graphics gems IV*. 1994; 994:24–26.