FASTER, MORE FLEXIBLE, PARTICLE SIMULATIONS: THE FUTURE OF MERCURYDPM

Anthony R Thornton[1,4], Mitchel Post[1], Luca Orefice[2,3], Paolo Rapino[4], Sudeshna Roy[1], Harmen Polman[1], M. Yousef Shaheen[1], Juan E. Alvarez Naranjo[1], Hongyang Cheng[1], Lu Jing [5,6], Hao Shi[1,4], Julius Mbaziira[1,4], Raïsa Roeplal[1], Thomas Weinhart[1,4]

[1]Multiscale Mechanics, Engineering Technology, MESA+, University of Twente
PO Box 217, 7500 AE Enschede, Netherlands

[2] Research Center Pharmaceutical Engineering (RCPE) GmbH, Inffeldgaße 13, 8010 Graz, Austria

[3] European Consortium on Continuous Pharmaceutical Manufacturing (ECCPM), 8010 Graz, Austria

[4] MercuryLab BV, Mekkelholtsweg 10, 7523 DE, Enschede, The Netherlands

[5]Department of Civil Engineering, The University of Hong Kong,
Haking Wong Building, Pokfulam Road, Hong Kong

[6]Department of Chemical and Biological Engineering, Northwestern University,
Evanston, IL 60208, USA

Keywords Granular Materials, DEM, DPM, MercuryDPM, Open-Source.

Abstract We focus on the main new developments underway in *MercuryDPM*. New features include deformable clusters (agglomerates), experimental coarse-graining, melting particles, particle-solid interactions, multi-resolution particle-fluid coupling, pressure-controlled Lees-Edwards boundaries, better hybrid openMP-MPI parallelisation, and more advanced STL/STEP readers for reading in industrial geometries. Some of these new features will be demonstrated for industrial relevant examples, such as industrial mixers, selective laser sintering, and a tunnel boring machine.

1 INTRODUCTION

We present several new features that have recently been developed in MercuryDPM. For a general introduction into *MercuryDPM*, we refer the reader to [27].

2 DEFORMABLE CLUSTERS (AGGLOMERATES)

This new feature of *MercuryDPM* allows the user to create agglomerates (or clusters) composed of individual elementary particles. Clusters are formed by compression, which cause the cluster particles to adhere to each together, as shown in figure 1, but their relative position is not fixed, making the agglomerates deformable and breakable. This feature is useful in simulations where such properties are required, such as particle breakage [1], tableting [2], granulation, simulating clay particles [3], etc.

A piecewise linear elasto-plastic-cohesive contact law [4] allows the particles to be in mechanical equilibrium despite having a finite overlap, and a finite tensile force is needed to pull them apart. The latter is what keeps agglomerates together, but also allows them to be deformed and broken when sufficiently strong external forces are exerted. As displayed in figure 1 clusters are mechanically stable before and after deformation. Computing their dynamics is computationally efficient, since there is no need to compute the agglomerate inertia tensor. This is not the case, if breakage would be modelled via multi-spheres.

Since the average overlap between particles within each cluster is known *before* the former is created, its size and porosity are *analytically* determined by the number of (monodispersed) particles per cluster, N, and their plasticity, ϕ, for spherical agglomerates (see figure 2).

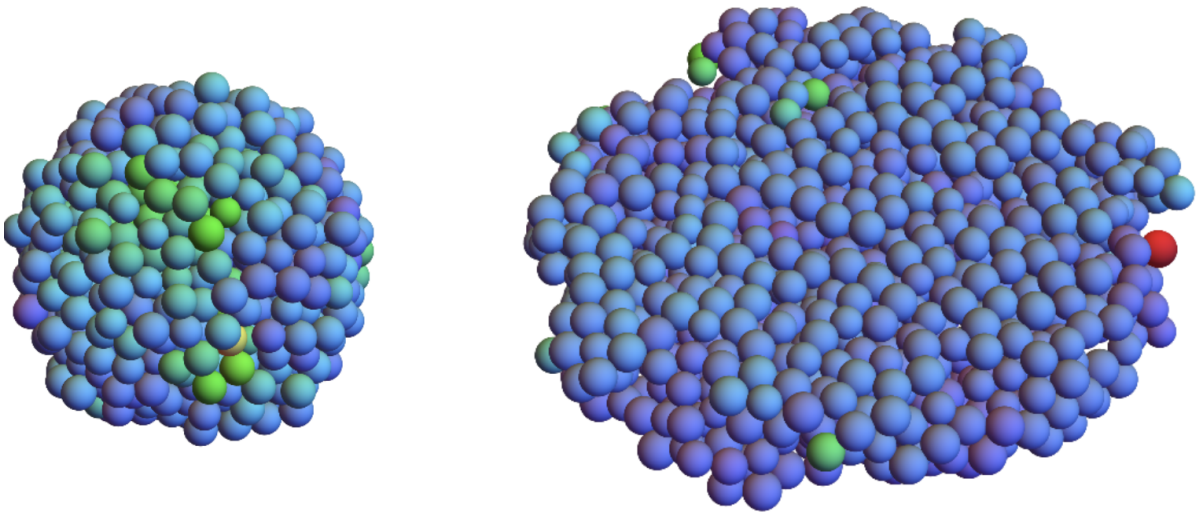

Figure 1 Cluster composed of N = 1000 monodispersed particles before (left) and after (right) uniaxial compression, colour indicating particles kinetic energy.

