Hopper flow of irregularly shaped particles (non-convex polyhedra): GPU-based DEM simulation and experimental validation

Nicolin Govender$^{1,2}$, Daniel N. Wilke$^3$, Chuan-Yu Wu$^1$, Johannes Khinast$^{2,4}$, Patrick Pizette$^5$, Wenjie Xu$^6$

$^1$Department of Chemical Engineering, University of Surrey, United Kingdom

$^2$Research Center Pharmaceutical Engineering, GmbH, Graz, Austria

$^3$Department of Mechanical and Aeronautical Engineering, University of Pretoria, South Africa

$^4$Technical University Graz, Graz, Austria

$^5$Laboratoire Genie Civil et geo-Environnement, IMT Lille Douai, France

$^6$Institute of Geotechnical Engineering, Department of Hydraulic Engineering Tsinghua University, Beijing, China

Abstract

Numerous practical applications of the Discrete Element Method (DEM) require a flexible description of particles that can account for irregular and non-convex particle shape features. Capturing the particle non-convexity is important since it allows to model the physical interlocking when the particles are in contact. To that end, the most flexible approach to capture the particle shape is via a polyhedron, which provides a faceted representation of any shape, albeit at a significant computational cost. In this study we present a decomposition approach to modeling non-convex polyhedral particles as an extension of an existing open source convex polyhedral discrete element code, BlazeDEM-GPU, which computes using general purpose graphical processing units (GPGPUs). Although the principle of decomposition of non-convex particles into convex particles is not new, its application by the discrete element modeling community has been rather limited. The non-convex extension of BlazeDEM-GPU was validated using a hopper flow experiment with identical convex and identical non-convex 3D printed particles. The experiment was designed around two sensitive flow points, with the convex particles following the intermittent flow and the non-convex particles forming stable arches. It was demonstrated that the DEM simulations can be applied to reproduce both the convex and the non-convex flow behavior using the same parameter set. This study is a significant step towards general computing of non-convex particles for industrial-scale applications using the GPGPUs.

1 Introduction
In nature and in industrial processes, particles have various shapes, as depicted Figures 1(a) - (d). A key ingredient in a DEM simulation is abstracting complex particles into shapes suitable for computation. Abstractions involve simple spheres, convex polyhedra, non-convex polyhedra and even non-convex large aspect ratio shapes, often resulting in entangled media. Some two-dimensional abstractions are depicted in Figures 2(a)-(d). Spherical particles can be representative of silica gel and ceramic granules in chemical processing or peas in the food industry, while salt and sugar crystals can be sufficiently approximated by convex polyhedral-shaped particles. Crystalline powders, crushed gravel and cement particles can be represented by non-convex polyhedra, while entangled media contain rigid staples, flexible woven fabrics and particles formed during the granulation process.

![Figure 1](image1.png)  
(a) (b) (c) (d)  
Figure 1: Four examples of typical particles shapes for powders created by (a) spray-drying, (b, c) crystallization and (d) dry granulation that could be approximated by spherical, convex polyhedral, non-convex polyhedral and entangled non-convex polyhedral representations.

![Figure 2](image2.png)  
(a) (b) (c) (d)  
Figure 2: Actual particle shapes abstracted as (a) spheroidal, (b) convex polyhedra, (c) non-convex polyhedra and (d) entangled non-convex polyhedra.

However, the geometrical abstraction of an actual particle is not the only consideration with regard to the DEM
simulations. The computational cost plays an important role in terms of selecting the shape abstraction. In addition, many simulation software codes can only perform particular shape abstractions. The DEM shapes are mainly limited to spherical particles or smoothed polyhedral particles, with spheres or cylinders used for geometric features modelling. A common technique to model non-convex particles is the multi-sphere approach, in which particle shapes lack angularity (often a critical particle property in many applications) [1, 2, 3]. Moreover, flat surfaces are approximated by spherical packing, resulting in artificial surface roughness effects. Triangulation of particle surfaces aims to address some of the clumped sphere shortcomings [4, 5]. However, both approaches involve a significant computational burden, which limits the number of particles that can realistically be considered. Consequently, the number of large-scale non-convex particle simulations is limited. In general, there have been few particle-shape studies in large-scale discrete element simulations despite their well-established industrial importance [1, 2, 3].

Large-scale industrial discrete element simulations can often only afford to abstract particle shapes as spheres or multi-spheres with few particles [2, 6, 7, 8]. Utilizing graphics processing units (GPUs) in the DEM may speed up computational times and allows more complex particle shapes to be used in large-scale industrial simulations. Recently, it has been demonstrated that convex polyhedral shape abstractions, which used to require CPU computing architectures [2, 9, 10, 11, 12, 13], can efficiently be included in large scale industrial simulations using GPUs [14, 15, 16]. This study extends the shape abstraction on the GPU architecture to include non-convex particle shapes and builds on the limited non-convex or concave particle shape DEM simulations that have been conducted on CPU architectures [5, 13, 17, 18, 19, 20, 21, 22]. The initial non-convex/concave particle shapes were limited to two-dimensional abstractions, e.g., in the work of Lim et al., Yang et al. and Alonso [18, 21, 23]. This was extended to three-dimensional smooth abstractions, as demonstrated by Lim et al. [18], which requires non-uniform rational b-spline (NURBS) surfaces to represent the particles and a contact grid to resolve contact. Fleissner et al. [4] extended non-convex abstractions to include faceted bodies that can be in contact with spherical particles by conducting a surface triangulation of the faceted bodies. Smeets et al. [5] generalized this approach by considering non-convex abstractions of particles and applying a surface triangulation to non-convex particles which allowed them to simulate up to 3000 non-convex particles. Recently, Rakotonirina et al. [13] rekindled the well-known convex decomposition proposed by Chazelle [20, 24, 25] as a strategy to extend standard convex polyhedral contact resolution in order to resolve non-convex polyhedral contact using CPU architectures with MPI for 250 non-convex particles.

