

A unified numerical framework for rigid and compliant granular materials

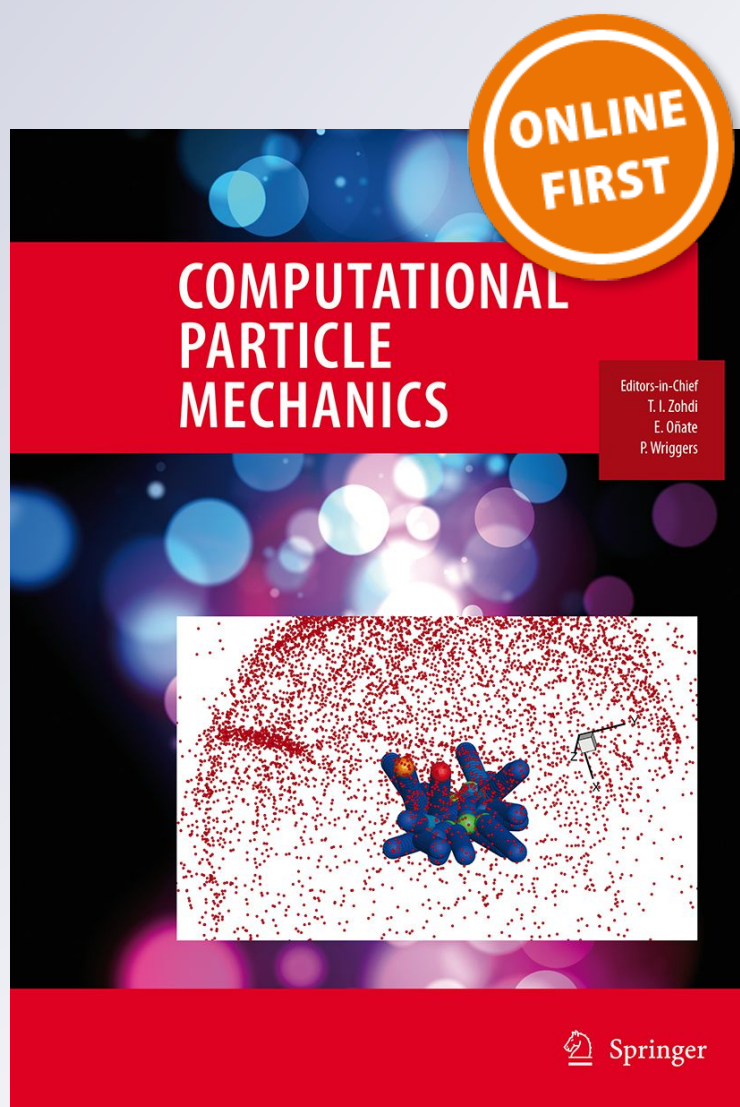
Guilhem Mollon

Computational Particle Mechanics

ISSN 2196-4378

Comp. Part. Mech.

DOI 10.1007/s40571-018-0187-6



Your article is protected by copyright and all rights are held exclusively by OWZ. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your article, please use the accepted manuscript version for posting on your own website. You may further deposit the accepted manuscript version in any repository, provided it is only made publicly available 12 months after official publication or later and provided acknowledgement is given to the original source of publication and a link is inserted to the published article on Springer's website. The link must be accompanied by the following text: "The final publication is available at link.springer.com".



A unified numerical framework for rigid and compliant granular materials

Guilhem Mollon¹ 

Received: 5 December 2017 / Accepted: 5 February 2018
© OWZ 2018

Abstract

A numerical framework for the simulation of granular materials composed of mixed rigid and compliant grains is presented in this paper. This approach is based on a multibody meshfree technique, coupled in a very natural way with classic concepts from the discrete element method. The equations of motion (for the rigid grains) and of continuum mechanics (for the compliant ones) are solved using an adaptive explicit scheme, in fully dynamic conditions. The parallelization strategy is described and tested on an illustrative simulation involving both kinds of grains.

Keywords Granular materials · Meshfree methods · Discrete element modelling · Multibody dynamics

1 Introduction

Numerical simulation of granular materials has been the subject of an important research effort from the scientific community for several decades. For standard granular materials (such as sand), the most relevant tool that arose from this effort is the discrete element method (DEM). It was first proposed in a celebrated paper by Cundall and Strack [1] and has been widely applied and improved ever since in order to deal with complex [2,3] and realistic [4,5] grain shapes, grain breakage [6], fluid coupling [7], and physico-chemical interactions [8], among others. The fields of application of DEM are very various, including geomechanics [9,10], geophysics [11], food and health industries [12], transportation industry [13], tribology [14,15], environmental sciences [16], physics of condensed matter [17], etc.

The main assumption of DEM is that each grain composing the granular material behaves as a rigid solid. Thanks to this assumption, the behaviour of a whole sample can be predicted using simple Newtonian dynamics and phenomenological contact laws (including such phenomena as friction, damping, adhesion, etc.). These dynamics are most often solved numerically using explicit solvers, although impulse-based implicit solvers do exist as well [18]. For this

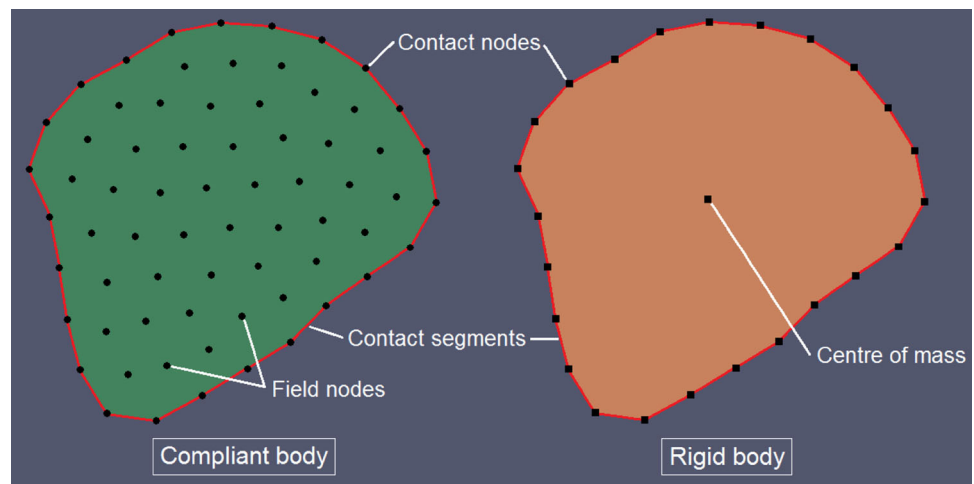
assumption to be acceptable, however, the individual deformation of each grain should take a negligible part into the deformation of the whole media, and most of the kinematics should be related to interparticle motions. This is true as long as the applied load remains low relatively to the stiffness of the material composing each grain.