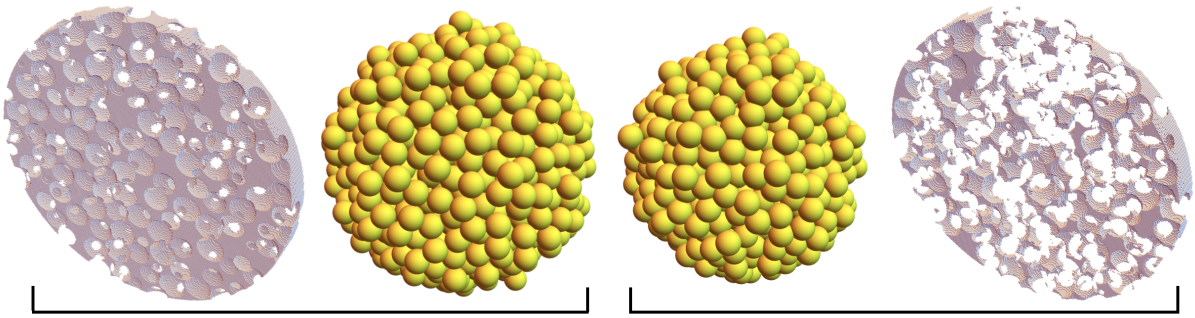

Figure 2 Spherical clusters of N = 1000 particles with $\phi = 0.1$ (left) and $\phi = 0.3$ (right). The different size due to bigger stable overlaps is evident when comparing their 3D image (in yellow) as well as their void fractions (in pink, plotted across a vertical slice).

3 EXPERIMENTAL COARSE-GRAINING

The *MercuryCG* toolbox was originally designed for coarse-graining data from DPM simulations [29], i.e. determining of continuum fields, but is applicable to discrete experimental data as well. The technique has two main advantages: first it results in experimental continuum fields that exactly conserve mass and momentum; secondly, it is always grid-free comparison between experimental and simulation data. Wrappers exist to analyse data from optical, Positron Emission Particle Tracking and Reflective Index Matched Scanning measurements, allowing for a much more detailed, grid-free comparison with simulation data.

The technique has now been applied to a rotating drum [30] and a split-bottom shear cell [31]. We will focus here on the split-bottom shear cell as an example: A shear band originates at the split between the two counterrotating basal rings and moves inwards, widening with increasing height. The goal of our experiments is to measure the shear band velocity profile at the free surface when the shear band is in steady-state. We use Particle Tracking Velocimetry (PTV) to track the (two-dimensional) position and velocity of the particles at the free surface.

To translate the discrete particle velocities to a continuous velocity field u, the particle positions and velocities are written into *MercuryDPM* data files and read into *MercuryCG*. The system is assumed to be ergodic, hence all data is assumed to represent steady state. This data is coarse-grained in space, resulting in a spatially-resolved velocity field [5,6]. We use a Gaussian smoothing function with a coarse-graining width of one particle diameter d_p. This way, we translate discrete velocity $\boldsymbol{v}_i$ data for each particle i to a continuous velocity field u as shown in figure 3.

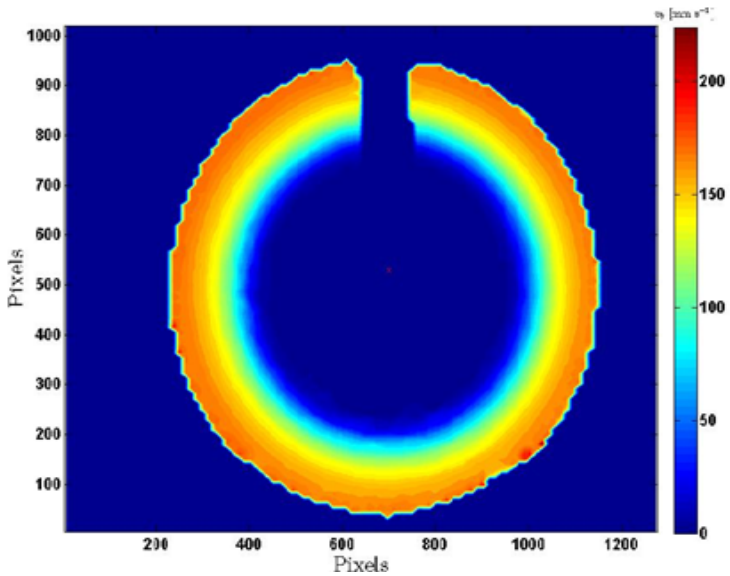

Figure 3 Coarse-grained velocity field as obtained using *MercuryCG* Toolbox by processing data from experiment of particles in split bottom shear cell, rotating at a frequency $f = 0.07\ s^{-1}$.

4 SINTERED PARTICLES

Solid-state sintering is a thermal treatment for bonding particles into a solid structure. Particles are sintered by heating particles beyond the glass temperature, but below the melting point of a material. This process is controlled by transport and diffusion of material along the particle's surface and volume, which leads to a reduction of the particle surface area. Solid-state sintering has three stages [8]. The first stage is neck formation: Matter from the particle is transported from regions of high chemical potential (contact region) to regions of low chemical potential (concave neck regions). In the second stage the diameters of the pores channels shrink until the pore structure changes. In the last stage isolated pores form. All stages are dominated by different transport mechanisms, and there is a strong dependence on temperature and initial particle size in the final stage of the process.

The solid-state sinter model in *MercuryDPM* describes the first stage, introducing a gradual increase in plastic overlap between particles at high temperatures. In [9], the model is applied to sintered polystyrene particles, and numerical and experimental indentation tests are executed and compared (figure 4).

