



# A GPU-accelerated simulator for the DEM analysis of granular systems composed of clump-shaped elements

Ruochun Zhang<sup>1</sup> · Colin Vanden Heuvel<sup>1</sup> · Alexander Schepelmann<sup>2</sup> · Arno Rogg<sup>3</sup> · Dimitrios Apostolopoulos<sup>4</sup> · Samuel Chandler<sup>4</sup> · Radu Serban<sup>1</sup> · Dan Negrut<sup>1</sup>

Received: 6 July 2023 / Accepted: 6 November 2023

© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2024

## Abstract

This paper discusses the use of the discrete element method (DEM) to simulate the dynamics of granular systems made up of elements with nontrivial geometries. The DEM simulator is GPU accelerated and can handle elements whose shape is defined as the union with overlap of diverse sets of spheres with user-specified radii. The simulator can also handle complex materials since each sphere in an element can have its own Young's modulus  $E$ , Poisson ratio  $\nu$ , friction coefficient  $\mu$ , and coefficient of restitution CoR. To demonstrate the simulator, this paper introduces a “digital simulant” (DS), a replica of the GRC-1 lunar simulant. The DS follows an element size distribution similar but not identical to that of GRC-1. The predictive attributes of the simulator are validated via several numerical experiments: repose angle, cone penetration, drawbar pull, and rover incline-climbing tests. Subsequently, a sensitivity analysis is carried out to gauge how the slope vs. slip curves change when the element shape, element size, and friction coefficient change. The paper concludes with a VIPER rover simulation that confirms a recently proposed granular scaling law. The simulation involves more than 11 million elements composed of more than 34 million spheres of different radii. The simulator works in the Chrono framework and can utilize two GPUs concurrently. The GPU code for the simulator and all numerical experiments discussed are open-source and available on GitHub for reproducibility studies and unfettered use and distribution. The DEM simulator introduced here: can handle complex particle geometry; can capture material breakage; can handle user-defined contact forces; allows for one DEM element to have different material properties in different regions of the element; is fast and runs on commodity hardware; can leverage two GPU cards simultaneously; it has a Python interface; and, it works in co-simulation mode with and augments Chrono, an established multi-body dynamics platform that supports the physics-based simulation of complex mechanical systems.

**Keywords** Discrete element method · Complex geometry · GRC-1 · Physics-based simulation · Extraterrestrial rover · GPU computing

---

✉ Dan Negrut  
negrut@wisc.edu

Ruochun Zhang  
rzhang294@wisc.edu

Colin Vanden Heuvel  
colin.vandenheuvel@wisc.edu

Alexander Schepelmann  
alexander.schepelmann@nasa.gov

Arno Rogg  
arno.rogg@nasa.gov

Dimitrios Apostolopoulos  
da1v@protoinnovations.com

Samuel Chandler  
samxchandler@protoinnovations.com

Radu Serban  
serban@wisc.edu

<sup>1</sup> Department of Mechanical Engineering, University of Wisconsin-Madison, 1513 Engineering Dr, Madison, WI 53706, USA

<sup>2</sup> Mechanisms and Tribology Branch, NASA Glenn Research Center, 23-217, 21000 Brookpark Road, Cleveland, OH 44135, USA

<sup>3</sup> VIPER Robotic Systems, KBRwyle, NASA Ames Research Center, MS 269-3, Moffett Field, CA 94035, USA

<sup>4</sup> Terrain Manipulation and Wheels, ProtoInnovations, 100 43rd St, Pittsburgh, PA 15201, USA

## 1 Introduction

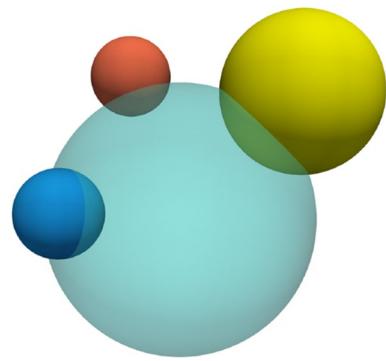
The discrete element method (DEM) is a computational approach used to predict the mechanical behavior of granular materials [1]. DEM keeps track of the motion of each individual element and models the elements' mutual interactions in a fully resolved fashion. DEM has been extended over time and has become a widely-used approach for predicting the dynamics of large granular systems [2], from mixing [3], and particulate flows [4], to landslides and other geomechanics phenomena [5–7], and astrophysical processes [8]. DEM has been used to simulate soil behavior [9], the tire-terrain interaction [10], and rover mobility in extraterrestrial conditions [11].

DEM simulations are generally slow due to two main reasons. First, the elements are often stiff and sometimes very small. This forces the time integrator to take very small time steps to maintain numerical stability. Secondly, collision detection is compute-intensive. To reduce simulation times, DEM has been accelerated via CPU-based parallel computing using OpenMP [12], see, for instance, [13, 14]; the Message Passing Interface (MPI) standard [15], for clusters with distributed memory [16]; and hybrid MPI–OpenMP parallelism [17–20]. An alternative computing architecture for parallel computing is provided by the Graphics Processing Unit (GPU), which has gained wide adoption in DEM, e.g., [21–24]. Whether using CPU or GPU computing, the number of DEM elements used in experiments reported in the literature has been relatively small—in the vicinity of  $10^3$  to  $10^5$  elements [24–39]. For comparison, in one cubic meter of sand, there are on the order of two billion elements and simulations on this scale have rarely been conducted [40].

In addition to utilizing parallel computing to control simulation times, practitioners typically rely on simple element geometries to avoid expensive collision detection calculations. By a large margin, the geometries employed in DEM are thus spheres, in many cases of equal radii. By exclusively dealing with spheres, the collision detection workload is significantly reduced. The software implementation process is also simpler, since handling nontrivial geometries is difficult [41]. However, in many applications, as will be demonstrated in this contribution, one cannot fall back on simple element geometry without compromising the system response [42–48]. Yet in studies in which nontrivial element shapes are brought to bear, the number of simulated DEM elements reported in the literature drops significantly. Commercial solutions to handle nontrivial shapes exist, e.g., [49], but they are closed-source, expensive, and limited in their ability to handle systems with large element counts. In this context, the solution proposed herein embraces the following

compromise: the number of DE can be large (into tens of millions) and while the shape of the elements is nontrivial, it is nonetheless assumed that each element is obtained as the union of a user-specified set of spheres of user-specified radii and material properties, see Fig. 1. The idea of using multiple spheres to generate nontrivial geometries is not new, see, for instance, the discussion in [46], where the authors called the technique the “multi-sphere method”, a member of the class of composite approaches in which elements' geometries are composed as the union of simple geometric primitives. However, previous attempts to use this multi-sphere method were limited to small systems. In [44, 50, 51], the DE systems have a number of elements in the low thousands; in [42], the authors work with hundreds of elements. The approach outlined here builds on two previous contributions: a fast way of doing collision detection, which follows in the step with the approach reported in [52]; and an aggressive use of shared memory along with a new way of representing state information using new data types, see [53, 54]. The simulator is demonstrated in conjunction with a terramechanics problem, in which the terrain is modeled as a collection of elements each composed of clumped spheres. Given that the terrain model is inspired by the GRC-1 lunar simulant, it is called “digital simulant” (DS).

The manuscript is organized as follows. Section 2 introduces the DEM model and the DEM simulator. Section 3 describes the digital simulant. Section 4 reports on validation studies, and touches on angle of repose, cone penetration, and drawbar pull simulations using the DS. The validation is done against experimental results that involved the GRC-1 simulant. Section 5 elaborates on the use of the simulator for single-wheel and full-rover tests of extraterrestrial rovers. Section 6 contains conclusions and outlines directions of future work.



**Fig. 1** In the proposed approach, each element in the DEM simulation is a clump of spheres. The different colors emphasize the fact that each sphere in the clump has its own physical attributes, i.e.,  $E$ ,  $v$ ,  $\mu$ , and CoR

## 2 Simulator overview: force model and implementation aspects

### 2.1 Contact force model and clump representation

The simulator described in this work is the new DEM simulator in Chrono [55]. It has a default force model based on Hertzian contact [56], and support for friction is implemented using the Mindlin model [57]. For an extended discussion see [58]. When bodies  $i$  and  $j$  are in contact, the mutual normal force  $\mathbf{F}_n$  is described via a spring–dashpot mechanism. The tangential friction force  $\mathbf{F}_t$  is evaluated based on material properties and local micro-deformations, and capped to satisfy the Coulomb condition through friction coefficient  $\mu$ . Specifically,

$$\mathbf{F}_n = f(\bar{R}, \delta_n)(k_n \mathbf{u}_n - \gamma_n \bar{m} \mathbf{v}_n), \quad (1a)$$

$$\mathbf{F}_t = f(\bar{R}, \delta_n)(-k_t \mathbf{u}_t - \gamma_t \bar{m} \mathbf{v}_t), \quad \|\mathbf{F}_t\| \leq \mu \|\mathbf{F}_n\|, \quad (1b)$$

$$f(\bar{R}, \delta_n) = \sqrt{\bar{R} \delta_n}, \quad (1c)$$

$$\bar{R} = R_i R_j / (R_i + R_j), \quad (1d)$$

$$\bar{m} = m_i m_j / (m_i + m_j), \quad (1e)$$

where the stiffness and damping coefficients  $k_n$ ,  $k_t$ ,  $\gamma_n$ , and  $\gamma_t$  are derived from material properties, i.e., Young's modulus  $E$ , Poisson ratio  $\nu$ , and the coefficient of restitution CoR [59]. The  $\bar{m}$  and  $\bar{R}$  terms represent the effective mass and effective radius of curvature for this contact, respectively. The fundamental assumption is that geometries are allowed to have small penetration  $\delta_n$  at the point of contact. The normal penetration vector is defined as  $\mathbf{u}_n = \delta_n \mathbf{n}$ . The relative velocity at the contact point  $\mathbf{v}_{\text{rel}} = \mathbf{v}_n + \mathbf{v}_t$  are evaluated as

$$\mathbf{v}_{\text{rel}} = \mathbf{v}_j + \boldsymbol{\omega}_j \times \mathbf{r}_j - \mathbf{v}_i - \boldsymbol{\omega}_i \times \mathbf{r}_i, \quad (2a)$$

$$\mathbf{v}_n = (\mathbf{v}_{\text{rel}} \cdot \mathbf{n}) \mathbf{n}, \quad (2b)$$

$$\mathbf{v}_t = \mathbf{v}_{\text{rel}} - \mathbf{v}_n, \quad (2c)$$

where  $\mathbf{v}_i$ ,  $\boldsymbol{\omega}_i$  and  $\mathbf{v}_j$ ,  $\boldsymbol{\omega}_j$  are the velocity at the center of mass and the angular velocity of bodies  $i$  and  $j$ , respectively. The position vectors  $\mathbf{r}_i$  and  $\mathbf{r}_j$  point from the center of mass of bodies  $i$  and  $j$  to the mutual contact point. The friction force  $\mathbf{F}_t$  depends on the tangential micro-displacement history  $\mathbf{u}_t$  [57, 59], the latter updated incrementally at each time step throughout the life of a contact event based on the tangential velocity  $\mathbf{v}_t$ . Let  $\mathbf{u}'_t$  be the updated tangential micro-displacement, then

$$\mathbf{u}' = \mathbf{u}_t + h \mathbf{v}_t, \quad (2d)$$

$$\mathbf{u}'_t = \mathbf{u}' - (\mathbf{u}' \cdot \mathbf{n}) \mathbf{n}, \quad (2e)$$

where  $h$  is the time step size. The strategy adopted to update  $\mathbf{u}'_t$  is borrowed from [59]. After the update, the new tangential micro-displacement  $\mathbf{u}'_t$  may need to be clamped, so the final  $\mathbf{u}_t$  for the next time step satisfies the capping condition  $\|\mathbf{F}_t\| \leq \mu \|\mathbf{F}_n\|$ :

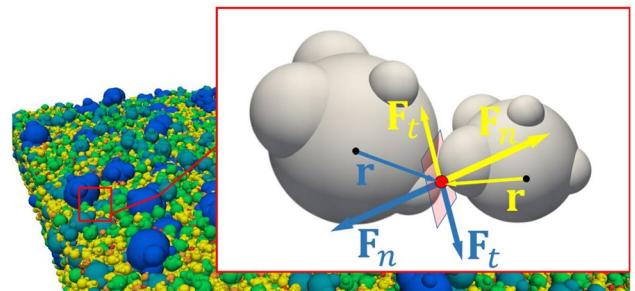
$$\mathbf{u}_t = \begin{cases} \mathbf{u}'_t & \text{if } \|\mathbf{F}_t\| \leq \mu \|\mathbf{F}_n\|, \\ \frac{\mu \|\mathbf{F}_n\|}{k_t} \frac{\mathbf{u}'_t}{\|\mathbf{u}'_t\|} & \text{otherwise.} \end{cases} \quad (2f)$$

The equations of motion for element  $i$  assume the expression

$$m_i \frac{d\mathbf{v}_i}{dt} = m_i \mathbf{g} + \sum_{j=1}^{nc} (\mathbf{F}_n + \mathbf{F}_t), \quad (3a)$$

$$I_i \frac{d\boldsymbol{\omega}_i}{dt} = \sum_{j=1}^{nc} (\mathbf{r}_j \times \mathbf{F}_t). \quad (3b)$$

Chrono's DEM simulator supports complex element geometry through clumps using an approach borrowed from [60]. A clump is a collection of potentially overlapping spheres representing an element shape, see Fig. 2. Herein, the word “clump” is used interchangeably with “element” when referring to complex shape DEM elements. One salient attribute of the model is that the  $E$ ,  $\nu$ ,  $\mu$ , and CoR parameters are associated with a sphere in an element/clump, and not with the element. In numerous instances, the inclusion of this extensive set of  $E$ ,  $\nu$ ,  $\mu$ , and CoR parameters for the spheres within the clump may be deemed excessive. However, it remains a viable choice for those seeking to implement a sophisticated discrete element system.