Following a convex decomposition approach, this study extends the convex polyhedral DEM code, BlazeDEM-GPU, to handle non-convex polyhedral-shaped particles on a GPU architecture. Our extension of BlazeDEM-GPU towards non-convex polyhedral shaped particles is validated against a carefully constructed experimental hopper discharge setup using monodispersed 3D printed particles. In addition, we demonstrate that we can compute a million non-convex polyhedral-shaped particles robustly and accurately within a realistic time frame due to the efficiency of our convex contact and decomposition implementation on the GPU.

1.1 Contact overview

In general, contacts between convex polyhedra are first detected during a broad phase and resolved during a narrow phase. The broad phase estimates which particles are likely to be in contact by conducting a crude but computationally fast estimate. The contact pairs identified during the broad phase are subsequently resolved during the narrow phase to compute the actual contact forces between the particles. These forces are estimated in various ways, including estimating the penetration distance [5, 26, 27], in vertex-face and edge-edge contacts. Alternatively, contact can be resolved by computing the overlapping volume [5, 10, 28] based on which the reaction forces are estimated.
1.2 Premise of this study

This study extends contacts between convex polyhedra to contacts between non-convex polyhedra following a decomposition approach. We constructed non-convex polyhedral particles from a number of convex polyhedral particles, which allowed to later decompose the non-convex particle into a number of convex particles. This multi-convex polyhedral approach is somewhat similar to the multi-sphere approach that decomposes complex shapes into spherical particles. The broad phase contact detection and narrow phase contact resolution are outlined in Figure 3. During the broad phase, the non-convex particles that are likely to be in contact are identified, as depicted in Figure 1.2(a). Contact between every identified non-convex particle contact pair is subsequently refined using a secondary broad phase, as outlined in Figure 3(b). In the secondary broad phase, the non-convex particles of an identified contact pair are decomposed into their respective convex constituents. This allows contact detection between the convex constituents of particle 1 and particle 2, following a secondary broad phase in which the convex constituents that are potentially in contact are established. Next, contact is resolved for the identified contact pairs between the convex constituents of particle 1 and particle 2, following a usual narrow phase contact resolution. In this study, we identified two decomposition strategies for the convex constituents that make up a non-convex particle: decomposition that has a zero or non-zero overlap amongst the convex constituents. Although this distinction has not been made before, it affects the details of computing the inertia tensor, resolving the contact volume and the potential implications of particle breakage when considering breakage models that break a non-convex particle into its convex constituents. First, we outline the state of the art in our current convex polyhedra implementation in BlazeDEM-GPU by focusing our attention on the narrow phase of convex polyhedral contact detection that accurately resolves the contact volume on the GPU.
Figure 3: Non-convex polyhedral contact detection follows a (a) broad phase contact detection between two non-convex particles, followed by a (b) secondary broad phase contact detection between the individual convex constituents of particle 1 and particle 2. This is then followed by the (c) narrow phase in which the contact is resolved between all convex constituents of particle 1 and particle 2 that have been identified to be in potential contact.
2 Convex polyhedra representation

![Figure 4: Typical polyhedral shaped particles include (a) tetrahedron (b) octahedron and (c) dodecahedron and (d) pentagonal pyramid (Johnson solid J2 [29]). The centroids are indicated by red dots.](image)

Recently, Boon et al. [30] demonstrated the possibility of employing a single-level data structure by only storing the faces (or half spaces), i.e., for each face a point on the face and the normal are stored, to define a convex domain that defines a convex particle as a set of linear inequalities. The overlap volume of two polyhedra in contact is then defined based on the combined set of linear inequalities between the two particles. Following linear programming terminology, the basic feasible solutions define the vertices of the overlap polyhedron. The convex hull of the vertices allows for the faces to be defined to ultimately resolve the reaction force magnitude and direction per particle. An alternative single-level data structure could be achieved by storing only the vertices that define a convex hull, which would require the convex hull to be constructed on the fly every-time a particle is evaluated in order to compute the reaction force magnitude and direction per particle. We circumvent the need for highly-efficient particle representations by making use of a particle library to store each particle type or template particle only once in the memory [14]. Each template particle is subsequently orientated via a rotation and translation to an actual particle in the simulation. This minimalistic representation allows to place data structures in the faster yet limited constant memory (48 KB) of the GPU [15].

Four typical polyhedral particles are outlined in Figures 4(a) - 4(d). The rigid body mass parameters which include mass $m$ and inertia tensor $I$ around the center of mass (COM) of the particles are required to compute linear momentum

$$L = mv,$$

for a particle translating at velocity $v$, and angular momentum

$$H = I\omega,$$

for a particle rotating with angular velocity $\omega$. For a particle with volume $V$, the particle mass $m$ given spatial mass density distribution $\rho(x, y, z)$ is obtained by integrating over the particle volume

$$m = \iiint_V \rho(x, y, z) dx dy dz,$$

while the inertia tensor $I_{ij}$ calculated at the COM is

$$I_{ij} = \iiint_V \rho(x, y, z)(r^2 \delta_{ij} - r_i r_j) dx dy dz,$$
with \( r_i = (r_{1i}, r_{2i}, r_{3i}) = (x, y, z) \), \( r^2 = x^2 + y^2 + z^2 \), \( \delta_{ij} \) the Kronecker delta symbol with three indices \( i = 1, 2, 3 \) and \( j = 1, 2, 3 \) following the usual index notation convention. The inertia tensor \( I_{ij} \) is computed at the COM can be calculated relative to point \( R = R_1x + R_2y + R_3z \), by the parallel axis theorem:

\[
J_{ij} = I_{ij} + m \left( |R|^2 \delta_{ij} - R_i R_j \right).
\] (5)

All these properties can be computed off-line for the various particle types employed in a simulation by following exact analytical integration or stochastic Monte Carlo integration. The approach we use to compute these parameters for a convex polyhedral particle is as follows:

1. decompose a convex polyhedral particle into tetrahedra with zero overlap between them,
2. compute the volume, center of mass (COM) and inertia tensor of each tetrahedra using analytically derived expressions,
3. assemble the volume, center of mass and inertia tensor of the polyhedral particle based on tetrahedral contributions.