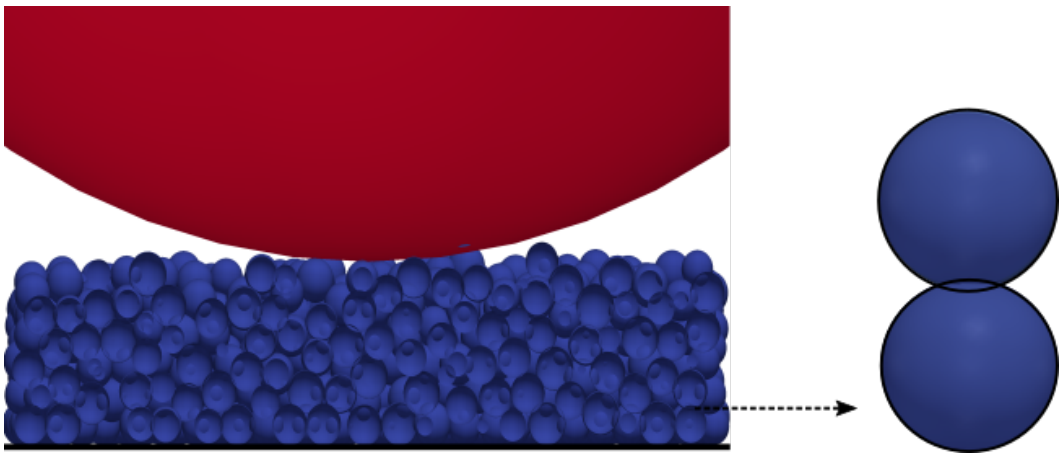

Figure 4 Vertical cut through centre of simulated sample during indentation.

5 MELTING PARTICLES

If particles are further heated, beyond their melting temperature, they start to melt.. The particles first melt at the surface, where the heat is applied, forming a melt layer that increases in thickness until the particles are fully melted. This process can now be modelled in *MercuryDPM* and was initially developed for additive manufacturing processes. In particular, we consider powder bed printing (PBP), where objects are produced by spreading successive layers of powdered material and hardening selected parts by partially or fully melting them with a laser. PBP processes are highly sensitive to the powder characteristics; therefore, the process parameters need to be optimised for each material. This is typically done by performing costly experimental trials, so developing a computational tool capable of capturing the stochastic nature of the process will help in reducing the amount of trials and thus lower manufacturing costs. In addition, particle-scale simulations of the spreading process can provide information on the powder layer behaviour and quality that is not accessible by experiments (porosity, particles segregation, etc). A parametric study of the influence of interparticle friction on the powder layer quality has been done by M.Y. Shaheen et al. [10].

The melting model is based on the model of Y. Gan et al. [11], which was applied to the formation and cyclic melting of faults during earthquakes. It is assumed that solid particles can melt during heating. On cooling these melt layers solidify potentially forming permanent bonds between the particles. The model was modified and extended to include thermal conduction, radiation and convection. Further extensions will include phase transformation and laser heat input modelling. Figure 5 shows a snapshot of particles partial melting, with particles diameter range 40-50 μm, and illustrates the heating and solidification of a new layer of particles.

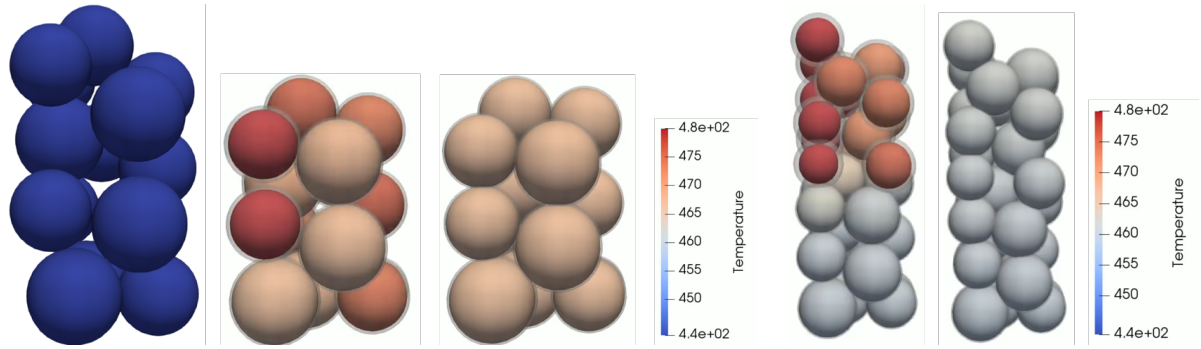

Figure 5 *Left:* Partial melting of particles,
Right: Adding a new layer of particles, heating it and allow it to cool down

6 PARTICLE-SOLID INTERACTION

MercuryDPM can now be coupled with oomph-lib. One of the new features this allows for is the coupling between solid particles and deformable, continuum bodies discretised by finite elements to simulate particle-structure/machinery interaction [12]. The versatile multi-physics *oomph-lib* library provides wrapper classes to solid elements (and the governing equations), where the coupling forces or kinematic constraints are interfaced to discrete particles.

For surface coupling, the oomph-lib geometry is mapped onto triangulated MercuryDPM walls, that can interact with the discrete particles. For each particle-wall interaction, the contact forces are added as external loads to the governing equations of *the oomph-lib* elements, and the wall positions and velocities in MercuryDPM are updated by the FEM solver.

Thanks to the multi-physics support of *oomph-lib*, applications of the coupled code include, but are not limited to: interactions between granular materials and deformable, fatigue-able structure/machinery [14], breakable discrete polygons [15], and fibre-particle mixtures [13].