**Fig. 2** The normal and tangential contact forces between clumps are calculated based on the penetration and displacement history of involved sphere components. An element (clump) has a mass and mass moment of inertia; however, each sphere in the clump has its own  $E$ ,  $\nu$ ,  $\mu$ , and CoR

## 2.2 DEM simulator overview

The new DEM simulator, whose design and implementation details are discussed at length in an upcoming contribution, is an upgrade to Chrono's previous DEM simulator Chrono::GPU [54]. Although fast [61], the latter only handled monodisperse granular systems [62]. For a detailed discussion of the new DEM simulator the reader is referred to [63]. Herein, the focus is on demonstrating Chrono's DEM simulator rather than discussing the design decisions that shaped its implementation. Note that the DEM simulator can run in standalone mode—it is built and can run separately from Chrono. When it runs in co-simulation mode with Chrono, the latter runs the dynamics of the machine system, while the DEM simulation handles the granular dynamics. This opens the door to simulating complex mechanical systems, e.g., rovers, construction equipment, interacting with granular materials—see numerical tests in Sect. 5.2.

If a different contact force model other than the default is preferred, the user can provide a C++ callback customized function that overwrites the default one. Typically, the updating of the active contacts set and the DEM force calculation are done sequentially and at each time step. The new simulator embraces a different strategy, which uses two distinct and parallel computational threads to update the active contacts set (done by the “kinematics thread”) and the integration of the equations of motion (done by the “dynamics thread”), respectively. The user has the option to update the collision pairs at each time step, or can carry out several time steps before updating the contact pairs. In the latter case, the active contacts set is kept the same for several time steps. Regardless of whether the active contacts set is re-evaluated or not at each time step, each active contact is employed *at each time step* to reassess the contact deformation  $\delta_n$  and the ancillary information defined in Eqs. 2. This active contacts set delayed re-evaluation strategy was implemented since it became apparent that identifying the active set represents a significant computational bottleneck. To avoid missing mutual contacts that might crop up between the moments the active contacts set is updated, for the task of updating the active contact set only, the solver artificially enlarges all contact geometries in the DEM system to preemptively detect potential contact pairs that might emerge in the near future.

At *each* time step, the dynamics thread uses the most recent active contacts set to update the penetration  $\delta_n$  and ancillary quantities (see Eqs. 2) needed to evaluate the set of frictional contact forces at work in the system. Note that by artificially enlarging the elements' geometries, the active contacts set will contain false positives and the dynamics thread will spend extra time calculating  $\delta_n$  for a contact only to notice that the gap between the bodies is larger than zero in which case the frictional contact force is set to zero. The thickness of the added safety margin, which is dictated by

the maximum clump velocity (bounded and known to the solver) and the time step size (typically small in large-scale DEM simulations), is a known value, e.g., for millimeter-sized granular material, it assumes values of the order of tens of microns. Overall, the overhead caused by the false-positive contacts does not offset the benefit of deferred updates of the active contacts set. This is because the kinematics and dynamics threads work concurrently and the computational effort dispensed to update the active contacts set by the kinematics thread happens “in the shadow” of the work done by the dynamics thread. Note that one still has the option to fall back on the traditional way of carrying out DEM simulation, in which the active contacts set is updated at each time step before computing the frictional contact forces and advancing the state of the system.

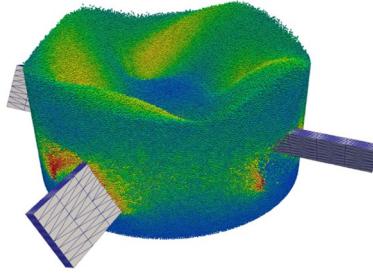
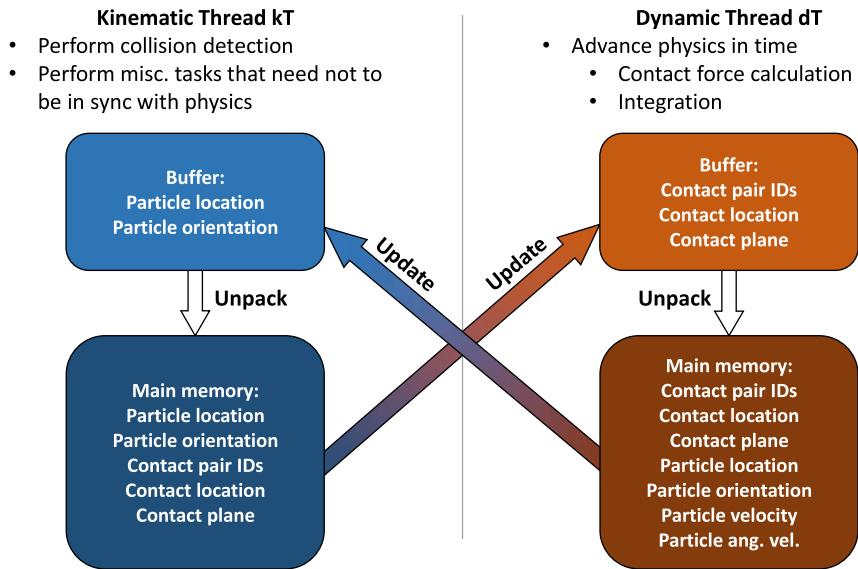
In terms of software implementation, the simulator is set up to use two NVIDIA GPUs—one by the kinematics thread, one by the dynamics thread. It is more efficient to allow each thread to run on a dedicated GPU, yet if only one GPU is available, the simulator makes the two threads share the only GPU available via CUDA streams [64]. As an alternative to using streams, starting with the NVIDIA Ampere architecture, one has the option to partition one GPU into several via NVIDIA's Multi-Instance GPU feature [65].

The collaboration pattern of the kinematics and dynamics threads is summarized in Fig. 3. After being produced by the kinematics thread, the active contacts set and associated bookkeeping information are written into a memory buffer to be read by the dynamics thread. The dynamics thread in turn updates the kinematics thread with new clump positions and velocities. Although logically there are two memory buffers, they are both allocated physically on the GPU mapped to the dynamics thread. This allows the dynamics thread to spend minimum time reading/writing data, thus speeding up the computation.

### 2.2.1 Computational performance

In this subsection, the aim is to assess the speed of the new DEM simulator by comparing its performance with that of Chrono::GPU, a fast solver for monodisperse granular dynamics. This comparison is particularly justified as a recent independent study has reported that Chrono::GPU performs two orders of magnitude faster than other established DEM packages [61]. Therein, for a 420,000-element pebble-packing simulation, Chrono::GPU running on a laptop GPU finished the simulation in an amount of time 261 times shorter than that required by LAMMPS [19], when the latter ran on 432 CPU cores of a cluster. For a 660,000-element pebble-packing simulation, Chrono::GPU executed 501 times faster than STAR-CCM+ [66], which ran on 160 CPU cores. In both tests, Chrono::GPU ran on the RTX 2060 Mobile NVIDIA GPU card of a laptop.

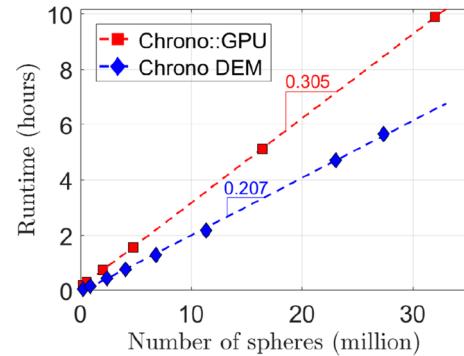
**Fig. 3** The collaboration pattern of the kinematic and dynamics thread. They each run on a dedicated GPU. Note if two GPUs are in use, the two buffers are both allocated physically on the GPU mapped to the dynamics thread



**Fig. 4** A rendering of the mixer scaling analysis test scene

The test used to compare the simulators is the particle mixer employed for the scaling analysis reported in [54]. A bladed mixer rotates at  $2\pi$  rad/s and interacts with a body of granular material, see Fig. 4. Since Chrono::GPU can only handle spheres, all DEM elements in the mixer have the same radius. In the scaling analysis study, increasing the number of spheres from test to test is accomplished by gradually reducing the radius of the spheres in the mixer. Note that the size of the mixer is kept constant. The total runtime for a three-second simulation is reported and plotted against the number of elements in Fig. 5. To ensure the tests display the same physics as in [54], the step size ( $10^{-5}$  s) and material properties from that reference are used. More details about this test can be found in [54].

Figure 5 shows that compared to Chrono::GPU, which is designed and therefore optimized for one case only, i.e., monodisperse granular dynamics, the new simulator is approximately 1.5 times faster. Chrono::GPU on average takes 0.305 hours for simulating one million DEM elements for three seconds. The new solver, on the other hand, requires 0.207 h. Both solvers scale linearly within the problem size. It should be noted that the new solver was run on



**Fig. 5** The scaling analysis results for Chrono::GPU and the new DEM simulator. The runtime for finishing each three-second simulation is plotted against the number of elements. Chrono::GPU runs on one A100, while the new simulator runs on two

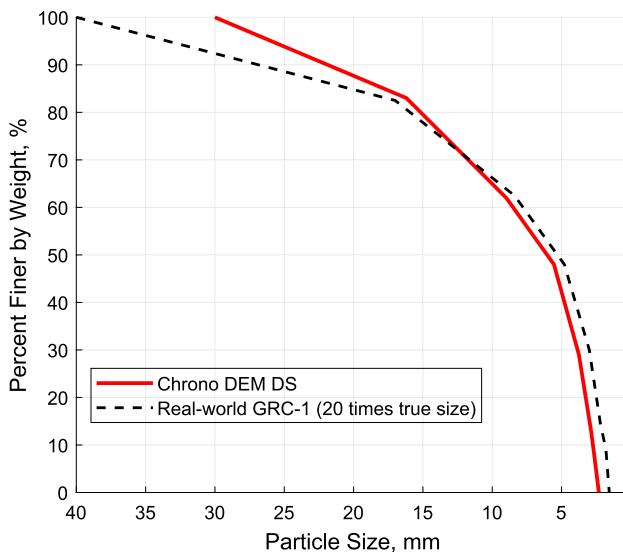
two A100s while Chrono::GPU ran on one A100. Owing to its ability to handle complex DEM particle shapes, the new simulator enhances the modeling capacity of the existing Chrono::GPU solution, without compromising efficiency as long as one uses two GPU cards. Note that the new DEM simulator can run even on one GPU card, yet this will come with a slowdown in the simulation speed.

### 3 Digital simulant used

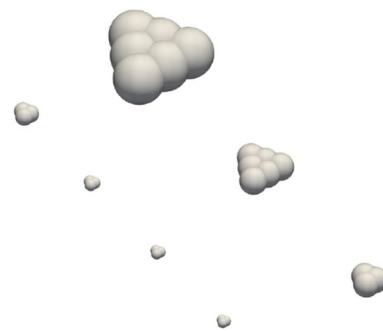
The new DEM simulator can work with complex particle geometries that are user-defined via overlapping spheres. However, in this paper, owing to the modeling interests, the focus is on the DEM simulation of a particular type of granular material. Specifically, the authors use a “digital simulant” (DS) inspired by GRC-1 [67], the latter developed

as a lunar soil replica for Earth testing of rovers and similar implements. The GRC-1 simulant contains cohesionless and frictional particles that have complex shapes and span a range of sizes. The fact that some GRC-1 particle sizes are micron-scale posed insurmountable challenges to the simulator. Despite using a customized way of storing contact information that goes beyond what the IEEE double precision data type can offer in C/C++, see [63], working with the actual particle size proved intractable owing to the broad spectrum of sizes—from microns to millimeters. As a compromise, the digital simulant maintained the statistical size distribution, yet uniformly increased by a factor of 20 the actual particle sizes encountered in GRC-1. In other words, the DS size distribution was identical to GRC-1, but the DS sizes were shifted to larger values. The value 20 mentioned above has no particular meaning other than being the smallest value that the representation had to be scaled by so that the DEM simulation became acceptable, both in terms of results accuracy and time to completion. The simulation experiments of Sect. 5.1.1 confirm that this “scaling up” of the particle sizes does not significantly compromise the accuracy of the results obtained through simulation.