After the dynamic mass properties have been calculated we can conduct a simulation by explicitly integrating the change in linear and angular momentum to obtain the velocities and positions of each particle. Particles can interact with each other and the “world” geometry that confines them to a spatial domain. Two proportionality scalings for the contact force in three dimensions are typically used i.e.,

1. penetration distance \([5, 31, 32, 33, 34, 20]\),
2. overlap volume \([5, 10, 28]\).

The penetration distance (relative overlap) is by far the most widely used contact modeling approach due to its computational efficiency, in particular for polyhedral particles. However, in Figure 5(a) and (b) the same force is resolved when considering the penetration depth although in Figure 5(a) the entire side of the polygon has penetrated the other polygon as opposed to merely the corner as depicted in Figure 5(b). In addition, the reaction force directions are crudely approximated using the penetration depth as opposed to computing the overlap volume as will become clear in the next Section. In contrast the overlap volume that we implemented results in an energy-conserving contact interaction scheme \([35]\). To the best of our knowledge the only two codes that resolve the overlap volume for convex polyhedra is the academic code BlazeDEM-GPU \([15]\) and commercial code XPS \([36]\).
Figure 5: Polyhedral contact with distinctly (a) larger and (b) smaller overlap volumes although the penetration depths are the same in both cases.

3 Non-convex polyhedra representation

In our study we model non-convex polyhedra by reusing the algorithms developed to handle convex polyhedra as much as possible. This is made possible by following the convex decomposition of non-convex particles. The downside of this approach is that additional memory is required to store the information regarding each convex particle, while the benefits of a convex decomposition approach are:

1. limited modifications to the existing algorithms to include non-convex capability,
2. improvements made for convex polyhedral particle contact that will also benefit non-convex particle contact,
3. supporting functions to compute particle properties for convex particles can be reused for non-convex particles,
4. incorporation of simplified particle breakage model that breaks a non-convex particle into its convex constituents.

The broad phase can be resolved using two potential approaches, namely by 1. encapsulating the non-convex polyhedra into a bounding primitive object, e.g. spheres or cylinders; or alternatively 2. identifying the separating planes for the convex hull representation of the non-convex particle.

In this study, we consider the former approach of bounding primitives by applying the broad phase twice. First, the broad phase is applied to the entire non-convex particle. If the possibility of contact between two particles is established, we refine the broad phase by determining which convex parts in each of the two particles are potentially in contact by identifying the bounding primitives of the convex parts that are potentially in contact. Once all potential convex-convex contact pairs in the two contacting non-convex particles have been established, we merely apply the narrow phase contact resolution between the convex-convex contact pairs to resolve in each
case the overlap volume that ultimately defines the elastic force contribution for each convex-convex contact pair. To enhance the flexibility of convex decompositions of a non-convex particle we consider decompositions that have both non-zero and zero overlap (or intersection) between the convex particles that make-up a non-convex particle. The importance of this distinction is that following the decomposition approach with some overlap between the convex constituents, both the computation of particle properties, such as the inertia tensor, and the contact force of the overlap volume need to consider the material that is accounted for multiple times between the multiple convex particles. The contact force computation may circumvent this complexity should the contact of multiple overlap volumes be statistically unlikely, while the particle properties can be computed for the non-convex particle rather than from the convex constituents. When there is no overlap between the convex particles, it is evident that each convex decomposition uniquely represents the material of the non-convex particle. The percentage of computational time spent in the detailed contact phase for hopper problems computed using convex polyhedra is typically around 80%. Evidently computing non-convex particles implies an additional burden on both memory and the computing resources. The need for additional memory resources is mitigated by the usage of a particle library. The worst case in terms of computational cost for non-convex to non-convex particle contact consists of \( n \times m \) checks for a convex to convex contact, where \( n \) and \( m \) are the number of convex particles making up the two respective non-convex particles. Hence, given total computational cost \( C_C \) for a convex polyhedral hopper simulation, the worst case total non-convex computational cost is estimated by

\[
C_{NC} = 0.2C_C + 0.8C_C \times m \times n.
\]

3.1 Decomposition without any intersection between convex particles

Figure 6(a) - (c) depicts the tetrahedral decomposition of a non-convex polyhedron into three tetrahedra, with zero intersection between the polyhedra when assembled to form the non-convex polyhedron. In addition to simplifying the computation of the particle properties and resolution of the contact this approach may also benefit simplistic particle breakage models in which a particle breaks into its convex polyhedral constituents. The computation of the inertia tensor of the non-convex particle is merely the inertia tensors of the individual convex particles combined using the parallel axis theorem.

Figure 6: (a) Tetrahedron with one surface dented that is (b) decomposed into three tetrahedra and (c) assembled such that the intersection between the tetrahedra is zero.

3.2 Decomposition with significant intersection between convex particles

Figure 7(a) - (c) depicts the tetrahedral decomposition of a non-convex polyhedron into two tetrahedra, with non-zero intersection between the convex polyhedra when assembled to form the non-convex polyhedron. Here, the
material associated with the overlapping volume between the two tetrahedra is duplicated. The computation of the inertia tensor of the non-convex particle is merely the inertia tensors of the individual convex particles combined using the parallel axis theorem with the inertia tensor of the overlap volume subtracted following the parallel axis theorem. Following this strategy of computing the overlap volumes between convex polyhedra we simplify the computation of particle properties such as the inertia tensor.

