The 7th International Conference on Discrete Element Methods
August 1 to August 4, 2016 Dalian, China
Paper Number: G010577

MercuryDPM: A fast and flexible particle solver
Part A: Technical Advances

T Weinhart, DR Tunuguntla[1], M van Schrojenstein-Lantman, AJ van der Horn, IFC Denissen, CR Windows-Yule, AC de Jong, AR Thornton

Abstract. *MercuryDPM* is an open-source particle simulation tool -- fully written in C++ -- developed at the University of Twente. It contains a large range of contact models, allowing for simulations of complex interactions such as sintering, breaking, plastic deformation, wet-materials and cohesion, all of which have important industrial applications. The code also contains novel complex wall generation techniques, that can exactly model real industrial geometries. Additionally, *MercuryDPM*s' state-of-the-art built-in statistics package constructs accurate three-dimensional continuum fields such as density, velocity, structure and stress tensors, providing information often not available from scaled-down model experiments or pilot plants. The statistics package was initially developed to analyse granular mixtures flowing over inclined channels, and has since been extended to investigate several other granular applications. In this proceeding, we review these novel techniques, whereas its applications will be discussed in its sequel.

Keywords: Discrete Particle Simulations, Open-Source Software, Contact Modelling, Micro-Macro, Coarse-Graining.

1. Background

For over decades, numerous numerical methods for quantifying particle-particle interactions have been formulated and utilised to investigate granular

[1] Corresponding author: d.r.tunuguntla@utwente.nl

Multiscale Mechanics, Engineering Technology/ MESA+, University of Twente, P.O. Box 217, 7500 AE Enschede, Netherlands

dynamics [1]. These include methods like cellular automata, direct simulation Monte Carlo method (DSMC), discrete particle/element method (DPM/DEM).

In lattice-based cellular automata (CA), which is a simple deterministic technique, the positions of the particles are computed using equations determined through experiments. These equations can either be physics-based or from rule-based mathematics. The technique utilises a grid like structure (lattice), where the physical domain is discretised into several cells, where each cell corresponds to one of the defined number of states, see [2]. CA has been used, previously, to understand granular flows in silos [3, 4], sand piles [2, 5], annular shear cells [6, 7], over inclined channels [8], including phenomenon such as particle segregation [9], rotating drums [10, 11], in hopper flows with irregular particle shapes [12]. Despite being a computationally fast technique, the granular dynamics are qualitative alone thereby lacking quantitative accuracy.

As an alternative, granular dynamics have also been predicted using a Direct simulation Monte Carlo (DSMC) approach where the evolution of the particulate system is based on the particle collision probability which is proportional to the relative velocity of the particles, see [13, 14]. The technique, first proposed by [13], is relatively simple to implement and computationally efficient. Generally, DSMC has been successfully employed to simulate dilute granular flows in both two- and three-dimensional vibrated containers [15] and Couette flows [16]. Moreover, recent studies have developed modified DSMC algorithms [17, 18] which are computationally more efficient than the traditional DSMC methods and consistent with the DEM results [18]. However, it is the discrete element method that is widely used because of its various advantages and is a promising alternative for modelling bulk handling materials.

In Discrete Particle Method (DPM), one models the dynamics of *each individual component* of a particulate system by solving Newton's laws of motion [19]. The method is computationally more expensive than continuum methods, due to the huge amount of particles involved. However, with recent advances in the state-of-the-art computing techniques [21-25] discrete particle simulations have advanced to a stage, where they are now being successfully utilised as design tools in some industries [20, 21]. Due to this increasing demand for utilising DPM, a few groups, in academia and industry, around the world have developed their own particle solvers. In the following, we illustrate some of the recent advances made in our cutting-edge solver called *MercuryDPM*.

2. *MercuryDPM* : Fast, flexible particle solver

MercuryDPM [26-28] is an open-source package designed for emulating the dynamics of granular mixtures. Basically, it is a discrete particle solver which computes the dynamics of particles or atoms when subjected to forces and torques.