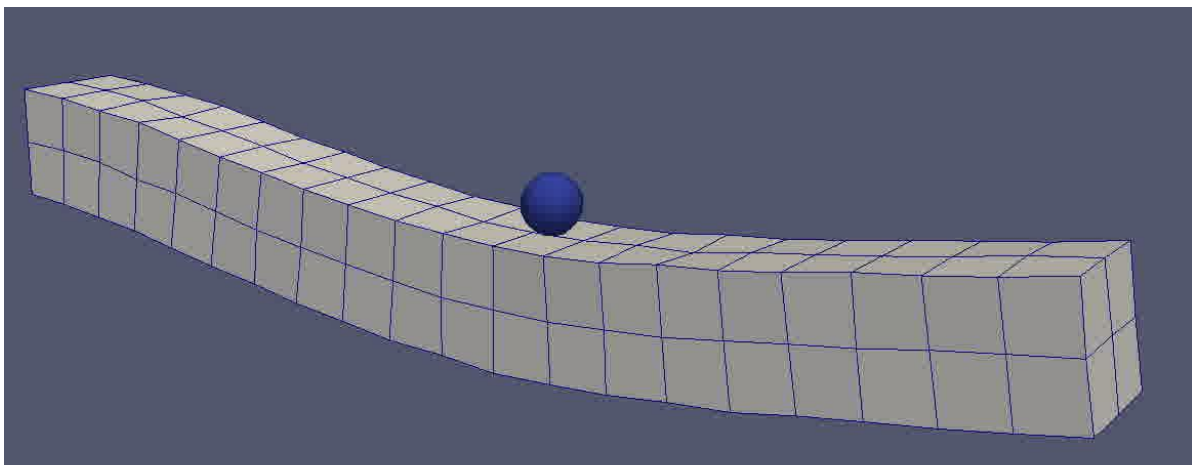

Figure 6 Single discrete particle interacting with a solid cantilever

7 MULTI-RESOLUTION PARTICLE-FLUID COUPLING

Coupling with oomph-lib also enables particle-fluid coupling, i.e. simulations of fluid particle mixtures (suspensions, dusty-gas, etc.). We are developing a multi-resolution coupling that adapts between full, partial and under-resolved situations, allowing particle-fluid coupling for arbitrarily large particle-size distributions, see figure 7.

Particle-fluid interactions play a major role in many industrial systems, for example pharmaceutical tableting. Tablets are often formed by compressing a powder in a die. In the die-filling process, air and agglomerated particles are interacting. While, during, and after compression; entrapped air pockets might result in capping (separated of either for the upper of lower part from the main body) and other failure modes.

In order to model tableting, *MercuryDPM* is coupled to oomph-lib [16], an open-source (FEM) package using a three resolutions approach. The Anderson and Jackson formulation [17] is used for under-resolved simulations. This introduces a voidage field, a measure for the fraction of total particle volume inside an element, to simulate the fluid flow. Coupling forces are be defined that specify the interaction between particles and fluid [18].

A different coupling method is semi-resolved, giving more detailed results than an under-resolved method but being much faster than a fully-resolved method. The voidage can be described as continuous function by coarse-graining the particles in space. The fluid elements can then simply evaluate this function at their location.

For fully resolved simulation there is a no-slip boundary condition for the fluid on the particle surface, and the coupling forces can be computed by integrating the pressure along the particle surface.

At the moment the three methods have been independent implemented; however, we are working on a fully adaptive methods where different particles sizes are resolved by different methods within one simulation. This will be fully adaptive and as the fluid grid resolving the method of coupling with change. Once this multi-resolution code is fully working we will be able to simulate particle-fluid simulations with arbitrary wide particle size distributions.

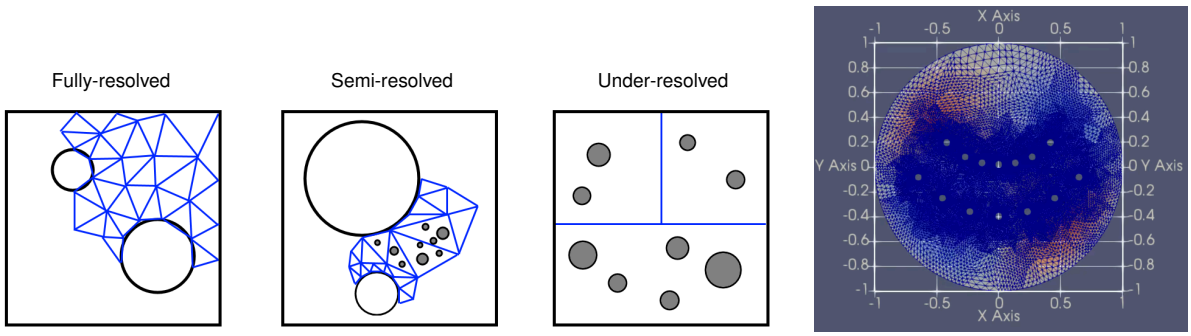

Figure 7 *Left*: Illustration of the fully-semi and under-resolved method, *Right*: fully-resolved simulation of sedimentation in a rotating drum, three cells of the wireframe correspond to one fluid element

8 PRESSURE-CONTROLLED LEES-EDWARDS BOUNDARIES

The `StressStrainControlBoundary` is a new boundary condition which combines a normal periodic boundary and a Lees-Edwards boundary [H1-3]. The user can specify a combination of targets for the stress and strain rate tensor.