The DS consists of seven DEM element types, each with a specific size and percentage of the total weight, see Table 1. The size distribution is plotted in Fig. 6. Therein, although the size distribution is not continuous, it is shown with seven line segments, each corresponding to the weight percentage contributed by each element type. Figure 7 illustrates the DS grain shapes. Out of the seven distinct grain types, the two larger ones are made of six overlapping spheres and have a flat triangular shape. The five smaller types are each made of three component spheres. All of them have a 120°



**Fig. 6** The size distribution of the DS, plotted against a scaled GRC-1 simulant size distribution



**Fig. 7** The seven clump shapes that show up in the DS

rotation symmetry. Having a triangular aspect, the particle size reported in Table 1 is measured as the diameter of the bounding sphere. The radii of the elements’ component spheres and the material properties used in the numerical tests throughout the paper are also summarized in Table 1.

The nontrivial shape of the elements turned out to play an important role in the dynamics of the granular material. Compared to monodisperse granular material, which is commonly used in DEM simulations, the DS allowed for geometric locking, thus shaping the granular material’s behavior, especially in shear-dominated experiments. Qualitatively, this observation is backed by the early results reported in Fig. 8a and b, wherein the same single-wheel slippage experiment was carried out once using the DS and then monodisperse granular material.

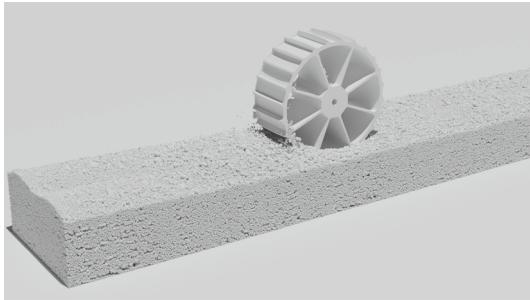
The simulations shown in the following sections consist of millions to tens of millions of DEM elements, a scale that is manageable on one or two modern GPU devices. If the true GRC-1 grain sizes were used, the number of elements would increase by close to four orders of magnitude, which places an insurmountable barrier to simulation owing to memory constraints, as well as arithmetic precision limits as some deformations  $\delta_n$  become prohibitively small. Note that the smallest sphere component in a DS clump had a radius of approximately 700  $\mu\text{m}$ .

## 4 Validation studies

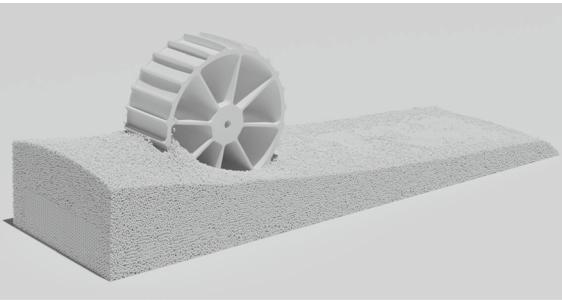
The validation and refinement of a DEM simulator is an arduous and lengthy process that spans an extended period. This contribution reports on three experiments that were used for the early validation of the new DEM simulator in Chrono. The results in this section represent a confidence-building step before using the DEM simulator to conduct additional studies for which there is limited or no experimental data available. The additional studies are discussed in Sect. 5. The code used in Sects. 4 and 5 is available in a GitHub public repository [68] for unrestricted use,

**Table 1** Weight distribution, percent-wise, by particle size. For all particle types,  $E = 10^9 \text{ N/m}^2$ ,  $\nu = 0.3$ ,  $\mu_s = 0.4$ , and CoR = 0.5

Type	1	2	3	4	5	6	7
Size (mm)	21	11.4	6.6	4.5	3	2.75	2.5
Component radius (mm)	3.6	1.95	1.81	1.24	0.82	0.75	0.7
%, by weight	17	21	14	19	16	5	8



(a) The single-wheel test on a  $25^\circ$  incline, with DS.



(b) The single-wheel test on a  $25^\circ$  incline with spherical DEM elements. Snapshot suggests that monodisperse granular material support for a single wheel test is qualitatively different than what behavior observed for DS testing.

**Fig. 8** The renderings show that, on the same  $25^\circ$  incline, the single-wheel simulation on spherical DEM elements leads to much more terrain shearing compared to simulation on DS

distribution, and reproducibility studies. The only caveat is that the wheel model shared in the public repository is only an approximation of the actual wheel used in this paper. The shared model was put together using data available in the public domain; the actual wheel model is not available for public access.

#### 4.1 Sample preparation

This subsection describes the preparation strategy employed to produce a bed of granular material. For monodisperse spherical DEM elements, one can create the initial elements using a sampling algorithm, such as hexagonal close-packed, with a separation factor slightly larger than the diameter of each sphere. For the DS, due to the irregular grain shape and large variation in grain size, the separation factor needs to be larger than the size of the largest element to prevent overlap at the start of the simulation. This leads to a low space occupancy during the initial sampling process, as shown in Fig. 9a. Therefore, several batches of clumps need to be generated and settled to form a terrain sample of a certain depth, as depicted in Fig. 9b and c. The element size distribution shown in Fig. 6 is retained during the spawning of each DS batch. After ten batches of clumps settle, the simulator produces the resultant sample illustrated in Fig. 9d.

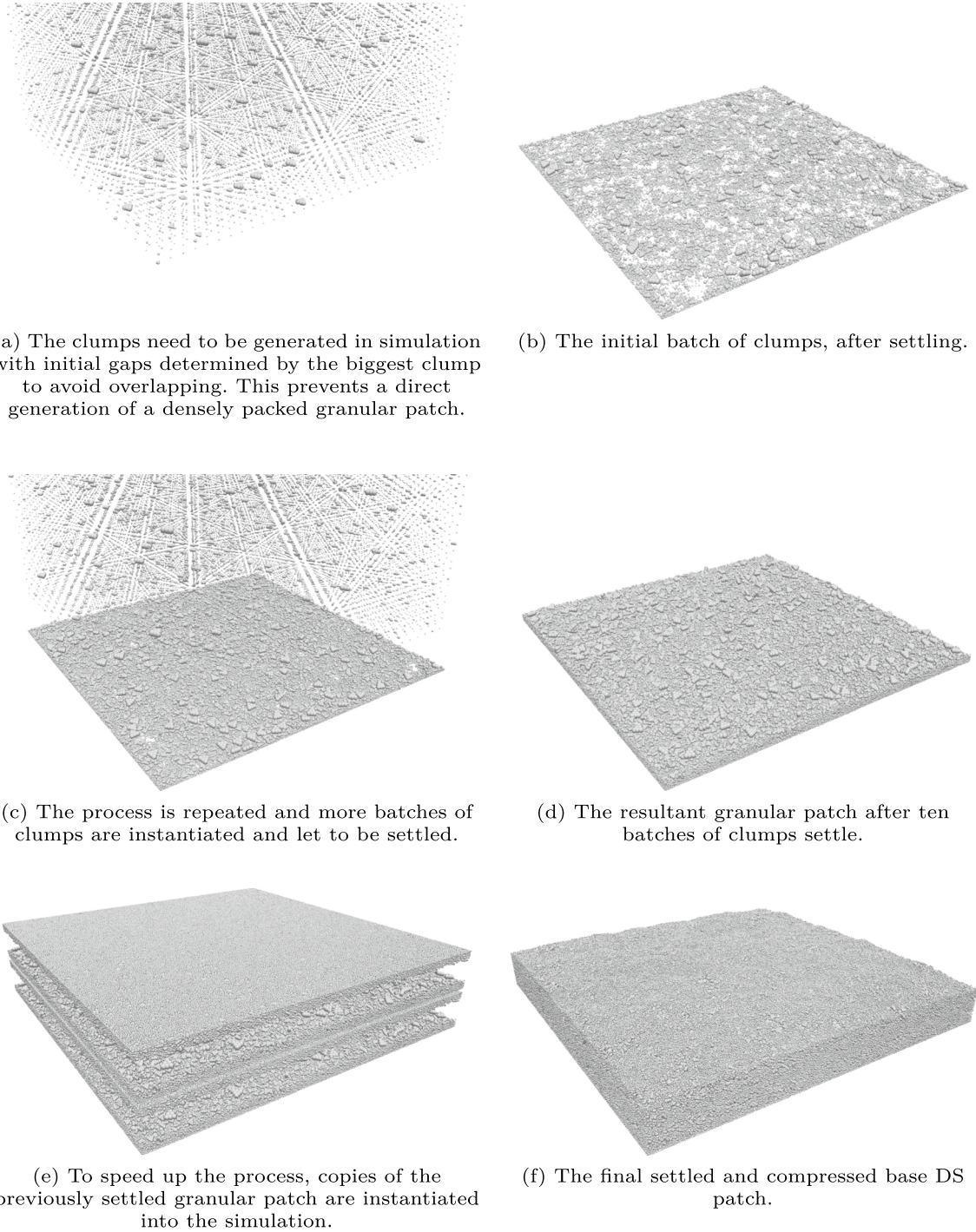
With this newly generated thin layer of clumps, a “copy-paste” process is used to accelerate the sample preparation. Several copies of the existing clumps are duplicated, then

added back to the simulation, as shown in Fig. 9e. After settling, a virtual compressor object is created to compress the sample to make the surface relatively even, see Fig. 9f. This terrain sample now is one meter by one meter in width and roughly 15 centimeters in height. Herein, this is referred to as a base DS patch. On two NVIDIA A100 GPUs, finishing the entire sample preparation process at time step size  $10^{-6}$  s takes approximately 12 h. There are 4,571,136 clumps, totaling 13,993,536 spheres that are present in the base DS patch.

The creation of this base patch is for the ease of scaling up the simulation. When a larger test bed is needed, several such patches can be instantiated, moved, and settled in the simulator to create the test bed. Conversely, a subset of this patch can be extracted to create a smaller test bed. All numerical experiments in this paper use test beds derived from the base patch, saving time that would otherwise be spent on settling the DEM terrain.

#### 4.2 Repose angle validation

The angle of repose is expected to be equal to the residual internal friction angle [69]. The latter serves as a proxy for a material’s shear strength that dictates a vehicle’s climbing ability in off-road conditions, a topic discussed in Sect. 5. For the repose angle test, the initial sample is prepared by taking a cylindrical portion out of a DS patch, making three extra copies, and then translating everything into a funnel defined via a mesh, see Fig. 10a. The material

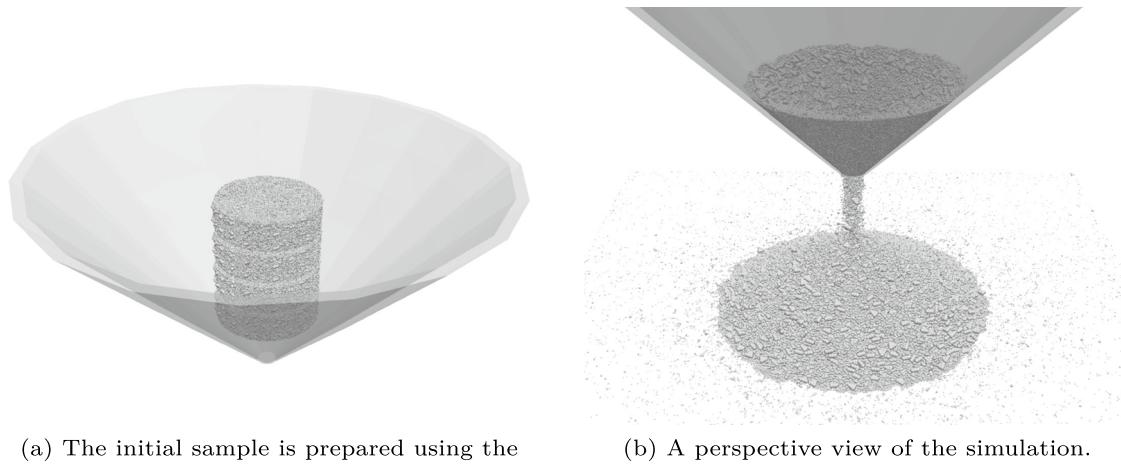
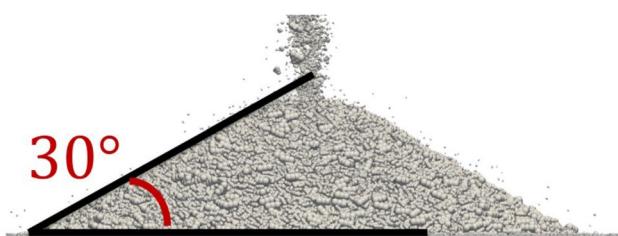
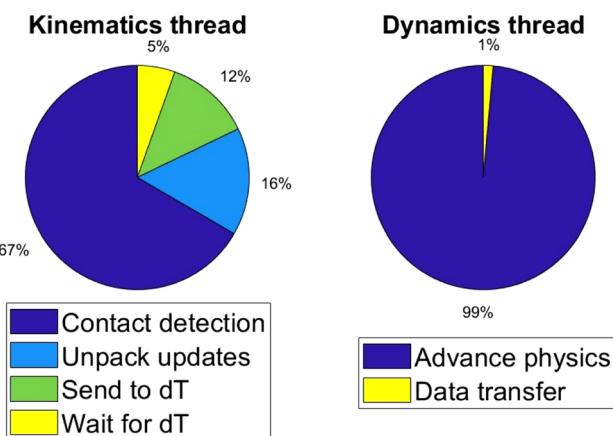


**Fig. 9** The rendering of DS preparation process

flows through the funnel under gravity. The pile formed underneath can be seen in Fig. 10b. Figure 11 displays a  $30^\circ$  angle of repose, in line with results reported in [67] which is  $29.8^\circ$ . In total, 731,060 clumps participate in the simulation. On two NVIDIA A100 GPUs, running this 30 s simulation at time step size  $10^{-6}$  s takes approximately

19 h. There are 2,234,544 spheres that combine to make up the clumps in the material used for this test.

