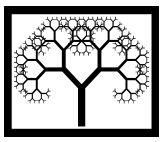
Paper 43



Proceedings of the Fourth International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering, P. Iványi and B.H.V. Topping (Editors), Civil-Comp Press, Stirlingshire, Scotland.

Parallel Simulation of Multidisperse Granular Flows using GPUs

I. Critelli[1], A. Tasora[2], M. Colledani[1] and H. Mazhar[3]
[1]Department of Mechanical Engineering, Politecnico di Milano, Italy
[2]Department of Industrial Engineering
Università degli Studi di Parma, Italy
[3]Department of Mechanical Engineering
University of Wisconsin-Madison, United States of America

Abstract

In this paper, we present an application of GPU-based parallel computation for the simulation of multidisperse granular flows. We also show an application for the case of the Corona Electrostatic Separation (CES) process used in the waste management industry, where a strong electric field is used to separate plastic from metal particles in dense multidisperse granular flow of oddly-shaped fragments. The two major bottlenecks of the simulation are the collision detection and the solution of a complementarity problem at each time step; this limits the number of particles that can be simulated in reasonable time frames on the CPU, so we ported our simulation software to a parallel computing architecture. A custom collision detection has been used, where both broad-phase and narrow-phase collision stages have been designed in order to exploit parallel computation; such an algorithm is able to deal with particles of different shape and size, as needed in multidisperse granular flow. Also, a custom solver has been developed for solving the complementarity problem on parallel hardware. Such a solver requires multiple kernels and complex computational primitives because the complementarity problem does not fit in the perfectly-parallel computational paradigm, moreover, special care must be used to exploit data coalescence as much as possible. Finally, external force fields have been introduced, to simulate and reproduce the physics of electrostatic forces in the CES separation process.

Keywords: GPU, multibody, collision detection, corona electrostatic separator, differential variational inequality.

1 Introduction

The simulation of dense granular flows with a large number of frictional contacts between particles is still an open problem in computational mechanics: our approach

uses an implicit integration is based on the recent Discrete Variational Inequalities (DVI) formulation. This leads to Cone-Complementarity Problems (CCP) and permits larger timesteps when compared to the explicit integration used in classical Discrete Element Methods (DEM). Typical numerical methods for the simulation of granular flows are based on large sets of Ordinary Differential Equations (ODEs), where interaction between particles are represented by nonlinear, yet smooth, force fields. Such contact forces are regularized versions of the original non-smooth forces. This is the most used approach in DEM software [1]. Although the regularization strategy can leverage on well tested ODE solvers [2], the high stiffness of the force fields require extremely small time steps, and this is even more critical when the size of the particles is small. To overcome these difficulties, an alternative approach has been used in this work: we implemented a simulation framework based on the DVI concept, where the non-smooth nature of contact forces is directly embedded in the formulation [3, 4, 5].

The DVI model is discretized in time steps, each step requiring the solution of a second-order CCP, a complementarity problem that is a special type of Variational Inequality (VI). This leads to an efficient integration method that is stable and robust even in case of large time steps [6].

In such framework, the two major computational bottlenecks are the solution of the CCP and the collision detection. The difficulty resides in the fact that both problems cannot be solved with algorithms that belong to the so-called embarassingly-parallel class, and special methods must be developed in order to take advantage of parallel architectures. In [7] we presented a parallel version of a fixed-point algorithm for the solution of CCP problems, and we implemented it on GPU architectures. In [8] we developed a parallel collision detection algorithm that can run on GPU and that can deal efficiently with millions of particles. A brief discussion on the parallel implementation of each algorithm will be provided in this document.

2 The corona electrostatic separation process

In the following we describe an engineering application that motivated the adoption of our parallel simulation software.

Mechanical recycling systems are multi-stage systems including size-reduction and separation technologies. In the de-manufacturing process-chain they are typically included between disassembly operations and, for some products, chemical recovery processes. Corona electrostatic separators (CES) are mainly used to separate conductors from insulators, small particles (typically below 1-2 mm), like copper from plastics in shredded waste of electrical and electronic equipment. In order to do so, first particles are charged and then passed through an electric field. More specifically, the separator uses corona charging or ion discharging to establish a charge on particles when particles pass between two high-voltage electrodes which create an electric field. The particles are brought through this field by an input hopper and a vibrating feeder, which regulates the material flow rate. Particles receive a discharge of elec-

tricity, which gives the non-metals a high surface charge, causing them to be remain stacked on the drum and fall in a different bin on the left upon the effect of a dedicated brush. Metal particles do not become charged, as the charge rapidly dissipates through the particles to the earthed rotor, and so they fall into the first bin on the right because they are charged only by electrostatic induction and projected towards the static electrode of opposite sign. Intermediate products, whose conductivity is between the two kinds of materials or non-liberated particles in the mixture, drop in the middle bin.