These forces and torques usually result from the external body forces, (e.g. gravity, electrical fields, magnetic fields, etc.), surrounding objects (side-walls, conveyors, hoppers) or inter-particle interactions. For granular materials, the forces are typically contact forces due to various type of particle interactions. Examples include elastic, viscous, frictional, plastic, cohesive, capillary and many more varied forms of interactions. Similarly, for materials on molecular scale, forces predominantly arise from interaction potentials (e.g. Lennard-Jones).

Often the method used in several simulation packages is referred to as the Discrete Element Method (DEM), which was originally designed for geotechnical applications. However, as MercuryDPM is designed for simulating particles, with special focus laid upon contact models, optimised contact detection for largely varying particle sizes, and in-built advanced data-analysis tools, we prefer the more general name Discrete Particle Method (DPM). Originally, MercuryDPM was developed for simulating granular flows over inclined channels, and has since evolved to simulate several other granular applications, which range from geological applications such as modelling of cinder cone creation to industrial applications such as optimising vibrofluidised waste separation devices, flows in silos, screw feeders, conveyor belts, drum mixers, etc.

Based on the range of applications MercuryDPM can be involved in, our recently established sister company *MercuryLab* (http://MercuryLab.org) offers a variety of training and consultancy packages for *MercuryDPM*. Furthermore, we have utilised *MercuryDPM* to optimise the efficiency of bulk handling processes for several companies. Our typical modus operandi consists of visiting your company, understanding your current device (handling equipment) and optimising the process under the constraints you specify. Previous clients have achieved a more than ten-fold increase in the efficiency of their redesigned equipment thanks to the intervention of *MercuryLab*.

3. Features

Since its establishment, *MercuryDPM* has advanced in several areas. Not only has it received worldwide attention but more importantly it has gained many novel features. These include, but are by no means limited to:

The hierarchical grid [29]: This neighbourhood search algorithm effectively computes interaction forces, even for a wide range of polydisperse particle size distributions, see Fig.1.

Analysis tools for understanding the bulk properties [30, 31]: *MercuryDPM* has a built-in advanced statistics package, *MercuryCG*, which constructs macroscopic continuum fields such as density, velocity, structure and stress tensors everywhere inside a 3-dimensional apparatus. This yields information that

is often not available from scaled-down model experiments or pilot plants, see Fig.4.

Access to continuum fields in real time: *MercuryDPM* allows you to see the bulk properties of your system 'live' as the code runs, allowing feedback between the system geometry and the flowing material. An example one could think of is a pressure-release wall, where the motion of the wall is determined by the macroscopic pressure induced by particle collisions, i.e. the wall moves due to the change in the system pressure and not because of some predefined function.

Contact laws for granular materials: Many granular contact forces are implemented, including elastic (linear or Hertzian), elastoplastic, cohesive (dry or wet), frictional (sliding/rolling/torsion), as well as temperature-, pressure-, and time-dependent forces (sintering). These forces can be combined together to model highly complex behaviour, enabling *MercuryDPM* to simulate complex industrial processes like sintering, tabletting, etc.

Multi-species systems: Mixtures can be modelled by grouping particles and walls into *species*, with different contact behaviour and particle properties defined for each species. This feature is hidden for the basic use of the code; however, it can be enabled with a single line of code.

Documentation, tutorials, training courses, and consultancy packages*:* Prospective users can easily learn how to read and write *MercuryDPM* simulation code using the online documentation and tutorials, available at http://docs.mercurydpm.org. Additionally, we provide training courses for the software (see http://mercurylab.org for details), or even complete consultancy services (no need for you to install and use the code at all).

Simple C++ implementation: All the features in *MercuryDPM* are implemented with the help of a series of C++ classes. This not only makes the code flexible, but also easy to use. Moreover, this enables the user to generate advanced applications with only a few lines of code.

Handlers: Besides simple implementation, the code provides handlers for particles, walls, boundaries and contact laws. Thus, allowing for each class object type to have a common interface. Note that individual class objects can have completely different properties. This also makes it easier for the user to create new objects.