There are several situations, however, for which this condition does not apply: compaction and sintering of ductile powders [19], biological fluids [20,21], tribological third-bodies [22], etc. When the material composing the grains is very compliant or the load is very high, the deformation of each grain should be accounted for. Finite-element method (FEM)-based approaches have been proposed [23–25], but they remain somewhat limited by the fact that common FEM softwares are not intended to deal with a very large number of bodies (typically more than ten thousands in DEM). Another approach based on a multibody meshfree technique was proposed recently [26]. It borrows from continuous modelling the idea of user-defined stress-strains relations (albeit at the grain scale instead of the macroscopic one), but retains the idea of a discrete description of the material in terms of grains interacting by the means of contacts. In this approach, each grain is a deformable body, and the displacement field is interpolated on the domain of each grain using a meshfree description based on a certain number of field nodes with two degrees of freedom in displacement each. The classical equations of continuum mechanics are solved using a weak formulation as in a typical FEM code, including geometric nonlinearities and inertial terms. In the first implemented

✉ Guilhem Mollon
guilhem.mollon@insa-lyon.fr

¹ Université de Lyon, LaMCoS, INSA-Lyon, CNRS
UMR5259, 69621 Lyon, France

Fig. 1 Typical compliant and rigid bodies



version of this method [26], contacts were accounted for in an augmented Lagrangian framework using spline contact elements, and the whole system was solved in time using a composite implicit Newmark–Euler scheme. This technique proved useful in the resolution of systems composed of a large number of compliant bodies.

In this paper, we describe some new developments of this numerical framework (including a different choice of the shape functions, a new adaptive explicit solver, a different contact resolution, a new constitutive law, and more complex contact laws), and we describe in which manner it can be coupled with classical DEM in order to deal within a single simulation with both rigid and compliant grains. We also detail the strategy employed to parallelize the code and demonstrate its capabilities on a complex simulation involving both rigid and compliant grains in a tribological framework.

2 Compliant bodies

The framework which was extensively described in [26] is employed in the present study, with several evolutions which are detailed in this subsection. The main principles are kept: the code is currently limited to 2D plane-strain problems, and each compliant body is represented by a collection of field nodes (some of them being on the external border of the body, and the others being in its bulk) which carry the degrees of freedom in displacement in the same way as in FEM (Fig. 1). Between these nodes, the continuous displacement field is interpolated using meshfree shape functions. In contrast with finite-element meshes which only provide a very local connectivity to each node, each of these shape functions possesses a domain of influence which typically encompasses between 10 and 25 field nodes. In the first implementation, radial point interpolation method (RPIM)

shape functions were used, as proposed by [27]. RPIM shape functions are convenient because they possess the Kronecker-delta property, but present two limitations: (i) they are not positive everywhere (which may trigger oscillations in the interpolated fields), and (ii) they are submitted to local loss of continuity and differentiability at the boundaries of the connectivity domains of the field nodes. These shortcomings (shown in details in Fig. 2) have limited consequences in linearized mechanics, but may trigger some instabilities in the case of large displacements and deformations. For this reason, the new version of the code employs moving least squares (MLS) shape functions and is thus closer to the initial version of the element-free Galerkin (EFG) method proposed in [28] and developed in [29]. MLS shape functions are positive and differentiable everywhere, but they do not possess the Kronecker-delta property (Fig. 2). This well-known shortcoming leads to difficulties in the application of Dirichlet boundary conditions, but simple solutions do exist to overcome them. While the first EFG papers did use Lagrange multipliers, a penalty method is employed here and gives satisfaction. However, one should keep in mind that, because of this property, the displacements \mathbf{u} at the nodes are not equal to the values \mathbf{U} of the degrees of freedom at these nodes.

In [26], a Saint Venant–Kirchhoff constitutive relation (which states the linearity between the Green–Lagrange and the Piola–Kirchhoff tensors) was used for the sake of simplicity. It was, however, discarded in the new implementation, because it presents a softening behaviour at large strains, which leads to severe instabilities under some loading conditions. In the present paper, a hyper-elastic neo-Hookean model was used instead, in order to retain the elasticity of the bodies while dealing with very large strains in a stable manner. Depending on the requirements of each field of application, more complex constitutive laws (e.g. visco-elastic or elasto-plastic) may be implemented as well.

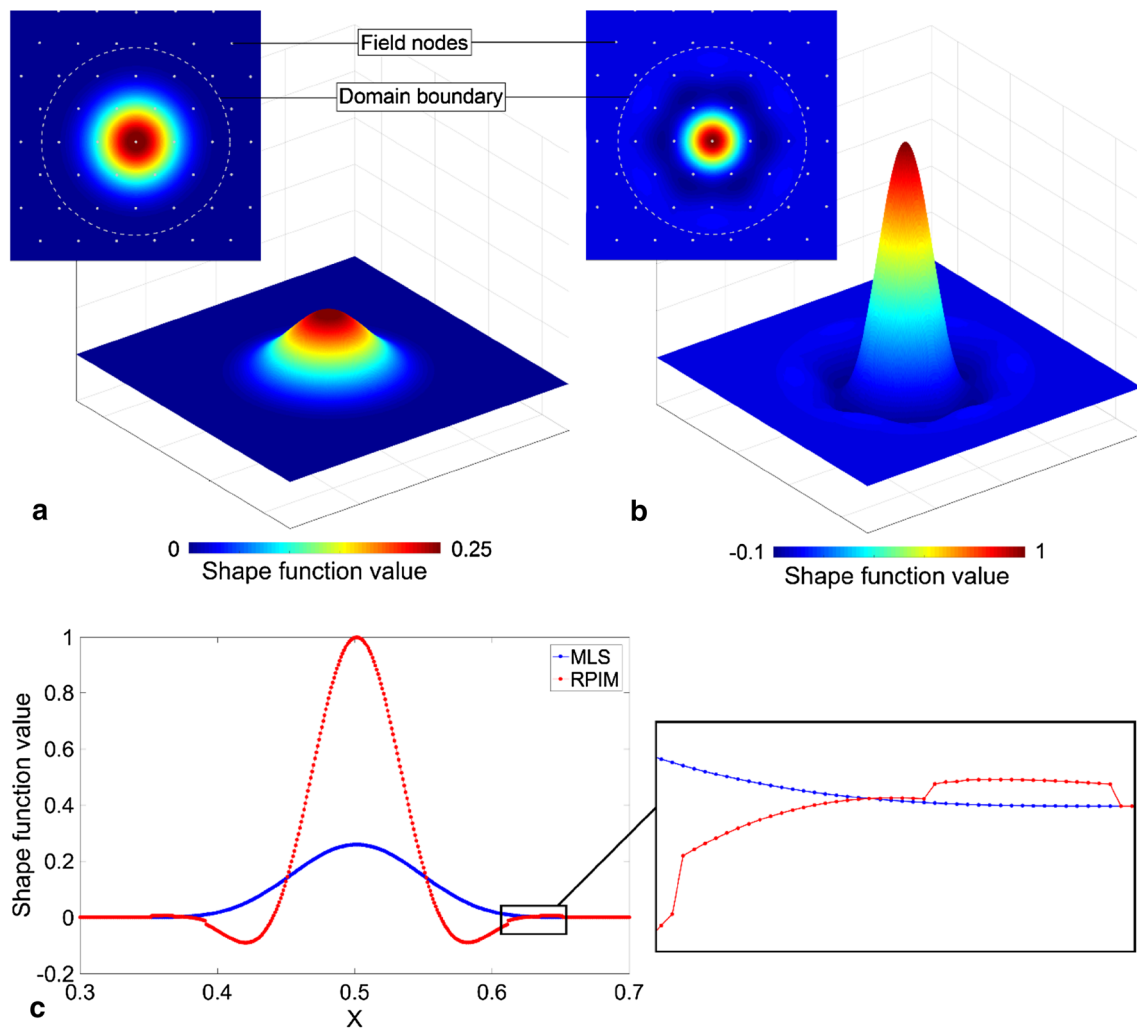


Fig. 2 Comparison between moving least square (MLS) and radial point interpolation method (RPIM) shape functions within a given compliant body; **a** MLS top and perspective views; **b** RPIM top and perspective views; **c** cross sections and detailed view showing the loss of continuity of RPIM shape functions

Based on the choices described above, the equations of continuum mechanics are expressed in a weak formulation, which is discretized with respect to the degrees of freedom of each field node. A Gaussian numerical integration performed on a triangular mesh mapped on the initial positions of the field nodes then leads to a system of ordinary differential equations, which is then solved explicitly in a manner described in Sect. 5.