In addition, should the overlap volume between two contacting particles be duplicated between the non-zero intersect convex constituents we can compute the convex overlap domain and subtract it from the overlap volume, which can be computationally costly. For some non-convex particle compositions these contributions are negligible and can be ignored to speed-up the contact computation.

![Figure 7: (a) Schönhardt polyhedron that is (b) decomposed into two tetrahedra and (c) assembled such that there is significant intersection between the tetrahedra.](image)

4 Contact between polyhedra

As stated above contact between convex polyhedral particles is resolved by finding the overlap volume of the two intersecting polyhedra. Although this is significantly more difficult than determining the overlap distance it does allow for both the direction and the magnitude to be resolved following an energy-conserving contact interaction scheme [35]. We outline the steps involved in computing the overlap volume by considering the intersection of two polyhedral cubes as depicted in Figure 8(a).

Before we continue our discussion we note that the intersection volume between two intersecting polyhedra is convex and given by the convex hull formed using the vertices at the intersections between the polyhedral edges and surfaces [37]. The first step is to find the intersecting vertices as depicted in Figure 8(b), after which we define the surfaces that form the convex hull of the overlap volume as depicted in Figure 9(a). The surface normal \( \mathbf{n}_j \) for each surface \( A_j \) of the convex hull is defined as depicted in Figure 9(b). If we consider contact volume in isolation as shown in Figure 10(a), it becomes clear that the surfaces of this volume are formed by the external surface of one polyhedra that is internal in the other polyhedra. We therefore need to identify which surfaces are internal to which body to ensure that we can compute the associated direction of the reaction force \( \mathbf{n}_i^f \) on particle \( i \). Once the surfaces have been associated with a respective particle as shown in Figure 10(b), where each colour (red and green) is associated with a polyhedral particle, the resultant force can be computed by integrating over the surface.
normals over the contact area as shown by the black lines. The resultant force and surface normals associated with each particle are depicted in Figures 11(a) and (b) respectively.

Before we proceed two additional properties of the contact volume needs to be computed which are required to resolve the elastic contact force i.e., the contact volume $V$ and the center of mass (COM) of the contact volume. As both $V$ and COM will change between contacts it is required to compute them efficiently on the GPU. The divergence theorem

$$
\iiint_V (\nabla \cdot F) \, V = \iiint_{S(V)} F \cdot S,
$$

(6)

allows us to transform the volume integral of the object volume $V$ into a surface integral with boundary surface $S(V)$. This can be done by appropriately choosing any vector field $F(x_1, x_2, x_3)$ that has a divergence of 1 i.e. $\nabla \cdot F = 1$. This gives the volume

$$
\iiint_S F \cdot S = \iiint_V V = V,
$$

(7)

as a surface integral. Alternatively, the contact volume can be computed by breaking it into tetrahedrons for which efficient closed form expressions exist in computing volumetric and inertial properties [38]. However, efficient inertial properties are not required since these can be computed off-line. In this study we compute the exact contact volume as opposed to approximating it [39]. Once, these quantities have been computed we can proceed to compute the direction of the reaction force

$$
n_i^f = \frac{\int_A n_A^i ds}{\int_A n_A^i ds} = \frac{1}{\sum_j A_j} \sum_j A_j n_A^i,
$$

(8)

acting on particle $i$ as shown in Figures 11(a) and (b) for the two particles. Here, the direction of the reaction forces are indicated by the dashed and solid black lines. In addition, the computed reaction force acts through the COM of the overlap volume, while the magnitude of the reaction force is proportional to the volume $V$ of the contact volume. The elastic force associated with contact volume, with volume $V$, for particle $i$ follows the constitutive relationship:

$$
F^i = kV n_i^f,
$$

(9)

where $k$ is the volumetric stiffness with units Newton per meter cubed. In addition, the viscous damping force that depends on the relative velocity and rotation between the particles as well as the friction forces completes the contact forces on the particle [15, 14].
Figure 8: (a) Two polyhedral cubes overlapping (b) intersection vertices are shown as dots.

Figure 9: Resulting (a) contact volume and (b) contact surfaces defined by the surface normals for two contacting polyhedral particles.
Figure 10: (a) Isolated contact volume with (b) resulting force resultant (black lines) and surface normals associated with each particle given by the red and green lines.

Figure 11: The decomposed surfaces (defined by the surface normals) of the contact volume into surfaces internal to the (a) cube at bottom left and the (b) cube on the top right.

5 Validation

The convex triangular prism and non-convex Schönhardt polyhedral hopper discharge simulations are validated via an experimental hopper discharge using 3D printed monodispersed particles made from polylactic acid (PLA).
5.1 Experimental setup

For this study we have selected the convex triangular prism and the non-convex Schönhardt polyhedron as depicted in Figures 12(a) and (b). The convex triangular prism consists of two triangular faces and three rectangular faces. The Schönhardt polyhedron can be obtained from the triangular prism by rotating the two triangular faces of the prism by 45° relative to each other, after triangulating the three rectangular faces of the triangular prism. In particular, the rectangular faces are triangulated via a triangulation across the diagonal of the rectangle from bottom left to top right.