Figure 12 shows how the kinematics and dynamics threads employed by the simulator spend their time. For the kinematics thread, around two-thirds of the runtime is spent on contact detection. As discussed, the kinematics thread's

**Fig. 10** The rendering of DS angle of repose test**Fig. 11** The settled DS forms a 30-degree pile**Fig. 12** The runtime breakdown for the kinematics and dynamics threads. The results shown are associated with the angle of repose simulation

buffer arrays are allocated on the dynamics thread's GPU, so it spends more time on obtaining the new geometry location information ("unpack updates") and sending the produced contact array ("send to dT") to the dynamics thread,

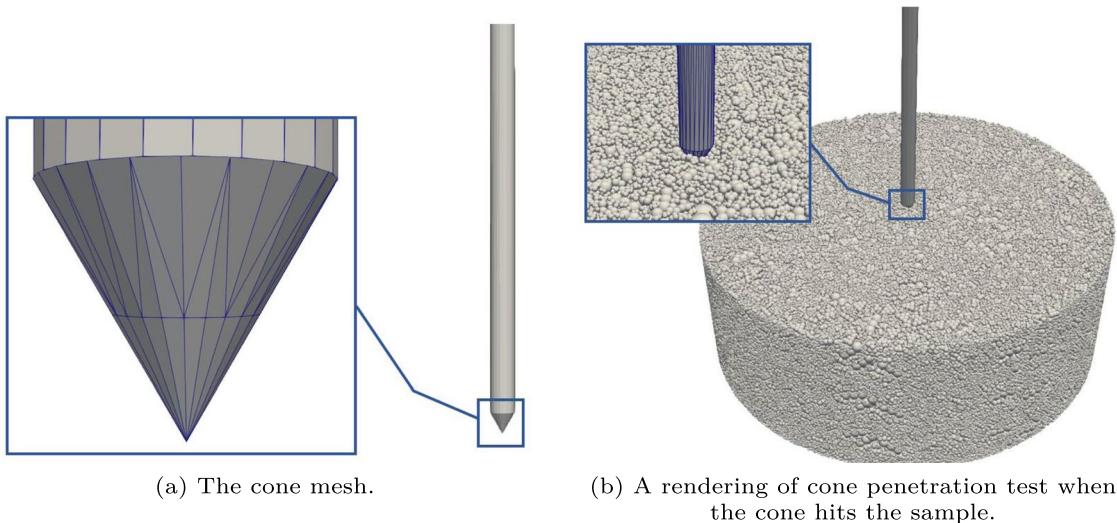
reaching 16% and 12% of the runtime, respectively. It also spends 5% of the runtime waiting for new "work orders" from the dynamics thread. The dynamics thread, on the other hand, spends nearly all runtime on advancing the physics, i.e., the state of the DEM elements. The 1% of runtime spent on data transfer includes sending the new geometry location information and receiving the produced contact array from the kinematics thread, as well as occasional waiting for the kinematics thread's update.

### 4.3 Cone penetration test

The cone penetration test is useful for gauging the soil's stiffness and density, as well as the stability of slopes and the potential for landslides [70]. Herein, the authors test in simulation the penetration experiments conducted with GRC-1 simulant reported in [67]. Same as in [67], the soil bin was selected to be 58.4 cm in diameter and 24 cm in depth, which is sufficient to minimize boundary effects [71].

Similar to the repose test, the initial sample is prepared by taking a cylindrical portion out of a DS patch, making two extra copies, and then translating everything into the cylindrical container and letting the material settle. Subsequently, a cone penetrates the sample with a constant velocity of 3 cm/s, see Fig. 13b. The cone, whose geometry is specified by a mesh, is shown in Fig. 13a. The cone has a base area of 323 mm<sup>2</sup> and an opening angle of 60°. The contact force on the cone is measured and plotted against the penetration depth in Fig. 14. The different bulk density densities of DS in Fig. 14 are reached by compressing the material bed before the simulation starts.

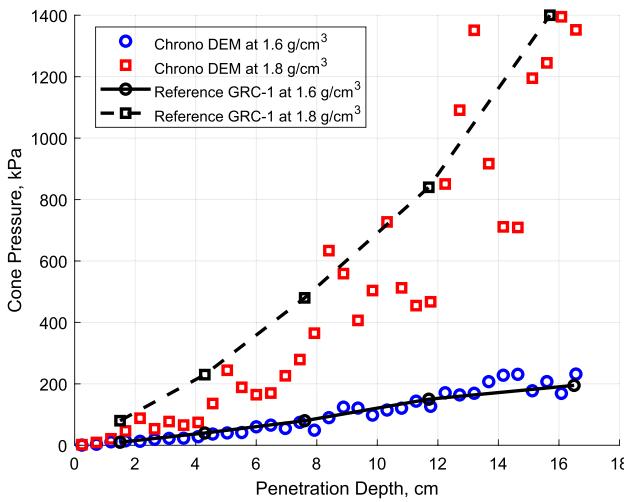
As shown in Fig. 14, the DS reproduces a depth-pressure relationship in the cone penetration tests similar to the one



**Fig. 13** Rendering of the cone penetration test using the DS

noted in real-world experiments [67]. The simulator also reproduces the drastic increase of the pressure on the cone when the simulant is compressed to a higher bulk density, from 1.6 to 1.8 g/mm<sup>3</sup>. In all the subsequent numerical tests presented in the paper, the DS operates at a bulk density around 1.6 g/mm<sup>3</sup>, which is close to what a naturally settled DS sample has.

In total, 773,097 clumps, which employ 2,362,698 spheres, participate in the simulation. On two NVIDIA A100 GPUs, running this 8 s simulation at time step size 10<sup>-6</sup> s takes approximately 4 h.



**Fig. 14** The DEM cone penetration test results with different bulk densities of the DS. They are plotted against the reference GRC-1 cone penetration experimental data. The experimental data are from GRC-1 simulant research paper [67]

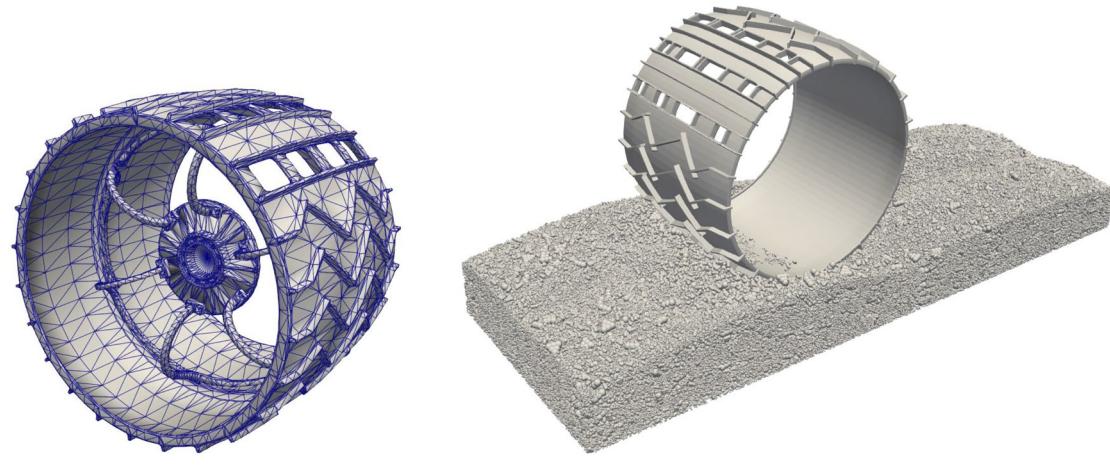
#### 4.4 Single-wheel drawbar pull test

In [72], a test rig named ARTEMIS was used to derive the force–slip relationship for a single Curiosity wheel operating on Mars soil simulant. Herein, the authors use the experimental setup described in [72] as a reference and demonstrate that the DS captures the terrain strength similar to real-world simulants. Note that there is a slight discrepancy between the actual granular material used in [72] and the DS used in the simulations. This aspect will be revisited shortly.

Reproducing drawbar pull tests in simulation involves firstly an abstraction of the test equipment. The terrain leveling and compacting process described in [73] has already been incorporated into the terrain preparation procedure discussed in Sect. 4.1. Additionally, enforcing a slip value does not require a mechanical system or attaching a hitch to the test vehicle as in the physical test; instead, specific linear and angular velocities are applied directly to the wheel.

This test is conducted in Earth's gravitational acceleration  $g = 9.81 \text{ m/s}^2$ . The Curiosity wheel is represented by the mesh shown in Fig. 15a. The grouser geometry is preserved exactly; the wheel has a radius of  $r = 0.25 \text{ m}$  and a weight of 750 N. The mesh file can be obtained from the Chrono repository [74], and readers are referred to [72] for comparison against the real Curiosity wheel. In this simulation, the wheel hub is removed to reduce the mesh size and save simulation time.

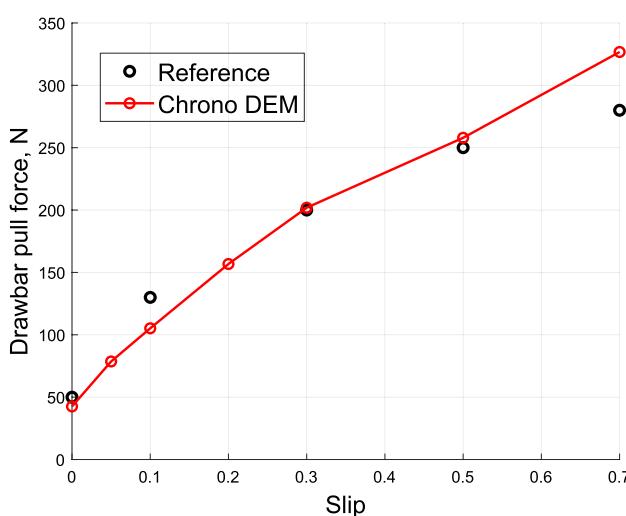
An overview of the simulation environment is illustrated in Fig. 15b. The granular terrain is prepared by making copies of the DS patch and concatenating these copies together so that a test bed is formed. A fixed angular velocity of  $\omega = (\pi/12) \text{ rad/s}$  is imposed on the wheel. The



(a) Open source mesh of the Curiosity wheel, publicly available in the Chrono repository [74].

(b) A perspective view of the wheel operating on the soil bed.

**Fig. 15** The mesh and the simulation scene of the drawbar pull test



**Fig. 16** Curiosity wheel drawbar pull DEM simulation result with a vertical load of 750 N, compared against the reference ARTEMIS experimental data [72]

linear velocity  $v$  in the forward direction is enforced as  $v = (1 - s)\omega r$ , where  $s$  is the desired slip ratio.

The simulation results are presented in Fig. 16. The reference experimental data from [72] is shown with black markers in the same plot. Note that the simulations in [72] use a custom-made blend of two types of sand that is different from GRC-1, and the authors do not have access to the drawbar pull data of GRC-1. Therefore, this comparison is qualitative only. Nonetheless, the drawbar pull–slip

curves exhibit notable similarities between our simulation and the reference experiment.

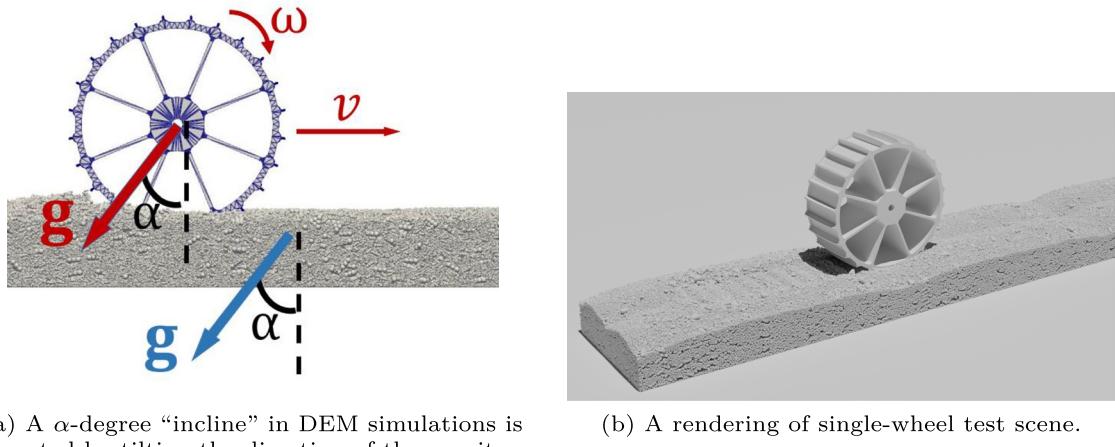
In total, 960,906 clumps employing 2,942,346 spheres participate in the simulation. On two NVIDIA A100 GPUs, running this 8 s simulation at time step size  $10^{-6}$  s takes approximately 5 h.

