A contact dynamics approach to the Granular Element Method

CrossMark

Keng-Wit Lim [a], Kristian Krabbenhoft [b], José E. Andrade [a],*

[a] Division of Engineering & Applied Science, California Institute of Technology, Pasadena, CA 91125, USA
[b] Centre for Geotechnical & Materials Modelling, University of Newcastle, NSW 2308, Australia

A R T I C L E   I N F O

A B S T R A C T

We present a contact dynamics (CD) approach to the Granular Element Method (GEM) Andrade et al. (2012) [1], abbreviated here as CD–GEM. By combining particle shape flexibility through Non-Uniform Rational Basis Splines, properties of implicit time-integration discretization (e.g., larger time steps) and non-penetrating constraints, as well as a reduction to a static formulation in the limit of an infinite time step, CD–GEM targets system properties and deformation regimes in which the classical discrete element method either performs poorly or simply fails; namely, in granular systems comprising of rigid or highly stiff angular particles and subjected to quasi-static or intense dynamic flow conditions. The integration of CD and GEM is made possible while significantly simplifying implementation and maintaining comparable performance with existing CD approaches.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

The objective of this paper is to develop a contact dynamics (CD) approach to the Granular Element Method (GEM) [1], abbreviated here as CD–GEM. CD–GEM targets system properties and deformation regimes in which the classical discrete element method (DEM) either performs poorly or simply fails; namely, in granular systems comprising of rigid or highly stiff angular particles and subjected to quasi-static or intense dynamic flow conditions. Within the context of such applications, we describe how CD–GEM offers a better solution in terms of particle morphology or shape representation and ease of implementation, while maintaining comparable performance with existing CD approaches. To motivate the development of our approach, we first refer the reader to Table 1 for a brief summary of the key features of and differences between CD and the classical DEM by Cundall and Strack [2]. In the following, we highlight the difficulties associated with CD and DEM followed by a description on how we eliminate them through CD–GEM.

The so-called Non-Smooth CD, originally developed by Moreau [6–9], is an alternative discrete approach to the DEM. The most prominent feature of CD, in contrast to that of classical DEM, is that the particles are considered perfectly rigid and the contact forces are determined as those that prevent interparticle penetration and at the same time satisfy the frictional stick–slip constraints. In their simplest forms, these contact laws are embodied in the so-called Signorini unilateral contact condition and classical Coulomb law, as shown in Fig. 1(a) and (b), respectively. Commensurate with these physical enhancements, however, is the need for both contact and constraint forces to be solved simultaneously or implicitly since the problem is nonlinear. The need for an implicit solution procedure till today remains the primary reason why CD is deemed much more complicated to implement than DEM. This has thwarted the wide adoption of CD despite the favorable performance that has been shown through a number of studies [10–20].

---

* Corresponding author.
   E-mail address: jandrade@caltech.edu (J.E. Andrade).

Table 1
Comparison of non-smooth contact dynamics and classical DEM.

Feature	N-S contact dynamics	Classic DEM
Normal contact	Rigid; unilateral contact[1,2] or non-penetration constraint directly included	Modeled using normal spring; particles overlap
Friction contact	Stick–slip frictional constraint[1,2] directly included	Imposes shear force incrementally using relative velocity from previous step
Time integration	Implicit, usually stable and with larger time step[3]	Explicit, with stability criterion; critical time step scales with inverse of spring frequency. Inefficient for highly stiff particles and cannot be applied to rigid particles
Collision response	Considers collisions and stick–slip frictional transitions simultaneously; velocities may be non-smooth	No real collisions and velocity jumps cannot occur due to continuous nature of contact spring
Damping	Numerical damping[4]	Through global and/or local damping devices, i.e., dashpots
Quasistatic limit	Can be directly included in formulation	Dynamic in nature; oscillations in solutions are typical; quasistatic limit is approached using global and/or local damping
Particle morphology representation[5,6]	Disc- or sphere-clustering and polyhedra	
Implementation difficulty	Intermediate to difficult[7]	Easy
Computational efficiency	Contact and constraint forces solved implicitly. Geometrical information (e.g., gap values and contact orientations) are stored in matrices as part of the solution procedure; higher memory requirement[8]	Contact forces are solved explicitly using particle overlap and previous velocities; time integration easily parallelized. Minimal storage of geometrical information; lower memory requirement

[1] Regularization to account for particle elasticity possible (see for e.g., [3]).
[2] See Fig. 1.
[3] Although the time step can be larger, it has to be reasonable so that collisions are properly resolved.
[4] Does not apply in the quasistatic limit.
[5] We list only those approaches, beyond ellipses/ellipsoids, that appear to be currently most widely applied.
[6] Improved using NURBS in CD–GEM.
[7] Made easier in CD–GEM.
[8] Managed in CD–GEM using efficient large-scale mathematical programming solvers (for e.g., [4,5]).
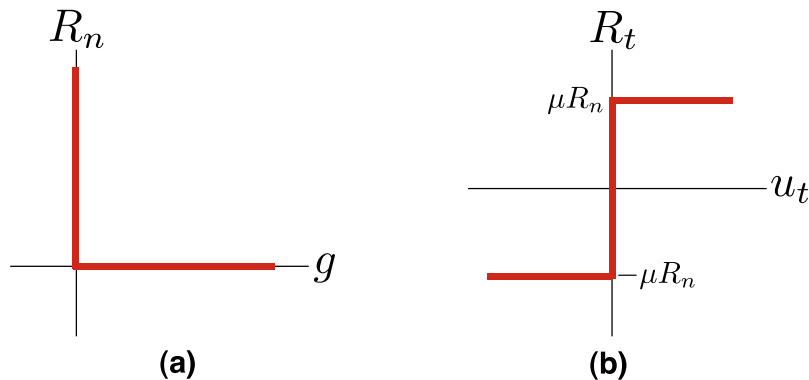
Fig. 1. Graph of non-smooth contact laws: (a) normal reaction force R_n against separation or gap g and (b) friction force R_t against slip displacement u_t; μ is the friction coefficient.

While there is wide applicability of DEM, its application has gone beyond its restriction as a tool that is strictly applicable only to materials with finite elasticity. For example, DEM is widely used as a tool to study real granular materials that are almost rigid or highly stiff in nature. Here, finite elasticity means that the contact interaction is essentially modeled using springs. Under explicit time integration algorithms that are typically used in DEM, the stable time step is restricted by the critical time step, which scales with the inverse of the contact spring-particle mass frequency. This results in infinitesimally small time steps if material parameters corresponding to highly stiff particles (e.g., rocks, sand, steel) are used. Although explicit integration algorithms can be easily parallelized, the runtime for stiff systems remains computationally prohibitive. One modeling technique commonly employed in practice to overcome this restriction is to simply reduce the contact stiffness, usually by two to four orders of magnitude, to the extent that particle kinematics obtained from simulations are still somewhat representative of the overall response of the actual system of interest. If quasi-static behavior is assumed to hold, usually

used in combination with stiffness tuning is mass scaling in which the particle masses are adjusted (usually increased) such that the combined spring-particle system frequency is lowered, increasing the time step size. In practice, model calibration by means of mass scaling and/or stiffness tuning is a delicate and cumbersome process. Another problem that is associated with the presence of contact springs and the dynamic nature of DEM is the introduction of unwanted oscillations or noise, with frequencies that increase with spring stiffness. This requires additional calibration of the global and/or local damping parameters. Moreover, under certain loading conditions (for e.g., strain-controlled and dynamic), either particle kinematics or contact forces obtained under such calibration procedures can be highly inaccurate [21].

Recent CD and DEM approaches include techniques to represent complex particle morphology or shape. In this aspect, recent trends show a clear dichotomy between the choice of shape representation technique. In DEM, the disc- or sphere-clustering technique (see for e.g., [22,23]) appears to have become fairly widespread because of its simplicity. The clustering technique approximates the particle shape by overlapping discs or spheres and the same disc-disc or sphere-sphere contact algorithm is reused over all potentially contacting pairs. In CD, the polyhedra approach has been become quite popular [24]. A key component in the CD formulation is the signed separation or gap, which is used in the determination of constraint forces to prevent particle interpenetration. Algorithms for the determination of signed separation operate under the assumption of the existence of a hyperplane separating two disjoint convex sets [25]. One such algorithm is the common plane approach [26] used in early DEM for convex polyhedra. The algorithm works by finding an initial separating plane that is approximately equidistant from each polyhedron of a potentially contacting pair. The orientation and position of the separating plane are then updated in subsequent time steps using an iterative technique so that the equidistant relationship is maintained [27]. More efficient and robust solution algorithms, however, have since been identified (see for e.g., [28–30]). While the polyhedra approach is considered as a more accurate shape representation technique than the clustering approach, its contact implementation remains complicated due to the need to enumerate all the various combinations of contact entities (node, edge, surface), as well as the complexities associated with the algorithms to estimate the penetration depth. As such, the simpler disc/sphere-clustering is favored over the more accurate polyhedra-based approach.

In the design of CD–GEM, we combine and refine two important developments that allow us to eliminate all the above difficulties:

- We simplify the formulation and implementation of CD significantly by generalizing a variational CD formulation recently developed for discs and spheres [3,31,32]. This particular formulation, which is employed in this paper, is appealing because it provides a way for CD to be easily implemented and solved using off-the-shelf mathematical programming solvers. The most prominent advantage of this formulation is its automatic inclusion of the quasi-static limit, enabling quasi-static modeling without the need for adjusting damping parameters or time step.
- We remove the complexities associated with polyhedra-based contact detection algorithms by adopting Non-Uniform Rational Basis Splines (NURBS) to describe arbitrary particle geometries, as used in GEM [1]. In addition to providing a significant enhancement in the representation of particle morphology (namely, sphericity and angularity [33]), NURBS also facilitates contact calculations through its smooth boundaries. The original contact algorithm as described in [1], however, is based on an explicit calculation of intersection between two arbitrary NURBS curves, which does not provide a signed separation and cannot be used in a CD setting. To overcome this problem, we develop a supporting separating hyperplane (SSH) contact algorithm for particles described using NURBS, which provides the required signed separation or gap required in the CD formulation. The proposed algorithm determines, in a consistent manner and simultaneously, the signed separation and contact normal. To further simplify its implementation, we cast the contact algorithm as a constrained minimization problem, which can then be solved using standard constrained optimization techniques. With the SSH contact algorithm in place, the integration of GEM into the CD formulation is shown to be simple and straightforward.

