A multibody meshfree strategy for the simulation of highly deformable granular materials

Guilhem Mollon

A multibody meshfree strategy for the simulation of highly deformable granular materials

Guilhem Mollon*,†

Univ Lyon, INSA Lyon, CNRS, LaMCoS, F-69621 cedex, Villeurbanne, France

In this paper, a multibody meshfree framework is proposed for the simulation of granular materials undergoing deformations at the grain scale. This framework is based on an implicit solving of the mechanical problem based on a weak form written on the domain defined by all the grains composing the granular sample. Several technical choices, related to the displacement field interpolation, to the contact modelling, and to the integration scheme used to solve the dynamic equations, are explained in details. A first implementation is proposed, under the acronym Multibody ELement-free Open code for DYnamic simulation (MELODY), and is made available for free download. Two numerical examples are provided to show the convergence and the capability of the method.

KEY WORDS: granular materials; meshfree methods; multibody dynamics

1. INTRODUCTION

Granular matter represents a particular class of materials of important interest in a large number of industrial fields: civil engineering, food industry, pharmaceutical engineering and powder industries (chemistry, cosmetics, etc.) among others. It is also pivotal in several scientific fields and across several scales (e.g. geophysics, geomechanics or tribology). When it comes to simulation, two main approaches are commonly used: continuous and discrete modelling.

-Continuous modelling [1, 2] consists in considering the granular matter as a continuum, with appropriate mass and momentum conservation laws, and in assigning to this continuum a constitutive model in order for it to reproduce some experimental observations. An appropriate numerical method such as finite element method (FEM) or finite differences method is then used to solve the set of partial differential equations so-obtained. This approach presents several advantages: it is simple to operate; its computational cost remains modest; and the choice of the constitutive law and of its parameters provides an appreciable freedom to the modeller. In most cases, however, a given constitutive law fails to cover the whole set of situations that a granular material may encounter (dense solid-like behaviour in cohesive or cohesionless conditions, dense liquid-like flow, collisional gas-like behaviour, etc.). Such modelling is hence limited to a given purpose for which enough experience has been gained and is generally applied in an engineering framework.

-Discrete modelling operates at a smaller scale and consists in predicting the motion of each grain as a rigid body. This motion is computed by a simple application of the laws of motion, based on the body forces applied to each grain and on those that appear between contacting grains. Appropriate resolution schemes, either explicit [3] or implicit [4], are then used to solve this set of stiff differential equations. Discrete modelling is very intuitive but, for reasons related to computational cost,

*Correspondence to: Guilhem Mollon, LaMCoS, INSA Lyon, CNRS, Univ. Lyon, F-69621 cedex, Villeurbanne, France.
†E-mail: guilhem.mollon@gmail.com

it is usually restricted to limited numbers of grains (i.e. rarely more than a few tens of thousands), and is a bit less used for engineering applications than continuous modelling. Its use in fundamental research, however, has made it possible in recent years to gain a much better understanding of the granular behaviours across the scales [5–7] and in very various conditions [8–10]. Indeed, this approach is more general than the continuous one because it does not rely on any assumption on the macroscopic behaviour of the material, which is entirely controlled by the choices made by the modeller at the grain scale (contact laws and grain shapes).

The success of discrete modelling is related to the fact that, on a macroscopic level, it seems realistic to assume that most of the deformations are related to relative motions of the grains rather than to their individual deformations. In this paper, however, we wish to present a numerical framework that makes it possible to take benefit from the main advantages of discrete modelling while relaxing its main hypothesis. Indeed, in several scientific and industrial applications, assuming that each grain is an undeformable and unbreakable solid strongly restricts the realism of discrete modelling. If the material composing the grains is brittle (e.g. silicates sands), small deformations of the grains may be interesting in order to compute their inner stress fields and to predict their possible breakage [11]. Alternatively, if this material is ductile (e.g. ductile powders, tribological third-body [12], biological fluids [13], etc.), then the large deformations of each grain might take a part in the macroscopic deformations that should not be ignored.

The proposed numerical framework to address these questions is a multibody meshfree approach. 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 sense, it shares some similarities with the multi-particle FEM used in [14–18] to deal with the compression of ductile powders. The approach proposed hereafter relies on the following technical choices:

- The granular material is composed of a set of individual grains, each of which is delimited by a closed boundary. The present approach is restricted to two dimensions.
- 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.
- Contact forces are computed on the basis of a two-pass node-to-segment formulation, hence considering the interaction between each field node located on the boundary of a given grain and the boundaries of the surrounding grains. A smoothed contact element is used, and an augmented Lagrangian strategy is applied to control the interpenetrations.
- At a given time step, a classical weak-form is considered for the whole problem. Numerical integration is performed using a Gauss quadrature on a triangular mesh. A Newton–Raphson scheme is used to account for the nonlinearities related to contacts, large displacements and possibly material behaviour.
- A stable composite Newmark–Euler scheme is used to solve the dynamic equations.

The remainder of this paper is devoted to the description of these choices. The important questions related to the appropriate selection of constitutive relations and contact laws in a specified framework are left for further studies.

2. SHAPE FUNCTIONS AND WEAK FORM

A displacement field is defined on each grain following a meshfree approach analogous to the classical element-free Galerkin (EFG) method [19]. Despite a slightly larger time cost, this technique is preferred to finite element interpolation because it provides more robustness under large deformations, easier adaptive refinement and a smoother representation of the stress fields with a limited number of nodes [19]. It is thus more appropriate to future evolutions of the code towards very strong deformations, plasticity localization, grain breakage and fluid coupling, among others. A thorough comparison between FEM and meshfree interpolations is provided in [20] in a

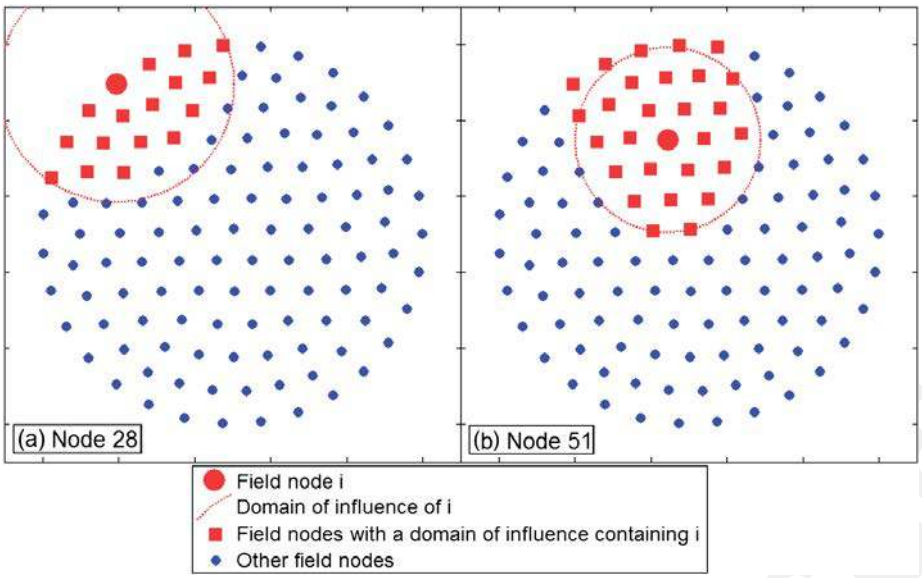

Figure 1. Field nodes and domains of influence on a disc, (a) for a node on the boundary and (b) for a node inside the domain.

multibody framework and clearly demonstrates that the meshfree method leads to a better accuracy for a given computation time. Indeed, it is more expensive than FEM at a given number of degrees of freedom but requires much less degrees of freedom to reach a certain level of accuracy. The main reason is that it postulates continuous stress fields, which is a property that FEM does not share.

In its initial formulation ([19], inspired by [21]), the EFG method was based on moving least-square (MLS) shape functions, which present nice smoothness characteristics but do not possess the Kronecker delta property: Based on some prescribed values at some selected field nodes, each MLS shape function provides an approximation, not an interpolation. This is the reason of well-documented complications in enforcement of Dirichlet boundary conditions [22, 23]. In the present work, however, a large number of contacts are expected to be solved, and interpolating shape functions are easier to handle. The so-called 'maximum entropy' class of shape functions [24, 25] is able to exhibit this property in a weak sense (i.e. only on the boundaries), but it is relatively complex to handle, especially in the case of non-convex bodies. To replace MLS, another relevant candidate called radial point interpolation method (RPIM) has been proposed in [26]. Like in MLS, RPIM is based on a set of N field nodes dispatched on the solid, and each field node is assigned a circular domain of influence with an appropriate radius. In the present study, this radius is chosen for each node in order to encompass exactly 20 other field nodes (Figure 1). Then, scalar fields may be expressed as

$$u\left(\vec{X}\right) = \sum_{i=1}^{n} R_i\left(\vec{X}\right) \cdot a_i + \sum_{j=1}^{m} p_j\left(\vec{X}\right) \cdot b_j . \tag{1}$$

In this expression, $\vec{X} = (x, y)$ is the position of the considered point in a 2D Cartesian frame, $R_i\left(\vec{X}\right)$ are radial basis functions (RBF), n is the number of field nodes for which the position $\vec{X}$ belongs to the influence domain, $p_j\left(\vec{X}\right)$ are monomials and a_i and b_j are scalar coefficients. Each RBF $R_i\left(\vec{X}\right)$ is only a function of the distance between the current point of coordinates $\vec{X}$ and a given field node of coordinates $\vec{X}_i$. Among many different kinds of existing RBF, we use here a simple Gaussian function:

$$R_i\left(\vec{X}\right) = r\left(d_i\right) = \begin{cases} exp\left(-\left(\frac{3d_i}{D_i}\right)^2\right) & if\ d_i \leqslant D_i\ and\ if\ \vec{X}\ belongs\ to\ the\ solid \\ 0 \quad otherwise \end{cases} \tag{2}$$