The quality of the separation is influenced by controllable parameters like electrode voltage and position, drum speed, splitters position, and feed rate. On the other end, non-controllable parameters affected the efficiency of the process as particle size, shape and material mixture. Moreover, due to randomness in the process, the CES provides contaminated output flows. Two major sources of randomness are relevant in this process, i.e. the presence of non-liberated particles in the mixture and the influence of particle-particle interactions and impacts among conductive/non-conductive particles. For these reasons, the study of metal and non-metal particle trajectories and particles interaction using numerical models are important because they influence the output of separation process.

Due to the complexity of the process, modeling the trajectories of particles within CES is of practical engineering interest. In the literature, several attempts have been made to simulate particle trajectories in roll-type electrostatic separators, for example, in [9, 10, 11]. As it can be noticed, existing recycling process models only focus on single particle trajectories and fail to model the major cause for loss of efficiency in the separation process, i.e. particle-particle and particle-equipment interactions and impacts. Instead, this model already implemented on the CPU [12, 13], considers all the aspects mentioned above.

3 DVI for particle simulations

The DVI formulation stems from the non-smooth nature of contact forces. When simulating granular flows, simultaneous contacts between pairs of particles might be in number of thousands, if not millions. Let $\mathbf{n}_i$ be the normal at the i-th contact point, given two particles A and B. Let $\mathbf{u}_i$ and $\mathbf{w}_i$ be two vectors in the contact plane such that $\mathbf{n}_i, \mathbf{u}_i, \mathbf{w}_i \in \mathbb{R}^3$ are orthogonal vectors.

Here we use Φ_i to represent the signed contact distance, assumed to be differentiable at least in the neighborhood of the contact.

A contact force is $\mathbf{F}_{i,N} = \widehat{\gamma}_{i,n}\mathbf{n}_i$, where $\widehat{\gamma}_{i,n} \geq 0$ is the multiplier that represents the modulus of the reaction, whereas friction force is represented by the multipliers $\widehat{\gamma}_{i,u}$, and $\widehat{\gamma}_{i,w}$ which lead to the tangential component of the reaction $\mathbf{F}_{i,T} = \widehat{\gamma}_{i,u}\mathbf{u}_i + \widehat{\gamma}_{i,w}\mathbf{w}_i$.

The Signorini conditions [3] express the normal contact model using complementarity constraints:

$$\widehat{\gamma}_{i,n} \geq 0 \quad \perp \quad \Phi_i(\cdot) \geq 0 \tag{1}$$

For tangential forces, arising from sliding or sticking frictional phenomena, we

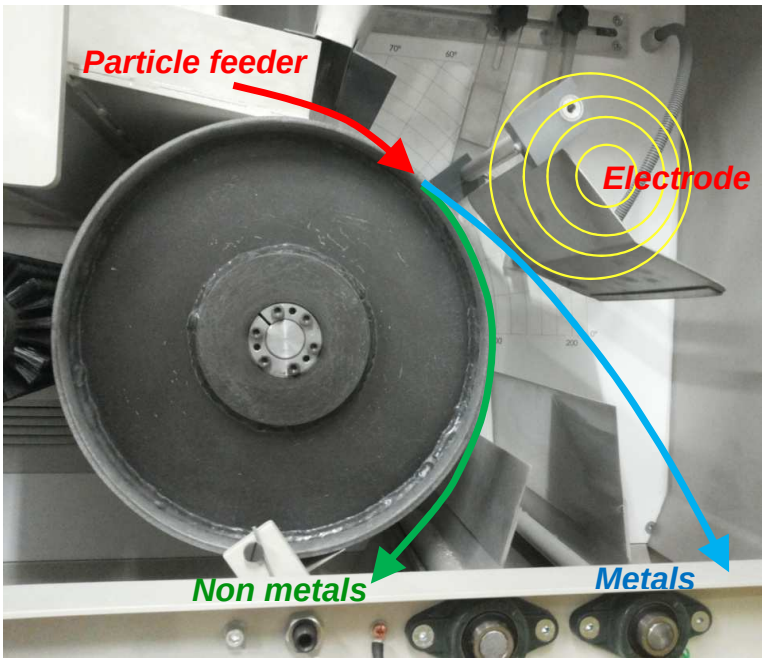


Figure 1: Corona Electrostatic separation machine

introduce the Amontons-Coulomb friction model, where the ratio between the normal and the tangential force is limited by friction coefficient μ_i. Also, the tangential force must have a direction that is opposite to the tangential speed $\mathbf{v}_{i,T}$, if any:

$$\begin{aligned} \mu_i \widehat{\gamma}_{i,n} \quad \geq \quad & \sqrt{\widehat{\gamma}_{i,u}^2 + \widehat{\gamma}_{i,w}^2} \quad , \quad ||\mathbf{v}_{i,T}|| \left(\mu_i \widehat{\gamma}_{i,n} - \sqrt{\widehat{\gamma}_{i,u}^2 + \widehat{\gamma}_{i,w}^2} \right) = 0, \\ & \langle \mathbf{F}_{i,T}, \mathbf{v}_{i,T} \rangle = - \, ||\mathbf{F}_{i,T}|| \;\; ||\mathbf{v}_{i,T}|| \end{aligned} \tag{2}$$

For the full multi-contact, multi-body problem, the system state is defined by the vector of generalized coordinates $\mathbf{q} \in \mathbb{R}^{m_q}$ and the vector of generalized speeds $\mathbf{v} \in \mathbb{R}^{m_v}$ of the particles. We remark that particles are not simple point-like masses: they can rotate in three-dimensional space, and their rotation is represented with an unit quaternion $\epsilon \in \mathbb{H}_1$ to avoid singularities in the parametrization of $\mathsf{SO}(\mathbb{R}, 3)$. The linear mapping $\dot{\mathbf{q}} = \Gamma(\mathbf{q})\mathbf{v}$ is described in [14].

We also introduce generalized force fields

$$\mathbf{f}_t(\mathbf{q}, \mathbf{v}, t) = \mathbf{f}_a(\mathbf{q}, \mathbf{v}) + \mathbf{f}_s(\mathbf{q}, \mathbf{v}, t) + \mathbf{f}_g(\mathbf{q}) + \mathbf{f}_c(\mathbf{q}, \mathbf{v})$$

including aerodynamic forces $\mathbf{f}_a(\mathbf{q}, \mathbf{v})$ acting on the particles, forces $\mathbf{f}_s(\mathbf{q}, \mathbf{v}, t)$ caused by particle interaction with the magnetic forces in the CES device, gyroscopic forces $\mathbf{f}_c(\mathbf{q}, \mathbf{v})$, and gravitational forces $\mathbf{f}_g(\mathbf{q})$. For details on the approximate, analytical expression of $\mathbf{f}_s$ and $\mathbf{f}_a$ that we use in our model we refer to [12].

We introduce the mass matrix $M(\mathbf{q}) \in \mathbb{R}^{m_v \times m_v}$, including all the masses and inertia tensors of the particles on the diagonal.

Bilateral constraints are introduced via a set $\mathcal{G}_B$ of scalar constraint equations, as-

sumed differentiable everywhere:

$$\Psi^i(\vec{q},t)=0, \quad i\in\mathcal{G}_{\mathcal{B}}. \tag{3}$$

We introduce $\nabla_q\Psi^i=[\partial\Psi^i/\partial\vec{q}]^T$ and $\nabla\Psi^{i^T}=\nabla_q\Psi^{i^T}\Gamma(\mathbf{q})$, to express the constraint (3) at the velocity level after differentiation:

$$\frac{d\Psi^i(\vec{q},t)}{dt}=\nabla\Psi^{i^T}\mathbf{v}+\frac{\partial\Psi^i}{\partial t}=0, \quad i\in\mathcal{G}_{\mathcal{B}}. \tag{4}$$

The term $\frac{\partial\Psi^i}{\partial t}$ is used for rheonomic (time-dependent) constraints such as, in our case, the constraint that imposes the rotation of the drum respect to the truss of the CES device.

Frictional unilateral contacts define a set $\mathcal{G}_A$. For each contact $i\in\mathcal{G}_A$, we introduce the tangent space generators, that can be derived as tangent constraint Jacobians: $\mathbf{D}^i_{\gamma_u}$, $\mathbf{D}^i_{\gamma_w}$.