![Figure 12: Visual illustration of one triangular face of (a) a triangular prism and (b) a non-convex Schönhardt polyhedron obtained by rotating the two triangular faces of the triangular prism by 45° relative to each other. In addition, the three rectangular faces of the triangular prism are triangulated across the diagonal before rotation.](image)

The two particles types were batch printed on a $9 \times 17$ grid of particles using four MBot Cube Duo 3D printers (Company: MBot, China) in the FabLab at IMT Lille Mines Douai. The overall geometric tolerance of the printed particles was within $\pm 0.1\text{mm}$, effectively rendering a monodispersed particle setup [40]. Representative individual particles are depicted in Figure 13(a) and (b). The dimensions, print volume fraction (PVF) and average particle mass (APM) of the triangular prisms and Schönhardt polyhedra are given in Table 5.1. The particles were weighed and their densities adjusted to ensure that all particles had the same weight in order eliminate one of the controllable variances between the two particle types. This is done by adjusting the “Print Volume Fraction” (PVF) of each particle i.e. the percentage volume of material present when a unit volume is printed. This is achieved by printing the interior of a particle that is porous as opposed to solid, as shown in Table 5.1. One thousand black and one thousand white particles of each particle type were printed.

<table>
<thead>
<tr>
<th>Particle</th>
<th>Height (mm)</th>
<th>Side Length (mm)</th>
<th>PVF (%)</th>
<th>APM (g)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Triangular prism</td>
<td>8.8±0.1</td>
<td>11.3±0.1</td>
<td>20</td>
<td>0.3469</td>
</tr>
<tr>
<td>Schönhardt polyhedron</td>
<td>8.8±0.1</td>
<td>11.3±0.1</td>
<td>100</td>
<td>0.3470</td>
</tr>
</tbody>
</table>

Table 1: Properties for the 3D printed particles.
In this study the hopper discharge of both the convex triangular prism and the non-convex Schonhardt polyhedron was investigated using a specifically designed hopper [16] as depicted in Figure 14 (b) with the dimensions indicated in Figure 14 (a). The 4cm discharge opening was carefully selected after extensive experimental investigation to determine an appropriate discharge dimension that ensures the formation of stable arches for the Schonhardt polyhedron, while quasi-stable arches for particle systems of triangular prisms are observed to collapse. Quasi-stable here implies a temporary arch that is present for less than 2 seconds before it collapses, i.e. 67% of the total discharge time of the experiment. For the triangular prisms quasi-stable arches were observed from the start right towards the end of discharge. Although this required great care and effort, the advantage of carefully selecting this intermittent flow domain for the triangular prism and the Schonhardt polyhedra is that it is sensitive i.e., slight changes result in the absence of stable or quasi-stable arches. More importantly it also serves as an additional robustness check for the contact algorithm, as minor instabilities in the contact algorithm or integration scheme would result in the collapse of stable or quasi-stable arches.

The experimental hopper discharge of both convex triangular prism and non-convex Schonhardt polyhedral particles was conducted and representative discharges where chosen for as comparison with our numerical simulations. Per simulation around 600 white particles and 400 black particles were used per discharge and packed in a layered fashion to allow us to better visualize the discharge. Due care was taken to isolate the setup from environmental vibrations, which is essential to capture stable arches and allow for higher repeatability in the experimental observations.

Figure 13: Actual 3D printed (a) triangular prism and (b) Schonhardt polyhedron particle.
5.2 Simulation setup

The geometry of the hopper is modeled as planes with dimensions given in Figure 14(a) and stiffness listed in Table 2. During filling an additional plane is modeled across the opening, which is removed to initiate discharge. The simulated triangular prism and the Schönhardt polyhedra are depicted in Figures 15(a)-(b), with the (c) superimposed projects of the triangular prism (green vertices) and Schönhardt polyhedron (red vertices). In addition the vertices for each particle are listed in Table 5.2.

Figure 15: Simulated (a) triangular prism and (b) the Schönhardt polyhedron, with the (c) superimposed projects of the triangular prism (green vertices) and Schönhardt polyhedron (red vertices). The particle vertices are listed in Table 5.2.
Table 2: Vertices for the triangular prism and the Schönhardt polyhedron.

<table>
<thead>
<tr>
<th>Number</th>
<th>Triangular prism</th>
<th>Schönhardt polyhedron</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>x (mm)</td>
<td>y (mm)</td>
</tr>
<tr>
<td>1</td>
<td>-3.67</td>
<td>-5.5</td>
</tr>
<tr>
<td>2</td>
<td>-3.67</td>
<td>-5.5</td>
</tr>
<tr>
<td>3</td>
<td>-3.67</td>
<td>5.5</td>
</tr>
<tr>
<td>4</td>
<td>-3.67</td>
<td>5.5</td>
</tr>
<tr>
<td>5</td>
<td>7.33</td>
<td>0.0</td>
</tr>
<tr>
<td>6</td>
<td>7.33</td>
<td>0.0</td>
</tr>
<tr>
<td>7</td>
<td>0.0</td>
<td>6.0</td>
</tr>
<tr>
<td>8</td>
<td>0.0</td>
<td>1.2</td>
</tr>
</tbody>
</table>

The DEM simulation parameters for the validation simulations are given in Table 5.2, which includes all particle specific parameters and general simulation parameters. A coefficient of restitution (COR) of 0.3 was used for the PLA printed particles which is within the 0.3 and 0.7 range for the experimentally estimated COR [41]. The simulations were conducted on a single NVIDIA Titan X GPU.