3 Rigid bodies

Along with the compliant bodies, the numerical approach proposed in this paper is requested to deal with rigid bodies. They may be used either to represent the boundaries of the domain of the granular sample (e.g. the rigid moving walls of a box), or to represent rigid grains interacting with

the compliant ones (Fig. 1). In any case, the dynamics of such bodies are quite straightforward, since they only rely on Newton's second law of motion. Each rigid body carries three degrees of freedom including a rotation (since the current version of the code only deals with 2D problems), and is characterized by a centre of mass, a mass, and a rotational inertia. This framework is thus identical to classical DEM implementations. Besides, each body is delimited by a collection of nodes located on its external border and linked by segments. Hence, the external frontier of a rigid grain is identical to that of a compliant grain, and the same contact algorithms (described in the next section) may be used. Contact forces are obtained by a penalization method (along with a user-defined contact law), and transferred to the centre of mass for solving. Thanks to this technique, grains with arbitrary shapes may be introduced in simulations, in a simpler way than in most DEM codes which require that a compli-

cated shape be replaced by a collection of overlapping discs [30].

4 Contacts

To account for contacts between bodies, the nodes located on the external border of each grain (either rigid or compliant) are linked by segments in order to close this border. The domain of each body is thus delimited by a piecewise linear frontier, which makes it possible to apply classical penalty-based contact algorithms. Because of the possible existence of sharp corners in the initial or deformed shapes of the contacting grains [26], a node-to-segment formulation is employed, based on a three-stage detection process.

The first stage is a broad proximity detection based on a sweep-and-prune algorithm [31], which was chosen because of its ability to deal efficiently with grains with arbitrary shapes. The purpose of this algorithm is to define pairs of close grains. Is considered as close any pair of grain which overlaps both when projected on the x - and on the y -axis (Fig. 3).

The second stage is a close proximity detection, with the purpose of defining potential contact between any node of the contour of a grain B and any segment of another grain A (A and B compose one of the close pairs defined in the broad detection stage). This is done by a set of quick geometric tests, as shown in Fig. 4: each segment $A_i A_{i+1}$ has its own detection zone (defined by the union of the two circles of radius $A_i A_{i+1}/4$ and centred on A_i and A_{i+1} , and of the associated rectangle of dimensions $(A_i A_{i+1}) \times (A_i A_{i+1}/2)$), and any point B_j entering into this zone is considered as potentially in contact with the segment $A_i A_{i+1}$. These first

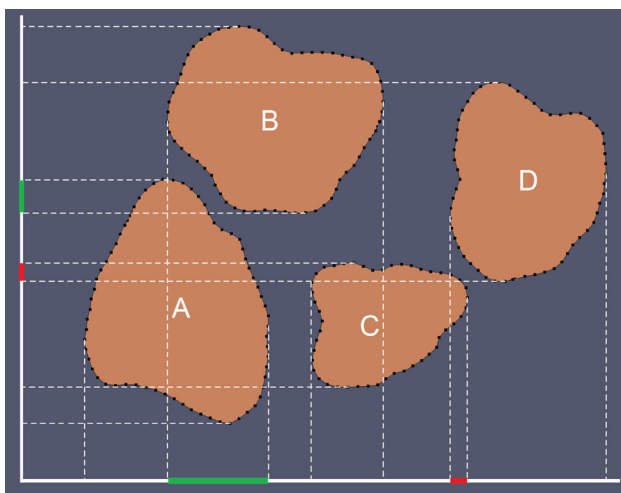


Fig. 3 Broad proximity detection stage: the only detected close pairs are A–B (green overlaps on the x - and y -axis) and C–D (red overlaps). (Color figure online)

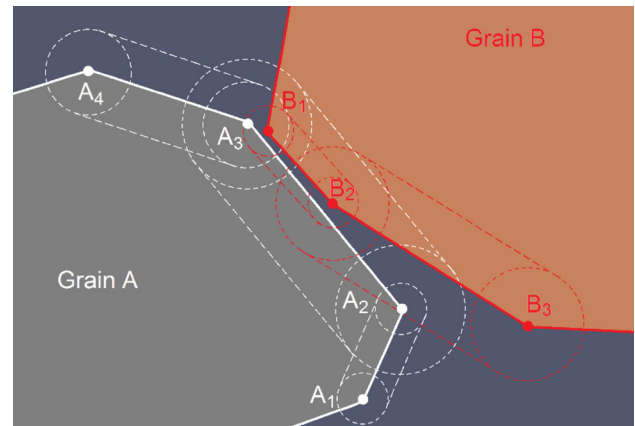


Fig. 4 Close proximity detection stage: node B_1 is detected as close to segments $A_2 A_3$ and $A_3 A_4$, node B_2 is detected as close to segment $A_2 A_3$, node A_2 is detected as close to segment $B_2 B_3$, and node A_3 is detected as close to segment $B_1 B_2$

two stages are not performed at every time step of the solver, but rather at a certain time-period which is user-defined in order to reduce the computational cost while ensuring that no contact is missed.

Conversely, the third stage is called at every time step and consists in performing the actual contact detection (Fig. 5). For any point B_j of a body B potentially contacting a segment $A_i A_{i+1}$ of a body A, its purpose is to determine the closest projection \tilde{B}_j , the signed gap γ_j (negative if B_j is inside the body A), the corresponding outward normal vector \vec{n}_j , and the tangential parameter ξ_j defining the position of \tilde{B}_j on the segment $A_i A_{i+1}$. In order to avoid loss of detection or double detection at the (possibly sharp) corners, a rigorous detection box is defined for each segment $A_i A_{i+1}$. This box is defined by the two bisecting lines arising from A_i and A_{i+1} , and by two lines parallel to the segment $A_i A_{i+1}$ at a distance of $A_i A_{i+1}/2$ in the outwards direction and $A_i A_{i+1}/10$ in the inward direction (the lower distance in the inward direction is designed to avoid false detection in the case of very thin structures). This contact box allows to determine rigorously a unique projection \tilde{B}_j of any node B_j (either orthogonally or towards one of the extremities of the segment) and to compute in a univocal way the quantities \vec{n}_j , γ_j and ξ_j which are required in order to apply a contact law.