We write (2) using the maximum dissipation principle, thus leading to a nonlinear program $(\widehat{\gamma}_{i,u},\widehat{\gamma}_{i,w})=\text{argmin } \mathbf{v}^T\left(\mathbf{D}_{\gamma_u}\widehat{\gamma}_{i,u}+\mathbf{D}_{\gamma_w}\widehat{\gamma}_{i,w}\right)$ subject to constraint $\sqrt{\widehat{\gamma}^2_{i,u}+\widehat{\gamma}^2_{i,w}}\leq\mu\widehat{\gamma}_{i,n}$.

An alternative expression for the friction model can be written by deriving the Fritz John optimality conditions for the nonlinear program, introducing a multiplier λ_v and obtaining

$$\nabla_{\gamma_u,\gamma_w}\mathbf{v}^T\left(\mathbf{D}_{\gamma_u}\widehat{\gamma}_{i,u}+\mathbf{D}_{\gamma_w}\widehat{\gamma}_{i,w}\right)-\lambda^i_v\nabla_{\gamma_u,\gamma_w}\left(\mu^i\widehat{\gamma}_{i,n}-\sqrt{\widehat{\gamma}^2_{i,u}+\widehat{\gamma}^2_{i,w}}\right)=0 \tag{5}$$

$$\mu^i\widehat{\gamma}_{i,n}-\sqrt{\widehat{\gamma}^2_{i,u}+\widehat{\gamma}^2_{i,w}}\geq 0, \quad \perp \quad \lambda^i_v\geq 0. \tag{6}$$

The complete model, including inertial effects, force fields, bilateral constraints, and unilateral frictional contacts, is the following DVI:

$$\begin{aligned}
i\in\mathcal{G}_{\mathcal{B}} \quad &: \quad \Psi^i(\mathbf{q},t)=0\\
i\in\mathcal{G}_{\mathcal{A}} \quad &: \quad \widehat{\gamma}^i_n\geq 0 \quad \perp \quad \Phi^i(\mathbf{q})\geq 0\\
&\quad \nabla_{\gamma_u,\gamma_w}\mathbf{v}^T\left(\mathbf{D}_{\gamma_u}\widehat{\gamma}_{i,u}+\mathbf{D}_{\gamma_w}\widehat{\gamma}_{i,w}\right)\\
&\quad -\lambda^i_v\nabla_{\gamma_u,\gamma_w}\left(\mu^i\widehat{\gamma}_{i,n}-\sqrt{\widehat{\gamma}^2_{i,u}+\widehat{\gamma}^2_{i,w}}\right)=0\\
&\quad \mu^i\widehat{\gamma}_{i,n}-\sqrt{\widehat{\gamma}^2_{i,u}+\widehat{\gamma}^2_{i,w}}\geq 0, \quad \perp \quad \lambda^i_v\geq 0\\
\dot{\mathbf{q}} \quad &= \quad \Gamma(\mathbf{q})\mathbf{v}\\
M(\mathbf{q})\frac{d\mathbf{v}}{dt} \quad &= \quad \sum_{i\in\mathcal{G}_{\mathcal{A}}}\left(\widehat{\gamma}_{i,n}\mathbf{D}^i_{\gamma_n}+\widehat{\gamma}_{i,u}\mathbf{D}^i_{\gamma_u}+\widehat{\gamma}_{i,w}\mathbf{D}^i_{\gamma_w}\right)+\\
&\quad +\sum_{i\in\mathcal{G}_{\mathcal{B}}}\widehat{\gamma}^i_{\mathcal{B}}\nabla\Psi^i+\mathbf{f}_t(t,\mathbf{q},\mathbf{v})
\end{aligned} \tag{7}$$

The former DVI can be discretized in time introducing a time step h. We set $\gamma=h\widehat{\gamma}$, we use an exponential map $\Lambda(\cdot)$ for incremental update of rotations in Lie groups

[15, 6], and we obtain:

$$
\begin{array}{rcl}
i \in \mathcal{G}_{\mathcal{B}} & : & \dfrac{1}{h}\Psi^i(\vec{q}^{(l)}) + \nabla\Psi^{i^T}\vec{v}^{(l+1)} + \dfrac{\partial\Psi^i}{\partial t} = 0 \\
i \in \mathcal{G}_{\mathcal{A}} & : & \gamma_{i,n} \geq 0 \quad \perp \quad \frac{1}{h}\Phi^i(\vec{q}^{(l)}) + \nabla\Phi^{i^T}\vec{v}^{(l+1)} \geq 0 \\
 & & \nabla_{\gamma_u,\gamma_w}\mathbf{v}^T\left(\mathbf{D}_{\gamma_u}\gamma_{i,u} + \mathbf{D}_{\gamma_w}\gamma_{i,w}\right) \\
 & & -\lambda_v^i\nabla_{\gamma_u,\gamma_w}\left(\mu^i\gamma_{i,n} - \sqrt{{\gamma_{i,u}}^2 + {\gamma_{i,w}}^2}\right) = 0 \\
 & & \mu^i\gamma_{i,n} - \sqrt{{\gamma_{i,u}}^2 + {\gamma_{i,w}}^2} \geq 0, \quad \perp \quad \lambda_v^i \geq 0 \\
\mathbf{q}^{(l+1)} & = & \Lambda(\mathbf{q}^{(l)}, \mathbf{v}^{(l+1)}, h) \\
M^{(l)}(\mathbf{v}^{(l+1)}) & = & \displaystyle\sum_{i\in\mathcal{G}_{\mathcal{A}}}\left(\gamma_{i,n}\mathbf{D}^i_{\gamma_n} + \gamma_{i,u}\mathbf{D}^i_{\gamma_u} + \gamma_{i,w}\mathbf{D}^i_{\gamma_w}\right) + \\
 & & + \displaystyle\sum_{i\in\mathcal{G}_{\mathcal{B}}}\gamma^i_{\mathcal{B}}\nabla\Psi^i + h\mathbf{f}_t(t,\mathbf{q},\mathbf{v}) + M^{(l)}\mathbf{v}^{(l)}
\end{array}
\tag{8}
$$

A relaxation of the original problem, that leads to a convex problem whose solution is guaranteed, has been proposed in [4] and it is used in our software. The convex form of the problem above is second-order Cone Complementarity Problem (CCP).

To express the CCP, we introduce the multidimensional cone obtained by performing the Cartesian product of all three-dimensional Lorentz-cones $\mathcal{FC}$ equivalent to Coloumb-Amontons friction cones, adding also $\mathcal{BC} = \mathbb{R}$ trivial cones for bilateral constraint forces, obtaining:

$$
\Upsilon = \left(\bigtimes_{i\in\mathcal{G}_{\mathcal{A}}}\mathcal{FC}^i\right)\bigtimes\left(\bigtimes_{i\in\mathcal{G}_{\mathcal{B}}}\mathcal{BC}^i\right). \tag{9}
$$

We denote the polar cone of Υ as Υ°. To make the expressions more compact, we group all the tangent space generators (jacobians) in the sparse $D_{\mathcal{E}}$, we append all the lagrangian multipliers γ_i in $\gamma_{\mathcal{E}}$, we append $\frac{\Psi^i}{h} + \frac{\partial\Psi^i}{\partial t}$ and all $\frac{\Phi^i}{h}$ in $\mathbf{b}_{\mathcal{E}}$, we set $\mathbf{k} = M\mathbf{v}^{(l)} + h\mathbf{f}_t^{(l)}$ and we write:

$$
\begin{bmatrix} M & D_{\mathcal{E}} \\ D_{\mathcal{E}}^T & 0 \end{bmatrix}\begin{Bmatrix} \mathbf{v}^{(l+1)} \\ \gamma_{\mathcal{E}}^{(l+1)} \end{Bmatrix} - \begin{Bmatrix} \mathbf{k} \\ -\mathbf{b}_{\mathcal{E}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \mathbf{u}_{\mathcal{E}} \end{Bmatrix} \tag{10}
$$

$$
\mathbf{u}_{\mathcal{E}} \in -\Upsilon^\circ \quad \perp \quad \gamma_{\mathcal{E}} \in \Upsilon \tag{11}
$$

One can see that in the much simpler case of bilateral constraint only, Eq.11 becomes $\mathbf{u}_{\mathcal{E}} = \mathbf{0}$, and Eq.10 becomes a simple linear problem of saddle-point type.

The previous CCP is expressed as a mixed-complementarity, but it can be expressed also as a pure CCP by introducing the Schur complement N; this is practicable because M is easily invertible:

$$
N = D_{\mathcal{E}}^T M^{-1} D_{\mathcal{E}} \tag{12}
$$

$$
\mathbf{r} = D_{\mathcal{E}}^T M^{-1}\mathbf{k} + \mathbf{b}_{\mathcal{E}} \tag{13}
$$

The VI now can be rewritten as a CCP:

$$(N\gamma_{\mathcal{E}} + \mathbf{r}) \in -\Upsilon^{\circ} \quad \perp \quad \gamma_{\mathcal{E}} \in \Upsilon. \tag{14}$$

For the solution of such CCP we implemented various types of numerical solution schemes. The fastest method is based on a fixed-point algorithm that solves the CCP by using a projected Gauss-Seidel iteration: it has slow convergence but its implementation can be parallelizable (as projected block-Gauss-Seidel or a projected Gauss-Jacobi) on GPU high-performance computing architectures [7].