This paper describes the details on how each of the above items are implemented and is structured as follows. In Section 2, we present the CD formulation for frictionless particles with only translational degrees-of-freedom followed by an extension to frictional particles including rotational degrees-of-freedom in Section 3. The formulation will show that the signed separation function provides the way through which GEM can be incorporated into CD. In Section 4, we describe the implementation of the proposed SSH contact procedure for particles described using NURBS. Finally, in Section 5, we present numerical examples to demonstrate the capabilities of CD–GEM before conclusions are drawn in Section 6. We limit our discussion to two-dimensional particles of strictly convex shapes in which the SSH theorem holds, and the contact algorithm can be concisely described and implemented.

Remark 1. An extended variational version of CD, that includes (as a regularization technique) particle elasticity, is presented in [3]. We focus our attention here on the case of perfectly rigid particles, but we note that the contact algorithm described in this paper can be directly incorporated in the extended case without any change.

Remark 2. There is another method, also called the Granular Element Method, which was proposed by Kishino [34]. The method, however, applies only to disks and spheres, and its solution procedure is different than the procedures described in this paper, and in [1,31].

2. Frictionless rigid particles

For clarity, we first consider the case in which the particles are frictionless and only the translational degrees-of-freedom are present. The extension to the frictional case including rotational degrees-of-freedom will be shown in the following section to be a simple extension of the case considered here.

2.1. Equations of motion

The equations of motion for a single frictionless rigid particle are given by

$$m\dot{\boldsymbol{v}}(t) = \boldsymbol{f}_{\text{ext}} \tag{1}$$

where $\boldsymbol{v}(t) = (v_x(t), v_y(t))^T$ are the linear velocities, m is the mass, and $\boldsymbol{f}_{\text{ext}} = (f_x, f_y)_{\text{ext}}^T$ are external forces.

2.2. Time discretization

Relating position to velocity by $\dot{\boldsymbol{x}}(t) = \boldsymbol{v}(t)$, the equations of motion can be written as

$$m\dot{\boldsymbol{v}}(t) = \boldsymbol{f}_{\text{ext}}$$
$$\dot{\boldsymbol{x}}(t) = \boldsymbol{v}(t) \tag{2}$$

These equations are discretized in time by the θ-method:

$$m\frac{\boldsymbol{v} - \boldsymbol{v}_0}{\Delta t} = \boldsymbol{f}_{\text{ext}}$$
$$\theta\boldsymbol{v} + (1-\theta)\boldsymbol{v}_0 = \frac{\boldsymbol{x} - \boldsymbol{x}_0}{\Delta t} \tag{3}$$

where $0 \leqslant \theta \leqslant 1$ and $\boldsymbol{x}_0$ and $\boldsymbol{v}_0$ are the known position and velocity at time t_0 while $\boldsymbol{x}$ and $\boldsymbol{v}$ are the corresponding quantities at time $t_0 + \Delta t$. We have here assumed that $\boldsymbol{f}_{\text{ext}}$ is constant, stemming, for example, from gravity. Straightforward manipulations lead to the following expressions for the displacements and velocities at time $t_0 + \Delta t$:

$$\bar{m}\Delta\boldsymbol{x} = \bar{\boldsymbol{f}}_0$$
$$\boldsymbol{v} = \frac{1}{\theta}\left[\frac{\Delta\boldsymbol{x}}{\Delta t} - (1-\theta)\boldsymbol{v}_0\right] \tag{4}$$

where

$$\bar{m} = \frac{1}{\theta\Delta t^2}m, \quad \bar{\boldsymbol{f}}_0 = \boldsymbol{f}_{\text{ext}} + \bar{m}\boldsymbol{v}_0\Delta t \tag{5}$$

and $\Delta\boldsymbol{x} = \boldsymbol{x} - \boldsymbol{x}_0$ are the displacements. The stability properties of the θ-method are well known [35]: for $\theta = \frac{1}{2}$ an unconditionally stable and energy preserving scheme is recovered, for $\theta > \frac{1}{2}$ the scheme is unconditionally stable and dissipative, and for $\theta < \frac{1}{2}$ stability depends on the time step. In the context of binary collisions, the algorithmic energy dissipation that occurs for $\theta > \frac{1}{2}$ can be related to the physical dissipation associated with impact and thus to the restitution coefficient. Indeed, as shown in [31], a value of $\theta = \frac{1}{2}$ corresponds to an elastic collision while $\theta = 1$ reproduces a perfectly inelastic collision.

2.3. Non-penetrating constraint

Consider two convex but otherwise arbitrary particles as shown in Fig. 2. The position of the particle centroids at time t_0 are given by $\boldsymbol{x}_0^i$ and $\boldsymbol{x}_0^j$. The signed separation g_0 can be found by finding two parallel supporting separating hyperplanes [25] (as indicated by the two parallel lines in Fig. 2), with support points $\boldsymbol{s}^i$ and $\boldsymbol{s}^j$, such that the distance between the hyperplanes is minimum when the particles do not overlap or maximum when the particles overlap. The notion of the signed separation and supporting hyperplanes will be described in detail in Section 4, but for the purpose of presenting the governing equations, the condition that the particles do not penetrate each other at time $t_0 + \Delta t$ can be stated simply as $g_0 \geqslant 0$. In contrast with the case of discs presented in [31], the signed separation in the case of arbitrary geometries cannot be described analytically.

The non-penetration condition is imposed in a fully explicit manner, i.e., it is assumed that the geometry does not change over one time step. Essentially, this implies the following linearized non-penetration condition:

$$\left(\boldsymbol{n}_0^{ij}\right)^T(\Delta\boldsymbol{x}^i - \Delta\boldsymbol{x}^j) \leqslant g_0 \tag{6}$$

which has the same form as that used in [31]. Here, $\boldsymbol{n}_0^{ij} = (\boldsymbol{s}^j - \boldsymbol{s}^i)/\|\boldsymbol{s}^j - \boldsymbol{s}^i\|$ and $\Delta\boldsymbol{x}_i$ and $\Delta\boldsymbol{x}_j$ are the incremental displacements. The explicit nature of the scheme does occasionally result in particle penetration. As discussed in [31], however, this error is very minor and can be reduced, or eliminated entirely, by adjusting the time step. Moreover, if particle penetration does
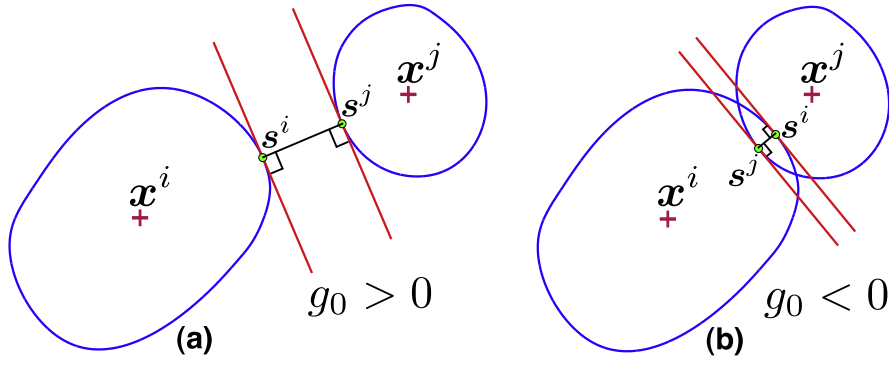
Fig. 2. Supporting separating hyperplanes for two convex particles: (a) positive separation and (b) negative separation.

occur in one time step, it will inevitably be corrected in the following step. In practice, we observe that this self-correction leads to a scheme that remains stable under reasonable time step sizes.

With the above imposed condition, the incorporation of GEM into the CD formulation is achieved by numerically computing the signed separation between two particles described using NURBS curves at the initial time t_0. The associated contact algorithm to determine signed separations for particles defined using NURBS curves is discussed in detail in Section 4.

2.4. Governing equations

For frictionless contact, the governing equations comprise of momentum balance for each particle (incorporating contact forces), the linearized non-penetration condition and the conditions that the contact forces are positive only if the gap is negative and otherwise zero. In the case of two particles, these requirements can be stated by the following discrete governing equations:

$$
\begin{aligned}
&\bar{m}^i \Delta \boldsymbol{x}^i = \bar{\boldsymbol{f}}_0^i - \boldsymbol{n}_0^{ij} p \\
&\bar{m}^j \Delta \boldsymbol{x}^j = \bar{\boldsymbol{f}}_0^j + \boldsymbol{n}_0^{ij} p \\
&\left(\boldsymbol{n}_0^{ij}\right)^T (\Delta \boldsymbol{x}^i - \Delta \boldsymbol{x}^j) \leqslant g_0 \\
&p \geqslant 0 \\
&p \left[\left(\boldsymbol{n}_0^{ij}\right)^T (\Delta \boldsymbol{x}^i - \Delta \boldsymbol{x}^j) - g_0 \right] = 0
\end{aligned}
\tag{7}
$$

where p is the contact normal force. The form of these equations is very similar to the fully implicit stress integration, or closest-point projection, scheme used in computational plasticity [36], and are thus amenable to similar solution procedures.

2.5. Variational formulation

To facilitate the variational formulation of the governing Eq. (7), we introduce the following matrix quantities that cover general n-particle systems:

$$
\begin{aligned}
&\bar{\boldsymbol{M}} = \operatorname{diag}(\bar{m}^1, \bar{m}^1, \ldots, \bar{m}^n, \bar{m}^n) \\
&\boldsymbol{x} = (\boldsymbol{x}^1, \ldots, \boldsymbol{x}^n), \quad \boldsymbol{v} = (\boldsymbol{v}^1, \ldots, \boldsymbol{v}^n) \\
&\boldsymbol{g} = (g^1, \ldots, g^N), \quad \boldsymbol{p} = (p^1, \ldots, p^N)
\end{aligned}
\tag{8}
$$

where n is the number of particles and N is the number of contacts. After collecting the normals associated with potential contacts in a matrix $\boldsymbol{N}$, the governing Eq. (7) can be written as:

$$
\begin{aligned}
&\bar{\boldsymbol{M}} \Delta \boldsymbol{x} = \bar{\boldsymbol{f}}_0 - \boldsymbol{N}_0 \boldsymbol{p} \\
&\boldsymbol{N}_0^T \Delta \boldsymbol{x} \leqslant \boldsymbol{g}_0 \\
&\boldsymbol{p} \geqslant \boldsymbol{0} \\
&\boldsymbol{P} \left(\boldsymbol{N}_0^T \Delta \boldsymbol{x} - \boldsymbol{g}_0 \right) = \boldsymbol{0}
\end{aligned}
\tag{9}
$$

where $\boldsymbol{P} = \operatorname{diag}(\boldsymbol{p})$ and subscripts 0 again refer to the known state. These equations constitute the first-order Karush–Kuhn–Tucker (KKT) optimality conditions associated with the following optimization problem [31,3]:

$$\min_{\Delta \boldsymbol{x}} \max_{\boldsymbol{p}} \quad \left\{ \frac{1}{2} \Delta \boldsymbol{x}^T \bar{\boldsymbol{M}} \Delta \boldsymbol{x} - \Delta \boldsymbol{x}^T \bar{\boldsymbol{f}}_0 \right\} + \left\{ \Delta \boldsymbol{x}^T \boldsymbol{N}_0 \boldsymbol{p} - \boldsymbol{g}_0^T \boldsymbol{p} \right\}$$

$$\text{subject to} \quad \boldsymbol{p} \geqslant \boldsymbol{0} \tag{10}$$

The first term in the objective function is a time discrete form of the action integral associated with a collection of non-interacting particles while the second term accounts for the effects of contact. This principle reproduces the governing Eq. (9) after which the velocities at time $t = t_1 = t_0 + \Delta t$ are calculated from the second equation of (4) and used to set up a new optimization problem to determine the displacements at time t_2, etc.