The advantage of this boundary is the freedom of choosing the control parameters freely, allowing the user to specify both a target stress tensor, $\boldsymbol{\sigma}$, and a strainrate tensor, $\dot{\boldsymbol{\varepsilon}}$, as input parameters. For example, for a constant-stress uniaxial compression one would specify controlled normal stress, $\boldsymbol{\sigma} = (\sigma_{xx}\ 0\ 0,\ 0\ 0\ 0,\ 0\ 0\ 0)$ and leave $\dot{\boldsymbol{\varepsilon}}$ to be zero; for a constant-rate uniaxial compression, one would set $\boldsymbol{\sigma}$ to be zero with $\dot{\boldsymbol{\varepsilon}} = (\dot{\varepsilon}_{xx}\ 0\ 0,\ 0\ 0\ 0,\ 0\ 0\ 0)$. Similarly, for isotropic compression, which is mostly used in sample preparation to achieve a homogeneous initial packing, the user could set $\boldsymbol{\sigma} = (\sigma_{xx}\ 0\ 0,\ 0\ \sigma_{yy}\ 0,\ 0\ 0\ \sigma_{zz})$ or $\dot{\boldsymbol{\varepsilon}} = (\dot{\varepsilon}_{xx}\ 0\ 0,\ 0\ \dot{\varepsilon}_{yy}\ 0,\ 0\ 0\ \dot{\varepsilon}_{zz})$. Additionally, if the user would like to achieve a constant volume simple shear deformation mode, the strainrate tensor has to be changed to $\dot{\boldsymbol{\varepsilon}} = (0\ \dot{\varepsilon}_{xy}\ 0,\ 0\ 0\ 0,\ 0\ 0\ 0)$ with zero stress tensor. This could be further extended to a constant stress simple shear by specifying $\boldsymbol{\sigma} = (\sigma_{xx}\ 0\ 0,\ 0\ \sigma_{yy}\ 0,\ 0\ 0\ \sigma_{zz})$ to have the stress adapted while shearing at a constant shear rate. Note that the same element in target stress tensor and strainrate tensor cannot be set simultaneously, e.g. the user could not set $\boldsymbol{\sigma} = (\sigma_{xx}\ 0\ 0,\ 0\ 0\ 0,\ 0\ 0\ 0)$ with $\dot{\boldsymbol{\varepsilon}} = (\dot{\varepsilon}_{xx}\ 0\ 0,\ 0\ 0\ 0,\ 0\ 0\ 0)$ at the same time, otherwise the two control targets will conflict with each other, resulting in an invalid deformation mode.

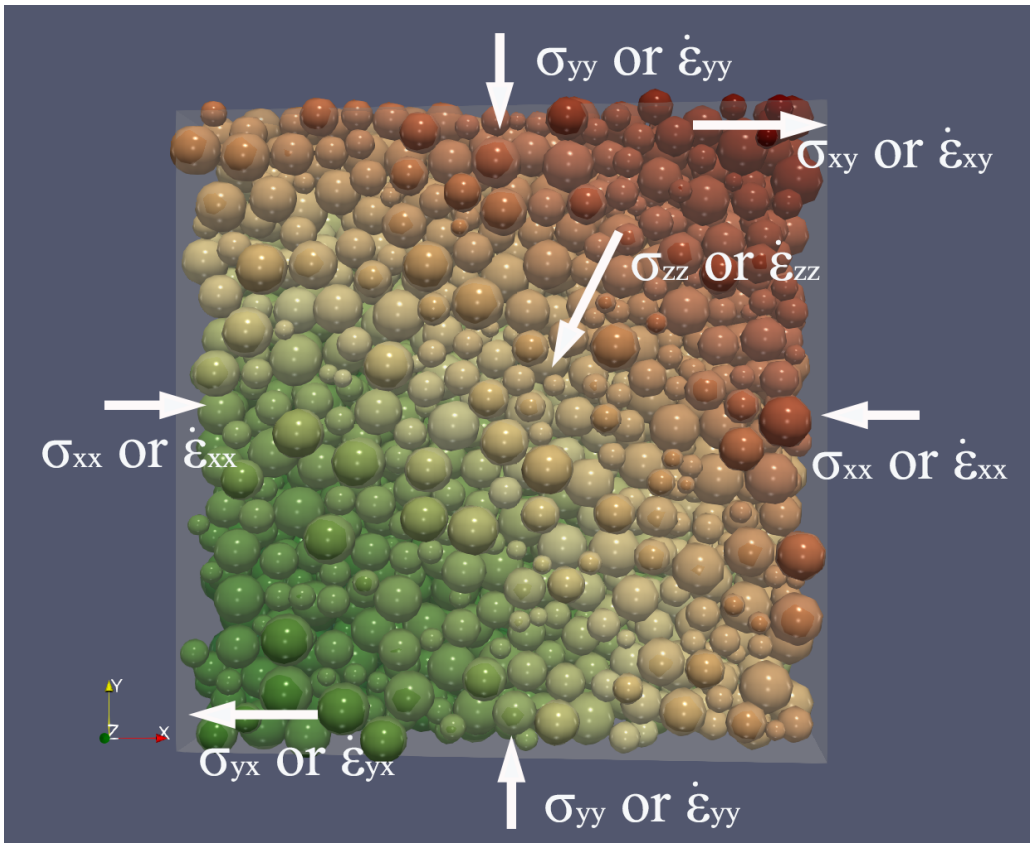

Figure 8 Stress-strain controlled periodic boundary condition in MercuryDPM and its possible deformation modes on a representative element volume (REV).

9 CHUTE FLOWS (INCLINED PLANES)

9.1 Controllable bottom roughness

A rough bottom is desired in included planes (chute flow) simulations to prevent slippage [19-21], but roughness is usually not controllable especially when flowing particles have various sizes (Fig. 9a; [22,23]). *MercuryDPM* provides several bottom generation methods, varying from flat/curved walls to immobile particles either randomly distributed or regularly placed on triangular lattice, in both 2D and 3D situations (Fig. 9). A parameter $R_a=(1+\epsilon)/(1+\lambda)$ characterises roughness for *any* arbitrarily generated bottom based on the bulk- to base-particle size ratio (λ) and local spacing (ϵ) (Fig. 9c inset; details in [21]), which effectively indicates whether slip occurs and, if so, how fast the slippage is (Fig. 9d). These advances make boundary roughness controllable in chute flow simulations using *MercuryDPM*.