Other more sophisticated solvers that we tested are based on modified Krylov subspace iterations, such as the Nesterov method and the Spectral Projected Gradient method [16]; in both cases the most relevant computational primitive to be parallelized is the sparse-matrix by vector multiplication.

4 Parallel implementation of the DVI

The Parallel DVI framework utilizes CUDA and OpenMP in order to accelerate the solution process. The parallel implementation consists of three key components, The data manager, the solver and the collision detection algorithm.

The purpose of the data manager is to store the relevant information for the simulation in a parallel friendly manner. In practice this means using large arrays to store information, enabling efficient cache usage, vectorization and parallelization. The data manager is a large structure containing several arrays storing collision geometry and contact information along with position, velocity and other state data. Memory management for the GPU is performed using the Thrust [17] template library which simplifies allocation and copy operations of GPU data structures along with providing implementations of data parallel algorithms required for the solver and collision detection.

For both Gauss-Jacobi and Krylov based methods the costly operations are generation of the jacobian matrices and in the case of the Krylov method, a sparse matrix-vector multiplication. While not compute intensive, these operations are memory bound and limited by the memory bandwidth of the platform. Because of this, GPU's with their higher memory bandwidth and parallel friendly cache structure greatly improve the performance of such operations. In this framework both the Jacobian computation and the linear algebra operations used in the solver have been parallelized.

5 Parallel collision detection

When dealing with complex simulations of the CES with hundred of thousands to millions of rigid bodies, the collision detection step, because of the large number of contact events, becomes a bottleneck when compared to other parts of the simulation process. However, because of the large number of objects and contacts, the problem

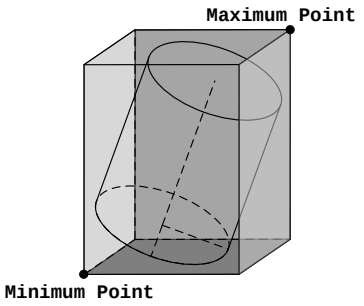


Figure 2: Ex: AABB generation for 3D cylinder.

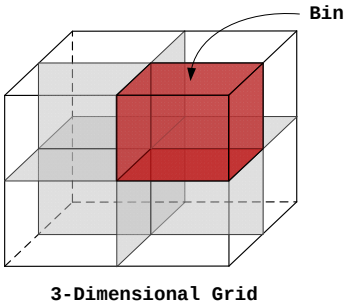


Figure 3: Ex: 3D space divided into bins.

fits into the Single Instruction Multiple Data (SIMD) paradigm. In general, for each body and contact the same sequence of instructions are performed. Consequently, an algorithm capable of leveraging SIMD hardware like commodity GPUs was developed to remove collision detection as the bottleneck for large granular dynamics simulations.

The collision detection algorithm is performed into two phases, Broad-Phase and Narrow-Phase. Broad-Phase is where the algorithm quickly determines potential contact pairs based on the current configuration of the system. Once the potential contact list has been generated the Narrow-Phase algorithm is used to determine the actual contact information. An outline of the parallel algorithm is presented below, for more details see [8].

The Broad-Phase algorithm computes whether two bodies are in contact at a given time. The goal is not to find actual contact information, but rather to determine if a contact between two objection could potentially occur based on the Axis Aligned Bounding Boxes of the bodies involved.

An Axis Aligned Bounding Box (AABB) is a bounding box that is always aligned to the global reference frame, i.e. does not rotate with the object, which simplifies collision detection. The consequence of using AABBs, is that the volume enclosed by the bounding box is equal to or greater than the volume of the shape it encloses. AABB generation is simple and parallelization can be done on a per object basis. See Fig. 2 for an example of AABB computation for a cylinder in $\mathbb{R}^3$.

An overview of the GPU based collision detection algorithm is as follows. The process beings with identifying intersections between AABBs and a fixed spatial grid. The size of each cell, or bin (see Fig. 3 for a visual representation of a bin), in this grid is determined by the average feature size for the simulation. The AABB-bin pairs are then sorted by their bin id using a parallel key-value sort algorithm [17]. Next, each bin's starting index is determined so that the bins' AABBs can be traversed sequentially. Because the starting index is determined per bin, each bin can be processed in parallel with an inner loop traversing the contained AABBs. All AABBs touching a bin are subsequently checked against each other for collisions.