## 5 Further DEM studies related to rover mobility

This section focuses on single-wheel and full-rover incline tests. The limited amount of experimental data used for validation was produced by the Simulated Lunar Operations (SLOPE) laboratory at NASA's Glenn Research Center using GRC-1 and GRC-3 simulants. The wheel geometry used in the tests of this section is from NASA's Moon Gravitation Representative Unit 3 (MGRU3).

### 5.1 Single-wheel slip test on incline

The inclines used in the simulation are created by altering the direction of gravity, see Fig. 17a. This approach to representing slopes requires minimum model adjustments when switching from one test scenario to another. Figure 17b is a screenshot of this test. The wheel in the test is subject to a fixed angular velocity  $\omega$  and is free to move linearly. The steady-state average linear velocity over 6 s  $v$  is measured, and subsequently used to obtain the slip ratio  $s = 1 - v/(\omega r)$ . The wheel used has a mass of 5 kg but may have a different “effective” mass in these simulations. The “effective” mass



**Fig. 17** The simulation setup for DEM single-wheel tests using DS

is introduced to account for the extra mass placed on top of each rover wheel. This modified mass is modeled as an extra gravity-aligned force applied to the center of the wheel. This force is dependent on the gravitational acceleration of the test scene. For example, if a wheel has an effective mass of 22 kg in a test that uses the Moon’s gravity, that means an extra force of  $17 \times 1.62$  N is applied to it. The granular terrain is prepared by making copies of the base DS patch and concatenating these copies together, similar to the previous drawbar pull test.

The ground-truth experimental data used for comparison is from the experiments done with MGRU3 climbing a “tilt bed” made with the GRC-1 simulant in the SLOPE lab. A picture of the test scene can be seen in Fig. 18, which is from a publicly available video of the test [75]. In this experiment, the wheels of MGRU3 were operating at  $\omega' = 0.8$  rad/s under Earth’s gravity  $g = 9.81$  m/s<sup>2</sup>. Note that the thickness

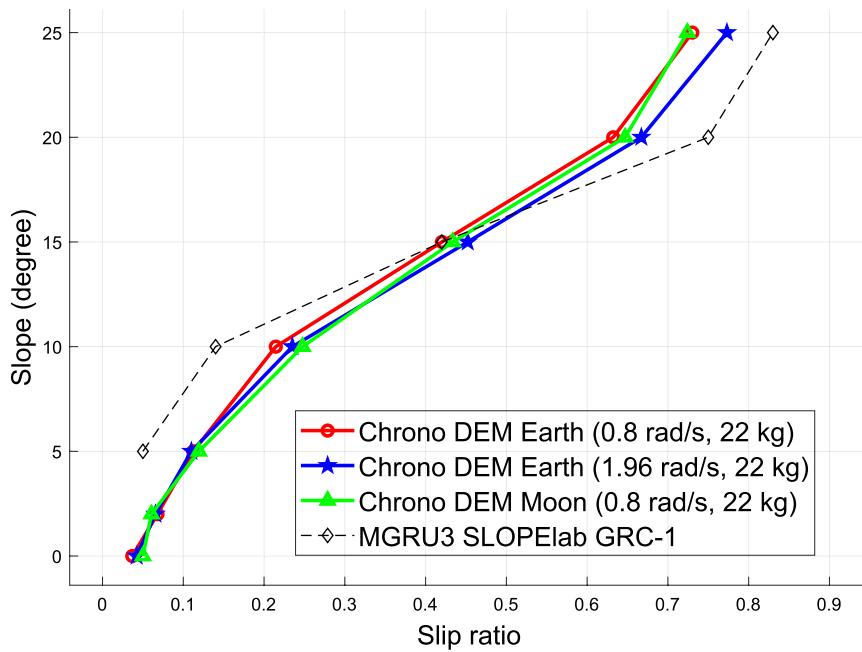
of the bed of granular material has an influence on the experimental results, and the sufficient thickness to eliminate the boundary effect could depend on the vertical load. However, this secondary aspect, which might produce slightly different data in both experiments and simulations, falls outside the scope of this contribution.

Figure 19a gives the slope–slip relationship for a wheel with an effective mass of 22 kg. The first DEM experiment is set to match the experimental baseline test condition. The result is shown with the red curve. Two additional single-wheel tests are subsequently conducted: the blue curve shows the result of a test with angular velocity  $\omega = 1.96$  rad/s under Earth’s gravity; the green curve is associated with a test in with angular velocity  $\omega' = 0.8$  rad/s under moon gravity  $g' = 1.62$  m/s<sup>2</sup>. The similarity of the results indicates that the simulator captures the scaling law for locomotion in granular media [76, 77], which states that,

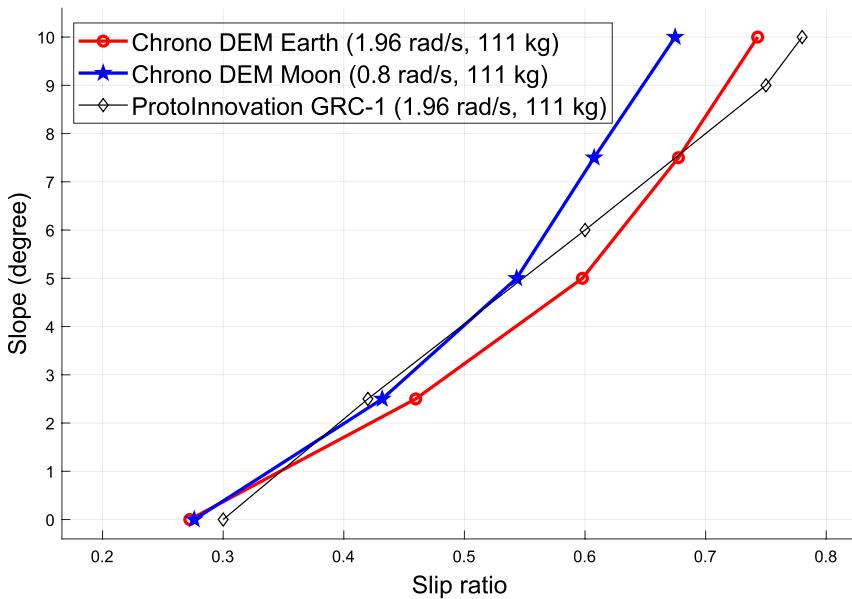
**Fig. 18** MGRU3 climbing a “tilt bed” in NASA’s SLOPE lab testing facility



**Fig. 19** The single-wheel slope vs. slip DEM test results with a variety of wheel effective mass, angular velocity, and gravitational pull



(a) Slip-on-incline test result with 22 kg wheel. The experimental data used for comparison (black line) are from Glenn Research Center's MGRU3 experiments with the GRC-1 simulant.

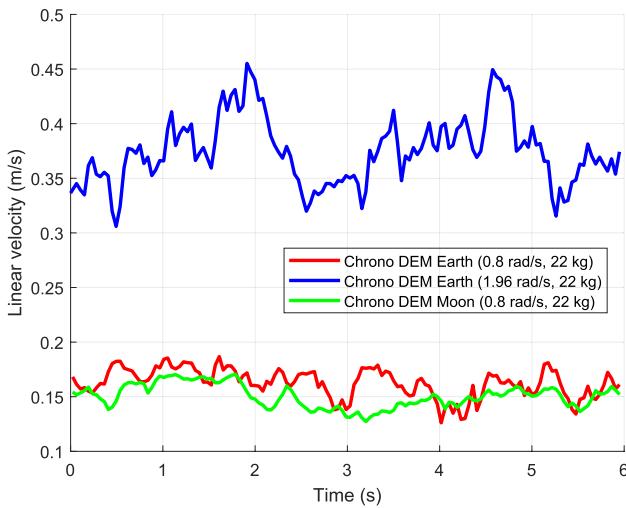


(b) Slip-on-incline test result with 111 kg wheel. The experimental data used for comparison (black line) are from ProtoInnovations' single-wheel experiments with the GRC-3 simulant.

if  $g'/\omega'^2 = g/\omega^2$ , the slip ratio at steady state is unchanged. At the same time, the fact that they are also close to the red curve suggests the slip ratio is relatively insensitive to the variation of the rotational speed, at least within the range tested herein. These three test scenarios demonstrate that it is viable to reproduce the terrain response on the Moon by using the same simulant on Earth while keeping the rover *mass* constant. Using the 10-degree simulation as an

example, in Fig. 20 the linear velocity of the wheel along the incline direction is plotted against the 6-second time span in which the average velocity  $v$  is measured. The linear velocity shows a well-defined mean value, meaning a steady state is reached during the measurement.

Next, another single-wheel test is shown in Fig. 19b to demonstrate that matching the *ground pressure* under a wheel on the Moon and Earth is not an accurate approach for predicting the traction capability of a rover. The wheel



**Fig. 20** The linear velocity of the wheel along the incline direction plotted against the 6-second time span used to derive the slip ratio, at a 10° incline

in this test has an effective mass of 111 kg. If matching the ground pressure on Earth would reproduce the rover behavior on the Moon, then one would expect the blue curve in Fig. 19b to show a similar slip ratio at a given slope, compared to the red curve in Fig. 19a. However, this is not the case. To negotiate the same slope, a lot more slip is observed in this 111 kg test because the terrain elements also shear a lot more easily in the low-gravity condition. Note that the scaling law holds in this test condition too, as shown by the similarity between the two curves in this plot. The ground-truth experimental data used for comparison (black dashed line) are from ProtoInnovations' single-wheel experiments with the GRC-3 simulant, because the experimental data for 111 kg wheel operating on GRC-1 simulant is not available. Note that GRC-3 simulant has in general comparable properties to GRC-1, but contains slit particles, and therefore this comparison is qualitative only. Those experiments were done using ProtoInnovations' in-house test bed, with the wheel operating at 1.96 rad/s under Earth's gravity  $g = 9.81 \text{ m/s}^2$

A number of 3,681,560 DEM elements employing 11,259,462 spheres are used in the simulations for which  $\omega = 1.96 \text{ rad/s}$ . On two NVIDIA A100s, running these 8.4 s simulations at time step size  $10^{-6} \text{ s}$  took approximately 15 h. For test cases where the angular velocity of the wheel is smaller, shorter test beds are used and the time cost is proportionally lower.

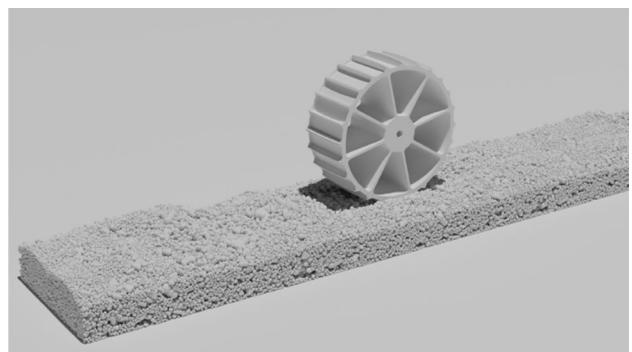
The simulation results obtained in this subsection are in agreement with the general scaling relation for locomotion in granular media [77] and they underline an important fact: to carry out single wheel tests on Earth that

display slope vs. slip curves similar to the ones obtained on the Moon, one should keep the same mass of the wheel. The remaining single-wheel simulations carried out in this section employ the 22 kg wheel and look into the sensitivity of the slope vs. slip curves with respect to three factors: the size of the clumps that make up the terrain, value of the friction coefficient, and clump shape.

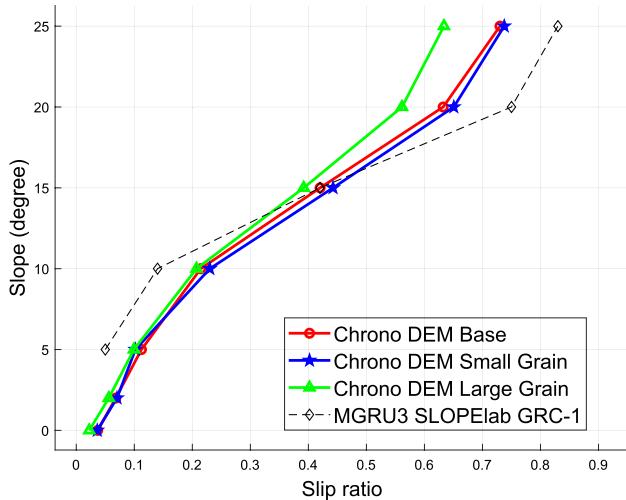
### 5.1.1 Sensitivity of slope vs. slip curve with respect to the clump size

In Sect. 3, it was pointed out that the DS is 20 times larger than the GRC-1 simulant. In this section, the focus is on demonstrating that shrinking the clump size within a certain range does not affect significantly the simulation accuracy.