Complex walls: One of the most sought after feature of the code is its support for complex geometries. Besides the simple flat wall geometry, MercuryDPM constructs, for example, axially-symmetric, polyhedral and helical screw walls in a very efficient manner, see Fig.3. With this, we can precisely simulate more complex industrial geometries, while most other codes approximate complex walls with flat triangular sections, which can lead to significant errors.

Specialised classes: *MercuryDPM* contains multiple pre-defined classes for common system geometries such as chute flows, silos, vibrofluidised systems and rotating tumblers. As such, anyone working with these widely-used systems can begin running simulations with minimal effort.

Demos and self-test suite: Every installation of *MercuryDPM* is provided with loads of self-tests and demo codes, which serve two sole purposes: 1) they allow the user to test and check the compatibility of both new and old features. Hence, ensuring that the code runs optimally and remains free of bugs; 2) The demos serve as tutorials for new users and teaches them to do different tasks using several features of MercuryDPM.

Interface to other particle simulation codes: The restarting interface is used to provide an interface to load external data (including experimental data [36], other particle codes such as EDEM [37], and molecular dynamics [38]) into *MercuryDPM* for advanced post-processing with *MercuryCG*.

Visualisation: The simulation output can be easily visualised using the open-source free packages ParaView and VMD (Visual Molecular Dynamics).

4. Advanced analysis with *MercuryCG*

To take full advantage of the data produced by *MercuryDPM* simulations, the macroscopic quantities have to be extracted from the available discrete particle data. This is done by our in-house statistical toolbox *MercuryCG*, which is one of the most significant parts of the *MercuryDPM* software package. *MercuryCG* can either be run as a post-processing step, or during the simulation. Where traditional techniques cannot accurately produce continuum fields near the edges of a system, *MercuryCG* uses advanced mathematics to allow continuum information to be extracted from microscopic data even within one particle diameter of a boundary [30].

An exemplary output of our statistics package is shown for a jet of particles impacting a rough inclined plane in Fig. 2. The flow height is obtained by assuming constant flow density across the height, and that flow is uniform and steady enough to have a lithostatic stress profile. Thereby, the flow height could be computed form the depth-averaged stress and density, as shown in Fig. 4. Given the height is determined, a depth-averaged velocity and the Froude number can be computed. A Froude number larger than unity denotes supercritical flow; otherwise the flow is subcritical. This allows us to determine the location of the shock (black line in right panel of Fig. 4). The statistics tool has been successfully utilised to analyse several other applications, e.g. [33,34,35].

5. Interested in *MercuryDPM*?

For more details regarding *MercuryDPM,* visit our website http://mercurydpm.org/. Moreover, you could also subscribe to our mailing list for

more information and latest update notifications. To do so, simply send an email to listserv@lists.utwente.nl with subject: `subscribe` and body: `MERCURYDPM-USER <your full name>`. This is a low volume mailing list and typically you will receive no more than one e-mail a month. The code itself is available through a public *svn* repository and details of how to obtain and install it can be found on the website, http://MercuryDPM.org

Originally, *MercuryDPM* was developed as a code purely for research purposes, to meet the requirement of a tool that was not available in any existing simulation codes. Since then it has grown and gained many external users in academia, research and industry. Therefore, we decided to make it freely available to both industry and academia via an open-source release (using the BSD license).

Financial Support

We acknowledge and thank the support of several research grants for their financial support to develop MercuryDPM.

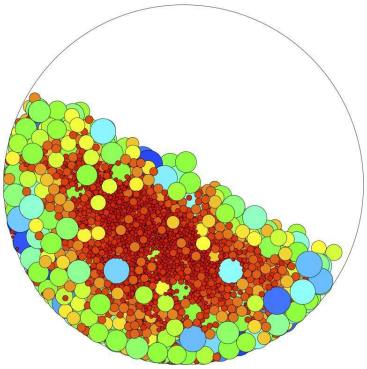

Fig. 1: Particle segregation in a mixture rotating in a drum. Note the wide particle size distribution. The varying colour denotes different particle size, with red and blue being the smallest and the largest respectively.