Table 3: Simulation parameters for the triangular prism and Schönhardt polyhedron particle systems. (PP) indicates particle-particle and (PW) indicates particle-wall settings.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Triangular prism volume</td>
<td>$V_{TP}$</td>
<td>$5.3240 \times 10^{-7}$</td>
<td>m$^3$</td>
</tr>
<tr>
<td>Schönhardt polyhedron volume</td>
<td>$V_S$</td>
<td>$2.7044 \times 10^{-7}$</td>
<td>m$^3$</td>
</tr>
<tr>
<td>Triangular prism mass</td>
<td>$m_{TP}$</td>
<td>0.3469</td>
<td>g</td>
</tr>
<tr>
<td>Schönhardt polyhedron mass</td>
<td>$m_S$</td>
<td>0.3470</td>
<td>g</td>
</tr>
<tr>
<td>Triangular prism density</td>
<td>$\rho_{TP}$</td>
<td>652</td>
<td>kg/m$^3$</td>
</tr>
<tr>
<td>Schönhardt polyhedron density</td>
<td>$\rho_S$</td>
<td>1283</td>
<td>kg/m$^3$</td>
</tr>
<tr>
<td>Volumetric stiffness (PP)</td>
<td>$k$</td>
<td>$98 \times 10^9$</td>
<td>N/cm$^3$</td>
</tr>
<tr>
<td>Volumetric stiffness (PW)</td>
<td>$k$</td>
<td>$98 \times 10^9$</td>
<td>N/cm$^3$</td>
</tr>
<tr>
<td>Restitution coefficient (PP)</td>
<td></td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>Restitution coefficient (PW)</td>
<td></td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>Static friction (PP)</td>
<td></td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>Kinetic friction (PP)</td>
<td></td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>Static friction (PW)</td>
<td></td>
<td>0.7</td>
<td></td>
</tr>
<tr>
<td>Kinetic friction (PW)</td>
<td></td>
<td>0.7</td>
<td></td>
</tr>
<tr>
<td>Explicit integration time step</td>
<td>$\Delta t$</td>
<td>$1 \times 10^{-5}$</td>
<td>s</td>
</tr>
</tbody>
</table>

5.3 Hopper discharge results

The particles were spaced on a regular grid of seven by two particles that subsequently fell from a height of 20cm under gravity. The particles were initially placed with a random orientation in the grid. In addition, the particles had random uniform initial horizontal velocities that ranged from 0 cm/s to 5 cm/s to simulate some of the variations in the experimental loading. The particle layers were colored post filling. The slight heaping to the right in the
simulation is due to the stochastic nature of packing, each time resulting in a slightly different packing due to the random orientations. This results in a flow pattern and arch that is slightly skewed to the left. Averaged over 5 samples the packing height was 124 (+/- 1) mm for the triangular prism and 129 (+/- 1) mm for the Schönhardt polyhedra.

Figure 5.3(a) and (d) depicts the hopper discharge of triangular prism and Schönhardt polyhedra at 0s, 0.8s, 1.6s and 2.4s respectively. Twenty experimental discharges were performed, with the depicted one chosen as being representative experimental discharges. It is evident that for the triangular prism quasi-stable arches form and are captured by Figure 5.3(a) 1.6s and 2.4s. However, in most cases these arches eventually collapsed and full discharge was observed. In turn, stable arches formed almost in every case of the Schönhardt polyhedra, in particular after due care was taken to isolate the experimental setup from any environmental vibration (slight disturbances resulted in additional discharge until the formation of the next stable arch). Figure 5.3(b) and (d) depict the simulated hopper discharges for the triangular prisms and the Schönhardt polyhedra respectively. As can be seen qualitatively good agreement is achieved between the experiment and simulated discharges.
In addition, a quantitative comparison between the mass discharge for the convex triangular prism and non-convex Schönhhardt polyhedron particle systems is depicted in Figure 17. Twenty experimental discharges were performed (indicated black dash lines) for each particle system with the simulated particle system indicated as a red solid line. It is evident that the experimental discharge is sensitive to the initial random packing of the particles, which is the only difference between the experimental runs. Random uniform initial horizontal velocities that ranged from 0 cm/s to 5 cm/s were introduced to simulate changes in the initial simulated packings that were
deemed representative of the initial experimental packings. It is evident that the simulated discharge rates reflect the experimental discharge rates.

For the triangular prisms we observe a quasi-stable arch forming between 1.6s and 2.4s. However experimentally quasi-stable arches were observed at numerous times as the flow for the triangular prisms is characterized by intermittent flow due to the formation of quasi-stable arches. For the Schönhardt polyhedra we observe a stable arch forming after 0.8s which coincided with our experimental observation. The initial shape of the arch coincides well with experimental observation. The arch remained stable over the duration of the simulation with the exception of deforming slightly over time, which is evident when looking at the shape of the arch at 0.8s, 1.6s and 2.4s. As the arch deformed over the 2.4 second simulation a total of ten particles discharged intermittently during the simulation without compromising the stability of the arch, whereas experimentally no additional particles were discharged and the arch not deforming over time as is evident from Figure 5.3(c). This is an important milestone in the simulation and prediction of stable arches in the hopper discharge.

![Graphs](image1.png)

Figure 17: Twenty independent experimental discharge observations (dash black lines) and five simulated (solid red line) discharge for the (a) convex triangular prism and (b) non-convex Schönhardt polyhedron particle systems. Note that the range of the percentage mass fraction for figures (a) and (b) are different.

5.3.1 Velocity field comparison between convex and non-convex (non-zero intersection) shaped particles

In addition, to comparing the discharges we analysed the differences in the velocity field between the convex triangular prisms and non-convex Schönhardt polyhedra discharges when quasi-stable and stable arches form as depicted in Figures 18(a) and (b) respectively. The initial velocity field for the triangular prisms is developed over a larger domain than the highly localized velocity field for the Schönhardt polyhedra after 0.001s. After 0.8s a stable arch was formed for the Schönhardt polyhedra which remains stable over the duration of the 2.4s simulation. In turn, a quasi-stable arch formed just after 0.8s for the triangular prism that destabilized around 1.3s.
Figure 18: Comparison of the translational velocity magnitude between the (a) triangular prism and (b) Schönhardt particle systems at 0.001s, 0.8s, 1.6s and 2.4s.
Figure 19: Comparison of the detailed translational velocity magnitude between the (a) triangular prism and (b) Schönhardt particle systems at 1.6s, 1.8s, 2.0s and 2.2s.
The development of the spatial flow field after 1.6s is well illustrated in Figures 19(a) and (b), in which we depict the velocity field after every 0.2s. The destabilization of the quasi-stable arch propagates towards the top right allowing for intermittent funnel flow to develop.