The quantity D_i is the radius of the domain of influence of node i, and d_i is the radial distance of the current point to this node:

$$d_i = \left\| \vec{X} - \vec{X}_i \right\|. \tag{3}$$

The monomials $p_j\left(\vec{X}\right)$ are taken in the basis $\left[1; x; y; x^2; y^2; xy\right]$, and we hence have $m = 6$. They are necessary to ensure reproducibility of displacement fields with polynomial precision. A set of n shape functions $\left\{\Phi_1\left(\vec{X}\right)\Phi_2\left(\vec{X}\right)\ldots\Phi_{\mathrm{n}}\left(\vec{X}\right)\right\}$ is then obtained by applying the following formula:

$$\left[\Phi_1\left(\vec{X}\right)\Phi_2\left(\vec{X}\right)\ldots\Phi_{\mathrm{n}}\left(\vec{X}\right)\Phi_{\mathrm{n+1}}\left(\vec{X}\right)\ldots\Phi_{\mathrm{n+m}}\left(\vec{X}\right)\right]^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{R_0} & \boldsymbol{P_m} \\ \boldsymbol{P_m^T} & \boldsymbol{0} \end{bmatrix}^{-1} \begin{Bmatrix} \boldsymbol{R}\left(\vec{X}\right) \\ \boldsymbol{p}\left(\vec{X}\right) \end{Bmatrix} \tag{4}$$

With:

$$\boldsymbol{R_0} = \begin{bmatrix} r_1(d_1) & \ldots & r_n(d_1) \\ \vdots & \ddots & \vdots \\ r_1(d_n) & \ldots & r_n(d_n) \end{bmatrix} \tag{5}$$

$$\boldsymbol{P_m} = \begin{bmatrix} 1 & x_1 & y_1 & x_1^2 & y_1^2 & x_1 y_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & y_n & x_n^2 & y_n^2 & x_n y_n \end{bmatrix} \tag{6}$$

$$\boldsymbol{R}\left(\vec{X}\right) = \left[\mathrm{R}_1\left(\vec{X}\right)\mathrm{R}_2\left(\vec{X}\right)\ldots\mathrm{R}_{\mathrm{n}}\left(\vec{X}\right)\right]^T \tag{7}$$

$$\boldsymbol{p}\left(\vec{X}\right) = \left[1 \;\; \mathrm{x} \;\; \mathrm{y} \;\; \mathrm{x}^2 \;\; \mathrm{y}^2 \;\; \mathrm{xy}\right]^T, \tag{8}$$

where $r_i\left(d_j\right)$ corresponds to the RBF of field node j applied to the position of field node i. Examples of interpolating shape functions such obtained are provided in Figure 2. The partial derivatives of the shape functions are obtained by

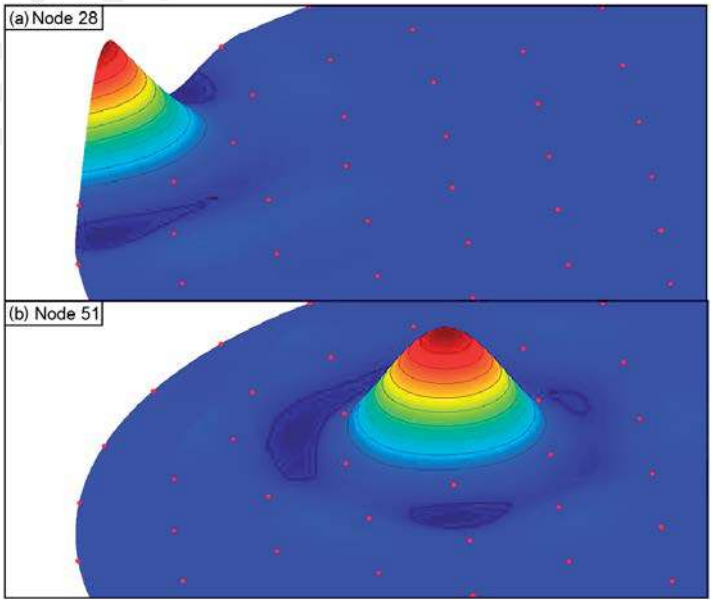

Figure 2. Shape functions on a disc, for a node on the boundary and a node inside the domain.

$$\left[\Phi_{1,\mathrm{x}}\left(\vec{X}\right)\Phi_{2,\mathrm{x}}\left(\vec{X}\right)\ldots\Phi_{\mathrm{n},\mathrm{x}}\left(\vec{X}\right)\Phi_{\mathrm{n}+1,\mathrm{x}}\left(\vec{X}\right)\ldots\Phi_{\mathrm{n}+\mathrm{m},\mathrm{x}}\left(\vec{X}\right)\right]^T=\begin{bmatrix}\boldsymbol{R_0} & \boldsymbol{P_m}\\ \boldsymbol{P_m^T} & \boldsymbol{0}\end{bmatrix}^{-1}\begin{Bmatrix}\boldsymbol{R}_{,\boldsymbol{x}}\left(\vec{X}\right)\\ \boldsymbol{p}_{,\boldsymbol{x}}\left(\vec{X}\right)\end{Bmatrix} \tag{9}$$

$$\left[\Phi_{1,\mathrm{y}}\left(\vec{X}\right)\Phi_{2,\mathrm{y}}\left(\vec{X}\right)\ldots\Phi_{\mathrm{n},\mathrm{y}}\left(\vec{X}\right)\Phi_{\mathrm{n}+1,\mathrm{y}}\left(\vec{X}\right)\ldots\Phi_{\mathrm{n}+\mathrm{m},\mathrm{y}}\left(\vec{X}\right)\right]^T=\begin{bmatrix}\boldsymbol{R_0} & \boldsymbol{P_m}\\ \boldsymbol{P_m^T} & \boldsymbol{0}\end{bmatrix}^{-1}\begin{Bmatrix}\boldsymbol{R}_{,\boldsymbol{y}}\left(\vec{X}\right)\\ \boldsymbol{p}_{,\boldsymbol{y}}\left(\vec{X}\right)\end{Bmatrix}. \tag{10}$$

From this point, any displacement field may be written:

$$\vec{u}\left(\vec{X}\right)=\begin{pmatrix}u_x\left(\vec{X}\right)\\ u_y\left(\vec{X}\right)\end{pmatrix}=\begin{pmatrix}\sum_{i=1}^{n}\Phi_i\left(\vec{X}\right)\cdot u_{xi}\\ \sum_{i=1}^{n}\Phi_i\left(\vec{X}\right)\cdot u_{yi}\end{pmatrix}, \tag{11}$$

where u_{xi} and u_{yi} are the horizontal and vertical displacements of node i.

In order to solve the dynamic equilibrium equations of a set of grains, a weak form is written in the classical way on the configuration of reference:

$$\begin{aligned}&\textit{find } \vec{u}\in V \textit{ such that}\\ &\int_\Omega \overline{\overline{S}}\left(\vec{u}\right):\overline{\overline{e}}\left(\vec{w}\right)d\Omega-\int_\Omega \rho\vec{f}\cdot\vec{w}\,d\Omega-\int_{S_N}\overrightarrow{T_N}\cdot\vec{w}\,dS-\int_{S_C}\overrightarrow{T_C}\cdot\vec{w}\,dS\\ &=-\int_\Omega \rho\frac{\partial^2\vec{u}}{\partial t^2}\cdot\vec{w}\,d\Omega-\int_\Omega c\frac{\partial\vec{u}}{\partial t}\cdot\vec{w}\,d\Omega \qquad \forall\vec{w}\in V_0\end{aligned} \tag{12}$$

In this expression, V is the Sobolev space of functions that are square-integrable with a square-integrable first derivative and respect the prescribed Dirichlet boundary conditions, V_0 is the same space but with homogeneous Dirichlet boundary conditions, Ω is the domain covered by the union of the grains, S_N is the domain of application of Neumann boundary conditions (i.e. external forces, noted $\overrightarrow{T_N}$) and S_C is the domain of possible contact surfaces (with associated forces noted $\overrightarrow{T_C}$). $\vec{f}$ represents the body forces (e.g. gravity) and ρ and c are the material unit weight and damping in the reference configuration. $\overline{\overline{S}}\left(\vec{u}\right)$ is the second Piola–Kirchhoff stress tensor associated with displacement $\vec{u}$ by the means of a chosen constitutive relation, and $\overline{\overline{e}}\left(\vec{w}\right)$ is the Green–Lagrange strain tensor associated with the virtual displacement $\vec{w}$. For the sake of simplicity, this weak form implicitly assumes that Neumann boundary conditions do not vary with the configuration (although they may evolve in time on a user-defined basis). In the current version of the code, a simple Saint Venant–Kirchhoff constitutive relation is used, which states the proportionality of the Green–Lagrange and the Piola–Kirchhoff tensors (with a properly defined fourth-order elasticity tensor $\mathcal{A}$), even in large deformations:

$$\overline{\overline{S}}\left(\vec{u}\right)=\mathcal{A}:\overline{\overline{E}}\left(\vec{u}\right). \tag{13}$$

Both $\vec{u}$ and $\vec{w}$ are expressed in terms of the meshfree shape functions using Equation (11). The surface integrations of Equation (12) are performed using a Gauss quadrature on a triangular mesh just as in FEM (Figure 3). A six-point quadrature is used here. This stage often leads scholars to state that EFG-related methods are not truly meshfree, but this is not an issue in the current framework because this mesh is only used for integration but does not affect the connectivity of the nodes.