The test where the 22 kg wheel operates at an angular velocity of 0.8 rad/s under Earth's gravity is repeated, with the terrain made of two new types of DEM elements. The blue curve shows the slip-slope relationship when each DS element's size is scaled by a 0.75 factor to obtain a different digital simulant. This reduces the apparent size of the smallest type of clumps from 2.5 to 1.875 mm. 2,846,244 clumps employing 8,707,887 are in the simulation and this 8.4 s simulation took approximately 11 h. Note that this size change requires re-running the entire sample generation and test bed preparation process before carrying out the slope vs. slip test. Compared to the base red curve, no significant changes were observed in the slope vs. slip results. Conversely, the green curve shows slope vs. slip results when the DS element size is increased by a factor of 2, with 299,929 clumps (917,907 sphere components) participating in the simulation. A time of 1.5 h is needed for this 8.4 s simulation. A rendering of this test is shown in Fig. 21. In simulations, larger clumps in this scenario reduced flowability with geometric locking, which led to the wheel slipping less, especially on steeper slopes.



**Fig. 21** A rendering of the single-wheel test with enlarged terrain clumps

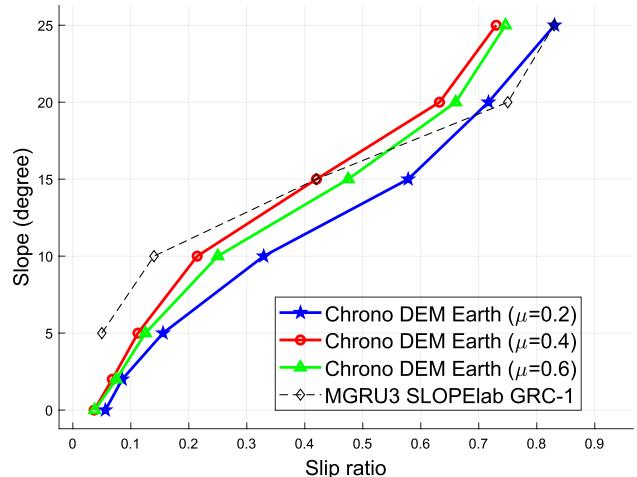


**Fig. 22** The single-wheel slope–slip test results with different selections of the clump sizes. The experimental data used for comparison (black line) are from Glenn Research Center’s MGRU3 experiments with the GRC-1 simulant

In conclusion, although DS has a particle size that is on average 20 times larger than the particle size in GRC-1, reducing the DS element size does not negatively impact the simulation fidelity. However, if the DS element sizes are increased by a factor of 2 (i.e. simulation resolution is reduced), the accuracy is hindered due to the induced low flowability and geometric locking (Fig. 22).

### 5.1.2 Sensitivity of slope vs. slip curve with respect to the friction coefficient

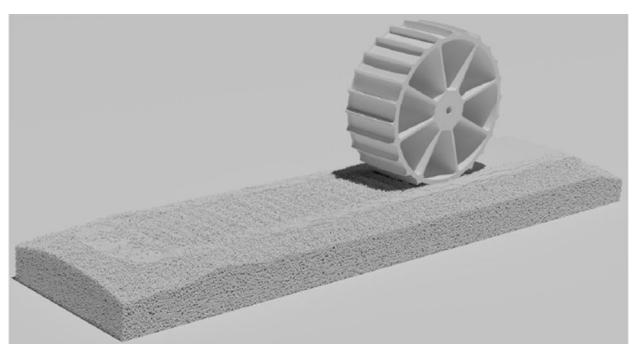
The choice of friction coefficient  $\mu = 0.4$  is a typical value for silica–silica contact. However, it is well-known that the actual value of the friction coefficient can vary depending on the environment, e.g., humidity and preparation of the facets that come in contact [78, 79]. In this subsection, the focus is on investigating the sensitivity of the slope vs. slip curve with respect to the friction coefficient. To that end, the authors repeat the test where the 22 kg wheel operates at an angular velocity of 0.8 rad/s under Earth’s gravity (red curve in Fig. 19a), with two different inter-element friction coefficient values. The blue curve in Fig. 23 shows results when  $\mu = 0.2$ —one can observe significantly more slip compared to the base case of  $\mu = 0.4$ . However, in the case where  $\mu = 0.6$ , the slip is comparable to the base case. This suggests that increasing  $\mu$  helps prevent the wheel from slipping on the incline, up to a certain value, beyond which the influence of  $\mu$  seems to diminish and other factors, e.g., particle locking, start playing an important role.



**Fig. 23** The single-wheel slope–slip test results with different selections of the friction coefficient. The experimental data used for comparison (black line) are from Glenn Research Center’s MGRU3 experiments with the GRC-1 simulant

### 5.1.3 Sensitivity of slope vs. slip curve with respect to the clump shape

The strength of the terrain in mobility tests is expected to change as the clump shapes move away from spheres. In this section, the simulation results that quantify this change are reported. One extreme is to have monodisperse granular material. To test this scenario, the authors repeat the single-wheel tests in Sect. 5.1 with the radius of the spherical elements being first 3 mm and then 6 mm. The wheel has a constant angular velocity of 0.8 rad/s and the gravitational acceleration is  $g = 9.81 \text{ m/s}^2$ . The material properties of the granular material are as in Table 1, except that the friction coefficient  $\mu$  is increased from 0.4 to 0.9. This change is motivated by difficulties making the spherical elements rest on a steep incline when using  $\mu = 0.4$ . The latter friction coefficient, associated with dry silica, is acceptable for the



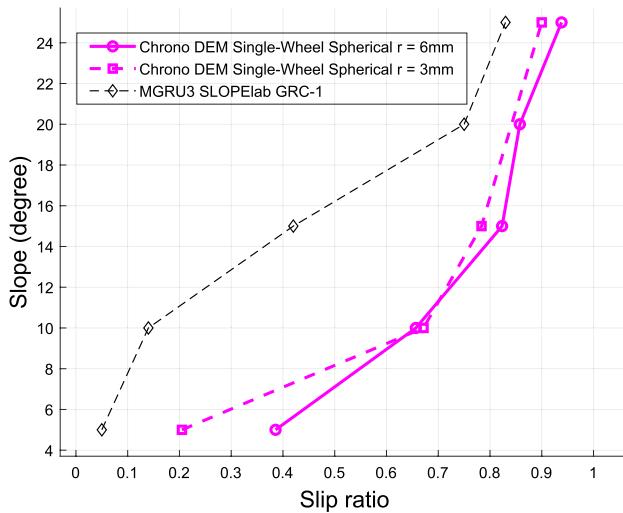
**Fig. 24** A rendering of the single-wheel test with the granular terrain represented by spherical elements, at 0° incline

DS tests since the elements' irregular shapes contribute to the locking of the grains.

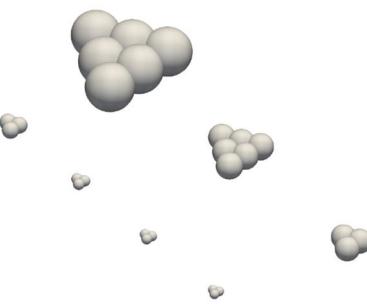
If the spherical elements are fine enough (3 mm in this test, the dashed magenta line), the sphere-based DEM appears to show a somewhat reasonable slip ratio in the flat terrain tests. A rendering is shown in Fig. 24. However, even for mobility tests on a gentle incline, the monodisperse terrain shears drastically more than the DS and the real-world GRC-1 simulant. The wheel is essentially spinning in place at a 15° incline. At a 25° incline, the elements immediately flow down the incline upon making contact with the wheel as previously shown in Sect. 1, Fig. 8b. The slope vs. slip relationship is summarized in Fig. 25.

The authors also present an opposite investigation where the aspect ratios of the clumps are enlarged by expanding the clumps without changing their thickness. This is accomplished by moving each component sphere slightly away from the geometric center of the clump while maintaining the planar overall shape of the clump, see Fig. 26 for new shapes. Quantitatively, keeping other conditions unchanged, the seven clump templates' aspect ratio was increased by about 25%. After creating a new DEM terrain using the new clumps, the single-wheel test is repeated with a wheel angular velocity of 0.8 rad/s and gravitational pull of  $g = 9.81 \text{ m/s}^2$ . It is observed in Fig. 27 that the slip, shown with the blue curve, is reduced compared to the base red curve, and this effect is more pronounced on steeper slopes.

Moving from spherical to clumped DEM elements has a significant impact on terrain strength in mobility simulations. The geometric locking effect that DS enables on incline tests cannot be replicated using monodisperse terrain.



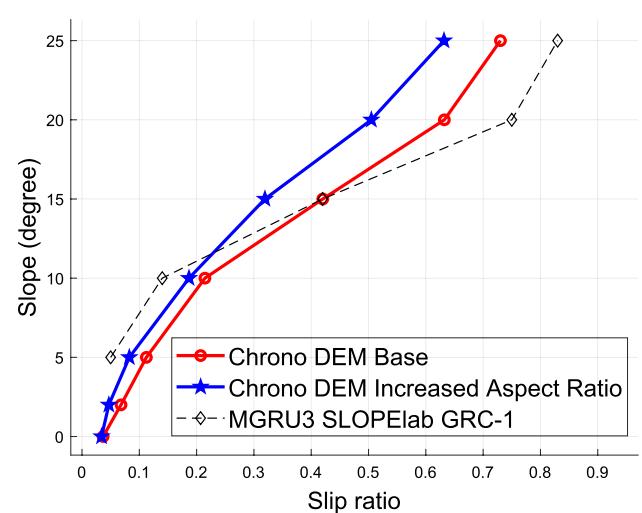
**Fig. 25** Slip-on-incline test result with spherical terrain elements of radius 6 mm and 3 mm. The experimental data used for comparison (black line) are from Glenn Research Center's MGRU3 experiments with the GRC-1 simulant



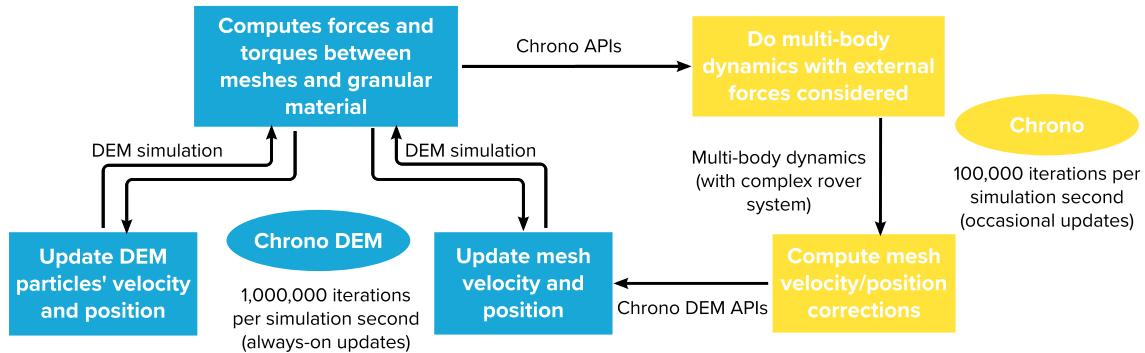
**Fig. 26** The clump shapes after enlarging their aspect ratio

The authors note the existence of studies on reproducing the locking effect in monodisperse material with modified material properties such as the rolling resistance [80]. Its applicability in rover mobility simulations is unclear, but even if feasible, it necessitates extra calibration stages that require field test data for each terrain configuration of interest. By using clumps, the DS does not require field test data but rather a good approximation of the elements' shapes.

Finally, note that dilating the clump at constant thickness also reduces the slope associated with a given slip, albeit not as significantly as noted for monodisperse material. The findings also suggest that if high fidelity is needed, one should pay attention to the shape of the elements. This can be nontrivial, particularly if the granular material does not have a known shape distribution, an aspect that can be addressed through a shape calibration process that falls outside the scope of this contribution.