9.2 MercuryCG for rough bottoms

MercuryCG has a unique feature facilitating treatment of rough bottoms, which evaluates stress fields near a boundary in a self-consistent way (details in [24]). High-resolution coarse graining allows "fuzzy" (very rough) boundaries to be defined [24].

9.3 Smart calibration of flow rule

MercuryDPM provides a smart approach for flow rule calibration (i.e. the H_{stop} curve [25]) in shallow granular flows. Following the routines indicated by left(up)-pointing triangular symbols in Fig. 9c, the demarcation curves between *flow* and *no flow* regimes can be automatically detected (filled circular symbols) within a single launch of the *MercuryDPM* program; see [19,20] for details.

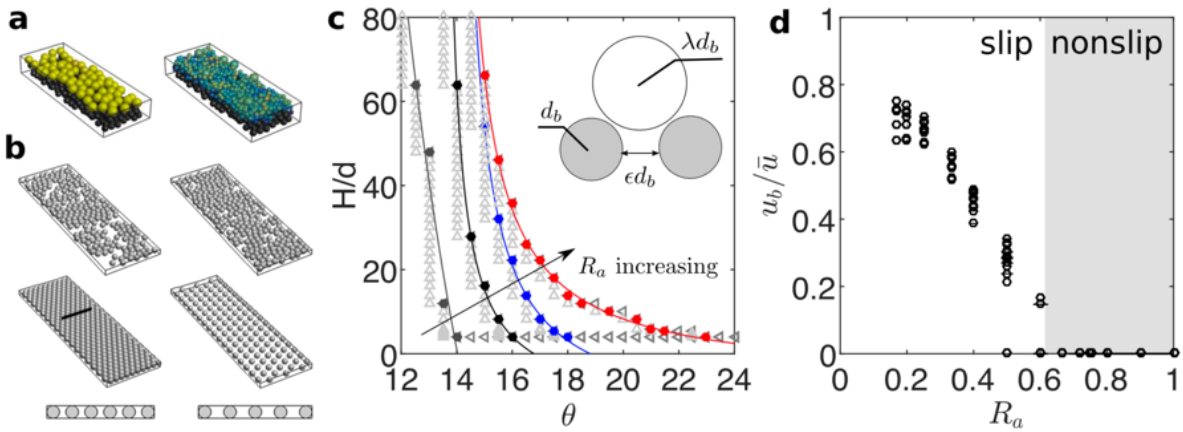

Figure 9 (a) Larger particles slip more easily over flat/bumpy bottom (brighter colour indicates higher velocity). (b) Generation of random- and ordered-packing bottoms in 3D and 2D. (c) Flow rule curves (below which no flow occurs) for different roughness (R_a) for 2D chute flows. Inset: 2D definition of R_a. (d) Normalized slip velocity vs R_a.

10 BETTER HYBRID OPENMP-MPI PARALLELISATION

Although *MercuryDPM* performs well for DPM simulations, due to advanced contact detecting and clever treatment of the walls, the computational power of a single processor (or thread) is limited. Thus, sequential DPM computations are limited to at most a few

million particles and a few minutes of process time. In order to finish the simulation in a reasonable time, for larger computations, parallel processing is required. Currently *MercuryDPM* uses a domain-decomposition based parallel computing algorithm utilising MPI [26]. The current domain decomposition is simple: the process domain is decomposed into n_x-by-n_y-by-n_z sub-domains of equal size (as specified by the user), and each processor computes the movement of particles in one sub-domain. To determine the contacts with particles from neighbouring sub-domains, a communication zone is established in the vicinity of the sub-domain boundaries, in which the processors communicate via MPI the location of the particles to their neighbours. This parallel computing algorithm can handle complex boundaries such as periodic boundaries, insertion/deletion boundaries and maser boundaries [28].

The current implementation works well for evenly distributed particle systems. However, for inhomogeneous systems, the workload of the processors is unbalanced, resulting in suboptimal scalability. The weaknesses of the current implementation will be addressed in the near future: one possibility to enhance load balancing is using a cyclic distribution with respect to particle identities (indices); a second possibility is to implement an adaptive mesh of differently-sized domains. These parallel-processing strategies for the MercuryDPM software package will be done using a combination of OPENMP and MPI for CPU parallel computing and OPENACC/CUDA for GPU computing.

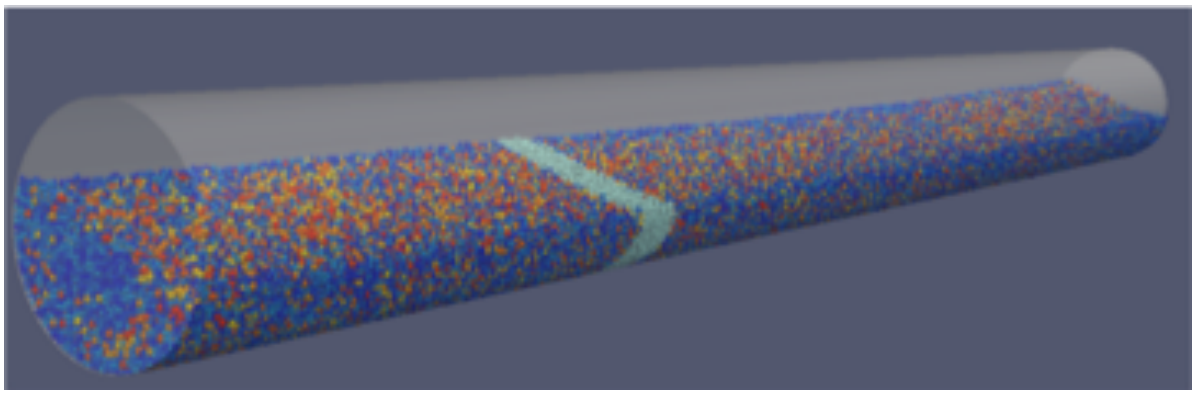

Fig 10: A parallel simulation of a rotating drum in MercuryDPM using 36 cores.
The light blue particles are computed on a single core. Image taken from [26].