The use of a triangular mesh mapped on the initial positions of the field nodes and of a constant number of Gauss points per integration cell ensures that the density of the Gauss points is somewhat proportional to that of the field nodes in any part of the system. Because the spatial rates of variations of the shape functions are related to this density, such a method ensures at least that the integration is performed in a consistent manner over the whole system. However, the choice of the proper fineness of this mesh and of the proper order of integration inside each cell, in relation with the size of the domains of influence of the field nodes, is an open topic that would deserve a comprehensive parametric study. This issue is kept for future studies.

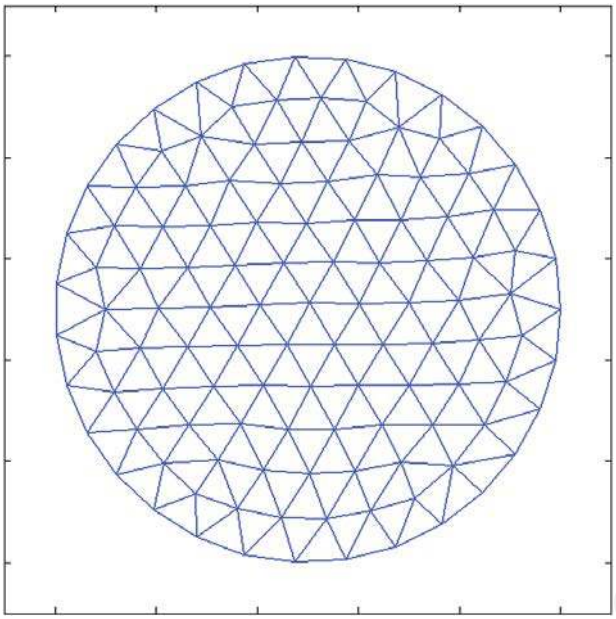

Figure 3. Typical mesh used for integration on the disc of Figures 1 and 2, with six Gauss points per triangle.

The result of these operations is a highly nonlinear system of equations in terms of the nodal displacements, velocities and accelerations, which has to be linearized for solving. The obtained tangent system has the general form:

$$\boldsymbol{K}\,\delta\boldsymbol{U} = \boldsymbol{R}. \tag{14}$$

In this expression, $\boldsymbol{K}$ is a $2N \times 2N$ consistent tangent matrix (which, in a dynamic framework, will contain elements originating from stiffness in the bulk and at contacts, from mass and from damping), $\boldsymbol{R}$ is the $2N \times 1$ residual vector (also called 'unbalanced forces vector', containing internal forces, body forces, external forces, contact forces, inertial forces and damping forces), and $\delta\boldsymbol{U}$ is an unknown $2N \times 1$ vector of displacement increment. Of course, both $\boldsymbol{K}$ and $\boldsymbol{R}$ are dependent on the nodal displacements, velocities and accelerations at the previous time step, as well as on the predicted displacements (in the Newton–Raphson sense) at the current time step. N is the total number of field nodes. Apart from the quantities related to contacts and to dynamic terms, the expressions of $\boldsymbol{K}$ and $\boldsymbol{R}$ are rather classical in an FEM framework (e.g. [27]), and their detailed expressions in the proposed meshfree framework are provided in Appendix A.

3. CONTACT MODELLING

If a large number of grains are to be simulated, this numerical framework is expected to be rather costly in terms of computation time. For this reason, it seems reasonable to try to limit as much as possible the number of field nodes contained in each particle. In the bulk of each grain, smooth stress fields are available without using too many field nodes, thanks to the use of meshfree shape functions. However, difficulties do arise at the boundaries of the grains, where contact conditions may be difficult to handle if the contours are represented only by linear segments connecting a small number of field nodes. Important angularities might appear at the boundaries, having two main consequences: (i) The standard contact element based on linear shape functions on boundary segments is often found to miss convergence, and (ii) the standard one-pass node-to-segment algorithm, based on a somewhat arbitrary choice of a master and a slave body, is not sufficient to prevent local interpenetrations.

To solve the first issue, a local regularization of angular sections of the grains boundaries seems necessary. A simple technique was proposed by Zavarise and De Lorenzis [28], which allowed to remain in the framework of a classical linear contact shape function. However, a more comprehensive and reliable solution is offered by the smoothed contact elements proposed by Nguyen *et al.* [29] and detailed in [30]. This approach is based on a classical 2D node-to-segment contact mod-

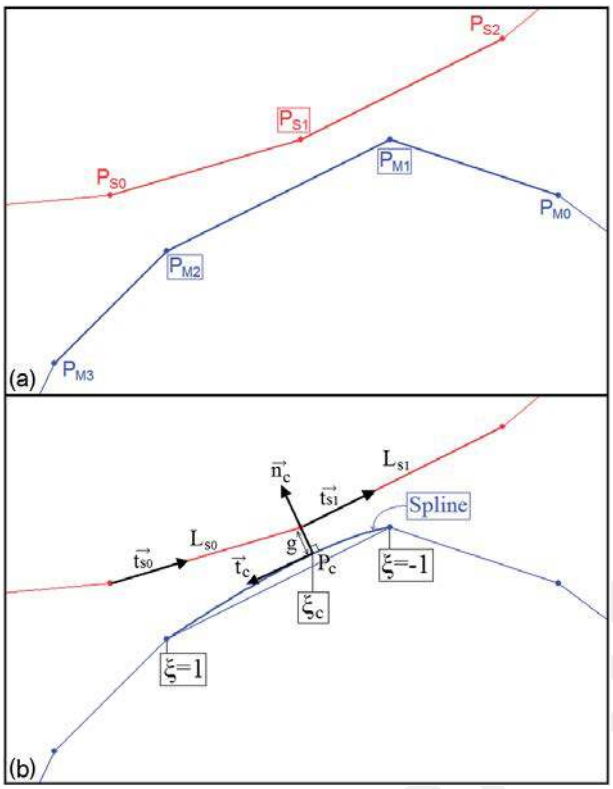

Figure 4. Smooth contact element; (a) position of the slave node and of the master segment; (b) Determination of the closest point and of the normal gap.

elling, into which the boundary of the master body is represented locally by a smooth spline with a compact support. The first stage is a broad proximity detection, in a manner similar to what is performed in usual DEM codes: for a given grain, a list of close grains is determined, for example, using a sweep-and-prune algorithm [31]. Then, for each couple of potentially contacting grains, a close proximity detection is performed, in a manner similar to common FEM codes. Suppose we attribute to one grain the role of slave body and to the other the role of master body: This stage consists in determining for each boundary node of the slave body (i.e. slave node) the potentially contacting boundary segment of the master body (i.e. master segment). A box-search [32] algorithm is used here to accelerate this detection. The resulting situation is provided in Figure 4(a) for an illustrative case: For a given slave node P_{s1}, the closest master node is P_{m1}, and the proper master segment is either $[P_{m0}\,P_{m1}]$ or $[P_{m1}\,P_{m2}]$. Choosing between these two possibilities is straightforward if classical linear elements are used (it is just a matter of point-to-line distance computation), but we are using here a different type of contact element, and both segments should be considered as potential master segments for P_{s1}. For the sake of clarity, we focus on the segment $[P_{m1}\,P_{m2}]$, keeping in mind that both segments will be checked. The proximity of sharp angles (in points P_{m1} and P_{m2}) that the slave node may cross during the time step if large slip occurs leads to replace the segment $[P_{m1}\,P_{m2}]$ by a spline, more specifically by a cubic Hermite polynomial (Figure 4(b)). Each point of this spline is defined by a parameter ξ and has the following coordinates:

$$\begin{cases} x\left(\xi\right) = \sum_{i=0}^{3} x_{mi} \cdot N_i\left(\xi\right) \\ y\left(\xi\right) = \sum_{i=0}^{3} y_{mi} \cdot N_i\left(\xi\right) \end{cases} with - 1 \leqslant \xi \leqslant 1 \tag{15}$$

The contact shape functions $N_i\left(\xi\right)$ are given by:

$$\begin{cases} N_0\left(\xi\right) = -\frac{\left(\xi^2-1\right)\left(\xi-1\right)}{16} \\ N_1\left(\xi\right) = \frac{\left(1-\xi\right)}{2} + \frac{\left(\xi^2-1\right)\left(\xi-1\right)}{8} + \frac{\left(\xi^2-1\right)\left(\xi+1\right)}{16} \\ N_2\left(\xi\right) = \frac{\left(1+\xi\right)}{2} - \frac{\left(\xi^2-1\right)\left(\xi-1\right)}{16} - \frac{\left(\xi^2-1\right)\left(\xi+1\right)}{8} \\ N_3\left(\xi\right) = \frac{\left(\xi^2-1\right)\left(\xi+1\right)}{16} \end{cases} \tag{16}$$

Such a spline has two major properties: It is entirely defined by the current positions (x_{mi}, y_{mi}) of the four master nodes P_{m0}, P_{m1}, P_{m2} and P_{m3}, and it is C^1-regular with the neighbouring splines (defined in the same way) at the two master nodes P_{m1} and P_{m2} (i.e. at $\xi = -1$ and $\xi = 1$).