2.6. Potential contact specification

In this work, the potential contacts at $t_0 + \Delta t$ are determined by an approximate nearest neighbor search procedure [37] on the basis of the positions at t_0. Alternatively, a Delaunay triangulation approach can be used, as in [31]. With either approach, the broad-phase contact detection or search constitutes a minor part of the overall computational effort.

3. Frictional rigid particles

The general case of frictional particles including rotational degrees-of-freedom is considered here following the approach presented in [31]. This case is obtained from the case in the previous section by adding appropriate normal contact forces and accounting for the moments induced by these forces, as well as the moments induced by the contact tangential forces. With these additional quantities, the resulting discrete mixed force–displacement problem then becomes

$$\min_{\Delta \boldsymbol{x}, \Delta \boldsymbol{\alpha}} \max_{\boldsymbol{p}, \boldsymbol{q}} \quad \left\{ \frac{1}{2} \Delta \boldsymbol{x}^T \bar{\boldsymbol{M}} \Delta \boldsymbol{x} - \Delta \boldsymbol{x}^T \bar{\boldsymbol{f}}_0 \right\}$$

$$+ \left\{ \frac{1}{2} \Delta \boldsymbol{\alpha}^T \bar{\boldsymbol{J}} \Delta \boldsymbol{\alpha} - \Delta \boldsymbol{\alpha}^T \bar{\boldsymbol{m}}_0 \right\}$$

$$+ \left\{ \Delta \boldsymbol{x}^T (\boldsymbol{N}_0 \boldsymbol{p} + \widehat{\boldsymbol{N}}_0 \boldsymbol{q}) - \boldsymbol{g}_0^T \boldsymbol{p} - \Delta \boldsymbol{\alpha}^T (\boldsymbol{R}_0^q \boldsymbol{q} + \boldsymbol{R}_0^p \boldsymbol{p}) \right\} \tag{11}$$

$$\text{subject to} \quad \|\boldsymbol{q}\| - \mu \boldsymbol{p} \leqslant \boldsymbol{0}, \quad \boldsymbol{p} \geqslant \boldsymbol{0}$$

where $\boldsymbol{q}$ are the shear forces tangential to $\boldsymbol{p}$ and $\boldsymbol{\alpha}$ are the angles of rotation. With a slight abuse of notation, we have denoted the vector containing the absolute values of the shear forces by $\|\boldsymbol{q}\|$. The matrix $\widehat{\boldsymbol{N}}$ is of the same form as $\boldsymbol{N}$ but contains entries $\boldsymbol{t} = (-n_y, n_x)^T$, i.e., the tangent vector defined as the 90° counterclockwise rotation of $\boldsymbol{n}$. Regarding the additional rotational terms implied by the presence of a shear force, the matrix $\bar{\boldsymbol{J}}$ contains the scaled mass moments of inertia:

$$\bar{\boldsymbol{J}} = \frac{1}{\theta \Delta t^2} \boldsymbol{J}, \quad \bar{\boldsymbol{m}}_0 = \bar{\boldsymbol{J}} \boldsymbol{\omega}_0 \Delta t \tag{12}$$

with $\boldsymbol{J} = \text{diag}(J^1, \ldots, J^n)$ being the mass moments of inertia. The vector $\bar{\boldsymbol{m}}_0$, with $\boldsymbol{\omega}_0$ being the angular velocities, is the effective rotational moment, analogous to the effective translational force vector $\bar{\boldsymbol{f}}_0$. In the above, the rotational terms have been discretized in time analogous to the translational terms and the angular velocities are thus calculated as

$$\boldsymbol{\omega} = \frac{1}{\theta} \left[\frac{\Delta \boldsymbol{\alpha}}{\Delta t} - (1 - \theta) \boldsymbol{\omega}_0 \right] \tag{13}$$

Let C_i be the set of potential contacts associated with particle i and denote by $I \in C_i$ a particular contact point in the set. As shown in Fig. 3, the contact plane (or supporting hyperplane) at the contact point I is described by its normal $\boldsymbol{n}_0$ and tangent $\boldsymbol{t}_0$. The matrix $\boldsymbol{R}_0^q$ contains the contribution of the total angular momentum balance from the tangential forces and contains entries $\boldsymbol{R}_{il}^T \boldsymbol{n}_0$, which is always positive for convex particles. Here, $\boldsymbol{R}_{il}$ is the moment arm extending from the centroid of particle i to the contact point I. The matrix $\boldsymbol{R}_0^p$ contains the contribution of the total angular momentum balance from the normal contact forces and contains entries $-\boldsymbol{R}_{il}^T \boldsymbol{t}_0$, which is a signed quantity depending on whether the contact normal force induces a positive (clockwise) or negative moment on the particle. Finally, the Coulomb criterion is imposed with $\mu = \tan \phi$ being the interparticle friction coefficient and ϕ is the friction angle. For further details on the above formulation (for discs and spheres) we refer the reader to [31].

3.1. Optimality conditions

Following the approach in [31], the first-order KKT conditions associated with Eq. (11) comprise of linear moment balance:

$$\bar{\boldsymbol{M}} \Delta \boldsymbol{x} + \boldsymbol{N}_0 \boldsymbol{p} + \widehat{\boldsymbol{N}}_0 \boldsymbol{q} = \bar{\boldsymbol{f}}_0 \tag{14}$$

balance of angular momentum:

$$\bar{\boldsymbol{J}} \Delta \boldsymbol{\alpha} - \boldsymbol{R}_0^q \boldsymbol{q} - \boldsymbol{R}_0^p \boldsymbol{p} = \bar{\boldsymbol{m}}_0 \tag{15}$$
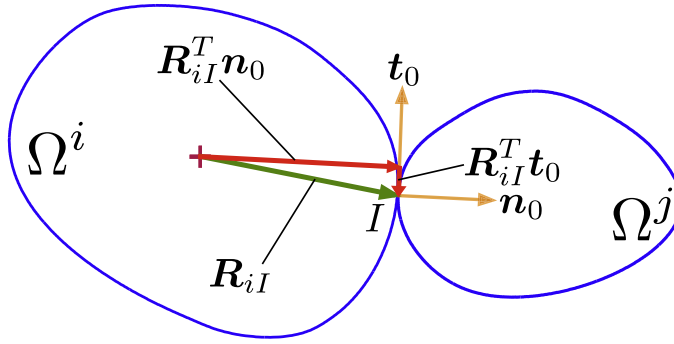
Fig. 3. Illustration of two particles (Ω^i and Ω^j) in contact showing contact normal $\boldsymbol{n}_0$ and tangent $\boldsymbol{t}_0$. The moment arm $\boldsymbol{R}_{il}$ emanates from the centroid of particle i to the contact point I and its projection on the normal and tangent are denoted by $\boldsymbol{R}_{il}^T\boldsymbol{n}_0$ and $\boldsymbol{R}_{il}^T\boldsymbol{t}_0$, respectively. Note that in the case shown here, $\boldsymbol{R}_{il}^T\boldsymbol{t}_0$ is negative.

sliding friction conditions:

$$\|\boldsymbol{q}\| - \mu\boldsymbol{p} + \boldsymbol{s} = \boldsymbol{0}, \quad \boldsymbol{s} \geqslant \boldsymbol{0} \tag{16}$$

$$\mathrm{diag}(\boldsymbol{s})\lambda = \boldsymbol{0}, \quad \lambda \geqslant \boldsymbol{0} \tag{17}$$

where $\boldsymbol{s}$ is the slack vector, introduced to enforce equality, and kinematics:

$$\boldsymbol{N}_0^T\Delta\boldsymbol{x} + \mu\lambda - \boldsymbol{g}_0 = \boldsymbol{0} \tag{18}$$

$$\widehat{\boldsymbol{N}}_0\Delta\boldsymbol{x} - \left(\boldsymbol{R}_0^{qT} + \boldsymbol{R}_0^{pT}\right)\Delta\boldsymbol{\alpha} - \mathrm{sgn}(\boldsymbol{q})\lambda = \boldsymbol{0} \tag{19}$$

where sgn is the signum function. The kinematics in Eqs. (18) and (19) pertain to the associated sliding rule, which leads to an apparent dilation proportional to the friction coefficient μ. However, as described in [31], this dilation can be viewed as an artifact of the time discretization which, with the exception of a few pathological cases, is gradually reduced as the time step is reduced. Moreover, it was shown in [31] that the dilation, even for rather large time steps, is negligible over a range of common conditions including both instances of highly dynamic and relatively unconfined flows as well as confined quasi-static deformation processes.

3.2. Force-based problem

Finally, it is possible to cast Eq. (11) in terms of the following force based problem:

$$
\begin{aligned}
\text{minimize} \quad & \tfrac{1}{2}\boldsymbol{r}^T\bar{\boldsymbol{M}}^{-1}\boldsymbol{r} + \tfrac{1}{2}\boldsymbol{t}^T\bar{\boldsymbol{J}}^{-1}\boldsymbol{t} + \boldsymbol{g}_0^T\boldsymbol{p} \\
\text{subject to} \quad & \boldsymbol{r} + \boldsymbol{N}_0\boldsymbol{p} + \widehat{\boldsymbol{N}}_0\boldsymbol{q} = \bar{\boldsymbol{f}}_0 \\
& \boldsymbol{t} - \boldsymbol{R}_0^q\boldsymbol{q} - \boldsymbol{R}_0^p\boldsymbol{p} = \bar{\boldsymbol{m}}_0 \\
& \|\boldsymbol{q}\| - \mu\boldsymbol{p} \leqslant \boldsymbol{0}, \quad \boldsymbol{p} \geqslant \boldsymbol{0}
\end{aligned}
\tag{20}
$$

where $\boldsymbol{t}$ are the dynamic forces associated with the rotations, i.e., torques.