5.3.2 Force chains

Part of the complexity of granular media is the heterogeneous spatial distribution of inter-particle forces. The COM of two particles are linked by an edge or force chain when they are in physical contact with each other, while thickness of the force chain is determined by the force magnitude using a linear scale. A quantitative force comparison between the triangular prism and Schönhardt polyhedra, depicted in Figures 20(a)-(b) and Figures 20(c)-(d) respectively, reveals that the developed normal forces between the triangular prisms is up to three times higher and much more localized than the normal forces between the Schönhardt polyhedra. In addition, for the Schönhardt polyhedra the developed force chain network gradually increases with height with a well interlocked network of forces, while for the triangular prism the larger forces are not restricted to the bottom of the hopper. Consequently, the force chain network is more localized for the triangular prism network allowing for easier destabilization of the packing that ultimately results in a higher probability of flow of the particles. In turn the dense force chain network for the Schönhardt polyhedra with a lower spatial force chain magnitude variation allows for a more stable packing. This is also evident for the stable arch that formed at 2.0s for the Schönhardt polyhedra as compared to a significant flow occurring at 2.0s for the triangular prism resulting in a disintegration of the lower force chain network.
5.4 Entangled non-convex polyhedra

To further demonstrate the capabilities of the proposed contact method for non-convex particles and the numerical stability of our approach we simulate entangled non-convex polyhedral particle systems by linking two hollowed triangular prism (HTP) particles, as depicted in Figure 5.4. Each HTP is decomposed into nine convex particles. This is representative of entangled particle systems, where each particle is linked to another particle and free to translate and rotate relative to its linked particle within the constraints imposed by direct contact.

Although entangled media are prevalent in nature, from micro, meso [42, 43, 44, 45] to macro [46, 22, 47]length scales, few studies have considered entangled non-convex polyhedra (e.g. Gravish et al. [22]), due to the computational complexity and the need for efficient robust contact detection algorithms to resolve the contact. Although these particles are idealized and far from industrially relevant they are easy to reproduce, adding to the scientific value as a test problem, as well as playing an important role towards verifying the numerical robustness of the BlazeDEM-GPU framework.

For the simulations described below we used 1024 hollow triangular prisms (HTP) and 1024 simplified dolos
particles as depicted in Figure 5.4(a) and (e) respectively. Of the 1024 HTP particles, 64 are linked to form 32 linked hollow triangular prisms (majority of the particles are single particles). The solid volume and bounding radius for the HTP and simplified dolos particles are tabulated in Table 5.4, with the geometrical details of each particle depicted in Figures 5.4(a) and (e) respectively.

<table>
<thead>
<tr>
<th>Shape</th>
<th>Solid Volume (m$^3$)</th>
<th>Bound Radius (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hollow triangular prism (HTP)</td>
<td>$0.25 \times 10^{-7}$</td>
<td>$0.98 \times 10^{-3}$</td>
</tr>
<tr>
<td>Simplified dolos</td>
<td>$0.27 \times 10^{-7}$</td>
<td>$1.02 \times 10^{-3}$</td>
</tr>
</tbody>
</table>

Table 4: Details of the hollow triangular prisms (HTP) and simplified dolos particles.

The flow pattern states after 0s, 2.0s and 4.0s are depicted for the HTP particle system, while the simplified dolos particles that displayed significant locking are depicted after 0s, 2.0s and 8.0s. The respective mass fractions as a function of time are depicted in Figure 22 for the first three seconds of discharge. The simplified dolos particle system exhibit significant entangled locking, while the linked HTP particle system exhibits intermittent interlocking.
Figure 21: (a) Hollow triangular prism particles and flow patterns for (b)-(d) and (e) simplified dolos particle and flow patterns for (f)-(h).
Finally, we demonstrate a simulation with the mixture of the HTP and simplified dolos particle systems, scaled down to 25% of the sizes depicted in Figures 5.4(a) and (e), to illustrate the ability to handle non-uniform particle shape and particle size. A total of 2048 particles are considered consisting of 1024 hollow triangular prisms and 1024 simplified dolos particles. The initial particle distributions are uniformly mixed. The flow pattern states after 0s, 0.5s, 2.0s and 3.5s are depicted for the mixed particle system in Figure 5.4(a)-(e), respectively. The mass fraction for the mixed particle system is depicted in Figure 22, clearly indicating a much faster discharge towards the end rather than in the beginning. This highlights some of the more complex discharge behaviors of complex particle systems that exemplifies the need to accurately model particle shapes in particular when they introduce new interaction mechanisms such as physical interlocking and entanglement.
Figure 23: Flow pattern of the mixed particle system, consisting of 1024 hollow triangular prisms and 1024 simplified dolos particles after (a) 0s, (b) 0.5s, (c) 2.0s and (d) 3.5s respectively.

Although these entangled particle illustrations are far from industrially relevant particle shapes they do offer a valuable insight into investigating entangled particle systems as highlighted by the developed force chains depicted in Figures 5.4(a)-(c) for the linked HTP, dolos and mixed particle systems after 2.0s of discharge. It is evident that the developed force chain network for the dolos particle system is significantly more developed and balanced over the spatial domain than the linked HTP and mixed particle systems. The implication is the force chain network is not so easily destabilized by small changes in a well developed force chain network as the force network offers multiple paths of support to particles over the hopper opening. In contrast the smaller particles of the mixed particle system after 2.0s is discharging freely as depicted in 5.4(c). The force chains link over smaller spatial distances as a result of particle size resulting finer detail in the force chain network. In addition, 5.4(c) that the force chain network is the most developed around the edges of the hopper opening but offers limited support to the particles over the hopper opening resulting in discharge of particles. The force network of the linked TP, depicted in 5.4(a), offers partial support for particles over the hopper opening from the left edge, while the force network support for particles from the right edge is partially degraded.