The so-called closest point P_c, defined on the spline by a parameter ξ_c, is found by minimizing the distance to the slave node P_{s1}. A simple Newton–Raphson (NR) scheme is used for this purpose, because there is no closed-form for this minimization. The scheme starts from an arbitrary value of ξ between -1 and 1 (for example $\xi^{(0)} = 0$) and applies recursively the following formula:

$$\xi^{(i+1)} = \xi^{(i)} - r^{(i)}/r'^{(i)}. \tag{17}$$

The residual $r^{(i)}$ in this scheme is the dot product of the current vector $\overrightarrow{P_{s1}P_c^{(i)}}$ and a tangent vector to the spline at $P_c^{(i)}$, that is,

$$r^{(i)} = \left(x\left(\xi^{(i)}\right) - x_{s1}\right) \cdot x'\left(\xi^{(i)}\right) + \left(y\left(\xi^{(i)}\right) - y_{s1}\right) \cdot y'\left(\xi^{(i)}\right). \tag{18}$$

The derivation of $r'^{(i)}$ is straightforward, as well as the first and second derivatives of the coordinates x and y:

$$\begin{cases} x'(\xi) = \sum_{i=0}^{3} x_{mi} \cdot N'_i(\xi) \\ y'(\xi) = \sum_{i=0}^{3} y_{mi} \cdot N'_i(\xi) \end{cases} \tag{19}$$

$$\begin{cases} x''(\xi) = \sum_{i=0}^{3} x_{mi} \cdot N''_i(\xi) \\ y''(\xi) = \sum_{i=0}^{3} y_{mi} \cdot N''_i(\xi) \end{cases} \tag{20}$$

If the final value of ξ_c is outside the interval $[-1, 1]$, it means that the chosen master segment ($[P_{m1}P_{m2}]$ in our illustrative case) is inadequate and the other one ($[P_{m0}P_{m1}]$) should be tested. In the previous NR scheme, a limited number of iterations are usually necessary to achieve convergence, but it should be pointed out that a very high precision in the determination of ξ_c is desirable in order to achieve global convergence of the main problem. In the current version of the code, a target value of 10^{-16} for the residual $r^{(i)}$ has been found to ensure global convergence without slowing too much the computations. When the closest point P_c is obtained, it is straightforward to compute the current gap g and the normal and tangent vectors at the contact: The tangential vector $\overrightarrow{t_c}$ is obtained by normalizing the vector of coordinates $\left[x'(\xi_c), y'(\xi_c)\right]$, and the normal vector $\overrightarrow{n_c}$ is obtained by a $\pi/2$ rotation of $\overrightarrow{t_c}$.

In the current version of the code, frictionless contacts without adhesion are considered, and an augmented Lagrangian strategy is used (e.g. [33, 34]). Hence, the contact force is directed along the outward normal direction $\vec{n_c}$ and is expressed by

$$T_C = < -\lambda - k \cdot g > . \tag{21}$$

In this expression, $<>$ are the Macaulay brackets (only retaining a positive value of their argument), k is a contact stiffness penalizing interpenetration of the slave and master bodies, g is the computed gap (negative in case of interpenetration, positive otherwise) and λ is a Lagrange multiplier obtained from a previous stage (negative for repulsive force, positive for attractive force) and taken as a constant. Because an NR scheme will be used for solving the problem, both a force vector and a consistent tangent matrix are needed. For a given contact, the elementary versions $\boldsymbol{R_{ce}}$ and $\boldsymbol{K_{ce}}$ of this vector $\boldsymbol{R_c}$ and this matrix $\boldsymbol{K_c}$ will concern seven nodes (P_{s0}, P_{s1}, P_{s2}, P_{m0}, P_{m1}, P_{m2} and P_{m3}), that is, 14 degrees of freedom. Their expressions are given in [30]:

$$\boldsymbol{R_{ce}} = T_C A_s \boldsymbol{B_n} \tag{22}$$

$$\boldsymbol{K_{ce}} = \boldsymbol{K_{c1}} + \boldsymbol{K_{c2}} + \boldsymbol{K_{c3}} + \boldsymbol{K_{c4}}, \tag{23}$$

with the expressions of A_s, $\boldsymbol{B_n}$, $\boldsymbol{K_{c1}}$, $\boldsymbol{K_{c2}}$, $\boldsymbol{K_{c3}}$ and $\boldsymbol{K_{c4}}$ provided in Appendix B. These local contributions are added to the contact vector $\boldsymbol{R_c}$ and tangent contact matrix $\boldsymbol{K_c}$ of the whole problem

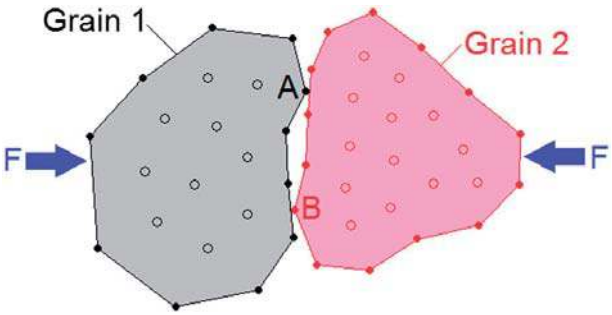

Figure 5. Example of pathological situation for the one-pass node-to-segment contact algorithm.

using the usual summation techniques. In a manner similar to classical DEM codes, wall elements are also used to ensure boundary conditions, and a wall-contact element is developed. Because it is very similar to that described previously, the details are provided in Appendix C.

As pointed out earlier, because of the possible presence of sharp angles in all grains, the one-pass node-to-segment is usually not sufficient to prevent large interpenetrations, whatever the choice of the role attributed to each member of a contacting couple of grains (i.e. master or slave). An example is provided in Figure 5, for which any distribution of the roles will lead to ignoring some penetration of a node in a grain. Hence, a two-pass node-to-segment approach is chosen, in order to restore symmetry and to avoid this kind of situation. Basically, it simply consists in computing $\boldsymbol{R_{ce}}$ and $\boldsymbol{K_{ce}}$ for all the boundary nodes of all the grains, instead of only doing so if these nodes belong to slave bodies in relation to a given master body. Hence, in the previous formalism, each node may successively occupy the position of a slave and of a master node. The two-pass strategy is known to induce some locking in the NR convergence, but this phenomenon was found to remain moderate in the present case (even in presence of very large deformations of the grains). In future versions of the code, modern approaches such as the mortar elements may be used to accelerate convergence.

4. TIME INTEGRATION SCHEME

The time stepping scheme developed to deal with the multibody meshfree problem exposed in the previous sections is based on a composite implicit dynamic scheme for which NR iterations are nested in the augmentation loops of the augmented Lagrangian procedure. The very first stage of this procedure is to compute the mass matrix $\boldsymbol{M}$ and the initial stiffness matrix $\boldsymbol{K_{i0}}$ (i.e. tangent stiffness matrix for zeros deformation) using the meshfree shape functions detailed in Section 2 and Appendix A. The damping matrix is then computed using the classical Rayleigh approach:

$$\boldsymbol{C} = \alpha \boldsymbol{M} + \beta \boldsymbol{K_{i0}}, \tag{24}$$

where α and β respectively control the damping at low and high frequencies and are material dependent. Initial values of field nodes positions, displacements, velocities and accelerations are stored in $2N \times 1$ vectors $\boldsymbol{X}$, $\boldsymbol{U}$, $\boldsymbol{U'}$ and $\boldsymbol{U''}$. Lagrange multipliers are set to zero for all boundary nodes of each grain, and a time step Δt is chosen.

As explained in [35], the classical Newmark implicit integration scheme with the so-called trapezoidal rule is sometimes found to be unstable in the case of strongly non-linear problems for which dynamic equilibrium is searched at each time step. This is the case in the present study. The backward differential formula (BDF) is not completely appropriate either, because the BDF-1 scheme is only first-order accurate, and higher-order BDF schemes require a fixed time step (which prevents time-refinement in case of no-convergence of the NR loops). Also, according to Hauret and Le Tallec [36], the popular Hilber-Hughes-Taylor scheme seems to lose some of its advantages in highly nonlinear problems.

For these reasons, a reliable scheme, apparently inspired by Bank *et al.* [37] and detailed in [35] in a mechanical framework, is used here. It consists in dividing each step Δt in two equal sub-steps $\Delta t/2$. In the first one, the Newmark trapezoidal rule is applied, while in the second one, a three-points Euler backward formula is employed. More specifically, let us assume that the displacement

vector ${}^{t}\boldsymbol{U}$, the velocity vector ${}^{t}\dot{\boldsymbol{U}}$ and the acceleration vector ${}^{t}\ddot{\boldsymbol{U}}$ are known at a time step t. In the first sub-step, the following relations hold:

$$ {}^{t+\Delta t/2}\dot{\boldsymbol{U}} = {}^{t}\dot{\boldsymbol{U}} + \frac{\Delta t}{4}\left({}^{t}\ddot{\boldsymbol{U}} + {}^{t+\Delta t}/\boldsymbol{2}\ddot{\boldsymbol{U}}\right) \tag{25} $$

$$ {}^{t+\Delta t/2}\boldsymbol{U} = {}^{t}\boldsymbol{U} + \frac{\Delta t}{4}\left({}^{t}\dot{\boldsymbol{U}} + {}^{t+\Delta t/2}\dot{\boldsymbol{U}}\right) \tag{26} $$

The displacement ${}^{t+\Delta t/2}\boldsymbol{U}$ at the end of the sub-step is obtained using an NR scheme of the form of Equation (14), which is detailed later on in this section. Nodal velocities ${}^{t+\Delta t/2}\dot{\boldsymbol{U}}$ and accelerations ${}^{t+\Delta t/2}\ddot{\boldsymbol{U}}$ are then directly obtained by the application of Equations (25) and (26). The second sub-step relies on the following relations:

$$ {}^{t+\Delta t}\dot{\boldsymbol{U}} = \frac{1}{\Delta t}{}^{t}\boldsymbol{U} - \frac{4}{\Delta t}{}^{t+\Delta t/2}\boldsymbol{U} + \frac{3}{\Delta t}{}^{t+\Delta t}\boldsymbol{U} \tag{27} $$

$$ {}^{t+\Delta t}\ddot{\boldsymbol{U}} = \frac{1}{\Delta t}{}^{t}\dot{\boldsymbol{U}} - \frac{4}{\Delta t}{}^{t+\Delta t/2}\dot{\boldsymbol{U}} + \frac{3}{\Delta t}{}^{t+\Delta t}\dot{\boldsymbol{U}} \tag{28} $$

Again, the final displacement ${}^{t+\Delta t}\boldsymbol{U}$ is obtained after NR iterations, and the velocities and accelerations are directly obtained by Equations (27) and (28).