3.3. Static limit

Omitting the dynamic forces $\boldsymbol{r}$ and $\boldsymbol{t}$ from Eq. (20) gives rise to the following static problem which is valid in the limit of Δt tending to infinity:

$$
\begin{aligned}
\text{minimize} \quad & \boldsymbol{g}_0^T\boldsymbol{p} \\
\text{subject to} \quad & \boldsymbol{N}_0\boldsymbol{p} + \widehat{\boldsymbol{N}}_0\boldsymbol{q} = \bar{\boldsymbol{f}}_{\mathrm{ext}} \\
& \boldsymbol{R}_0^q\boldsymbol{q} + \boldsymbol{R}_0^p\boldsymbol{p} = \boldsymbol{0} \\
& \|\boldsymbol{q}\| - \mu\boldsymbol{p} \leqslant \boldsymbol{0}, \quad \boldsymbol{p} \geqslant \boldsymbol{0}
\end{aligned}
\tag{21}
$$

The above principle is useful for quasi-static problems governed by an internal pseudo-time rather than physical time. Examples include common soil mechanics laboratory tests such as triaxial tests, quasi-static soil–structure interaction problems such as cone penetration, and various applications in the earth sciences where the time scales are such that the deformations are of a quasi-static nature, e.g., [38,39]. We note that in the quasi-static formulation, the accuracy of the

scheme would then depend on the increment size of the applied boundary conditions (for e.g., wall displacements or stresses). In this work, the primal–dual interior-point solver in MOSEK [5] is employed for the solution of either (20) or (21).

Remark 3. The solution and storage costs of the minimization problem (20) or (21) are usually justified by the larger analysis steps that can be taken when using implicit algorithms. This is more so for systems comprised of rigid or highly stiff particles in which explicit solution procedures perform poorly or simply fails.

Remark 4. Large-scale mathematical programming solvers with sparse storage (for e.g., [4,5]) are becoming widely available and increasingly efficient and robust. More recent solvers such as MOSEK [5] also include multi-core or multi-threaded capabilities. The performance of the primal–dual interior-point method is described in detail in [31] and the following properties are summarized: (1) insensitivity of iteration count to problem size and problem type (either linear or quadratic programming), (2) arithmetic complexity that is equivalent to standard Newton–Raphson schemes, and (3) highly robust (almost never fails or stalls). For details on the fundamental theory and implementation of interior-point methods, we refer the reader to [40].

4. Supporting separating hyperplane contact algorithm

In this section, we present a supporting separating hyperplane (SSH) approach to determine the signed separation between two strictly convex NURBS curves. We build upon an algorithm from the field of robotics tracking [41], which finds an SSH in a given fixed direction for convex NURBS objects, and develop a simple procedure to iterate an initial SSH to its required direction meeting certain geometric conditions. We begin by reviewing the essential components of NURBS in the context of CD–GEM.

4.1. Non-Uniform Rational Basis-Splines (NURBS)

The literature on NURBS is extensive and relatively mature, and our purpose here is not to present all of its elements but rather those that are needed for completeness of presentation. Here, we present a brief discussion of NURBS following the presentation format in [1]. For an exhaustive description of NURBS the reader is referred to [42–45], whose presentation and notational convention we follow closely. In this paper, we adhere to the convention in the computational geometry literature where the degree $p = 0, 1, 2, 3$, etc. refers to constant, linear, quadratic, cubic, etc., polynomials, respectively. In the following description of NURBS, we split the acronym into three parts: Non-Uniform (NU), Rational (R), and Basis-Splines (BS), describing these parts in reverse order and incrementally for clarity of presentation.

4.1.1. Basis-Splines (BS)
Referring to Fig. 4, the three main elements of Basis-Splines (B-Splines) are [44]:

1. A set of $n + 1$ control points $\mathbf{P}_i$ ($0 \leqslant i \leqslant n$).
2. A knot vector U of $m + 1$ knots ($0 = u_0 \leqslant u_1 \leqslant \cdots \leqslant u_{m-1} \leqslant u_m = 1$), and
3. A degree p satisfying $m = n + o$ where $o = p + 1$ is the order of the curve.

The last requirement means that the number of knots $m + 1$ must be equal to the number of control points $n + 1$ plus the curve order $p + 1$. The control points need not lie on the curve itself. Rather, control points act as a scaffold that controls the geometry [45]. Piecewise linear interpolation of the control points effectively furnishes a control polygon bounding the geometry.

Given the aforementioned paramters, a (non-rational, polynomial) B-Spline curve is defined parametrically such that

$$\mathbf{C}(u) = \sum_{i=0}^{n} N_{i,p}(u) \mathbf{P}_i \tag{22}$$

where u is the curve parameter, and $N_{i,p}$ is the i-th B-Spline basis function of degree p defined recursively as follows:

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u \in [u_i, u_{i+1}) \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \tag{23}$$

The above is also known as the Cox-de Boor recursion formula [46,47].

4.1.2. Rational B-Splines (RBS)
A known limitation of (non-rational) B-Splines, as defined in Eq. (22), is their inability to capture conic sections (e.g., circles and ellipses). This limitation stems from the simple polynomial form of B-Splines. To be able to represent conic
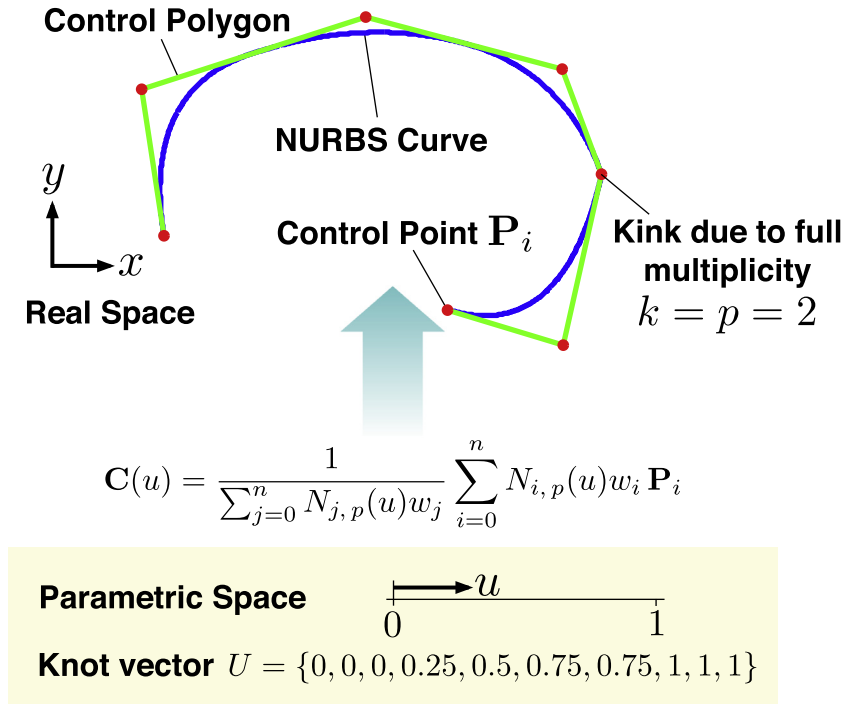
Control Polygon

NURBS Curve

y

x

Real Space

Control Point $\mathbf{P}_i$

Kink due to full multiplicity
$k = p = 2$

$$\mathbf{C}(u) = \frac{1}{\sum_{j=0}^{n} N_{j,p}(u) w_j} \sum_{i=0}^{n} N_{i,p}(u) w_i \, \mathbf{P}_i$$

Parametric Space u

0 1

Knot vector $U = \{0, 0, 0, 0.25, 0.5, 0.75, 0.75, 1, 1, 1\}$

Fig. 4. Schematic illustration of a NURBS curve. The curve degree p is 2 (quadratic). The weights w_i are 1 in the case of (non-rational, polynomial) B-Spline. The full multiplicity $k = p = 2$ refers to the knot/parameter value $u = 0.75$ being repeated $p = 2$ times.

sections, the parametric form would need to be *rational*, i.e., the quotient of two polynomials. A rational B-Spline (RBS) is furnished by adding a weight $w_i \geq 0$, which provides an additional degree of freedom for geometry manipulation. Hence, the curve equation becomes

$$\mathbf{C}(u) = \frac{1}{\sum_{j=0}^{n} N_{j,p}(u) w_j} \sum_{i=0}^{n} N_{i,p}(u) w_i \mathbf{P}_i \tag{24}$$

$$= \sum_{i=0}^{n} R_{i,p}(u) \mathbf{P}_i \tag{25}$$

where $R_{i,p}(u) = N_{i,p}(u) w_i / \left(\sum_{j=0}^{n} N_{j,p}(u) w_j \right), 0 \leq i \leq n$, are the rational basis functions. Since $R_{i,p}(u)$ is rational, the exact description of conic sections becomes possible. When all weights are equal to unity, Eq. (25) reduces to Eq. (22).

The weight w_k affects the effective contribution of control point $\mathbf{P}_k$ on the overall shape of the curve $\mathbf{C}(u)$. Making w_k smaller corresponds to 'pushing' the curve away from the control point $\mathbf{P}_k$. In the extreme, when $w_k = 0$, the term $w_k \mathbf{P}_k$ is annihilated from the equation of the curve and the contribution of the control point is obviously nullified. The other extreme is obtained by making w_k very large relative to other weights. Dividing Eq. (25) by w_k gives

$$\mathbf{C}(u) = \frac{\sum_{i \neq k}^{n} N_{i,p}(u) w_i / w_k \mathbf{P}_i + N_{k,p}(u) \mathbf{P}_k}{\sum_{i \neq k}^{n} N_{i,p}(u) w_i / w_k + N_{k,p}(u)} \tag{26}$$

where one can see that as w_k is increased, the curve $\mathbf{C}(u)$ is 'pulled' towards the control point $\mathbf{P}_k$.

4.1.3. Non-Uniform (NU) Rational B-Splines (NURBS)

The NU portion in NURBS is furnished by the knots in the knot vector U of the B-Splines. The non-decreasing knots $u_i, i = 0, 1, \ldots, m$ partitions the parameter space into segments of half-open intervals $[u_i, u_{i+1})$, which are also called knot spans. The knot span can be of zero length since the knots need not be distinct, i.e., they can be repeated. The number of times a knot value repeats itself is called multiplicity k. We use the non-periodic (or clamped) B-Splines, in which the knot vector takes the form

$$U = \{\underbrace{0, 0, \ldots, 0}_{p+1}, u_{p+1}, \ldots, u_{m-p-1}, \underbrace{\alpha, \alpha, \ldots, \alpha}_{p+1}\} \tag{27}$$

If the internal knots are uniformly spaced, the curve is infinitely continuously differentiable in the interior of a knot span, and $(p - k)$-times continuously differentiable at a knot. If $k = p$, we say that the knot has full multiplicity; the multiplicity cannot be greater than the degree. Multiplicity of knots provides a way to specify the continuity between segments. For example, a full multiplicity knot in the knot vector (away from the ends) means that a kink or cusp is present in the curve. In general, the knots can be unequally spaced, i.e., the knot vector is non-uniform, and hence, we get non-uniform B-Splines (the NU part in NURBS).