Figure 24: Developed force chains for the (a) linked hollow triangular prisms (b) dolos and (c) mixed particle systems after 2.0s of discharge.
6 Large scale simulation of convex and non-convex (zero intersection) shaped particles

We now consider the discharge of a convex tetrahedral particle system and non-convex dented tetrahedral particle system constructed from three tetrahedra without overlap as depicted in Figure 6. The dimensions of the convex tetrahedron and non-convex dented tetrahedron are identical except that one face of the dented tetrahedron is indented by one third of the height from the base to the apex. Each edge of the outer dimensions of the tetrahedron is 11.3 mm. In both simulations 262144 particles were used.

![Tetrahedron and disassembled dented tetrahedron](image)

(a) (b)

Figure 25: (a) Tetrahedron and (b) disassembled dented tetrahedron, with edge lengths of 11.3mm, used in the large scale simulation.

The flow rates for the convex tetrahedral particle system and non-convex dented tetrahedral particle system with the corresponding flow velocity fields and patterns are depicted in Figures 27 and 26 respectively. The mass fraction saturates at around 10%, as particles settle to the left and the right of the hopper basin. The two tetrahedra are identical except the dented tetrahedron has one surface dented inwards making it non-convex. The slower flow rate of the dented tetrahedron is evident in addition to a change in the discharge slope around 300 ms. The importance of, modeling non-convexity is highlighted since it introduces a significantly different mechanism, namely physical interlocking, that may even be the dominating factor to model. This highlights the importance of physical interlocking as a mechanism present in particle flows in addition to demonstrating that constructing non-convex particles from convex components with zero or non-zero intersection between the convex components is a viable and effective means of extending a convex polyhedral simulation platform.
Figure 26: Mass fraction over time for the tetrahedron and dented tetrahedron particle systems.
7 One Million Particle Drop Test

To conclude this study we considered a large scale packing to demonstrate the robustness of the convex and non-convex particle-particle and particle-wall contacts. The aim of the investigation is to ensure that the final number of particles after each simulation is indeed 1 million particles, in particular when subjected to this drop test where multiple impulse loads at various angles are imposed onto the particles. In addition, the drop test is conducted over a large surface area to allow for unstable particles to escape the packing should they become unstable. Although this mass balance test is rudimentary it remains an important test when polyhedral particles are considered to ensure robust and well resolved particle-particle contact, since stability can be challenging [48].

In this study in two separate simulations we packed 1 million convex triangular prisms and 1 million non-convex Schönhardt polyhedra in a 5m by 5m by 2m container. The initial particle velocities are 2 m/s and falling from a
height of 1.75m. Each layer of particles consists of a regular grid of particles that is 4m by 4m and contains 26569 triangular prisms or 28561 Schönhardt polyhedra per layer.

The large scale packing depicted in Figures 28(a), (c) and (e) demonstrates robustness of the convex triangular prism particle-particle and particle-wall interactions, while the Figures 28(b), (d) and (f) demonstrate the non-convex Schönhardt polyhedra particle packing. The final number of particles in both simulations was 1 million particles i.e. no particles were lost over the course of the packing due to instabilities.

Figure 28: Packing of $10^6$ convex triangular prism particles after (a) 0.1s, (c) first impact with floor (0.385s) and (e) impact of last layer (1.165s) and $10^6$ non-convex Schönhardt polyhedral particles after (b) 0.1, (d) first impact with floor (0.385s) and (f) impact of last layer (1.100s).

8 Conclusions

This study demonstrated that non-convex particles can be modeled efficiently on a GPU computing architecture following a convex decomposition approach to model contact between non-convex particles. The benefit of this
approach is that a significant portion of already developed convex contact machinery for polyhedral particles can be reused when available.

The results were validated in a carefully constructed experimental study. In particular, the experiment was designed such that the convex particles exhibit quasi-stable arches, while the non-convex particles exhibit stable arches. The ability of a discrete element simulation to capture fully stable arches is a significant validation test. Since explicit time integration inherently results in slight vibrations amongst the particles that may destabilize stable arches to exhibit quasi-stable arches or fully developed flow even when stable arch formation can be expected. In addition, simulating stable arches serves as a significant test of the quality of the contact algorithms employed i.e. to be able to simulate stable arches the discrete element simulation needs to be devoid of any contact instabilities that would otherwise destabilize any arch formation. The stability of our contact algorithm was further stress tested by considering a 1 million particle drop test for both the convex and non-convex particle systems.

The results demonstrate that quasi-stable and stable arches can be simulated using our approach. The difference in the force chain networks between quasi-stable and stable arches was investigated, which clearly pointed to a dense network in the non-convex particle system in compared to the convex particle system. This investigation pointed towards the benefit of simulation, allowing to visualize quantities not readily measurable experimentally. In addition, entangled non-convex particle systems are considered that demonstrates the potential for highly complex particle systems that exhibit significantly different hopper discharge behaviour. Lastly, as non-convex particles can become quite complex and difficult to reproduce by other researchers, we stress the importance of using reproducible particles by other researchers. In this study we utilized 3D printed particles to allow for a rigorous validation that allows for other researchers to reproduce the experimental findings. This is critical in terms of moving DEM research forward [49], especially as the particle systems under investigation become more complex.

Acknowledgments

This work was supported by the MARIE Sklodowska-CURIE Individual Fellowships with acronym DECRON, funded through the People Programme (MARIE Sklodowska-CURIE Actions) of the European Union’s H2020 under REA grant agreement No. 747963. We gratefully acknowledge the support of the NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research, as well as the 3D printer equipment supplied by the FabLab des Mines de Douai. The financial support of the National Research Foundation (NRF) of South Africa is acknowledged.
References