For each of these two sub-steps, the convergence is obtained by applying several NR iterations nested in augmentation loops. More specifically, starting from known values of $\boldsymbol{U}$, $\dot{\boldsymbol{U}}$, $\ddot{\boldsymbol{U}}$ and of the Lagrange multipliers at all the boundary nodes of each grain, an NR scheme is performed. In this scheme, the Lagrange multipliers are considered as constant. The dynamic terms used for the computation of the tangent matrix and of the residual vectors depend on the sub-step and are provided in Appendix D. When this scheme has converged, the Lagrange multipliers are augmented, that is, each multiplier λ is replaced by $- < -\lambda - k \cdot g >$. In this expression, k is the contact stiffness, and g is the gap of the corresponding contact. Then, a new NR scheme is performed, followed by a new augmentation, and so on. The augmentation loops are stopped when, at a given augmentation stage, the maximum gap (for which there is contact, i.e., for which $\lambda < 0$) in absolute value is smaller than some prescribed value.

This classical scheme (see [33, 34] for example) ensures a complete control on dynamic equilibrium and on interpenetrations between grains, with a larger robustness and ease of implementation than the strict use of Lagrange multipliers in the tangent system. The price to pay, however, is an increase of the number of iterations at each time step. This is especially the case if very small gaps are targeted, because of some convergence locking phenomena related to the two-pass contact algorithm. A good way to accelerate convergence while using relatively large time steps is to use a rather small value k_0 of the contact stiffness k (e.g. of the order of magnitude of the material Young modulus) but to increase this stiffness by a certain factor α_k (e.g. equal to 2) at each augmentation loop. Of course, k should be reset to its initial value k_0 at the beginning of each sub-step. This technique seems to counteract the effect of the two-pass contact algorithm without degrading the robustness. The whole solver is summarized in the flowchart of Figure 6.

5. EXAMPLES

A prototype version of the code called Multibody ELement-free Open code for DYnamic simulation (MELODY) was implemented in a MATLAB [38] environment in order to evaluate the convergence of the NR loops inside an implicit scheme. This implementation is available for download at http://guilhem.mollon.free.fr. A first example of simulation involves the fall and bouncing of a single circular grain on a fixed horizontal wall. The material parameters (Young modulus, Poisson coefficient, density, damping coefficients) are set in order for the bouncing to imply a somewhat large

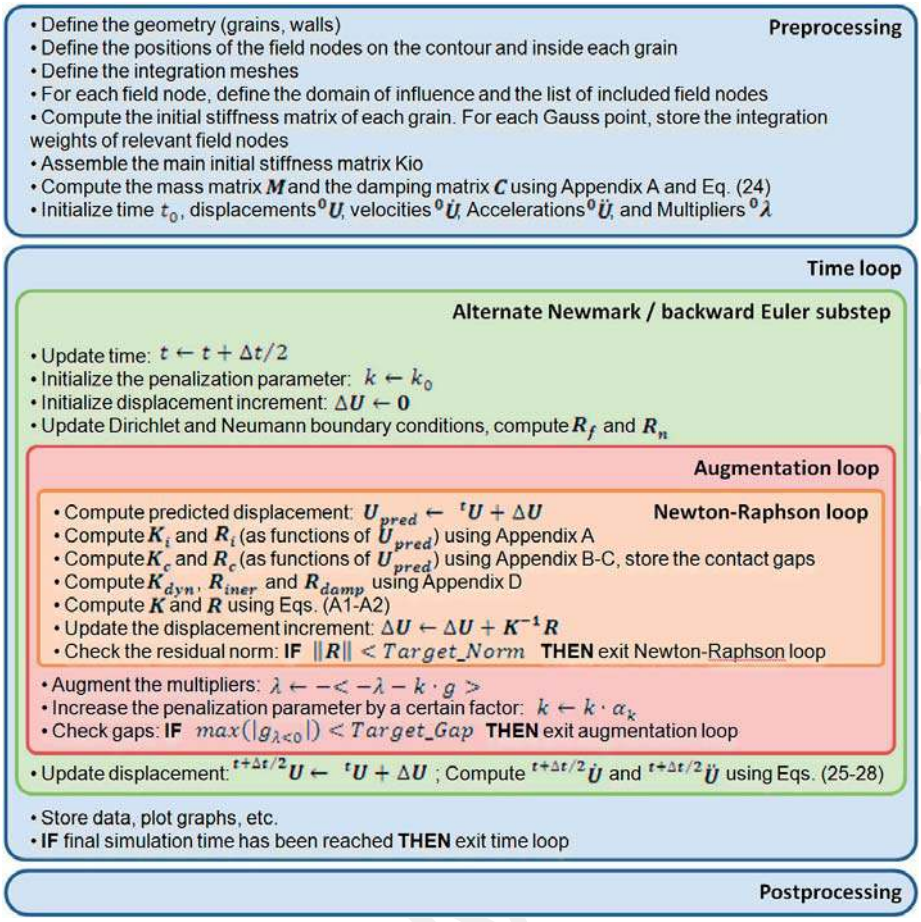

Figure 6. Main algorithm of the code.

Table I. Parameters of the two numerical examples.

	Example 1	Example 2
Number of bodies	1	100
Diameters (mean; standard deviation)	(2; 0)	(17.6; 3.2)
Young modulus	10^6 Pa	10^3 Pa
Poisson coefficient	0.49	0.49
Density	1 000 kg/m^3	1 kg/m^3
Alpha Rayleigh damping	0.01	0.01
Beta Rayleigh damping	0.1	0.1
Number of nodes	105	9 761
Number of integration cells	172	16 444
Number of Gauss points	1 032	98 664
Shape functions	RPIM	RPIM
Neighbours in the influence domain	20	20
Penalization factor	10^8	10^4
Prescribed residual norm	10^{-12}	10^{-12}
Prescribed maximum gap	10^{-12}	10^{-6}
Time step	Variable (Figure 8)	0.2 s
Number of time steps	Variable (Figure 8)	1 800
Simulated time	3 s	360 s
Computation time	A few minutes	$\cong$ 3 weeks

The computation times were obtained on a single CPU in a MATLAB environment.

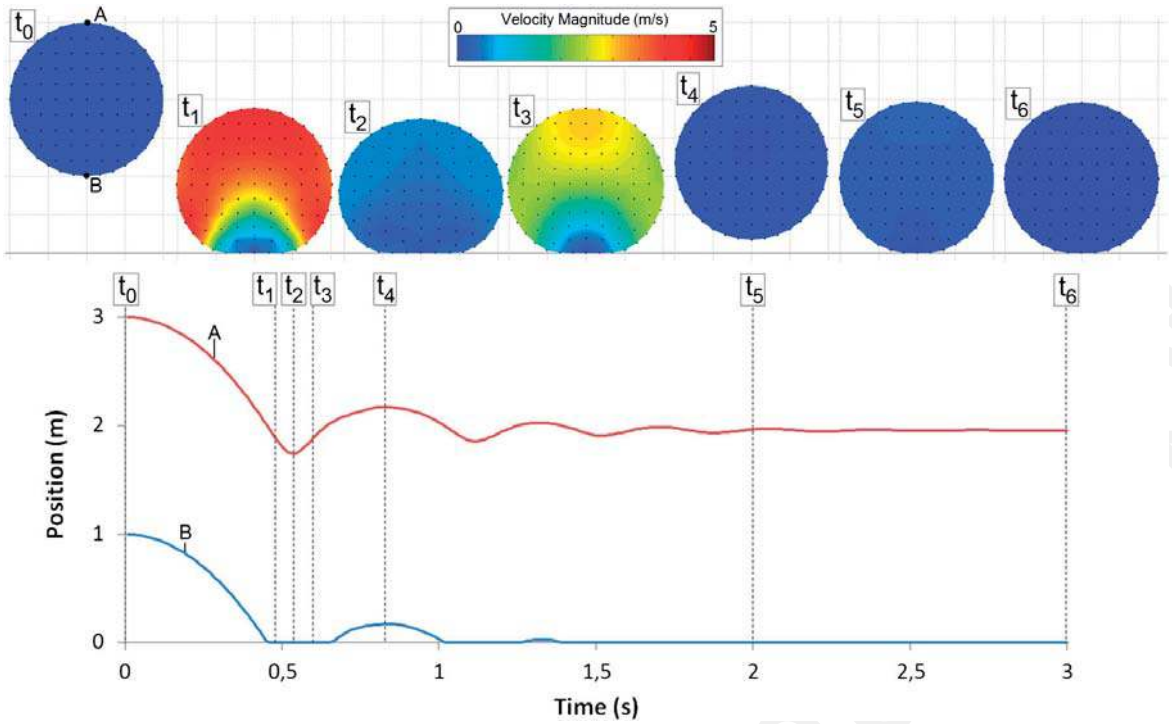

Figure 7. Vertical position along time of the upper (A) and lower (B) field nodes of a bouncing disc with damping and associated velocity fields.

deformation of the grain and a stabilization under gravity after a few oscillations (Table I). Figure 7 shows the velocity field and the deformed shape of the disc at several stages of the simulation, as well as two curves showing the vertical positions of points A and B, located at the upper and lower extremities of the grain. The stability of the numerical scheme is further assessed in Figure 8, which focuses on the vertical motion of point B for $t > 2s$. This time corresponds to a period for which this point is supposed to remain still, thanks to the damping applied to the disc material. More specifically, a target contact gap of 10^{-12} is chosen, meaning that the vertical position of point B should remain in this order of magnitude. The composite scheme (Figure 8(d–f)) described in Section 4 is compared with a classical Newmark scheme with trapezoidal rule (Figure 8(a–c)), with two different time steps for each integrator. These time steps are chosen in order for the comparison to be fair, that is, the time step used for the Newmark scheme is half that used for the composite scheme (because the latter divides each time step in two equal sub-steps). The results clearly show the superiority of the composite scheme. Figure 8(d) demonstrates that the vertical position of point B remains below 10^{-12} in absolute value, which is consistent with the chosen target gap. In turn, the vertical velocity remains lower than 2.10^{-10}, and the vertical acceleration remains lower than 3.10^{-7}, both in absolute value. This demonstrates an excellent stability of the simulation. In contrast, the classical Newmark scheme shows some clear signs of instability. In Figure 8(a), it appears that the vertical position of point B locally exceeds 10^{-5}, meaning a temporary loss of contact. This is related to strong oscillations of the velocity and acceleration signals (Figure 8(b–c)).