Finally, we close the curve using the following approach: the first and last control points are made coincident, i.e., $\boldsymbol{P}_0 = \boldsymbol{P}_n$ and the two coincident and adjacent control points are made collinear to maintain curvature continuity.

4.2. Supporting separating hyperplane for convex particles

Let Ω be a convex particle and $\boldsymbol{s}$ be a point on its boundary. If a given direction $\boldsymbol{\eta}$ is such that

$$\boldsymbol{\eta}^T \boldsymbol{x} < \boldsymbol{\eta}^T \boldsymbol{s} \quad \forall \boldsymbol{x} \in \Omega \tag{28}$$

then the hyperplane $\{\boldsymbol{x} | \boldsymbol{\eta}^T \boldsymbol{x} = \boldsymbol{\eta}^T \boldsymbol{s}\}$ is called the supporting hyperplane to Ω at the support point $\boldsymbol{s}$ [25]. The normal of the hyperplane is $\boldsymbol{\eta}$. The problem of finding the support point in a given direction $\boldsymbol{\eta}$ can therefore be written as a problem of finding the maximal value of the support function of the particle Ω, h [41]:

$$h(\boldsymbol{\eta}) = \underset{\boldsymbol{x} \in \Omega}{\text{maximize}} \ \boldsymbol{x}^T \boldsymbol{\eta} \tag{29}$$

Consider now the case with convex particles i and j as shown in Fig. 2. If the particles are disjoint, i.e., $\Omega^i \cap \Omega^j = \emptyset$, we can define two separating hyperplanes, one on each particles, such that their hyperplane normals and support points satisfy the following geometric conditions [41]:

$$h_i(\boldsymbol{\eta}) = (\boldsymbol{s}^i)^T \boldsymbol{\eta} \quad \text{and} \quad h_j(-\boldsymbol{\eta}) = (\boldsymbol{s}^j)^T (-\boldsymbol{\eta}) \quad \Omega^i \cap \Omega^j = \emptyset \tag{30}$$

with $\boldsymbol{s}^i$ and $\boldsymbol{s}^j$ being the corresponding support points on particles i and j, respectively, and $\boldsymbol{\eta} = \boldsymbol{s}^j - \boldsymbol{s}^i$. Since the particles are disjoint, they do not overlap and the separation g_0 is positive. This condition is illustrated in Fig. 2(a). For the purpose of determining the signed separation, we also need the case when the particles overlap in which case, the above geometric conditions become

$$h_i(-\boldsymbol{\eta}) = (\boldsymbol{s}^i)^T (-\boldsymbol{\eta}) \quad \text{and} \quad h_j(\boldsymbol{\eta}) = (\boldsymbol{s}^j)^T \boldsymbol{\eta} \quad \Omega^i \cap \Omega^j \neq \emptyset \tag{31}$$

as illustrated in Fig. 2(b). Since the particles now overlap, the separation g_0 is negative.

4.3. Implementation

The basic routines required for the implementation of the proposed contact algorithm are the closest point projection (CPP) and determination of the SSH given a trial direction, as described in Algorithms 1 and 2. The minimization problems in Algorithms 1 and 2 can be solved using either gradient-based or derivative-free constrained optimization procedures. In this paper, we use the derivative-free procedure since it is simpler and does not require the evaluation of the derivatives of the objective function. Using these two components, Algorithm 3 describes a procedure to iterate an initial guess direction of the hyperplane normal until it converges to the required geometric conditions described in Eqs. (30) and (31). The curve derivative $\boldsymbol{C}'(u)$ used in Algorithm 3 is simply another NURBS curve and the associated algorithms for its computation are described in full detail in [42]. For simplicity, we generate the initial guess direction using the centroids of potentially contacting particles.

As illustrated in Fig. 5 the CPP operation on the midpoint generates two closest projected points, one on each curve, and one might be tempted to use these points to generate the trial direction $\boldsymbol{\eta}$. Degeneracy of the trial direction, however, occurs when these two points become nearly or exactly coincident. The use of the normal at a closest projected point removes this degeneracy problem, as well as eliminating robustness issues under near-tangential contact conditions, i.e., contact at a single point, associated with using intersection algorithms based on recursive subdivision [48]. Depending on the relative curvatures at the two closest projected points, the algorithm can generate an SSH that lies outside the overlapping region as shown in Fig. 6(a), which leads to nonconvergence of Algorithm 3. This case can be detected after one iteration by checking the curvatures at the closest projected points and corrected by restarting Algorithm 3 with the master particle being the one having the lower curvature. The result of this correction is shown in Fig. 6(b). We note that the proposed contact algorithm recovers the result described in [49] in the case of tangential or grazing intersection.

Although a more rigorous analysis is possible, we provide here a heuristic argument that the algorithm converges: at each step, the midpoint and CPP operations rotate both hyperplanes closer to the required geometric conditions stated in Eqs. (30) and (31). Considering the midpoint as a trial state, the algorithm is reminiscent of a CPP stress-return operation used for constitutive model updates in computational plasticity [36].
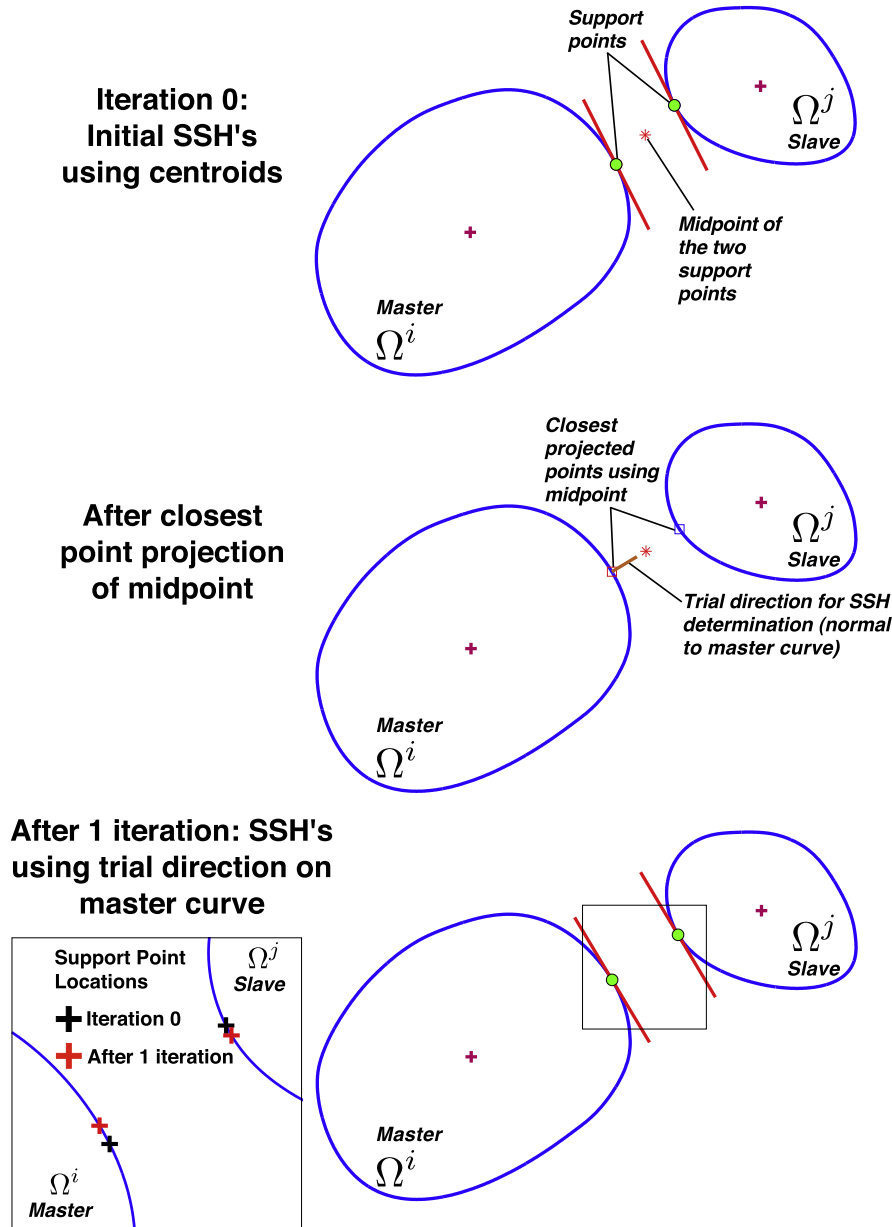
Fig. 5. Determination of separating supporting hyperplanes in one iteration of Algorithm 3. Insert on the left bottom corner is a closeup showing the locations of support points at iterations 0 and 1.

Remark 5. The proposed contact algorithm can also be used for GEM in the context of explicit time discretization [1]. Unlike the original, the current algorithm does not require a recursive subdivision process and handles near-tangential or grazing intersections robustly.

Remark 6. Polygonal bodies can be generated using NURBS curves having degree $p = 1$ and full multiplicity $k = p$ at each internal knot. In this case, existing polyhedra-based contact algorithms are fully applicable. In this paper, we focus on the case of smooth convex bodies described using NURBS, with $p > 1$.

Remark 7. The extension of the SSH contact algorithm to three-dimensions would follow similar lines to the approach outlined in this section but using surface gradients and curvatures. It is well known, however, that the merging of multiple NURBS patches is problematic and cumbersome [50]. In this regard, a superset of NURBS, called T-Splines [51], may offer an advantage over NURBS surfaces by treating issues related to presence of extraordinary points and surface continuity requirements across multiple patches. Further research is required to assess the potential of using T-Splines for granular contact algorithms.