In all the three stages described above, there is no univocal master and slave status for contacting bodies since a two-pass algorithm is used: each body A or B within a close pair will successively be master and slave, as each node of A is tested against all segments of B and each node of B against all segments of A during the second stage. This precaution is necessary to avoid undesired penetrations of sharp edges (in contrast with classical FEM contacts into which sharp corners are usually avoided by appropriate refinement and curvature and a one-pass logic is more appropriate). Several contact laws are implemented in the current version of the

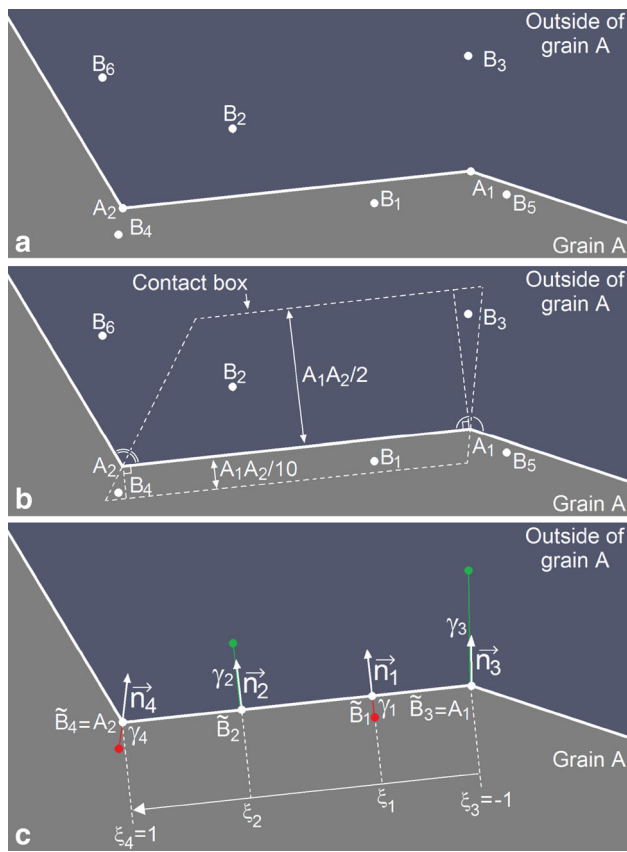


Fig. 5 Contact detection stage; **a** we suppose that nodes B_1 to B_6 of a grain B were detected as close to segment A_1A_2 of a grain A during the close-detection stage; **b** only nodes B_1 to B_4 are actually within the contact box of this segment; **c** closest projections (on the segment for B_1 and B_2 , on the nodes for B_3 and B_4) allow computation of local contact quantities (outward normal vector \vec{n}_j , signed gap γ_j , local coordinate ξ_j)

code: frictionless, Mohr–Coulomb, cohesive, cohesive with fatigue damage, etc. Each one of these laws will provide a contact pressure (in contrast with the classical DEM framework, which deals with contact forces), and the actual contact force is the product of this contact pressure by the contour length associated with the penetrating node (i.e. the sum of the half-lengths of the two segments which the node belongs to). On the segment submitted to penetration, the inverse contact force is shared by the two nodes based on the value of the ξ_j parameter. Finally, if bodies A and B are compliant, the MLS-based meshfree framework requires to distribute these forces on the nodes located in the neighbourhood of the three nodes A_i , A_{i+1} , and B_j , according to the nodal values of their respective shape functions [26].

5 Solver

An explicit solver is used to integrate in time the motion and the deformation of each body. It is based on the following algorithm, with a time step Δt :

$$\mathbf{M}^{-1} \cdot {}^t\mathbf{F} \rightarrow {}^t\ddot{\mathbf{U}} \quad (1)$$

$${}^t\ddot{\mathbf{U}} + \Delta t \cdot {}^t\ddot{\mathbf{U}} \rightarrow {}^{t+\Delta t}\ddot{\mathbf{U}} \quad (2)$$

$${}^t\mathbf{U} + \Delta t \cdot {}^{t+\Delta t}\ddot{\mathbf{U}} \rightarrow {}^{t+\Delta t}\mathbf{U} \quad (3)$$

$${}^{t+\Delta t}\mathbf{U} \rightarrow {}^{t+\Delta t}\mathbf{u} \quad (4)$$

In these expressions, ${}^t\mathbf{F}$ is the vector of the generalized forces applied to each degree of freedom (DOF) of the system (i.e. one of the two translational DOF of a field node of a compliant body, one of the two translational DOF of a rigid body, or its rotational DOF) at the time t , and ${}^t\mathbf{U}$ is the corresponding degree of freedom. \mathbf{M}^{-1} is the inverse of the mass matrix for all the DOF (with a diagonal lumping for the DOF related to the compliant bodies, as usually done in explicit solvers). This matrix is diagonal and is computed only once at the beginning of the simulation and stored for use at each time step. Equation (4) is related to the computation of the actual displacements ${}^{t+\Delta t}\mathbf{u}$ of each node (i.e. field nodes for compliant bodies and border nodes for rigid ones) from the degrees of freedom (i.e. DOF at field nodes for compliant bodies and at the centre of mass for rigid ones).

This scheme is well known for being conditionally stable, with a critical time step strongly related to the density of the material, to its stiffness, to the distance between neighbouring nodes, and to the value of the contact stiffness used in the penalty method. In order to optimize the numerical efficiency of the code, this solver is rendered adaptive by computing at each time step an error estimator. For this purpose, the scheme is first applied exactly in the manner described in Eqs. (1–3), and the resulting values of the displacements ${}^{t+\Delta t}\mathbf{u}_1$ are stored in memory. Then, the scheme is applied with a time step divided by two. Displacements ${}^{t+\Delta t/2}\mathbf{u}$ are thus obtained, and used to update the forces ${}^{t+\Delta t/2}\mathbf{F}$. The scheme is applied again between times $t + \Delta t/2$ and $t + \Delta t$, in order to obtain a new estimation of the displacements ${}^{t+\Delta t}\mathbf{u}_2$. An error estimate is then constructed as:

$${}^{t+\Delta t}\text{err} = \max \left| ({}^{t+\Delta t}\mathbf{u}_2 - {}^{t+\Delta t}\mathbf{u}_1) ./ \mathbf{d} \right| \quad (5)$$

In this expression, the symbol $./$ corresponds to the term-by-term division, and \mathbf{d} is a vector containing typical nodal distances in the neighbourhood of each node. Hence, this error is defined as the maximum of the absolute values of the differences between the displacements of the nodes obtained with the time steps Δt and $\Delta t/2$, normalized by the local nodal distances. This error is compared to a target value $\text{target}^{\text{err}}$, and the following test is performed by the algorithm at the end of each time step:

- If ${}^{t+\Delta t}\text{err} > \alpha \cdot \text{target}^{\text{err}}$, the step is rejected, and computed again.
- Otherwise, the computation goes on.