A second example (detailed in Table I) is proposed in order to evaluate the ability of the implicit scheme to converge in case of large numbers of grains, of large deformations and of dynamic situations. A sample of 100 grains with sand-like shapes is generated using the 'Packing2D' algorithm (described in [39] and available for download at http://guilhem.mollon.free.fr), and field nodes are positioned using the meshing tool proposed by Persson and Strand [40]. An average number of 97 field nodes are used in each grain, with a minimum number of 24 for the smallest grain. As described in [39], the initial sample is generated in a square domain, without any contact between the grains and with the four surrounding walls. This situation is shown in Figure 9(a). The grains are first compacted under gravity (Figure 9(b)) and are submitted to uniaxial compression (i.e. the upper wall moves down, Figure 9(c)). The sample is then submitted to biaxial shearing at constant volume

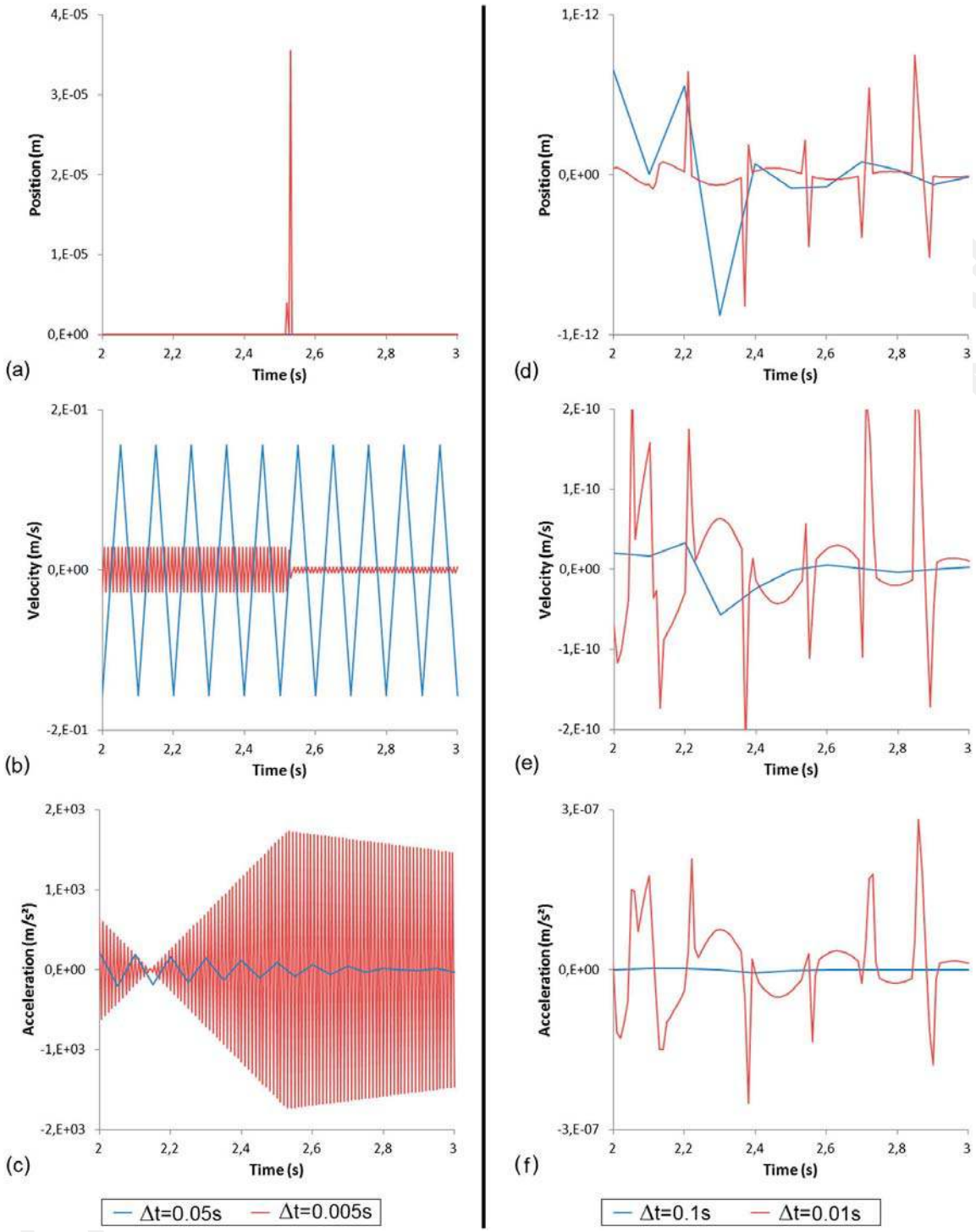

Figure 8. Vertical position, velocity and acceleration of node B between $\mathbf{t_5}$ and $\mathbf{t_6}$, for two different time steps; (a–c) Newmark trapezoidal rule; (d–f) composite Newmark–Euler integration scheme.

(i.e. the lateral walls are moved horizontally, and the position of the vertical wall is adjusted in order for the volume of the domain to remain constant, Figure 9(d)). Figure 10 shows the evolution of the number of boundary nodes in a contact state during simulation. In the gravitational deposition stage, this proportion goes from 0% to 13% (with some oscillations related to the damping of kinetic energy), and it reaches about 37% after the uniaxial compression phase. During shearing and relaxation stages, the proportion of contacting boundary nodes remains rather constant, between 35% and 40%. During all this simulation, the implicit solver finds convergence with the expected quadratic rate, despite the strong nonlinearities brought by the large number of contacts, the large displacements and large deformations and the dynamic terms. This excellent robustness is directly related

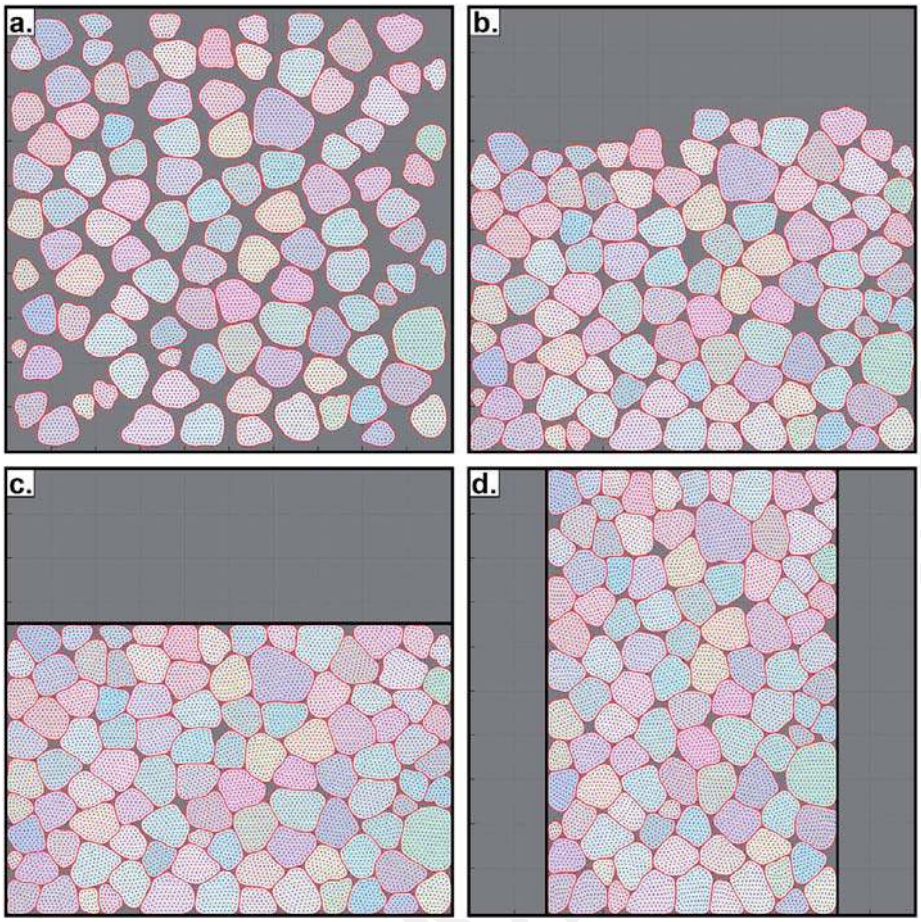

Figure 9. Sample of 100 grains submitted to mechanical solicitations. (a) Step 0: initial state; (b) Step 600: compaction under gravity; (c) Step 900: uniaxial vertical compression; (d) Step 1600: biaxial shearing at constant volume.