Algorithm 1. Closest point projection (CPP) algorithm

Algorithm CPP($\boldsymbol{C}, \boldsymbol{x}$)
Input: Curve $\boldsymbol{C}$ and point $\boldsymbol{x}$
Output: Closest projected point on curve $\boldsymbol{c}$

1: Initialize $\hat{f}$ to some large value
2: **for** $i = 0, \ldots, m - p - 1$ **do** {loop over internal knot spans}
3:    Solve for u^* : $\underset{u}{\text{minimize}} \, f(u)$ subject to $u_i \leqslant u \leqslant u_{i+1}$ where $f(u) = \|\boldsymbol{C}(u) - \boldsymbol{x}\|$
4:    **if** $f(u^*) < \hat{f}$ **then**
5:      $\hat{f} \leftarrow f(u^*)$
6:      $\boldsymbol{c} \leftarrow \boldsymbol{C}(u^*)$
7:    **end if**
8: **end for** {end internal knot span loop}

Algorithm 2. Supporting separating hyperplane (SSH) algorithm

---

Algorithm SSH($\boldsymbol{C}, \boldsymbol{\eta}$)
Input: Curve $\boldsymbol{C}$ and normal direction of hyperplane $\boldsymbol{\eta}$
Output: Supporting point of hyperplane $\boldsymbol{s}$

1: Initialize $\hat{f}$ to some large value
2: **for** $i = 0, \ldots, m - p - 1$ **do** {loop over internal knot spans}
3:    Solve for u^* : $\underset{u}{\text{minimize}} \, f(u)$ subject to $u_i \leqslant u \leqslant u_{i+1}$ where $f(u) = \boldsymbol{C}'(u)^T \boldsymbol{\eta}$
4:    **if** $|f(u^*)| < \text{TOLERANCE}$ and $(\boldsymbol{\eta} \times \boldsymbol{C}'(u))_3 > 0$ **then**
5:      $\boldsymbol{s} \leftarrow \boldsymbol{C}(u^*)$
6:      **break**
7:    **end if**
8: **end for** {end internal knot span loop}

Algorithm 3. Contact algorithm to find signed separation

1: Run broad-phase contact detection
2: **for all** Potentially contacting pairs of NURBS curves **do**
3:    Let one of the curves be the master $\boldsymbol{C}^i$ and the other the slave $\boldsymbol{C}^j$, with centroids $\boldsymbol{x}^i$ and $\boldsymbol{x}^j$, respectively
4:    Compute initial guess of SSH normal direction as $\boldsymbol{\eta} = \boldsymbol{x}^j - \boldsymbol{x}^i$
5:    **for** $k = 1, \ldots, N_{\text{MAX}}$ **do** {iteration loop}
6:      Find the support point for each curve using Algorithm 2:
      $\boldsymbol{s}_k^i = \text{SSH}(\boldsymbol{C}^i, \boldsymbol{\eta})$
      $\boldsymbol{s}_k^j = \text{SSH}(\boldsymbol{C}^j, -\boldsymbol{\eta})$
7:      Compute midpoint of the support points:
      $\boldsymbol{m}_k = \frac{1}{2}\left(\boldsymbol{s}_k^i + \boldsymbol{s}_k^j\right)$
8:      Compute closest projection point of $\boldsymbol{m}_k$ onto master curve using Algorithm 1: $\boldsymbol{c}_k^i = \text{CPP}(\boldsymbol{C}^i, \boldsymbol{m}_k)$
9:      Store previous direction: $\boldsymbol{\eta}_{\text{prev}} \leftarrow \boldsymbol{\eta}$
10:     Set SSH direction of next iteration to curve normal at $\boldsymbol{c}_k^i$: $\boldsymbol{\eta} \leftarrow \boldsymbol{n}(\boldsymbol{c}_k^i)$
11:     **if** $\|\boldsymbol{\eta} - \boldsymbol{\eta}_{\text{prev}}\|_\infty < \text{TOLERANCE}$ **then**
12:       Compute signed separation: $g = \text{sgn}(\boldsymbol{d}^T \boldsymbol{\eta})\|\boldsymbol{d}\|$ where $\boldsymbol{d} = \boldsymbol{s}_k^j - \boldsymbol{s}_k^i$
13:       **break**
14:     **end if**
15:    **end for** {end iteration loop}
16: **end for** {end contacting pairs loop}
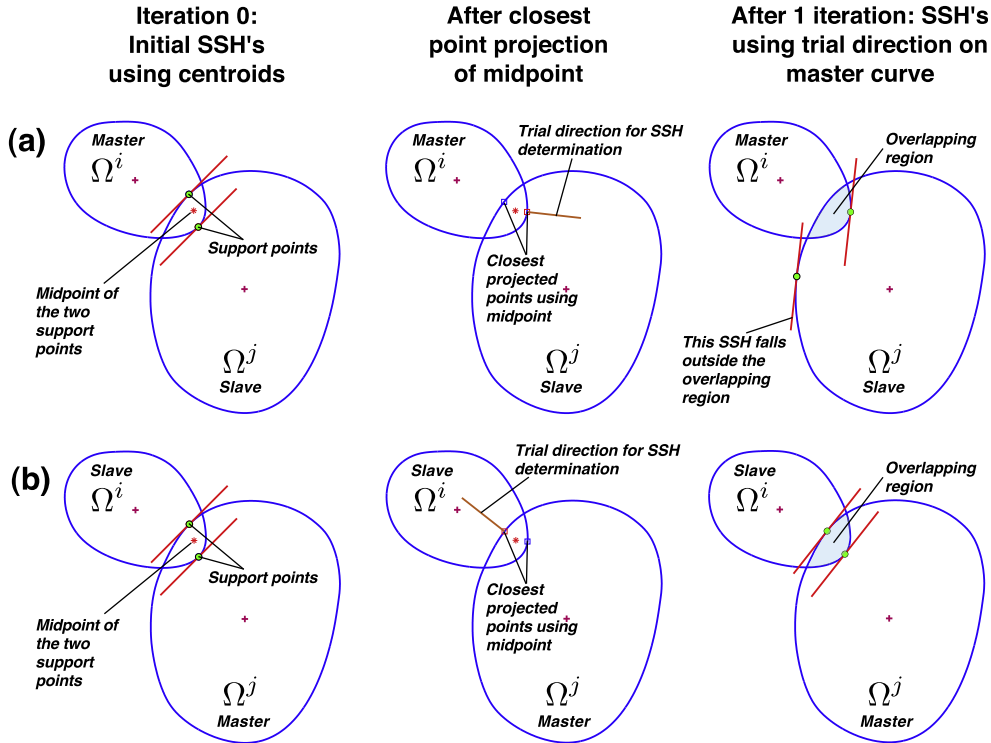
Fig. 6. Separating supporting hyperplanes after one iteration: (a) using Ω^i as master and Ω^j as slave and (b) using Ω^i as slave and Ω^j as master.

4.4. Computational complexity

The computational complexity of the proposed contact algorithm is measurably higher than contact approaches in classical disc or sphere-based DEM in which the contact overlap can be evaluated analytically and without any iterations. For discrete methods with particle morphology or shape representation techniques, iterative procedures are required to isolate potential contact points. As such, a more appropriate comparison would be to compare the proposed contact algorithm with polyhedra-based approaches.

By considering the knot spans as segments, our proposed contact algorithm has essentially the same computational complexity as polyhedra-based approaches. We note that for a given prescribed resolution, NURBS has the advantage of carrying significantly less geometric information compared to polyhedra, i.e., less number of NURBS control points compared to number of polyhedra nodes [1]. To further enhance the efficiency of the proposed contact algorithm, a bounding-volume tree or hierarchy identical to that used in contact detection algorithms for geometries using polyhedra [52] can be applied to the control polygon of each particle. The tree allows one to first isolate knot spans that are near the intersection points of the control polygons of potentially contacting particles. These spans are then used in the span search range of Algorithms 1 and 2.

Aside from the foregoing, two additional optimizations can be made. First, the bounding-volume tree can be constructed, at the expense of memory, statically, i.e., does not change throughout the simulation, as a one-time pre-processing step prior to starting a simulation. With this approach, contact detection is performed by traversing the static tree but performing the intersection calculations using bounding boxes that are mapped from the initial configuration to the current configuration using the total translation vector and rotation matrix computed from kinematics update. Second, instead of performing a full-depth traversal of the bounding-volume tree, an incremental approach similar to that of [53] may be taken.

5. Numerical examples

In the following, we present two examples demonstrating the capabilities of CD–GEM. In both examples, the assemblies are comprised with rigid grains and we highlight the effect of particle angularity or lack of sphericity by comparing the macroscopic responses of disc and GEM (angular) assemblies.

5.1. Column drop test

We consider a column with an initial height-radius ratio of h_0/r_0 of approximately 1.51. Two columns, one using GEM particles and another using discs, are prepared by dropping particles into a box and letting them settle under gravity. The

GEM column is constructed using particles from fifteen different shape categories, which are the same as those presented in [1]. The shapes are described by NURBS curves of cubic ($p = 3$) degree and have between 19 and 32 control points. For the disc column, the particle sizes have a max–min diameter ratio of $d_{max}/d_{min} = 0.927/0.5 = 1.854$. The number of particles are 1440 and 1400 for the GEM and disc assemblies, respectively.

The friction coefficient of the base supporting the column is $\mu_{base} = 0.5$ while a smooth vertical wall, representing a symmetry boundary, is placed on one side of the column. The drop test is conducted by removing one of the side walls of the box and letting the column spread under gravity. The simulations are carried out using $\theta = 0.7$ and a time step of $\Delta t = 0.02$ for a total of 650 steps and 1300 steps for interparticle friction coefficients $\mu = 0.5$ and $\mu = 0$, respectively.

The initial and final configurations for the case of $\mu = 0.5$ is shown in Fig. 7. The slopes of the final spreads of the GEM and disc assemblies are approximately 12° and 9° corresponding to dimensions (height by width) of 35.4 by 162 and 31 by 194, respectively. The spread for the disc assembly is approximately 20% wider than that of the GEM assembly, which is consistent with the larger rolling resistance provided by angular particles in the GEM assembly.

The final configurations for the case of $\mu = 0$ is shown in Fig. 8. Without rolling resistance, the disc assembly simply 'melts' away to a final layer thickness of about two particles. On the other hand, the GEM assembly maintains a well-defined spread, even at zero interparticle friction, due to the rolling resistance provided by angular particles.

5.2. Biaxial test

A two-dimensional biaxial test is carried out for a rectangular assembly with 960 GEM particles as shown in Fig. 9. This test is performed under the static limit described in Section 3.3. Also shown in the figure is a disc assembly for response comparison. The particle shapes and sizes are the same as those described in the column drop test. Both assemblies have an initial porosity of approximately 0.175. Three different interparticle friction coefficients $\mu = 0.364$, 0.577, and 0.839 corresponding to friction angles 20°, 30°, and 40° are used to gauge the effect of interparticle friction. The upper wall is moved downwards while the pressure on the right wall is maintained at 125 units of pressure. A total of 300 steps are used to impose a total axial strain of approximately 0.15. The left and bottom walls are stationary and all walls are frictionless.