**Fig. 27** Slip-on-incline test result with the clumps that have larger aspect ratios, compared against the base clump shape. The experimental data used for comparison (black line) are from Glenn Research Center's MGRU3 experiments with the GRC-1 simulant



**Fig. 28** The co-simulation workflow between the multi-body system simulated by Chrono and the new DEM simulator

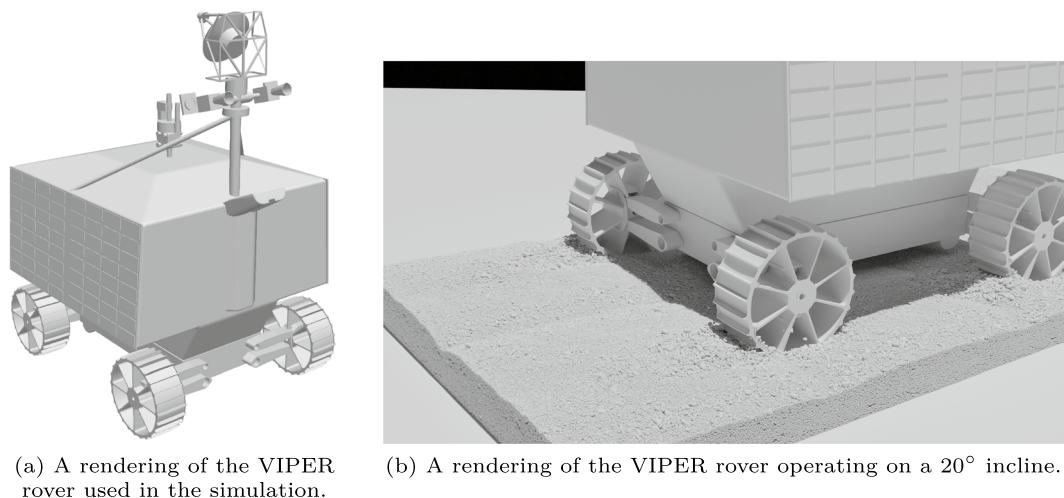
## 5.2 Full-rover slip test on incline

This section discusses a slope vs. slip test carried out for a full rover. This calls for a co-simulation between a multi-body system (the rover) and a DEM system (the terrain). For the multi-body system, since the MGRU3 CAD model was inaccessible, the authors instead used a similar VIPER rover model publicly available in the latest Chrono distribution [55].

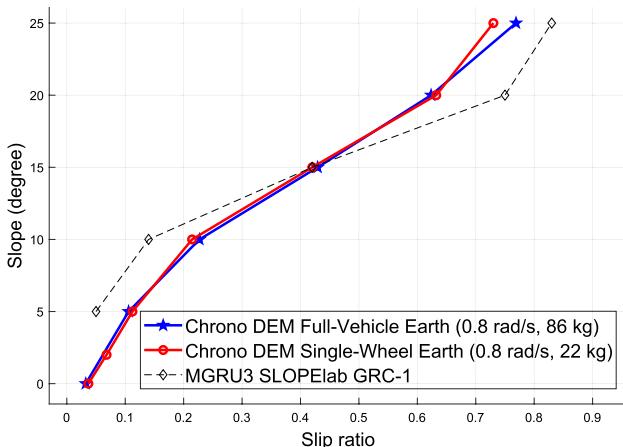
The co-simulation setup is shown in Fig. 28. The DEM simulator handles the evolution of the granular terrain, while Chrono deals with the rover dynamics. The two simulators are bridged through the meshes that represent the wheels. The DEM simulator calculates the force exerted by the terrain on the wheel mesh. The force information is used when the Chrono numerical integrator propagates forward in time the evolution of the meshes. In turn, the new position of the wheels will become a set of updated boundary conditions for the granular material. The rover's mobility is also influenced by forces that crop up in the chassis and suspension

that are independent of the motion of the granular terrain. In this co-simulation, the multi-body rover system in Chrono is advanced by one time step for every ten DEM time steps. Shown in Fig. 29a, the rover is roughly 2.1 m in length, 1.5 m in width, and 1.4 m in height (excluding the antenna). Four wheels are connected to the rover body through revolute joints. The wheel geometry is from MGRU3, the same as that in Sect. 5.1. The rover moves around by prescribing all its four wheels a 0.8 rad/s angular velocity on inclines of 0°, 5°, 10°, 15°, 20°, and 25°. A rendering of a test scene is provided in Fig. 29b.

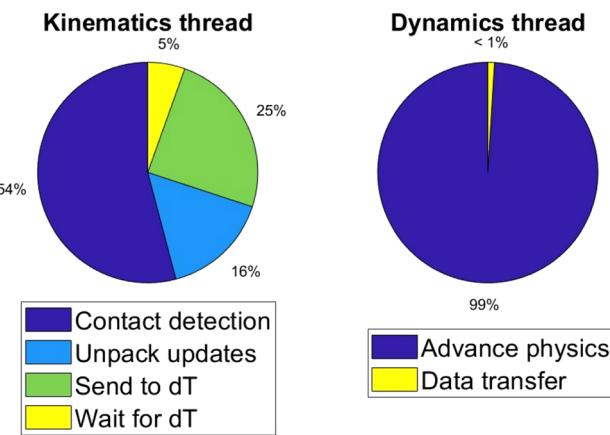
Note that the full-rover slip data shown in Fig. 30 displays no notable difference compared to the single-wheel counterpart. This suggests that the more expeditious single-wheel tests are likely sufficient to gain insights into the rover's mobility attributes. The slip ratio increases relatively slowly with the slope angle in the interval between 0° and 10°. Past 10°, this rate of increase escalates, and the rover almost fails to climb on a 25° incline. Finally, the DS for this test uses 11,336,638 DEM elements that employ 34,691,952



**Fig. 29** The full-rover test using the DS



**Fig. 30** VIPER rover slip-on-incline test result, compared against the single-wheel test done in the same conditions. The experimental data used for comparison (black line) are from Glenn Research Center's MGRU3 experiments with the GRC-1 simulant



**Fig. 31** The runtime breakdown for the kinematics and dynamics threads, during the lifespan of the full-rover simulation

component spheres. On two NVIDIA A100 GPUs, a 15-second simulation requires approximately 109 h of run time.

Figure 31 shows the itemized time consumption associated with the important steps in the kinematics and dynamics threads' work cycles. The analysis is similar to that shown in Sect. 4.2. For the rover simulation, the amount of mutual contact data produced is relatively large, causing the kinematics thread to spend more time on transferring it to the dynamics thread, reaching 25% of the former thread's total runtime. The dynamics thread continues to spend minimal time on transferring data. This was done by design to enable the dynamics thread to almost exclusively focus on advancing the state of the system forward in time.

## 6 Summary and future work

This work reports results obtained with a new GPU-accelerated DEM solver that uses a clump-based representation of the discrete elements. The new DEM simulator is showcased in conjunction with a granular material similar to the GRC-1 simulant which is modeled using elements of nontrivial shapes. Given that particle sizes in GRC-1 go down to the micron level, the DS used herein qualitatively embraces the element size distribution of GRC-1 but increases all element sizes by a factor of 20. Despite this size augmentation, the DS is shown to behave similarly to GRC-1 in several tests, i.e., angle of repose, cone penetration, drawbar pull, and single-wheel and full-rover incline slip vs. slope tests. The DEM simulations scale up to tens of millions of elements that have nontrivial shapes obtained by clumping together spheres of various radii. The DEM simulator works in the Chrono framework and can leverage two GPUs simultaneously. The simulator and the code for all simulations discussed are open-source and available at [63].

A salient observation is that the simulation methodology advanced by the DEM simulator is physics-based. One only needs to specify the parameters  $E$ ,  $v$ ,  $\mu$ , and CoR to be able to carry out the simulation. Moreover, these parameters can be specified on a per-sphere basis rather than on a per-element basis, which opens the door to simulating complex discrete elements that involve particles of different shapes and different materials. To the best of our knowledge, the simulator is unique in this regard. A detailed description of the implementation of the new simulator, which touches on its data structures, Python wrapper, the different data types used to store state information, gains associated with using two computational threads (the kinematics and dynamics one, that is) will be provided in a follow-up contribution. Finally, particle breakage and custom contact force models are supported, and they will also be discussed in this follow-up contribution.

There are three directions in which this effort will be continued. First, the authors plan to generate models associated with other extraterrestrial simulants, e.g., GRC-3 and Mojave Mars simulant [81], and then assess their terramechanics attributes following the workflow established in this contribution. In this context, when moving to full rover analysis or heavy wheels, one will have to also increase in simulation the depth of the bed of granular material. For the 22 kg single wheel tests, the thickness of the bed was 15 cm. For heavy implements, this value will have to be increased to avoid boundary effects noted in practice. Second, the authors are in the process of carrying out a systematic study to gauge how various wheel design attributes, e.g., wheel shape/size/width, and grouser count/height/width/shape, influence the

mobility of a rover operating on lunar terrain. The focus is on how a wheel design changes rover attributes such as energy consumption, steering, slip initiation, and climbing ability. Lastly, it would be insightful to better understand how the high-fidelity DEM simulation results can be used to design data-driven, expeditious wheel–terrain interaction methods.

**Acknowledgements** The authors would like to thank Colin Creager of NASA Glenn for his insights and feedback on this manuscript.

**Funding** This work has been partially supported by NSF projects OAC2209791 and CISE1835674, US Army Research Office project W911NF1910431, and NASA Small Business Technology Transfer (STTR) Sequential Contract #80NSSC20C0252.

**Code availability** The code used to reproduce the results in this work is available in a GitHub public repository [68]. The DEM simulator used in this work is also available in a GitHub public repository [63] for unrestricted use and distribution.

**Data availability** The datasets in this study can be generated by running the open-source code. The raw data analyzed during the study are also available from the corresponding author upon reasonable request.

## Declarations

**Conflict of interest** The authors have no competing interests to declare that are relevant to the content of this article.

## References

- Cundall PA, Strack OD (1979) A discrete numerical model for granular assemblies. *Geotechnique* 29(1):47–65
- Pöschel T, Schwager T (2005) Computational granular dynamics: models and algorithms. Springer, Berlin
- Lemieux M, Léonard G, Doucet J, Leclaire L-A, Viens F, Chaouki J, Bertrand F (2008) Large-scale numerical investigation of solids mixing in a v-blender using the discrete element method. *Powder Technol* 181(2):205–216
- Apostolou K, Hrymak A (2008) Discrete element simulation of liquid-particle flows. *Comput Chem Eng* 32(4–5):841–856
- Tang C-L, Hu J-C, Lin M-L, Angelier J, Lu C-Y, Chan Y-C, Chu H-T (2009) The Tsaoling landslide triggered by the Chi-Chi earthquake, Taiwan: insights from a discrete element simulation. *Eng Geol* 106(1–2):1–19
- Salciarini D, Tamagnini C, Conversini P (2010) Discrete element modeling of debris-avalanche impact on earthfill barriers. *Phys Chem Earth Parts A/B/C* 35(3–5):172–181
- O’Sullivan C (2011) Particle-based discrete element modeling: geomechanics perspective. *Int J Geomech* 11(6):449–464
- Sánchez P, Scheeres DJ (2011) Simulating asteroid rubble piles with a self-gravitating soft-sphere distinct element method model. *Astrophys J* 727(2):120
- Foldager FF, Munkholm LJ, Balling O, Serban R, Negruț D, Heck RJ, Green O (2022) Modeling soil aggregate fracture using the discrete element method. *Soil Tillage Res* 218:105295
- Recuero AM, Serban R, Peterson B, Sugiyama H, Jayakumar P, Negruț D (2017) A high-fidelity approach for vehicle mobility simulation: nonlinear finite element tires operating on granular material. *J Terramech* 72:39–54. <https://doi.org/10.1016/j.jterra.2017.04.002>
- Johnson JB, Kulchitsky AV, Duvoy P, Iagnemma K, Senatore C, Arvidson RE, Moore J (2015) Discrete element method simulations of Mars exploration rover wheel performance. *J Terramech* 62:31–40
- OpenMP (2021) Specification Standard 5.2. Available online at <http://openmp.org/>. Accessed 4 Jan 2024
- Amritkar A, Deb S, Tafti D (2014) Efficient parallel CFD-DEM simulations using OpenMP. *J Comput Phys* 256:501–519
- Knuth MA, Johnson J, Hopkins M, Sullivan R, Moore J (2012) Discrete element modeling of a mars exploration rover wheel in granular material. *J Terramech* 49(1):27–36
- Message Passing Interface Forum: MPI: a message-passing interface standard version 3.0. Chapter author for Collective Communication, Process Topologies, and One Sided Communications (2012)
- Yan B, Regueiro RA (2018) A comprehensive study of MPI parallelism in three-dimensional discrete element method (DEM) simulation of complex-shaped granular particles. *Comput Part Mech* 5(4):553–577
- Checkaraou AWM, Rousset A, Besseron X, Varrette S, Peters B (2018) Hybrid MPI+ OpenMP implementation of extended discrete element method. In: 2018 30th International symposium on computer architecture and high performance computing (SBAC-PAD), Lyon, France. IEEE, pp 450–457
- LIGGGHTS (2013) Open source discrete element method particle simulation code. <https://www.cfdem.com/liggghts-open-source-discrete-element-method-particle-simulation-code>. Accessed 4 Jan 2024
- LAMMPS (2013) A molecular dynamics simulator. <http://lammps.sandia.gov/>. Accessed 4 Jan 2024
- Serban R, Olsen N, Negruț D (2017) High performance computing framework for co-simulation of vehicle-terrain interaction. In: NDIA ground vehicle systems engineering and technology symposium
- Xu J, Qi H, Fang X, Lu L, Ge W, Wang X, Xu M, Chen F, He X, Li J (2011) Quasi-real-time simulation of rotating drum using discrete element method with parallel GPU computing. *Particuology* 9(4):446–450
- Govender N, Wilke D, Kok S (2016) Blaze-DEMGPU: modular high performance DEM framework for the GPU architecture. *SoftwareX* 5:62–66
- Gan J, Zhou Z, Yu A (2016) A GPU-based DEM approach for modeling of particulate systems. *Powder Technol* 301:1172–1182
- He Y, Evans T, Yu A, Yang R (2018) A GPU-based DEM for modeling large scale powder compaction with wide size distributions. *Powder Technol* 333:219–228
- Iwashita K, Oda M (1998) Rolling resistance at contacts in simulation of shear band development by DEM. *J Eng Mech* 124(3):285–292
- Renzo AD, Maio FPD (2004) Comparison of contact-force models for the simulation of collisions in DEM-based granular flow codes. *Chem Eng Sci* 59(3):525–541
- Cruz F, Emam S, Prochnow M, Roux JN, Chevoir F (2005) Rheophysics of dense granular materials: discrete simulation of plane shear flows. *Phys Rev E* 72:021309. <https://doi.org/10.1103/PhysRevE.72.021309>
- Rycroft CH, Grest GS, Landry JW, Bazant MZ (2006) Analysis of granular flow in a pebble-bed nuclear reactor. *Phys Rev E* 74:021306
- Kruggel-Emden H, Sturm M, Wirtz S, Scherer V (2008) Selection of an appropriate time integration scheme for the discrete element method (DEM). *Comput Chem Eng* 32(10):2263–2279
- Wasfy TM, Wasfy HM, Peters JM (2014) Coupled multibody dynamics and discrete element modeling of vehicle mobility on