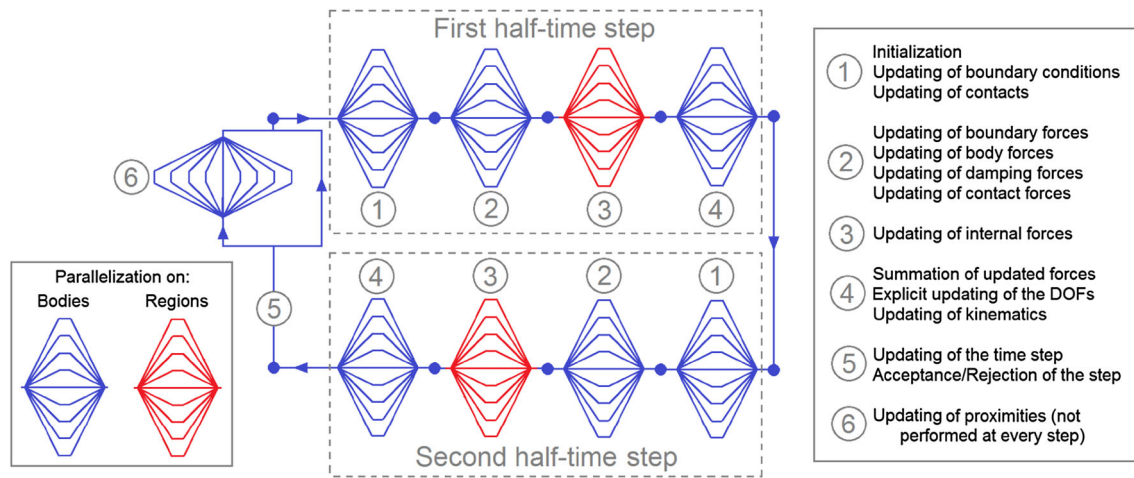


Fig. 6 Solver algorithm with fork-join parallelization strategy on a time step

- In both cases, the time step Δt is updated using the following heuristic:

$$\Delta t \rightarrow \Delta t \cdot (t + \Delta t \text{ err} / \text{target err})^{-\beta} \quad (6)$$

The parameters target err , α , and β are user-defined. Typical values of 10^{-4} , 2 and 0.2, respectively, seem to provide satisfactory results. This procedure makes it possible to accelerate or decelerate the computation automatically depending on the events occurring in the simulation, while ensuring that Δt remains smaller than the critical time step.

6 Parallelization strategy

Discrete element modelling is notoriously costly in terms of computation time, and the proposed approach is even worse because (i) the number of degrees of freedom per grain is larger (two DOFs per field node for compliant bodies), (ii) the number of contacts to solve is more important (contacts are based on nodes and segments, not simply on bodies), and (iii) the continuum equations are more demanding in terms of elementary arithmetic operations. Hence, while typical DEM simulations are nowadays typically restricted to about 10^5 grains, the proposed framework can reasonably deal with only a few thousands compliant grains (i.e. a few hundreds of thousands of DOFs) and even such a task cannot be reasonably performed on a single processor.

The proposed framework was implemented in a C++ code called MELODY (“Multibody ELEMENT-free Open code for DYNAMIC simulations”), and parallelized in an Open-MP (i.e. shared-memory) framework. A classical fork-join paradigm is employed, as described in Fig. 6. For each half-time step of the adaptive solver, four successive parallel stages are performed. In the first stage, time-varying

boundary conditions are updated and the contact detection is performed. In the second stage, nodal generalized forces related to boundary conditions, damping and contacts are computed. The third stage is (by far) the longest one, and consists in computing the strain-related nodal generalized forces in compliant bodies. Finally, the fourth stage sums up all the generalized forces applied on each node, and applies Eqs. (1–4) in order to compute the new values of the degrees of freedom and of the nodal displacements.

The logic of the implementation makes the parallelization quite straightforward, since all the operations related to a given body can be attributed to a given processor in an independent way, without any risk of conflict in memory access. This attribution is performed in a dynamic way (i.e. as soon as a processor is inactive, it receives a new task related to a given body, until all the bodies are treated). Hence, the stages 1, 2, and 4 are parallelized on a body basis. Conversely, the stage 3 (computation of internal forces of compliant bodies) requires a specific strategy for load-balance optimization, because some compliant bodies may be much larger than some other ones (for example in the case of the interaction between a single large compliant structure and a sample of rigid grains). Hence, this stage is parallelized on a “region” basis: a region is either a compliant body or a part of it. The logic is thus to divide large bodies in smaller regions to smooth the load between the processors and optimize the computational resources. This parallelization strategy was tested on the computational cluster of the LaMCoS on machines with up to 32 processors, and led to a computational efficiency of about 90–99%.

7 Illustrative case

In order to demonstrate the capabilities of the code, a simulation involving both rigid and compliant bodies is proposed

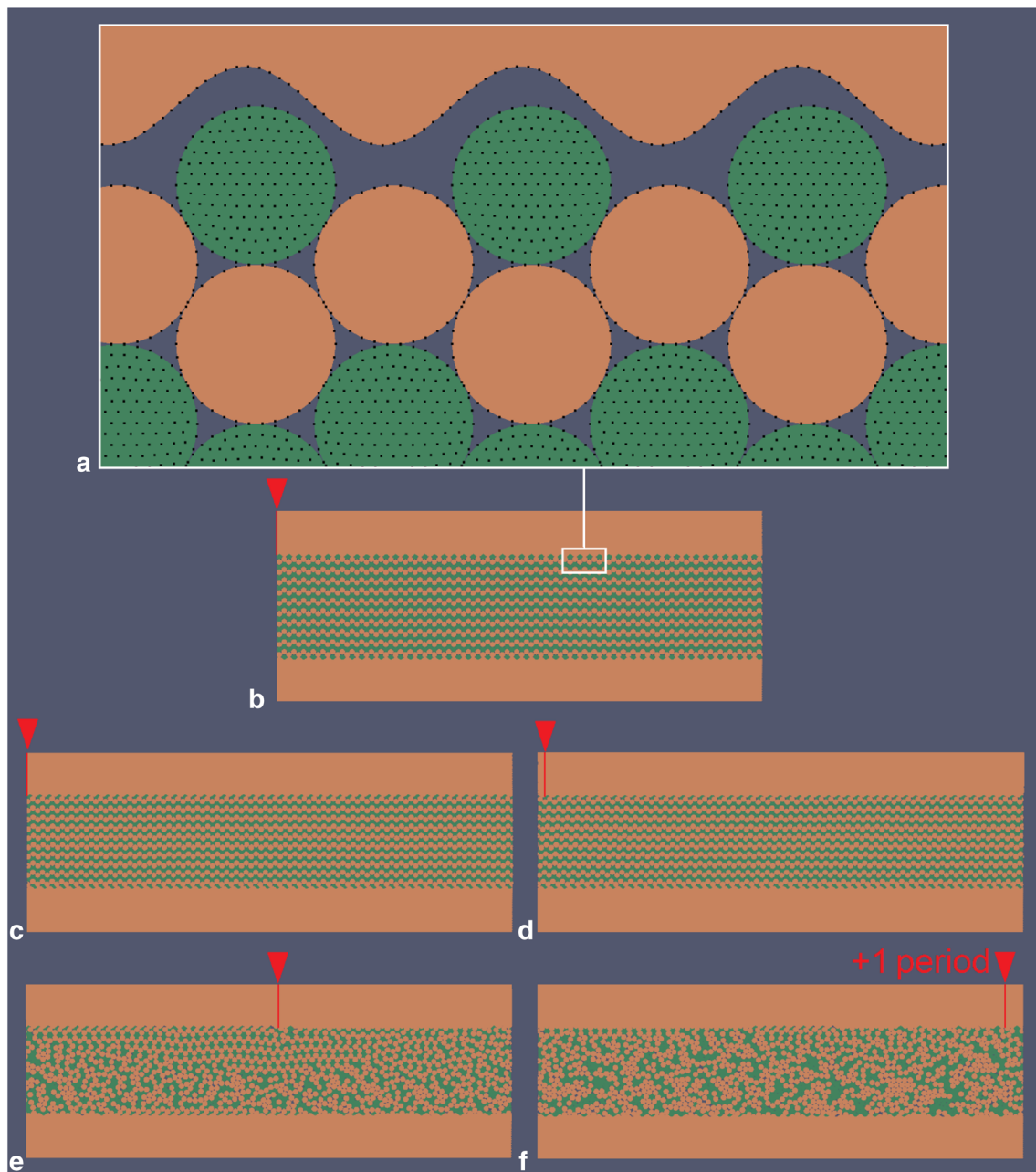


Fig. 7 Overview of the simulation; **a** zoomed view of the initial state; **b** general view of the initial state; **c** $t = 1 \mu\text{s}$ (end of vertical compaction); **d** $t = 1.5 \mu\text{s}$ (beginning of shearing); **e** $t = 6 \mu\text{s}$ (shearing on half a spatial period); **f** $t = 18.5 \mu\text{s}$ (end of simulation)