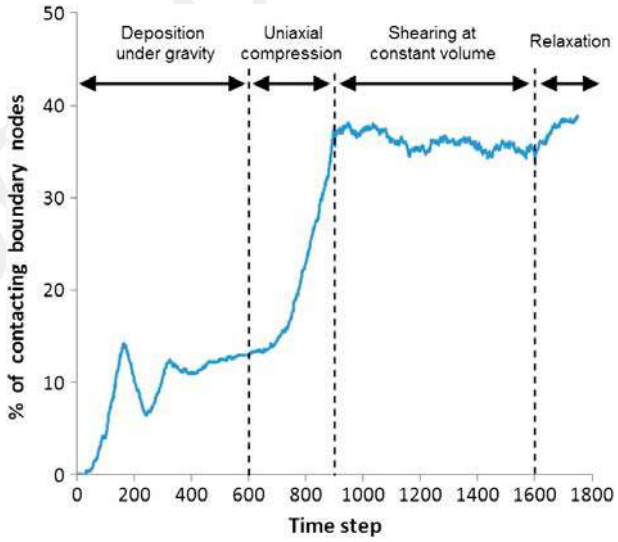

Figure 10. Proportion of boundary nodes in a contact state during the different stages of the simulation.

to the rigorous derivation of a consistent stiffness matrix at each iteration (Appendices A–D), to the use of an augmented Lagrangian strategy coupled with smooth contact elements and to the stability of the chosen composite time integrator. The shear stress field obtained at the end of the compaction phase is provided in Figure 11 and demonstrates the quality of the solution in terms of smoothness.

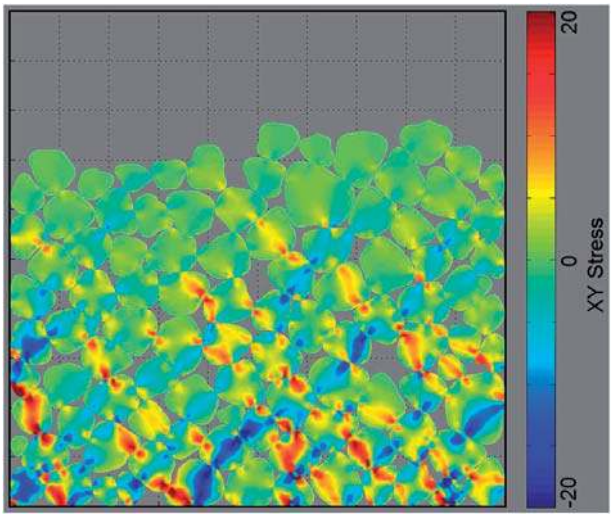

Figure 11. Shear stress at the end of the compaction under gravity (Step 600).

6. CONCLUSION AND PERSPECTIVES

In this paper, a numerical framework has been proposed in order to simulate the behaviour of highly deformable granular materials. Each grain is considered as a deformable body. The displacement field on each grain is interpolated by the means of meshfree shape functions, and the interactions of each body with its neighbours are dealt with using a node-to-segment smooth contact elements with C^1-continuity on the whole grains contours. A classical Newton–Raphson scheme nested in augmentation loops is applied to solve at each time step this nonlinear problem, enforcing prescribed conditions on the maximum force residual and contact interpenetration. Time integration is performed using a composite Newmark–Euler scheme, ensuring stability while keeping the possibility to refine the time steps in case of lack of convergence. The framework is applied on two examples to demonstrate its potential.

As mentioned in the previous section, the current version of the code is developed in a MATLAB environment, under the acronym MELODY, and is available for free download at http://guilhem.mollon.free.fr. This prototype version made it possible to demonstrate the feasibility of the method, and a more efficient implementation in compiled language may now be considered. The computational cost (Table I) is very high in the prototype version because of the use of an interpreted language. Future evolutions towards a rewritten version of the code in a compiled language, some optimizations of the proximity and contact detections and some relevant parallelization routines might increase the computational efficiency by a couple of decades.

To the author's best knowledge, experimental results of the compression and shear of highly deformable frictionless granular materials with local measurements (e.g. grains kinematics) are not available in the scientific literature, preventing an experimental validation of the method. It seems that, at the moment, the only way to validate the code and to quantify its accuracy is to perform a convergence-in-mesh study (or, more appropriately, a study of the convergence in nodal density). Also, comparisons may be made with DEM codes in the limit of very rigid grains, and with FEM codes in the limit of small numbers of grains. This work is kept for future studies.

Besides the issues of computational efficiency and accuracy, a large number of improvements might be interesting for practical use of the code in a scientific or industrial context:

- The Rayleigh damping currently used is not objective, that is, it is frame dependent. Indeed, Rayleigh damping is usually used in the context of linear structures and is not adapted to large displacements because it tends to damp rigid body motions. Hence, an objective formulation of the damping matrix will have to be performed, with updating of this matrix at each iteration (e.g. [41]). Failing this, the code is currently restricted to undamped systems, or to quasi-static situations for which the damping is only used to dissipate kinetic energy.

- As an alternative to the previous point, a full viscoelastic constitutive model may be considered. Hyperelasticity with a condition of incompressibility would also be welcome, because solids subjected to very large deformations often exhibit this behaviour. In the long run, implementation of plasticity seems necessary, as well as phenomena such as localized deformations and grain breakage.
- It will be necessary to develop contact elements with tangential forces, including phenomena such as friction and adhesion, which are pivotal to the proper modelling of granular behaviours. The spline elements presented in the present study seem to be a good basis for this perspective.
- Multi-physics coupling will be made possible in the proposed framework, involving, for example, heat creation and conduction or electrical phenomena. A rigorous fluid–grain coupling is also considered in order to deal with complex fluids, such as grease and biological fluids.
- A challenging long-term purpose is to extend this code to the 3D case, in order to take advantage of 3D realistic grains generators developed in previous studies [42, 43]. A full 3D code will also make it possible to simulate realistic granular situations for which top of the range measurements was performed, using, for example, the recent progress in X-ray microtomography techniques [44]. This may allow future progress in a better understanding of granular behaviours in complex situations.

APPENDIX A: UNBALANCED FORCES VECTOR, CONSISTENT TANGENT MATRIX, AND MASS MATRIX

The consistent tangent matrix used in Equation (14) is the sum of several terms:

$$\boldsymbol{K} = \boldsymbol{K_i} + \boldsymbol{K_c} + \boldsymbol{K_{dyn}}. \tag{A.1}$$

$\boldsymbol{K_c}$ is the contribution of the contacts (described in Section 3 and Appendix B), and $\boldsymbol{K_{dyn}}$ is the contribution of dynamic effects and depends on the chosen integration scheme (Section 4). Hence, this appendix focuses on the computation of $\boldsymbol{K_i}$, which is the classical 'tangent stiffness matrix' accounting for the constitutive model and for the large deformations and displacements effects in the body. The computation of the mass matrix $\boldsymbol{M}$ is also covered here. In the same manner, the whole residual vector is the sum of several terms:

$$\boldsymbol{R} = \boldsymbol{R_i} + \boldsymbol{R_f} + \boldsymbol{R_n} + \boldsymbol{R_c} + \boldsymbol{R_{iner}} + \boldsymbol{R_{damp}}, \tag{A.2}$$

where $\boldsymbol{R_i}$ corresponds to the internal forces, $\boldsymbol{R_f}$ to the body forces, $\boldsymbol{R_n}$ to the external forces (Neumann boundary conditions), $\boldsymbol{R_c}$ to the contact forces, $\boldsymbol{R_{iner}}$ to the inertial forces and $\boldsymbol{R_{damp}}$ to the damping forces. The contact forces are dealt with in Section 3 and Appendices B and C, and the dynamic (inertial and damping) forces are covered in Section 4 and Appendix D. Hence, this appendix only deals with the terms $\boldsymbol{R_i}$, $\boldsymbol{R_f}$ and $\boldsymbol{R_n}$.

As explained in Section 2, a Gauss quadrature integration is used to compute the four surface integrals of Equation (12). The term $\int_\Omega \overline{\overline{S}}\left[\vec{u}\right] : \overline{\overline{e}}\left[\vec{w}\right] d\Omega$ is related to $\boldsymbol{K_i}$ and $\boldsymbol{R_i}$, while the other surface integrals only require the computation of the mass matrix $\boldsymbol{M}$. For the computation of $\boldsymbol{K_i}$ and $\boldsymbol{R_i}$, each Gauss point of the mesh has the following contributions:

$$\boldsymbol{K_{ie}} = \left(\boldsymbol{B_t}^T \boldsymbol{A}\, \boldsymbol{B_t} + \boldsymbol{G}^T \boldsymbol{B_p}\right) w_e J_e, \tag{A.3}$$

$$\boldsymbol{R_{ie}} = \left(\boldsymbol{B_t}^T \boldsymbol{S}\right) w_e J_e, \tag{A.4}$$

where w_e is the weight of the Gauss point in the quadrature used for the current triangular element, and J_e is the Jacobian of the affine transform from the reference triangle to the current triangular element. $\boldsymbol{A}$ is the 3×3 elasticity matrix. It is simply a rewriting of the elasticity tensor $\boldsymbol{A}$, which is kept constant in the present study because a Saint Venant–Kirchhoff constitutive model is chosen. $\boldsymbol{S}$ is the second Piola–Kirchhoff stress tensor, given in engineering notation by

$$\boldsymbol{S} = \left[S_{xx}, S_{yy}, S_{xy}\right]^T = \boldsymbol{A}\boldsymbol{E}. \tag{A.5}$$