The deformed configurations and evolution of shear stress against axial strain are shown in Figs. 9 and 10, respectively. As expected, the peak strength for each assembly at low levels of strain becomes more pronounced as the interparticle friction coefficient is increased, while the final residual, or critical state, strength is relatively insensitive to that parameter. A
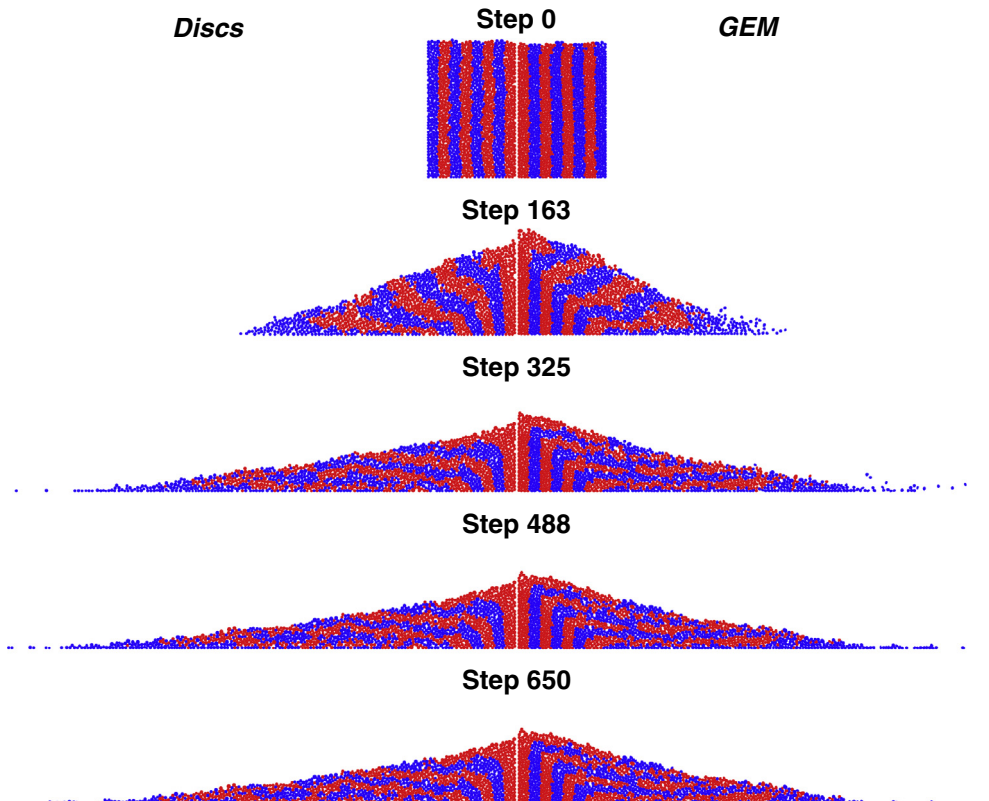

Fig. 7. Collapse of granular column with interparticle friction coefficient $\mu = 0.5$. The base is rough with $\mu_{base} = 0.5$.
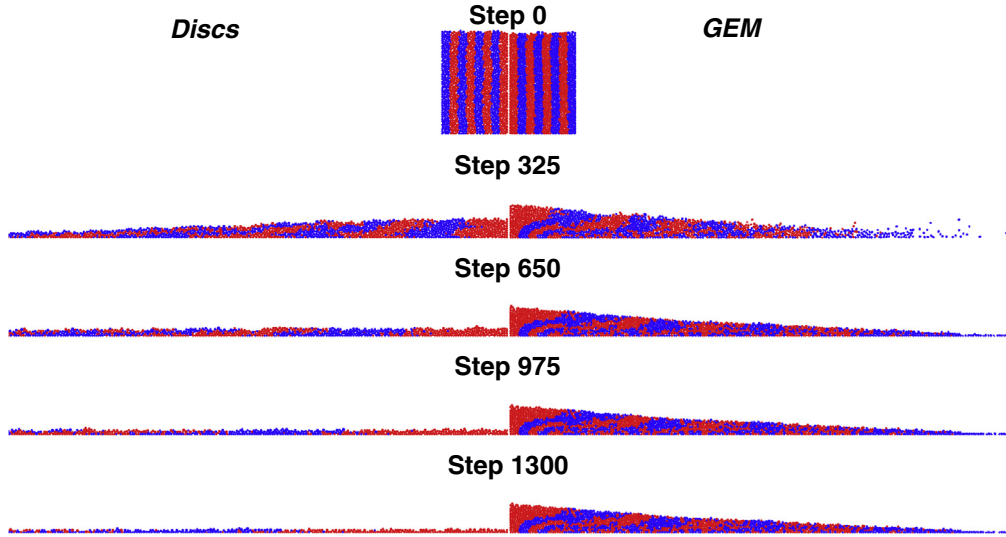
Fig. 8. Collapse of granular column with interparticle friction coefficient $\mu = 0$. The base is rough with $\mu_{\text{base}} = 0.5$.
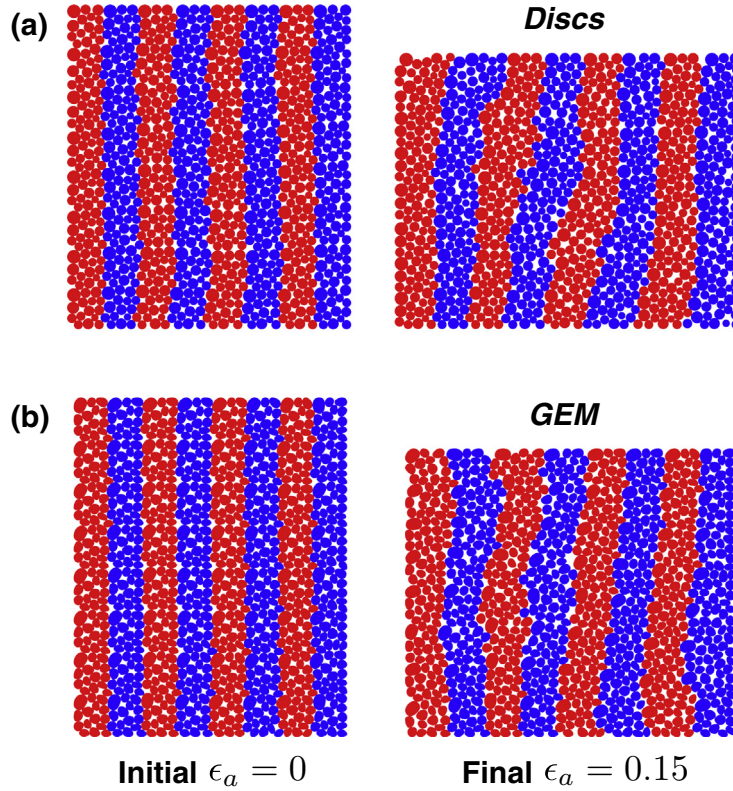

Fig. 9. Initial and deformed configurations for the biaxial test ($\mu = 0.577$ case): (a) disc assembly and (b) GEM assembly.

comparison between the two assemblies, however, show that the both the peak strength and dilatancy (as measured by the initial slope in the volumetric strain plot, i.e., volumetric change near the peak strength) in the GEM assembly are higher than those in the disc assembly. Also, very importantly, we see that the critical strength of the material seems to increase with particle angularity, not interparticle friction. This is very important to understand the micro-mechanical basis of critical state theory, which is still elusive. These observations are consistent with the increase in angularity of the particles.
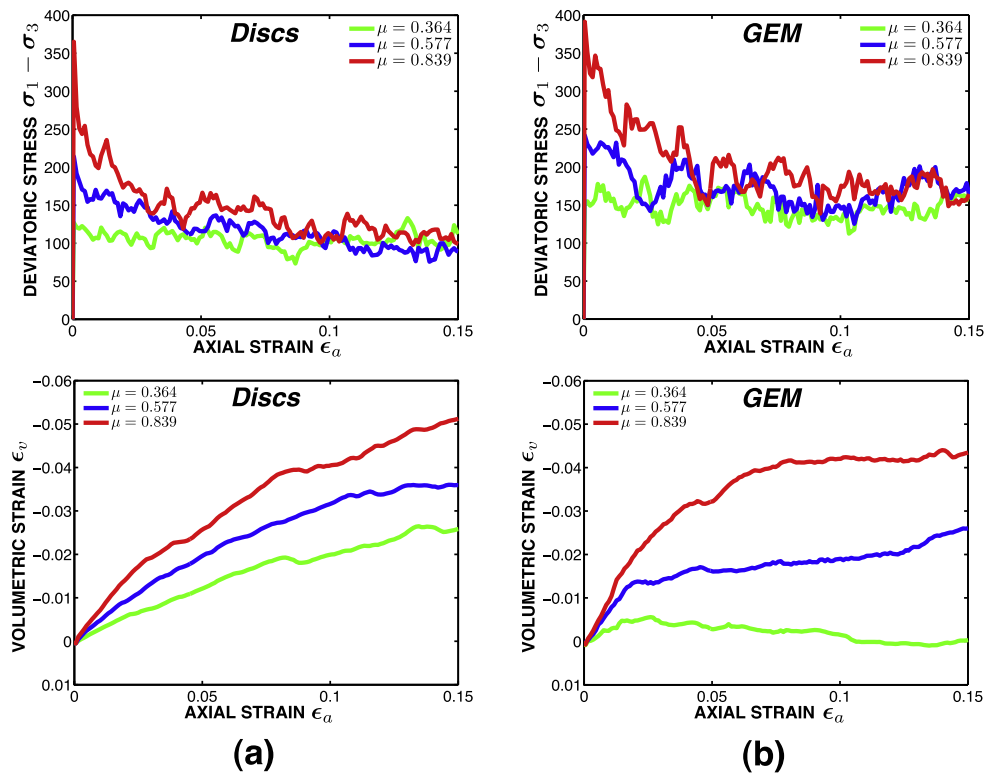
Fig. 10. Shear stress against axial strain in biaxial test: (a) disc assembly and (b) GEM assembly.

6. Conclusion and Future Directions

We have presented a contact dynamics (CD) approach to the Granular Element Method (GEM), which we abbreviated here as CD–GEM. The implementation CD–GEM is made simple by adopting a variational framework of contact dynamics and the resulting discrete model can be readily solved using off-the-shelf mathematical programming solvers. Non-Uniform Rational Basis Splines (NURBS) provides an accurate particle morphology representation and facilitates contact calculations through its smooth boundaries. With these two ingredients in place, a supporting separating hyperplane (SSH) contact algorithm for NURBS is devised to complete the integration of CD and GEM.