in this section. It aims to represent in an idealized way the local behaviour of a very highly loaded dry contact, as often performed in the tribological literature. Most tribological experiments [22] indeed reveal that, if a dry contact is sufficiently loaded and mature enough, the two contacting bodies get separated by a micrometric layer of solid matter (the so-called third body, coming either from the degradation of the contacting surfaces or from matter coming from outside of the contact) which transmits the load and accommodates the velocity jump between the two first bodies. Because of its

discontinuous character, the third body has often been modelled as a granular material in a DEM framework [8,14,15]. However, in typical mechanical contacts into which the local pressure can easily exceed 1 GPa, the assumption of rigid grains does not hold any more and the introduction of compliant grains is desirable. Besides, since the contacting bodies may not be composed of the same material, there may be a large difference of rigidity between the third body particles coming from each body. In that case, mixing rigid and compliant grains may be interesting.

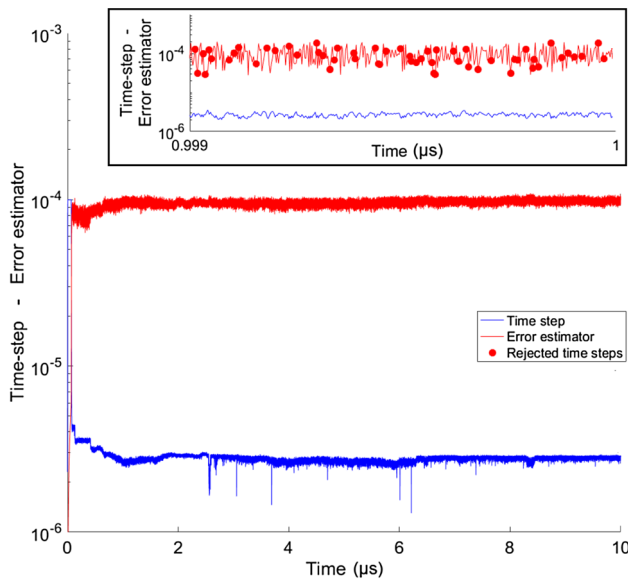


Fig. 8 Behaviour of the adaptive solver during the simulation

In the proposed simulation, we focus on a small area (with a length of $87\text{ }\mu\text{m}$) of the interface between two rigid contacting bodies. Between these first bodies, a third body composed of 2000 grains is introduced. 1000 of these grains are rigid, and 1000 are compliant (Hyper-elastic Neo-Hookean behaviour with $E = 2\text{ GPa}$ and $\nu = 0.4$). All grains have a diameter of $1\text{ }\mu\text{m}$, and interact by the means of a purely cohesive contact law with a cohesion of 200 MPa in order to mimic in a phenomenological way the attractive interactions (e.g. van der Waals or electrostatic forces) which exist between micrometric particles at this scale. The initial packing is shown in Fig. 7a. The lower body is fixed, and the upper one is submitted to a vertical pressure of 1 GPa and to a horizontal velocity of 10 m/s . Periodic boundary conditions on the lateral boundaries allow matter getting out from one side to reenter from the other. Overall, the simulation contains 2002 bodies, for about 220,000 nodes and 340,000 degrees of freedom.

Five successive stages of the simulation are shown in Fig. 7b–f, showing the shearing of the third body. The progressive mixing of the two phases exhibits some interesting patterns: At low shear (Fig. 7d), the initial dense hexagonal packing retains its integrity, while in a later stage (Fig. 7e) it is progressively destroyed and replaced by a disordered state. At very high strains (Fig. 7f), the initial packing has almost totally disappeared and the two phases seem to be somewhat organized in clusters of 10–20 identical grains.

During the simulation, the solver automatically adapts the time step to the events occurring in the simulated system, with a target error estimator of $\text{target err} = 10^{-4}$. The evolutions in time of the error estimator and of the time step are provided in Fig. 8. This figure shows the benefit of using

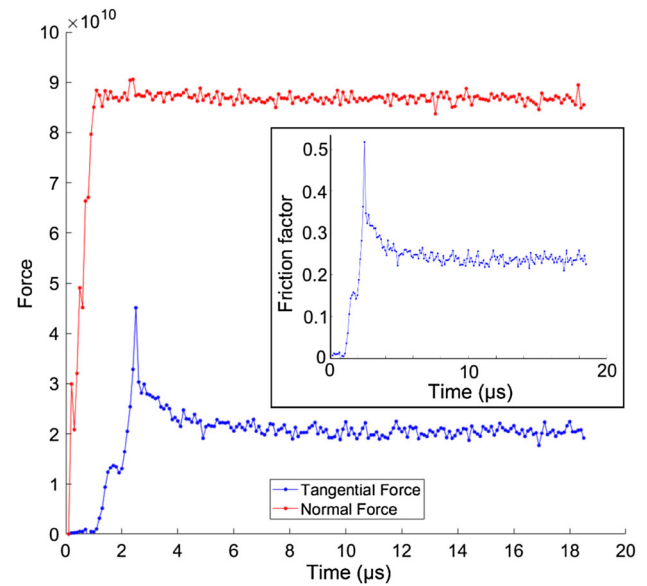


Fig. 9 Normal and tangential forces on the fixed boundary of the lower body, and corresponding friction factor

an adaptive solver, since the error estimator remains close to its target value while the corresponding time step does vary much more. In the first stages of the simulation (compression of the third body and beginning of shearing), the time step decreases from 3.5×10^{-6} to about 1.8×10^{-6} and then stabilizes around 2×10^{-6} . But even after this stabilization there are a large number of events during which the time step suddenly decreases up to 5×10^{-7} (most critical and short events do not appear on the curves because they were smoothed to ease the reading). Hence, it is evident that an adaptive solver is necessary in order to ensure both the stability of the simulation and its computational efficiency. The insert of Fig. 8 shows the same data on a much more narrow time window and also exhibits the rejected time steps (i.e. those for which the error estimator $\text{err}^{t+\Delta t}$ is larger than $\alpha \cdot \text{target err}$, with $\alpha = 2$ in this case). Such rejections happen about nine times every 100 steps in average in the present case, but do not happen at all in some other cases.