Once potential contacts have been determined the list of contacts is sent to the Narrow-Phase algorithm which processes each possible contact and determine if it

actually occurs. To this end an algorithm capable of determining contacts between convex geometries was implemented on the GPU. This algorithm, called "XenoCollide" [18], is based upon Minkowski Portal Refinement (MPR) [19].

6 Results

Figure 4 shows a typical result from our simulations. The granular flow is multidisperse because it represents the outcome of previous shredding processes involving different materials, therefore we developed an algorithm that can create a constant flow of particles whose sizing, density, shape and other properties can follow prescribed probability distributions.

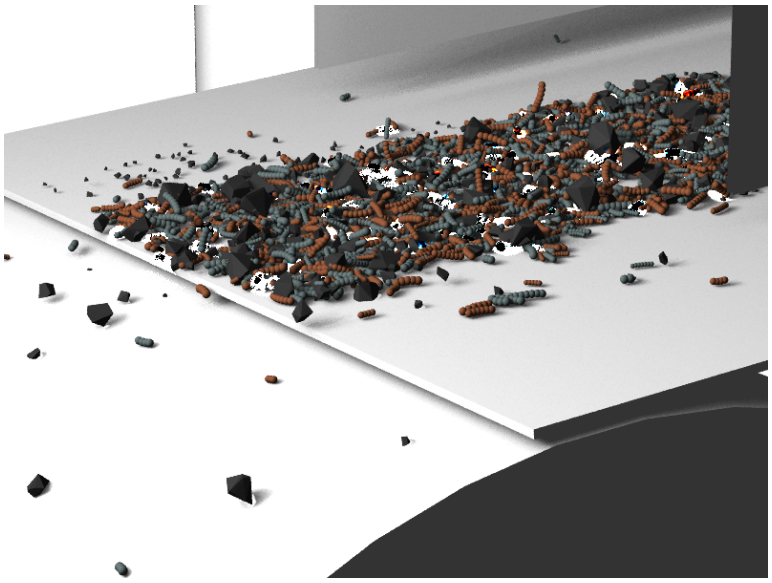


Figure 4: Example of simulated multidisperse granular flow

Figure 5 shows two examples of output mass distributions and simulations parameters. Through data post-processing is possible to obtain various mass distributions according to changes in most machine parameters.

Furthermore the position of two splitter between the collecting boxes of conducting, middling and nonconducting products are considered. They can move along the horizontal coordinate at fixed height, and the variation for both splitters is in the $[0, 200]$ mm range, divided in 60 possible positions. In this case, for simplicity, the splitters are not physical bodies such as particles or drum, and the interaction of the particles with these volumes is neglected; splitters are virtual elements considered only in the post processing phase.

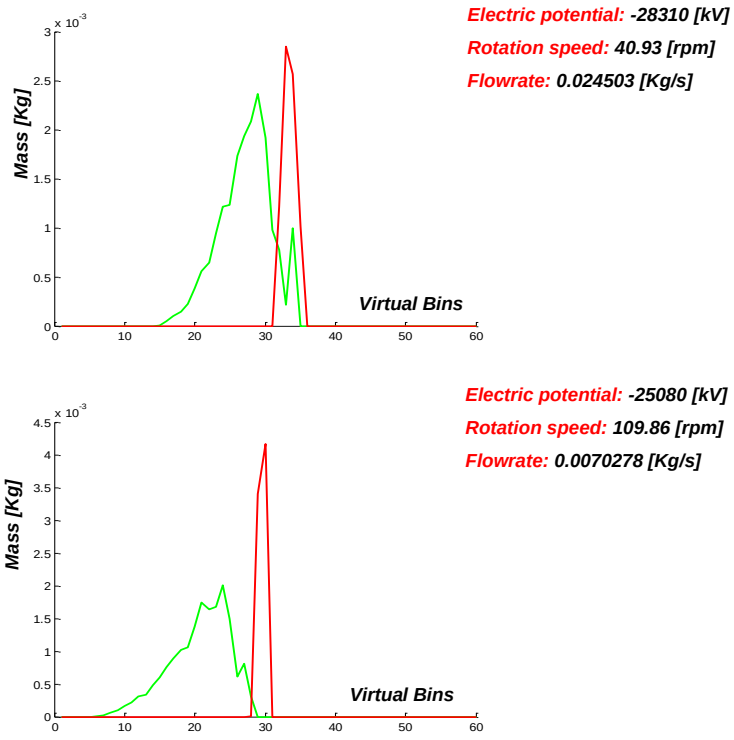


Figure 5: Example of two output mass distributions from CES simulator

7 Conclusion

We developed a DVI method for simulating multi-disperse granular flows, and we applied it to the case of a CES machine. The method has been implemented in our simulation software, and a parallel version has been developed. The computational bottlenecks of the simulation loop are the solution of the CCP problem and the collision detection; in case of granular flows with high number of particles, both problems can benefit from the adoption of GPU parallel processing.