11 STL/STEP READERS FOR READING IN INDUSTRIAL GEOMETRIES

MercuryDPM contains packages for curved geometries (cylinders, cones, coils, screws) which can be adjusted and combined to set up a geometry which satisfies the user's requirements. A new development in *MercuryDPM* is the possibility to import geometries as STL, STEP, or VTU files. This feature has been used in concrete mixing simulations, where a variety of custom designed rotating blades are required.

In binary STL files, the surface of the geometry is represented as collection of triangulated walls (Figure 10a). The STL/VTU reader `readTriangleWalls` imports the walls and sets the material properties, a scale factor (to convert from STL units to DPM units), velocity, and angular velocity with a corresponding centre of rotation for each set of imported walls. This provides the user with the freedom to assign different properties to selected parts of the

geometry. For example, in the case of the concrete mixer, the mixing blade and the housing has been imported separately so that the blades are allowed to rotate while the housing remains stationary. We are currently working on an STEP reader that keeps all the curvature information, leading to quicker and more accurate simulations.

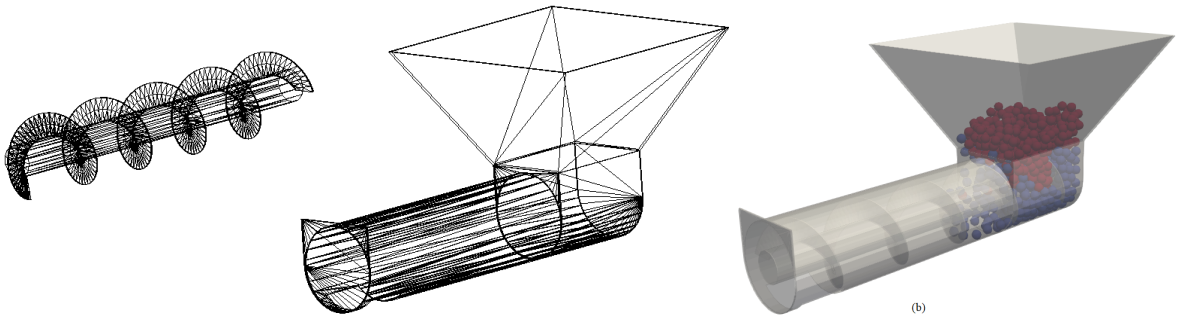

Figure 10 *Left/centre*: Illustration of two parts of a custom geometry created using CAD software which are exported to separate STL files. *Right*: The combined geometry, with particles inserted in *MercuryDPM*.

12 SUMMARY

A quickly growing developer's community has allowed the development of several new *MercuryDPM* features, some of which have been described in the above chapters. Many more are in development. In particular, we expect intense development of features on multiscale and particle-fluid coupling, as these are needed to solve the many unsolved multi-physics, multiscale problems in the granular field. Calibration of DPM simulation from bulk measurements is also a topic under development, where we aim to fully integrate the grain-learning package [Hong cite]. Another focus of our future work will be the development of user interfaces and parallel computing capabilities, with the aim of developing virtual prototypes of real, full-scale industrial machines.

ACKNOWLEGEMENTS

We acknowledge the support of the following grants: NWO VIDI 13472 Shaping segregation; NWO-TTW 15050 Multiscale Modelling of Agglomeration; NWO VIDI 16604 Virtual Prototyping of Particulate Processes.

REFERENCES

[1] Furukawa, R., Kadota, K., Noguchi, T., Shimosaka, A., Shirakawa, Y. (2017). DEM modelling of granule rearrangement and fracture behaviours during a closed-die compaction, *AAPS PharmSciTech*, 18, 2368-2377.

[2] Skelbæk-Pedersen, A., Vilhelmsen, T., Wallaert, V., Rantanen, J. (2019). Quantification of Fragmentation of Pharmaceutical Materials After Tableting, *J. Pharmaceutical Sciences*, 108, 1246-1253.

[3] Mašín, D. (2013). Double structure hydromechanical coupling formalism and a model for unsaturated expansive clays, *Engineering Geology*, 165, 73-88.

[4] Luding, S. (2008). Cohesive frictional powders: contact models for tension, *Granular Matter*, 10, 235-246.

[5] Weinhart, T., Hartkamp, R., Thornton, A.R., Luding, S. (2013). Coarse-grained local and objective continuum description of three-dimensional granular flows down an inclined surface. *Physics of Fluids* 25(7), 070605

[6] Weinhart, T., Thornton, A.R., Luding, S., Bokhove, O. (2012). From discrete particles to continuum fields near a boundary, *Granular Matter* 14(2), 289-294.

[7] Tunuguntla, D. R., Thornton, A. R., Weinhart, T. (2016). From discrete particles to continuum fields: Extension to bidisperse mixtures. *Comp. Part. Mech.* 3(3), 349-365.

[8] Blendell, J. E. (2001). Solid-state Sintering. *Encyclopedia of Materials: Science and Technology*, 8745-8750.