By combining particle shape flexibility, properties of implicit time-integration discretization (e.g., larger time steps) and non-penetrating constraints, as well as a reduction to a static formulation in the limit of an infinite time step, CD–GEM targets system properties and deformation regimes in which the classical discrete element method either performs poorly or simply fails; namely, in granular systems comprising of rigid or highly stiff angular particles and subjected to quasi-static or intense dynamic flow conditions. Through numerical simulations of the column drop and biaxial tests, we demonstrated and highlighted the ability of CD–GEM to capture the effect of increased rolling resistance associated with angular particles in the macroscopic response. These are manifested macroscopically by a smaller spread under a column drop test and as an increase in the mobilized strength and dilatancy under shear.

We note in closing that the combination of a variational update structure and SSH contact algorithm provides considerable latitude in the development of contact algorithms for non-smooth frictional bodies [54] and collision integrators within the framework of constrained variational integrators [55,56]. These areas are to be explored in the context of granular mechanics simulations in the future.

References

[1] J.E. Andrade, K.-W. Lim, C.F. Avila, I. Vlahinich, Granular element method for computational particle mechanics, Computer Methods in Applied Mechanics and Engineering 241–244 (2012) 262–274.

[2] P.A. Cundall, O.D.L. Strack, A discrete numerical model for granular assemblies, Géotechnique 29 (1979) 47–65.

[3] K. Krabbenhoft, A.V. Lyamin, J. Huang, M. Vicente da Silva, Granular contact dynamics with particle elasticity, Granular Matter 14 (2012) 607–619.

[4] J.F. Sturm, SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones, Optimization Methods and Software 11–12 (1999) 625–653.

[5] MOSEK ApS. MOSEK. http://www.mosek.com/.

[6] J.J. Moreau, Bounded variation in time, in: J.J. Moreau, P.D. Panagiotopoulos, G. Strang (Eds.), Topics in Nonsmooth Mechanics, vol. 1, Birkhauser Verlag, Basel-Boston-Stuttgart, 1987, pp. 1–74.

[7] J.J. Moreau, Unilateral contact and dry friction in finite freedom dynamics, in: J.J. Moreau, P. Panagiotopoulos (Eds.), Nonsmooth Mechanics and Applications, CISM Courses and Lectures, vol. 302, Springer, Berlin, 1988, pp. 1–82.

[8] J.J. Moreau, Some numerical methods in multibody dynamics: application to granular materials, European Journal of Mechanics – A/Solids 13 (1994) 93–114.

[9] M. Jean, The non-smooth contact dynamics method, Computer Methods in Applied mechanics and Engineering 177 (3–4) (1999) 235–257.

[10] C. Nouguier-Lehon, B. Cambou, E. Vincens, Influence of particle shape and angularity on the behaviour of granular materials: a numerical analysis, International Journal for Numerical and Analytical Methods in Geomechanics 27 (14) (2003) 1207–1226.

[11] S. McNamara, H. Herrmann, Measurement of indeterminacy in packings of perfectly rigid disks, Physical Review E 70 (2004) 061303.

[12] M. Renouf, F. Dubois, P. Alart, A parallel version of the non smooth contact dynamics algorithm applied to the simulation of granular media, Journal of Computational and Applied Mathematics 168 (12) (2004) 375–382.

[13] L. Staron, E.J. Hinch, Study of the collapse of granular columns using two-dimensional discrete-grain simulation, Journal of Fluid Mechanics 545 (2005) 1–27.

[14] A. Taboada, K.-J. Chang, F. Radjai, F. Bouchette, Rheology, force transmission, and shear instabilities in frictional granular media from biaxial numerical tests using the contact dynamics method, Journal of Geophysical Research: Solid, Earth 110 (B9) (2005).

[15] G. Saussine, C. Cholet, P.E. Gautier, F. Dubois, C. Bohatier, J.J. Moreau, Modelling ballast behaviour under dynamic loading. Part 1: A 2d polygonal discrete element method approach, Computer Methods in Applied Mechanics and Engineering 195 (2006) 2841–2859.

[16] A. Ries, D.E. Wolf, T. Unger, Shear zones in granular media: three-dimensional contact dynamics simulation, Physical Review E 76 (2007) 051301.

[17] L. Staron, E.J. Hinch, The spreading of a granular mass: role of grain properties and initial conditions, Granular Matter 9 (2007) 205–217.

[18] R. Farhang, R. Vincent, Contact dynamics as a nonsmooth discrete element method, Mechanics of Materials 41 (6) (2009) 715–728.

[19] N. Estrada, E. Azéma, F. Radjai, A. Taboada, Identification of rolling resistance as a shape parameter in sheared granular media, Physical Review E 84 (2011) 011306.

[20] P.-Y. Lagree, L. Staron, S. Popinet, The granular column collapse as a continuum: validity of a two-dimensional Navier-Stokes model with a μ(I)-rheology, Journal of Fluid Mechanics 686 (2011) 378–408.

[21] X. Tu, J.E. Andrade, Criteria for static equilibrium in particulate mechanics computations, International Journal for Numerical Methods in Engineering 75 (2008) 1581–1606.

[22] A.K. Ashmawy, B. Sukumaran, A.V. Hoang, Evaluating the influence of particle shape on liquefaction behavior using discrete element method, in: Proceedings of the Thirteenth International Offshore and Polar Engineering Conference (ISOPE 2003) Honolulu, Hawaii, 2003.

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

[24] F. Dubois, M. Jean, The non smooth contact dynamic method: recent lmgc90 software developments and application, in: Peter Wriggers, Udo Nackenhorst (Eds.), Analysis and Simulation of Contact Problems, volume 27 of Lecture Notes in Applied and Computational Mechanics, Springer, Berlin Heidelberg, 2006, pp. 375–378.

[25] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, New York, NY, USA, 2004.

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

[27] D. Baraff, Curved surfaces and coherence for non-penetrating rigid body simulation, SIGGRAPH Computer Graphics 24 (4) (1990) 19–28.

[28] E.G. Gilbert, D.W. Johnson, S.S.A. Keerthi, A fast procedure for computing the distance between complex objects in three- dimensional space, IEEE Journal of Robotics and Automation 4 (2) (1988) 193–203.

[29] K. Chung, W. Wang, Quick collision detection of polytopes in virtual environments, in: Proceedings of Symposium on Virtual Reality Software and Technology, 1996, pp. 125–132.

[30] B. Mirtich, V-clip: fast and robust polyhedral collision detection, ACM Transactions on Graphics 17 (3) (1998) 177–208.

[31] K. Krabbenhoft, A.V. Lyamin, J. Huang, M. Vicente da Silva, Granular contact dynamics using mathematical programming methods, Computers and Geotechnics 241–244 (2012) 262–274.

[32] J. Huang, M. Vicente da Silva, K. Krabbenhoft, Three-dimensional granular contact dynamics with rolling resistance, Computers and Geotechnics 49 (2013) 289–298.

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

[34] Y. Kishino, Computer analysis of dissipation mechanism in granular media, in: Powders and Grains, A.A. Balkema, Rotterdam, 1989, pp. 323–330.

[35] W.L. Wood, Practical Time-Stepping Schemes, Oxford University Press, Oxford, 1990.

[36] J.C. Simo, T.J.R. Hughes, Computational Inelasticity, Springer-Verlag, New-York, 1998.

[37] D.M. Mount, S. Arya, ANN: a library for approximate nearest neighbor searching. http://www.cs.umd.edu/mount/ANN/.

[38] P. Souloumiac, Y.M. Leroy, B. Maillot, K. Krabbenhoft, Predicting stress distributions in fold-thrust and accretionary wedges by optimisation, Journal of Geophysical Research 114 (2009) B09404.

[39] P. Souloumiac, K. Krabbenhoft, Y.M. Leroy, B. Maillot, Failure in accretionary wedges with the maximum strength theorem, Computers & Geosciences 19 (2010) 793–811.

[40] S.J. Wright, Primal-dual interior-point methods, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1997.

[41] Colin Turnbull, Stephen Cameron, Computing distances between nurbs-defined convex objects, in: IEEE International Conference on Robotics and Automation, 1998, pp. 3685–3690.

[42] L. Piegl, W. Tiller, The NURBS Book, second ed., Springer-Verlag New York Inc., New York, NY, USA, 1997.

[43] C.-K. Shene, CS3621 Introduction to computing with geometry, 2011, http://www.cs.mtu.edu/shene/COURSES/cs3621/NOTES/.

[44] J. Lowther, J. Fisher, C.-K. Shene, If you know B-Splines well, you also know NURBS!, SIGCSE Bulletin 36 (2004) 343–347

[45] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Computer Methods in Applied Mechanics and Engineering 194 (2005) 4135–4195.

[46] M.G. Cox, The numerical evaluation of B-splines, Technical report, National Physics Laboratory DNAC 4, 1971.

[47] C. De Boor, On calculation with B-splines, Journal of Approximation Theory 6 (1972) 50–62.

[48] S. Krishnan, D. Manocha, An efficient surface intersection algorithm based on lower-dimensional formulation, ACM Transactions on Graphics 16 (1) (1997) 74–106.

[49] R.P. Markot, R.L. Magedson, Solutions of tangential surface and curve intersections, Computer Aided Design 21 (7) (1989) 421–429.

[50] T.W. Sederberg, G.T. Finnigan, X. Li, H. Lin, H. Ipson, Watertight trimmed nurbs, ACM Transactions on Graphics 27 (2008) 79:1–79:8.

[51] T.W. Sederberg, D.L. Cardon, G.T. Finnigan, N.S. North, J. Zheng, T. Lyche, T-spline simplification and local refinement, ACM Transactions on Graphics 23 (2004) 276–283.

[52] C. Ericson, Real-time collision detection, The Morgan Kaufmann Series in Interactive 3-D Technology, 2004, Morgan Kaufman Publishers Inc., San Francisco, CA, USA.

[53] T.-Y. Li, J.-S. Chen, Incremental 3D collision detection with hierarchical data structures, in: Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST '98, ACM, 1998, pp. 139–144.

[54] A. Pandolfi, C. Kane, J.E. Marsden, M. Ortiz, Time-discretized variational formulation of non-smooth frictional contact, International Journal for Numerical Methods in Engineering 53 (8) (2002) 1801–1829.

[55] R.C. Fetecau, J.E. Marsden, M. Ortiz, M. West, Nonsmooth lagrangian mechanics and variational collision integrators, SIAM Journal on Applied Dynamical Systems 2 (3) (2003) 381–416.

[56] S. Leyendecker, J.E. Marsden, M. Ortiz, M. West, Variational integrators for constrained dynamical systems, ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik 88 (9) (2008) 677–708.