References

[1] P.A. Cundall, O.D.L. Strack, "A discrete numerical model for granular assemblies", *Geotechnique*, 29(1): 47–65, 1979.

[2] E. Hairer, S.P. Nørsett, G. Wanner, *Solving Ordinary Differential Equations*, Springer, 2010.

[3] D.E. Stewart, J.C. Trinkle, "An implicit time-stepping scheme for rigid-body dynamics with inelastic collisions and Coulomb friction", *International Journal for Numerical Methods in Engineering*, 39: 2673–2691, 1996.

[4] M. Anitescu, A. Tasora, "An iterative approach for cone complementarity prob-

lems for nonsmooth dynamics", *Computational Optimization and Applications*, 47(2): 207–235, 2010.

[5] A. Tasora, M. Anitescu, "A convex complementarity approach for simulating large granular flows", *Journal of Computational and Nonlinear Dynamics*, 5(3): 1–10, 2010.

[6] A. Tasora, M. Anitescu, "A matrix-free cone complementarity approach for solving large-scale, nonsmooth, rigid body dynamics", *Computer Methods in Applied Mechanics and Engineering*, 200(5-8): 439 – 453, 2011, ISSN 0045-7825.

[7] A. Tasora, D. Negrut, M. Anitescu, "Large-scale parallel multi-body dynamics with frictional contact on the graphical processing unit", *Journal of Multi-body Dynamics*, 222(4): 315–326, 2008.

[8] H. Mazhar, T. Heyn, D. Negrut, "A scalable parallel method for large collision detection problems", *Multibody System Dynamics*, pages 1–19, 2011, ISSN 1384-5640, 10.1007/s11044-011-9246-y.

[9] J. Li, H. Lu, Z. Xu, Y. Zhou, "A model for computing the trajectories of the conducting particles from waste printed circuit boards in corona electrostatic separators", *Journal of Hazardous Materials, Volume 151, Issue 1*, 28 February 2008.

[10] M. Younes, A. Tilmatine, K. Medles, M. Rahli, L. Dascalescu, "Numerical Modeling of Conductive Particle Trajectories in Roll-Type Corona-Electrostatic Separators", *IEEE transactions on industry applications vol. 43, no. 5*, 2007.

[11] H. Lu, J. Li, J. Guo, Z. Xu, "Movement behavior in electrostatic separation: Recycling of metal materials from waste printed circuit board", *Journal of materials processing technology, Elsevier*, 2008.

[12] I. Critelli, A. Degiorgi, A. Tasora, M. Colledani, "A simulation model of Corona Electrostatic Separation (CES) for the recycling of Printed Circuit Boards (PCBs).", in *Symposium of urban mining*, 2014.

[13] I. Critelli, A. Tasora, A. Degiorgi, M. Colledani, "A multi-body simulation model for Corona Electrostatic Separator machine.", in *Computational, Engineering & Technology Conferences and Publications*, 2014.

[14] A. Shabana, *Multibody Systems*, John Wiley and Sons, New York, 1989.

[15] A. Iserles, H.Z. Munthe-Kaas, S.P. Nrsett, A. Zanna, "Lie-group methods", *Acta Numerica*, 9: 215–365, 2000.

[16] T. Heyn, M. Anitescu, A. Tasora, D. Negrut, "Using Krylov subspace and spectral methods for solving complementarity problems in many-body contact dynamics simulation", *International Journal for Numerical Methods in Engineering*, 95(7): 541–561, 2013, ISSN 1097-0207.

[17] J. Hoberock, N. Bell, "Thrust: A Parallel Template Library", 2010, URL `http://thrust.github.io/`, Version 1.7.0.

[18] G. Snethen, "XenoCollide Website", Sept. 2007, http://www.xenocollide.com.

[19] G. Snethen, "XenoCollide: Complex Collision Made Simple", in S. Jacobs (Editor), *Game Programming Gems 7*, pages 165–178. Charles River Media, 2008.