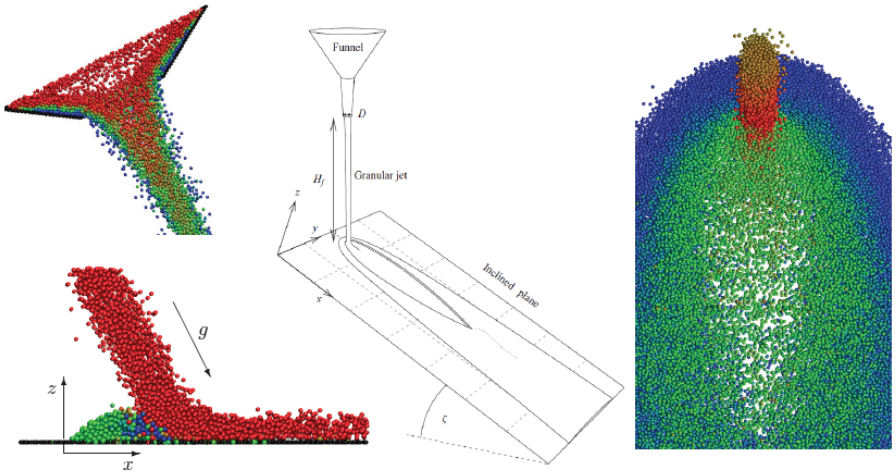

Fig. 2: Top left illustrates the flow through a hopper, whereas the bottom left focusses on the impact region. In the middle, we have a schematic of the original experiment and in the right, is the top view of the full simulation (~500k particles). Black particles indicate fixed particles; all other colours indicate speed, with blue low and red high speed.

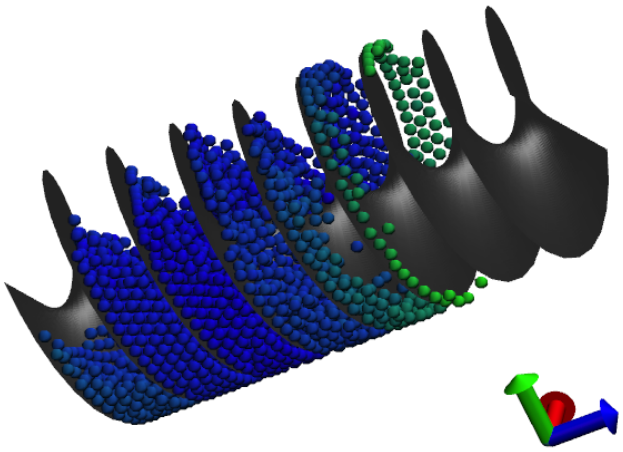

Fig. 3: Illustrates a simulation of a screw conveyor simulation, colour coded by the particle's velocity. The screw transports the particles through the tube..

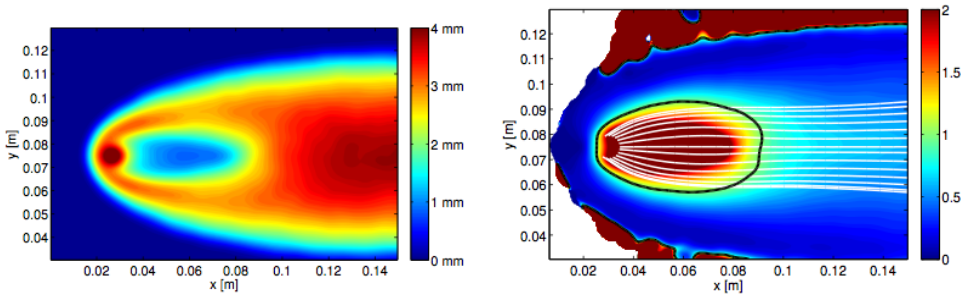

Fig. 4: Coarse grained macroscopic fields created using MercuryDPM's own statistics toolbox (MercuryCG). On the left, we have the flow height in millimetres with its corresponding local Froude number shown in the right. The white lines indicate velocity streamlines whereas the black line indicates the location of a granular jump/shock.