- cohesive granular terrains. In: ASME 2014 international design engineering technical conferences and computers and information in engineering conference. American Society of Mechanical Engineers, pp 006–1005000610050. <http://proceedings.asmedigitalcollection.asme.org/proceeding.aspx?articleid=2091049>. Accessed 4 Jan 2024
31. Lommen S (2014) DEM speedup: stiffness effects on behavior of bulk material. *Particuology* 12(1):107–112
  32. Utili S, Zhao T, Housby GT (2015) 3D DEM investigation of granular column collapse: evaluation of debris motion and its destructive power. *Eng Geol* 186:3–16
  33. Potticary M, Zervos A, Harkness J (2015) An investigation into the effect of particle plattness on the strength of granular material using the discrete element method. In: IV International conference on particle-based methods—fundamentals and applications. <https://eprints.soton.ac.uk/394117/1/particles2015.pdf>. Accessed 4 Jan 2024
  34. Michael M, Vogel F, Peters B (2015) DEM-FEM coupling simulations of the interactions between a tire tread and granular terrain. *Comput Methods Appl Mech Eng* 289:227–248
  35. Ciantia M, Arroyo M, Butlanska J, Gens A (2016) DEM modelling of cone penetration tests in a double-porosity crushable granular material. *Comput Geotech* 73:109–127
  36. Zheng Z, Zang M (2017) Numerical simulations of the interactions between a pneumatic tire and granular sand by 3D DEM-FEM. In: 7th International conference on discrete element methods, pp 289–300. [https://link.springer.com/chapter/10.1007/978-981-10-1926-5\\_32](https://link.springer.com/chapter/10.1007/978-981-10-1926-5_32). Accessed 4 Jan 2024
  37. Parteli E, Poschel T (2016) Particle-based simulation of powder application in additive manufacturing. *Powder Technol* 288:96–102
  38. Kivugo R (2017) Tire-soil interaction for off-road vehicle applications. Phd, Politecnico di Milano. <https://www.politesi.polimi.it/handle/10589/136229>
  39. Calvetti F, Prisco C, Vairaktaris E (2016) DEM assessment of impact forces of dry granular masses on rigid barriers. *Acta Geotech* 12:129–144
  40. Furuchi M, Nishiura D, Kuwano O, Bauville A, Hori T, Sakaguchi H (2018) Arcuate stress state in accretionary prisms from real-scale numerical sandbox experiments. *Sci Rep* 8:8685
  41. Ericson C (2005) Real time collision detection. Morgan Kaufmann, San Francisco
  42. Favier J, Abbaspour-Fard M, Kremmer M, Raji A (1999) Shape representation of axi-symmetrical, non-spherical particles in discrete element simulation using multi-element model particles. *Eng Comput* 16(4):467–480
  43. Hilton J, Cleary P (2011) The influence of particle shape on flow modes in pneumatic conveying. *Chem Eng Sci* 66(3):231–240
  44. Kiangi K, Potapov A, Moys M (2013) DEM validation of media shape effects on the load behaviour and power in a dry pilot mill. *Miner Eng* 46:52–59
  45. Ren B, Zhong W, Jin B, Shao Y, Yuan Z (2013) Numerical simulation on the mixing behavior of corn-shaped particles in a spouted bed. *Powder Technol* 234:58–66
  46. Zhong W, Yu A, Liu X, Tong Z, Zhang H (2016) DEM/CFD-DEM modelling of non-spherical particulate systems: theoretical developments and applications. *Powder Technol* 302:108–152
  47. Kawamoto R, Andò E, Viggiani G, Andrade JE (2018) All you need is shape: predicting shear banding in sand with LS-DEM. *J Mech Phys Solids* 111:375–392
  48. Marteau E, Andrade JE (2021) An experimental study of the effect of particle shape on force transmission and mobilized strength of granular materials. *J Appl Mech* 88(11):111009
  49. Altair (2023) EDEM software. <https://altair.com/edem>. Accessed 4 Jan 2024
  50. Maione R, Kiesgen De Richter S, Mauviel G, Wild G (2015) DEM investigation of granular flow and binary mixture segregation in a rotating tumbler: influence of particle shape and internal baffles. *Powder Technol* 286:732–739. <https://doi.org/10.1016/j.powtec.2015.09.011>
  51. Guo X, Zheng Z, Zang M, Chen S (2022) A multi-sphere DE-FE method for traveling analysis of an off-road pneumatic tire on irregular gravel terrain. *Eng Anal Bound Elem* 139:293–312. <https://doi.org/10.1016/j.enganabound.2022.04.003>
  52. Mazhar H, Heyn T, Negru D (2011) A scalable parallel method for large collision detection problems. *Multibody Syst Dyn* 26:37–55. <https://doi.org/10.1007/s11044-011-9246-y>
  53. Kelly C, Olsen N, Negru D (2020) Billion degree of freedom granular dynamics simulation on commodity hardware via heterogeneous data-type representation. *Multibody Syst Dyn* 50:355–379
  54. Fang L, Zhang R, Vanden Heuvel C, Serban R, Negru D (2021) Chrono::GPU: an open-source simulation package for granular dynamics using the discrete element method. *Processes*. <https://doi.org/10.3390/pr9101813>
  55. Tasora A, Serban R, Mazhar H, Pazouki A, Melanz D, Fleischmann J, Taylor M, Sugiyama H, Negru D (2016) Chrono: an open source multi-physics dynamics engine. In: Kozubek T (ed) High performance computing in science and engineering. Lecture notes in computer science. Springer, Berlin, pp 19–49
  56. Hertz H (1882) Ueber die verdunstung der flüssigkeiten, insbesondere des quecksilbers, im luftleeren raume. *Annalen der Physik* 253(10):177–193. <https://doi.org/10.1002/andp.18822531002>. <https://onlinelibrary.wiley.com/doi/pdf/10.1002/andp.18822531002>. <https://arxiv.org/abs/onlinelibrary.wiley.com/doi/pdf/10.1002/andp.18822531002>
  57. Mindlin R, Deresiewicz H (1953) Elastic spheres in contact under varying oblique forces. *J Appl Mech* 20:327–344
  58. Fang L, Negru D (2021) Producing 3D friction loads by tracking the motion of the contact point on bodies in mutual contact. *Comput Part Mech* 8:905–929. <https://doi.org/10.1007/s40571-020-00376-9>
  59. Fleischmann J, Serban R, Negru D, Jayakumar P (2016) On the importance of displacement history in soft-body contact models. *J Comput Nonlinear Dyn* 11(4):044502
  60. Price M, Murariu V, Morrison G (2007) Sphere clump generation and trajectory comparison for real particles. In: Proceedings of discrete element modelling 2007
  61. Reger D, Merzari E, Balestra P, Stewart R, Strydom G (2023) Discrete element simulation of pebble bed reactors on graphics processing units. *Ann Nucl Energy* 190:109896. <https://doi.org/10.1016/j.anucene.2023.109896>
  62. Kelly C, Olsen N, Vanden Heuvel C, Serban R, Negru D (2019) Towards the democratization of many-body dynamics: billion degree of freedom simulation of granular material on commodity hardware. In: Proceeding of the ECCOMAS multibody dynamics conference, Duisburg, Germany
  63. Zhang R, Vanden Heuvel C, Negru D (2022) DEM-engine, a multi-GPU DEM solver with complex geometry support. Simulation-Based Engineering Laboratory, University of Wisconsin-Madison. <https://github.com/projectchrono/DEM-Engine>
  64. NVIDIA Corporation (2021) Compute unified device architecture toolkit documentation. <https://docs.nvidia.com/cuda>. Accessed 4 Jan 2024
  65. NVIDIA (2023) NVIDIA multi-instance GPU user guide. NVIDIA. Available online at <https://docs.nvidia.com/datacenter/tesla/mig-user-guide/>. Accessed 4 Jan 2024
  66. SIEMENS (2021) Simcenter STAR-CCM+ software. <https://plm.sw.siemens.com/en-US/simcenter/fluids-thermal-simulation/star-ccm>. Accessed 4 Jan 2024

67. Oravec HA, Zeng X, Asnani VM (2010) Design and characterization of GRC-1: a soil for lunar terramechanics testing in earth-ambient conditions. *J Terramech* 47(6):361–377. <https://doi.org/10.1016/j.jterra.2010.04.006>
68. Zhang R, Vanden Heuvel C, Negrut D (2022) DEM-Engine's GRC-1-inspired digital simulant scripts. Simulation-Based Engineering Laboratory, University of Wisconsin-Madison. [https://github.com/projectchrono/DEM-Engine/tree/Paper\\_Digital\\_Simulant](https://github.com/projectchrono/DEM-Engine/tree/Paper_Digital_Simulant)
69. Beakawi Al-Hashemi HM, Baghabra Al-Amoudi OS (2018) A review on the angle of repose of granular materials. *Powder Technol* 330:397–417. <https://doi.org/10.1016/j.powtec.2018.02.003>
70. Steiner A, Kopf AJ, Henry P, Stegmann S, Apprioual R, Pelleau P (2015) Cone penetration testing to assess slope stability in the 1979 Nice landslide area (Ligurian Margin, SE France). *Mar Geol* 369:162–181. <https://doi.org/10.1016/j.margeo.2015.08.008>
71. Oravec H, Asnani V, Zeng X (2008) The development of a soil for lunar surface mobility testing in ambient conditions. In: *Earth & space 2008: engineering, science, construction, and operations in challenging environments*. American Society of Civil Engineers, pp 1–10. [https://doi.org/10.1061/40988\(323\)12](https://doi.org/10.1061/40988(323)12)
72. Senatore C, Stein N, Zhou F, Bennett K, Arvidson R, Trease B, Lindemann R, Bellutta P, Heverly M, Iagnemma K (2014) Modeling and validation of mobility characteristics of the mars science laboratory curiosity rover. In: *Proceedings of the 12th international symposium on artificial intelligence, robotics and automation in space (i-SAIRAS)*
73. Creager C, Asnani V, Oravec H, Woodward A (2017) Drawbar pull (DP) procedures for off-road vehicle testing. Technical Report GRC-E-DAA-TN31725, NASA
74. Project Chrono (2020) Chrono: an open source framework for the physics-based simulation of dynamic systems. <http://projectchrono.org>. Accessed 03 Mar 2020
75. Simulated Lunar Operations Laboratory: NASA's VIPER prototype motors through moon-like obstacle course. <https://www.nasa.gov/feature/ames/nasas-viper-prototype-motors-through-moon-like-obstacle-course>. Accessed 02 Apr 2023
76. Slonaker J, Motley DC, Senatore C, Iagnemma K, Kamrin K (2016) Geometrically general scaling relations for locomotion on granular beds. *Phys Rev E* 95:052901
77. Slonaker J, Motley DC, Zhang Q, Townsend S, Senatore C, Iagnemma K, Kamrin K (2017) General scaling relations for locomotion in granular media. *Phys Rev E* 95:052901. <https://doi.org/10.1103/PhysRevE.95.052901>
78. Bowden FP, Tabor D (1951) The friction and lubrication of solids. *Am J Phys* 19(7):428–429. <https://doi.org/10.1119/1.1933017>
79. Deng K, Ko WH (1992) A study of static friction between silicon and silicon compounds. *J Micromech Microeng* 2:14–20
80. Xie C, Ma H, Zhao Y (2019) Investigation of modeling non-spherical particles by using spherical discrete element model with rolling friction. *Eng Anal Bound Elem* 105:207–220. <https://doi.org/10.1016/j.enganabound.2019.04.013>
81. Peters GH, Abbey W, Bearman GH, Mungas GS, Smith JA, Anderson RC, Douglas S, Beegle LW (2008) Mojave Mars simulant—characterization of a new geologic Mars analog. *Icarus* 197(2):470–479. <https://doi.org/10.1016/j.icarus.2008.05.004>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.