Figure 9 shows some quantitative results of interest in the field of tribology. The resulting normal force on the fixed boundary of the lower body increases during the compression of the third body (from 0 to $1\text{ }\mu\text{s}$) and then stabilizes. Meanwhile, The tangential force increases during the first stages of shearing until it reaches a quite sharp peak when the initial hexagonal packing suddenly breaks. The tangential force then decreases and reaches a stable plateau for the rest of the simulation. The insert of Fig. 9 shows the friction factor, defined as the ratio between the tangential and the normal forces. It appears that the friction factor first increases up to a peak at about 0.5 (sometimes called “static friction

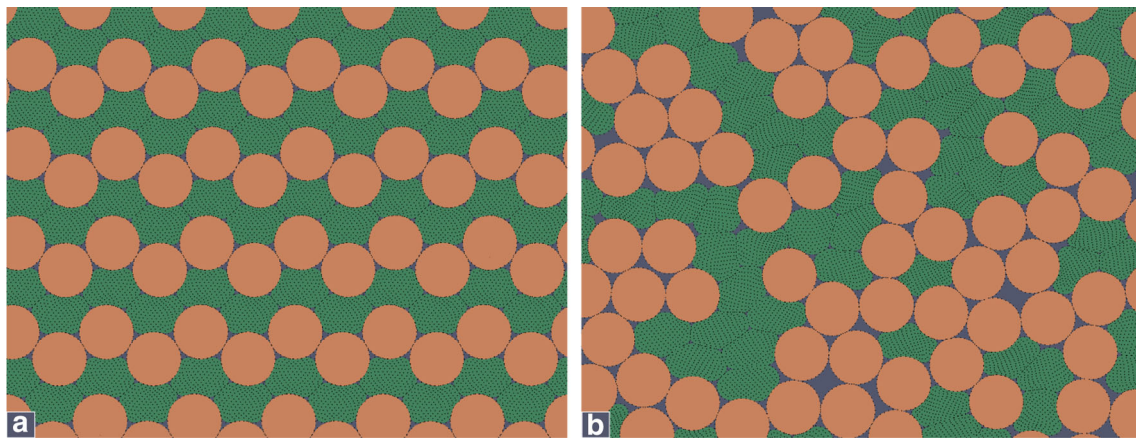


Fig. 10 Location of rigid and compliant grains; **a** $t = 1.5 \mu\text{s}$; **b** $t = 18.5 \mu\text{s}$

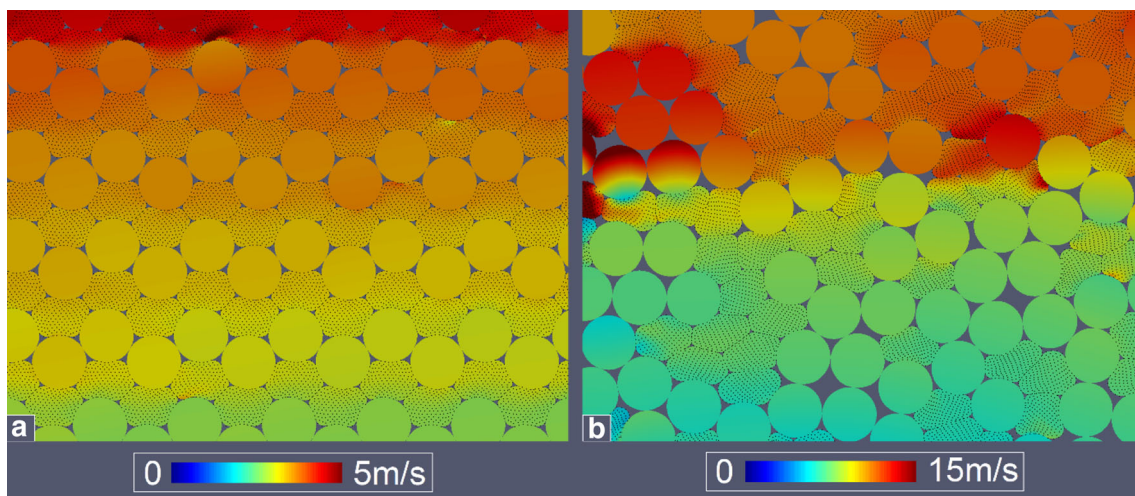


Fig. 11 Velocity norm field; **a** $t = 1.5 \mu\text{s}$; **b** $t = 18.5 \mu\text{s}$

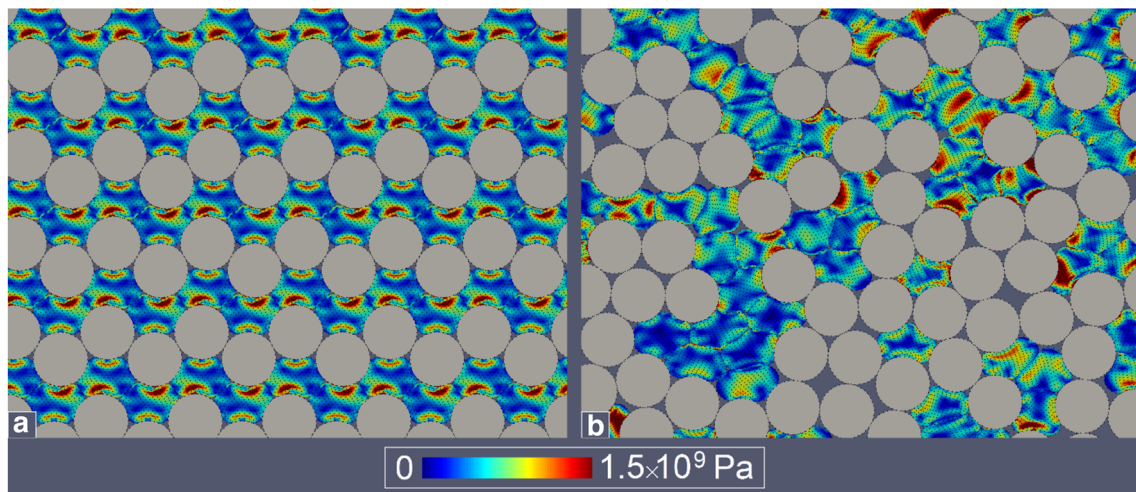


Fig. 12 Von Mises equivalent stress field; **a** $t = 1.5 \mu\text{s}$; **b** $t = 18.5 \mu\text{s}$

coefficient”), and then reaches a plateau at about 0.25 (sometimes called “dynamic friction coefficient”). These values are

very typical of those commonly encountered in tribological experiments.

Figures 10, 11 and 12 show zoomed views of a small area of the third body at two stages of the simulation (i.e. at the beginning of the shearing and at the end of the simulation). More specifically, Fig. 10 shows the locations of the different types of grains, from the initial hexagonal packing in Fig. 10a (already deformed by the compression and the beginning of the shearing) to the steady disordered state in Fig. 10b. The organization in clusters of identical grains (i.e. rigid or compliant) is evident, although the explanation of this behaviour remains unknown and shall be investigated in further studies. A possible explanation may be the high deformability of the compliant grains which allow them to increase their contact area and thus the total attractive force between them, and may lead them to cluster in some way. Figures 11 and 12 show the velocity norm and equivalent Von Mises stress fields at the same instants and demonstrate the level of detail and the quantity of rich data that such a simulation may bring (e.g. local velocity gradients and stress concentrations).