References

1. Herrmann and Luding, Continuum Mechanics and Thermodynamics, 10(4):189-231, 1998.
2. E Goles. Sandpile automata. In Annales de l'IHP Physique theorique, volume 56, pp. 75-90. Elsevier, 1992.
3. Karolyi et al, Europhysics Letters, 44(3):386, 1998.
4. Tejchman. Simulations of flow pattern with cellular automaton. In Confined Granular Flow in Silos, pp. 455-492. Springer, 2013.
5. Alonso and Herrmann, Physical Review Letters, 76(26):4911, 1996.
6. Jasti and Higgs III, Granular Matter, 12(1):97-106, 2010.
7. Marinack and Higgs, Powder Technology, 2015.
8. LaMarche et al., Granular Matter, 9(3-4):219-229, 2007.
9. Marks and Einav, Granular Matter, 13(3):211-214, 2011.
10. T Yanagita, Physical Review Letters, 82(17):3488, 1999.
11. Santomaso et al., Chemical Engineering Science, 90:151-160, 2013.
12. Baxter and Behringer, Physical Review A, 42(2):1017-1020, 1990.
13. GA Bird, Molecular gas dynamics and the direct simulation of gas flows, Claredon, Oxford (1994).
14. M□uller and Herrmann. In Physics of Dry Granular Media, pp. 413-420. Springer, 1998.
15. M□uller et al., Simulations of vibrated granular media in 2d and 3d. 1997.
16. Reyes et al. Physical Review E, 83(2):021302, 2011.
17. Du et al. Chemical Engineering Science, 66(20):4922-4931, 2011.
18. Pawar et al., Chemical Engineering Science, 105:132-142, 2014.
19. Windows-Yule et al., Computational Particle Mechanics, 3(3):311-332, 2016.
20. Feng et al., Engineering Computations, 30(2):157-196, 2013.
21. S Luding, European Journal of Environmental and Civil Engineering, 12(7-8):785-826, 2008.

22. Ketterhagen et al., Journal of Pharmaceutical Sciences, 98(4):442-470, 2009.
23. Guo and Curtis, Annual Review of Fluid Mechanics, 47:21-46, 2015.
24. Windows-Yule et al., Computational Particle Mechanics, 3(3): 311-332, 2016.
25. Zhu et al., Chemical Engineering Science, 62(13):3378-3396, 2007.
26. Zhu et al., Chemical Engineering Science, 63(23):5728-5770, 2008.
27. Thornton et al., Newsletter EnginSoft 10(1), 48–53 (2013)
28. Thornton et al., Proceedings 6th InternationalConference on Discrete Element Methods (DEM) pp. 393–399 (2013)
29. Thornton, A.R. and Weinhart, T. et al., Mercurydpm. http://MercuryDPM.org/ (2009-2016)
30. Ogarko and Luding, Computer Physics Communications, 2012. 183(4): p. 931-936.
31. Weinhart et al., Granular Matter, 14(2):289-294, 2012.
32. Tunuguntla et al., Computational Particle Mechanics, 3(3): 349-365, 2016.
33. Thornton et al., Eur Phys J E Soft Matter, 2012. 35: p. 9804.
34. Weinhart et al., Granular Matter, 2012. 14(4): p. 531-552.
35. Thornton et al., International Journal of Modern Physics C, 2012. 23(08): p. 1240014.
36. Tunuguntla et al., Journal of Fluid Mechanics, 749:99-112, 2104.
37. Windows Yule et al., Physical Review Letters, 112, 2014.
38. Weinhart et al., Powder Technology, 293:138-148, 2015.
39. Hartkamp et al., Journal of Chemical Physics, 137(4), 2012.