[9] Fuchs, R., Weinhart, T., Ye, M., Luding, S., Butt, H. J., Kappl, M. (2017). Initial stage sintering of polymer particles–Experiments and modelling of size-, temperature-and time-dependent contacts. *EPJ Web of Conferences* (140), 13012.

[10] Shaheen, M.Y., Thornton, A.R., Luding, S., Weinhart, T. (2017). "Discrete particle simulation of the spreading process in additive manufacturing" *Proc. 8th Int. Conf. Discrete Element Methods*.

[11] Gan, Y., Rognon, P. and Einav, I. (2012). Phase transitions and cyclic pseudotachylyte formation in simulated faults. *Philosophical Magazine*, Vol. 92, pp. 3405-3417.

[12] Cheng, H., Yamamoto, H., Guo, N., Huang, H. (2017). A simple multiscale model for granular soils with geosynthetic inclusion. In: Li, X., Feng, Y., Mustoe, G. (eds.), *Proc. 7th Int. Conf. Discrete Element Method*s, pp. 445–453. Springer.

[13] Cheng, H., Yamamoto, H., Thoeni, K., Wu, Y. (2017). An analytical solution for geotextile-wrapped soil based on insights from DEM analysis. *Geotext. Geomembr.* 45(4), 361–376

[14] Dratt, M., Katterfeld, A. (2017). Coupling of FEM and DEM simulations to consider dynamic deformations under particle load. *Granular Matter* 19(3), 49.

[15] Ma, G., Zhou, W., Chang, X.L., Yuan, W. (2014) Combined FEM/DEM modeling of triaxial compression tests for rockfills with polyhedral particles. *Int. J. Geomech.* 14(4), 1–12

[16] Heil, M., Hazel, A.L. (2006). oomph-lib – an object-oriented multi-physics finite-element library. In Hans-Joachim Bungartz and Michael Schäfer, Editors, *Fluid-Structure Interaction*, pages 19–49, Springer Berlin Heidelberg.

[17] Anderson, T.B., Jackson, R. (1967). Fluid mechanical description of fluidized beds. Equations of motion. Industrial & Engineering Chemistry Fundamentals, 6(4):527–539

[18] Davies, C.N. (1979). Particle-fluid interaction. *J. Aerosol Science*, 10(5):477 – 513.

[19] Weinhart, T., Thornton, A.R., Luding, S., Bokhove, O. (2012). Closure relations for shallow granular flows from particle simulations, *Granular Matter* 14, 531–552.

[20] Thornton, A.R., Weinhart, T., Luding, S., Bokhove, O. (2012). Frictional dependence of shallow-granular flows from discrete particle simulations, *Eur. Phys. J. E* 35.

[21] Jing, L., Kwok, C.Y., Leung, Y.F., Sobral, Y.D. (2016). Characterization of base roughness for granular chute flows, *Phys. Rev. E*. 94, 052901.

[22] Jing, L., Kwok, C.Y., Leung, Y.F., Sobral, Y.D. (2017). Effect of geometric base roughness on size segregation, *EPJ Web Conf*. 140, 03056.

[23] van der Vaart, K., Thornton, A.R., Johnson, C.G., Weinhart, T., Jing, L., Gajjar, P., Gray, J.M.N.T., Ancey, C. (2018). Breaking size-segregation waves and mobility feedback in dense granular avalanches, *Granular Matter* 20.

[24] Weinhart, T., Thornton, A.R., Luding, S., Bokhove, O. (2012). From discrete particles to continuum fields near a boundary, *Granular Matter* 14(2), 289-294.
[25] Pouliquen, O. (1999) Scaling laws in granular flows down rough inclined planes, *Physics of Fluids* 11, 542–548.
[26] van Schrojenstein Lantman, M.P. (2019). A study on fundamental segregation mechanisms in dense granular flows*, Ph.D. Thesis*, University of Twente
[27] Weinhart, T., Post, M., Denissen, I.F.C., Grannonio, E., Barbosa, J., den Otter, W., Thornton, A.R. (2019). Fast, flexible particle simulations: An introduction into MercuryDPM. *Proc. 8th Int. Conf. Discrete Element Methods*.
[28] Weinhart, T., Tunuguntla, D.R., van Schrojenstein Lantman, M.P., Denissen, I.F.C., Windows-Yule, C.R., Polman, H., Tsang, J.M.F., Jin, B., Orefice, L., van der Vaart, K., Roy, S., Shi, H., Pagano, A., den Breeijen, W., Scheper, B.J., Jarray, A., Luding, S., Thornton, A.R. (2017). MercuryDPM: Fast, flexible particle simulations in complex geometries Part B: Applications, *Proc. Int. Conf. Particle-Based Methods*.
[29] Miller, T, Rognon, P., Einav, I. (2013). The Stadium Shear Device: A Novel Apparatus For Studying Dense Granular Flows. *AIP Conference Proceedings*. 1542. 483.
[30] Weinhart, T., Tunuguntla, D. R., van Schrojenstein Lantman, M., van der Horn, A. J., Denissen, I. F. C., Windows-Yule, C. R. K., de Jong, A. C., Thornton, A. R. (2016), MercuryDPM: A fast and flexible particle solver Part A: Technical Advances, *Proc. 7th Int. Conf. Discrete Element Methods*.
[31] Roy, S., Scheper, B., Polman, H., Thornton, A.R., Tunuguntla, D.R., Luding, S., Weinhart, T. (2019). Surface flow profiles for dry and wet granular materials by Particle Tracking Velocimetry; the effect of wall roughness. *European Physical Journal E*, 42: 14