8 Conclusion and perspectives

The numerical framework presented in this paper allows to combine in a same simulation a large number of rigid and compliant bodies. A meshfree interpolation of the continuous fields, a robust contact algorithm and an adaptive explicit solver provide the code with a sufficient stability to deal with such complex situations, while an efficient parallelization strategy makes it possible to deal with pretty large systems. Besides, the discretization technique makes it possible to introduce grains with arbitrary shapes, such as those generated by Fourier–Voronoi techniques [32,33]. This will make it possible to drive interesting studies on the influence of the grains shapes on the macroscopic behaviour of granular materials, either composed of rigid or compliant grains.

Although it was primarily developed for tribological purposes, it is expected that this framework will find some applications in a number of technical and scientific fields such that geomechanics (e.g. interaction between a soil and a compliant structure) or powder technologies (e.g. compaction of mixed powders with grains of different materials). A fully functional version of the C++ code (sources and windows executable) as well as its matlab preprocessor is available for free download at the URL <http://guilhem.mollon.free.fr>. Although already capable of dealing with a lot of situations, the code still has a lot of room for improvement. Next numerical developments will be dedicated to the implementation of more complex contact laws and constitutive behaviours (e.g. viscoelasticity or elastoplasticity). A possible coupling with XFEM would allow to study in a realistic manner the breakage of the grains based on their stress fields. Eventually, a 3D version including fluid coupling is foreseen in order to

deal with more complex systems, both in geomechanics and in tribology.

Compliance with ethical standards

Conflict of interest The author acknowledges that this study contains original material, as a result of a purely academic study without any kind of private funding or conflict of interest. Its publication has been approved tacitly by the responsible authorities at the institute where the work has been carried out.

References

1. M Cundall PA, Strack ODL (1979) A discrete numerical model for granular assemblies. *Géotechnique* 29(1):47–65
2. Azema E, Radjai F, Saussine G (2009) Quasistatic rheology, force transmission and fabric properties of a packing of irregular polyhedral particles. *Mech Mater* 41:729–741
3. Mollon G, Richefeu V, Villard P, Daudon D (2015) Discrete modelling of rock avalanches: sensitivity to block and slope geometries. *Granul Matter* 17(5):645–666
4. Stahl M, Konietzky H (2011) Discrete element simulation of ballast and gravel under special consideration of grain-shape, grain-size and relative density. *Granul Matter* 13:417–428
5. Mollon G, Zhao J (2013) Characterization of fluctuations in granular hopper flow. *Granul Matter* 15(6):827–840
6. Herbst JA, Potapov AV (2004) Making a discrete grain breakage model practical for comminution equipment performance simulation. *Powder Technol* 143:144–150
7. Zhao J, Shan T (2013) Coupled CFD-DEM simulation of fluid–particle interaction in geomechanics. *Powder Technol* 239:248–258
8. Richard D, Iordanof I, Renouf M, Berthier Y (2008) Thermal study of the dry sliding contact with third-body presence. *ASME J Tribol* 130(3):031404
9. O’Sullivan C (2011) Particle-based discrete element modeling: geomechanics perspective. *Int J Geomech* 11(6):449–464
10. Darve F, Duriez J, Wan R (2016) DEM modelling in geomechanics: some recent breakthroughs. In: *Proceedings of the 7th international conference on discrete element method*, pp 3–12
11. Cagnoli B, Piersanti A (2015) Grain size and flow volume effects on granular flow mobility in numerical simulations: 3-D discrete element modeling of flows of angular fragments. *J Geophys Res Solid Earth* 120(4):2350–2366
12. Tijskens E, Ramon H, De Baerdemaeker J (2003) Discrete element modelling for process simulation in agriculture. *J Sound Vib* 66:493–514
13. Ouhbi N, Voivret C, Perrin G, Roux JN (2016) Railway ballast: grain shape characterization to study its influence on the mechanical behaviour. *Procedia Eng* 143:1120–1127
14. Fillot N, Iordanof I, Berthier Y (2004) A granular dynamic model for the degradation of material. *ASME J Tribol* 126(3):606–14
15. Mollon G (2015) A numerical framework for discrete modelling of friction and wear using Voronoi polyhedrons. *Tribol Int* 90:343–355
16. Richards K, Bithell M, Dove MT, Hodge RA (2004) Discrete-element modelling: methods and applications in the environmental sciences. *Philos Trans R Soc A Math Phys Eng Sci* 362(1822):1797–1816
17. Da Cruz F, Emam S, Prochnow M, Roux JN, Chevoir F (2005) Rheophysics of dense granular materials: discrete simulation of planar shear flows. *Phys Rev E* 72:021309

18. Jean M (1999) The non-smooth contact dynamics method. *Comput Methods Appl Mech Eng* 177(3–4):235–257
19. Gethin DT, Lewis RW, Ransing RS (2003) A discrete deformable element approach for the compaction of powder systems. *Model Simul Mater Sci Eng* 11:101–114
20. Mirea DA, Trunfio-Sfarghiu A-M, Matei CI, Munteanu B, Piednoir A, Rieu JP, Blanchin MG, Berthier Y (2013) Role of the biomolecular interactions in the structure and tribological properties of synovial fluid. *Tribol Int* 59:302–311
21. Duvernois V, Marsden AL, Shadden SC (2013) Lagrangian analysis of hemodynamics data from FSI simulation. *Int J Numer Methods Biomed Eng* 29:445–461
22. Descartes S, Saulot A, Godeaux C, Bondeux S, Dayot C, Berthier Y (2011) Wheel flange/rail gauge corner contact lubrication: tribological investigations. *Wear* 271:54–61
23. Zhang J (2009) A study of composite particles by multi-particle finite element method. *Compos Sci Technol* 69:2048–2053
24. Harthong B, Jerier J-F, Richefeu V, Chareyre B, Doremus P, Imbault D, Donzé F-V (2012) Contact impingement in packings of elastic–plastic spheres, application to powder compaction. *Int J Mech Sci* 61:32–43
25. Gustafsson G, Haggblad H-A, Jonsen P (2013) Multi-particle finite element modelling of the compression of iron pellets with statistically distributed geometric and material data. *Powder Technol* 239:231–238
26. Mollon G (2016) A multibody meshfree strategy for the simulation of highly deformable granular materials. *Int J Numer Methods Eng* 108(12):1477–1497
27. Liu GR, Zhang GY, Gu YT, Wang YY (2005) A meshfree radial point interpolation method (RPIM) for three-dimensional solids. *Comput Mech* 36:421–430
28. Nayroles B, Touzot G, Villon P (1992) Generalizing the finite element method: diffuse approximation and diffuse elements. *Comput Mech* 10:307–318
29. Belytschko T, Lu YY, Gu L (1994) Element-free Galerkin methods. *Int J Numer Methods Eng* 37:229–256
30. Ferrellec J-F, McDowell G (2010) A method to model realistic particle shape and inertia in DEM. *Granul Matter* 12:459–467
31. Cohen J, Lin MC, Manocha D, Ponamgi MK (1995) I-COLLIDE: an interactive and exact collision detection system for large scale environments. In: *ACM interactive 3D graphics symposium*, Monterey, USA
32. Mollon G, Zhao J (2012) Fourier–Voronoi-based generation of realistic samples for discrete modelling of granular materials. *Granul Matter* 14:621–638
33. Mollon G, Zhao J (2014) 3D generation of realistic granular samples based on random fields theory and Fourier shape descriptors. *Comput Methods Appl Mech Eng* 279:46–65