$\boldsymbol{E}$ is the Green–Lagrange strain tensor, obtained in a classical way from the gradient tensor $\boldsymbol{F}$:

$$\boldsymbol{E} = \begin{bmatrix} E_{xx} \\ E_{yy} \\ 2E_{xy} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} F_{xx}^2 + F_{yx}^2 - 1 \\ F_{yy}^2 + F_{xy}^2 - 1 \\ F_{xx}F_{xy} + F_{yy}F_{yx} \end{bmatrix} \tag{A.6}$$

The term $\boldsymbol{G}$ is expressed in the following manner:

$$\boldsymbol{G} = \begin{bmatrix} \Phi_{1,\mathrm{x}} & 0 & \cdots & \Phi_{\mathrm{ie,x}} & 0 & \cdots & \Phi_{\mathrm{ne,x}} & 0 \\ \Phi_{1,\mathrm{y}} & 0 & & \Phi_{\mathrm{ie,y}} & 0 & & \Phi_{\mathrm{ne,y}} & 0 \\ 0 & \Phi_{1,\mathrm{x}} & & 0 & \Phi_{\mathrm{ie,x}} & & 0 & \Phi_{\mathrm{ne,x}} \\ 0 & \Phi_{1,\mathrm{y}} & & 0 & \Phi_{\mathrm{ie,y}} & & 0 & \Phi_{\mathrm{ne,y}} \end{bmatrix} \tag{A.7}$$

In this expression, the partial derivatives of the shape functions at the Gauss point are computed using Equations (9) and (10), and the index $1 \leqslant i_e \leqslant n_e$ indicates that the only shape functions considered in the expression are those that contribute to the Gauss point (i.e. those for which the Gauss point belongs to the domain of influence). The displacements (obtained from the result ${}^{(i-1)}\boldsymbol{U}$ of the previous NR iteration) of the nodes associated with these shape functions are stored in the $2n_e \times 1$ column vector:

$$\boldsymbol{U_e} = \begin{bmatrix} u_{x1} & u_{y1} & \cdots & u_{xie} & u_{yie} & \cdots & u_{xne} & u_{yne} \end{bmatrix}^T. \tag{A.8}$$

We then define a 4×1 vector $\boldsymbol{H} = \boldsymbol{G}\boldsymbol{U_e} = [H_1 H_2 H_3 H_4]^{\boldsymbol{T}}$, and the gradient tensor $\boldsymbol{F}$ is given by

$$\boldsymbol{F} = \begin{bmatrix} H_1 + 1 & H_2 \\ H_3 & H_4 + 1 \end{bmatrix} = \begin{bmatrix} F_{xx} & F_{xy} \\ F_{yx} & F_{yy} \end{bmatrix}. \tag{A.9}$$

The matrix $\boldsymbol{B_t}$ is the sum of two terms $\boldsymbol{B_e}$ and $\boldsymbol{B_u}$:

$$\boldsymbol{B_e} = \begin{bmatrix} \Phi_{1,\mathrm{x}} & 0 & \cdots & \Phi_{\mathrm{ie,x}} & 0 & \cdots & \Phi_{\mathrm{ne,x}} & 0 \\ 0 & \Phi_{1,\mathrm{y}} & & 0 & \Phi_{\mathrm{ie,y}} & & 0 & \Phi_{\mathrm{ne,y}} \\ \Phi_{1,\mathrm{y}} & \Phi_{1,\mathrm{x}} & & \Phi_{\mathrm{ie,y}} & \Phi_{\mathrm{ie,x}} & & \Phi_{\mathrm{ne,y}} & \Phi_{\mathrm{ne,x}} \end{bmatrix} \tag{A.10}$$

$$\boldsymbol{B_u} = \begin{bmatrix} H_1\Phi_{1,\mathrm{x}} & H_3\Phi_{1,\mathrm{x}} & \cdots & H_1\Phi_{\mathrm{ne,x}} & H_3\Phi_{\mathrm{ne,x}} \\ H_2\Phi_{1,\mathrm{y}} & H_4\Phi_{1,\mathrm{y}} & & H_2\Phi_{\mathrm{ne,y}} & H_4\Phi_{\mathrm{ne,y}} \\ H_1\Phi_{1,\mathrm{y}} + H_2\Phi_{1,\mathrm{x}} & H_3\Phi_{1,\mathrm{y}} + H_4\Phi_{1,\mathrm{x}} & & H_1\Phi_{\mathrm{ne,y}} + H_2\Phi_{\mathrm{ne,x}} & H_3\Phi_{\mathrm{ne,y}} + H_4\Phi_{\mathrm{ne,x}} \end{bmatrix}. \tag{A.11}$$

Finally, the matrix $\boldsymbol{B_p}$ is expressed by

$$\boldsymbol{B_p} = \begin{bmatrix} S_{xx}\Phi_{1,\mathrm{x}} + S_{xy}\Phi_{1,\mathrm{y}} & 0 & \cdots & S_{xx}\Phi_{\mathrm{ne,x}} + S_{xy}\Phi_{\mathrm{ne,y}} & 0 \\ S_{xy}\Phi_{1,\mathrm{x}} + S_{yy}\Phi_{1,\mathrm{y}} & 0 & & S_{xy}\Phi_{\mathrm{ne,x}} + S_{yy}\Phi_{\mathrm{ne,y}} & 0 \\ 0 & S_{xx}\Phi_{1,\mathrm{x}} + S_{xy}\Phi_{1,\mathrm{y}} & & 0 & S_{xx}\Phi_{\mathrm{ne,x}} + S_{xy}\Phi_{\mathrm{ne,y}} \\ 0 & S_{xx}\Phi_{1,\mathrm{x}} + S_{xy}\Phi_{1,\mathrm{y}} & & 0 & S_{xx}\Phi_{\mathrm{ne,x}} + S_{xy}\Phi_{\mathrm{ne,y}} \end{bmatrix}. \tag{A.12}$$

The elementary $2n_e \times 2n_e$ tangent matrix $\boldsymbol{K_{ie}}$ of the current Gauss point should be added to the global tangent stiffness matrix $\boldsymbol{K_i}$, at the $2n_e$ appropriate positions of the concerned degrees of freedom. The same operations should be performed for the elementary $2n_e \times 1$ residual vector $\boldsymbol{R_{ie}}$ as well in order to construct the global vector of internal forces $\boldsymbol{R_i}$. Because $\boldsymbol{K_i}$ and $\boldsymbol{R_i}$ are displacement dependent, it is necessary to recompute them at each iteration of the NR solver, which would imply to compute the first derivatives of the shape functions a tremendous number of times in a simulation. A good way to speed up the computations is to compute only once the values and first derivatives of the shape functions $\left(\Phi_{\mathrm{ie}}, \Phi_{\mathrm{ie,x}}, \Phi_{\mathrm{ie,y}}\right)_{1 \leqslant ie \leqslant ne}$ at each Gauss point of the integration mesh and to store these values in memory. They may then be conveniently called when needed during simulation and also during post-processing (e.g. stress fields plotting).

The surface integrations related to the mass matrix $\boldsymbol{M}$ and the body forces $\boldsymbol{R_f}$ are performed on the same basis, that is, using an elementary mass matrix $\boldsymbol{M_e}$ and an elementary body force vector $\boldsymbol{R_{fe}}$ for each Gauss point, with

$$\boldsymbol{M_e} = \rho_e w_e J_e \begin{bmatrix} \Phi_1\Phi_1 & 0 & \dots & \Phi_1\Phi_{ne} & 0 \\ 0 & \Phi_1\Phi_1 & & 0 & \Phi_1\Phi_{ne} \\ \vdots & & \ddots & & \vdots \\ \Phi_{ne}\Phi_1 & 0 & \dots & \Phi_{ne}\Phi_{ne} & 0 \\ 0 & \Phi_{ne}\Phi_1 & & 0 & \Phi_{ne}\Phi_{ne} \end{bmatrix} \tag{A.13}$$

$$\boldsymbol{R_{fe}} = w_e J_e \begin{bmatrix} f_x\,\Phi_1 & f_y\Phi_1 & \dots & f_x\Phi_{ne} & f_y\Phi_{ne} \end{bmatrix}^{\boldsymbol{T}}, \tag{A.14}$$

where ρ_e is the unit weight of the material at the Gauss point, and (f_x, f_y) are the components of the body force density. After assembling, the mass matrix $\boldsymbol{M}$ such obtained is used to compute the damping matrix $\boldsymbol{C}$, and both of them are used to compute the dynamic (i.e. inertial and damping) nodal forces, as explained in Section 4. Finally, the nodal forces $\boldsymbol{R_n}$ related to prescribed Neumann boundary conditions are integrated on the relevant segments of the grain boundaries in the usual manner.

APPENDIX B: CONTACT FORCES VECTOR AND ASSOCIATED CONSISTENT TANGENT MATRIX

As derived in Section 3, the contribution of a given slave node to the global force vector concerns seven nodes (P_{s0}, P_{s1}, P_{s2}, P_{m0}, P_{m1}, P_{m2} and P_{m3}), that is, 14 degrees of freedom. This ordering is followed in the following equations. The terms used in Equations (22) and (23) are given in [30] and provided hereafter with minor typos corrected and notations adapted to the present work:

$$\boldsymbol{K_{c1}} = -\frac{T_C \boldsymbol{A}_s g}{x'^2 + y'^2}\left[\left(n_x x'' + n_y y''\right)\boldsymbol{D} + \boldsymbol{B_{n,\xi}}\right]\cdot\left[\left(n_x x'' + n_y y''\right)\boldsymbol{D} + \boldsymbol{B_{n,\xi}}\right]^T \tag{B.1}$$

$$\boldsymbol{K_{c2}} = -T_C A_s\left[-\boldsymbol{B_{n,\xi}}\boldsymbol{D}^T - \boldsymbol{D}\boldsymbol{B}_{\boldsymbol{n,\xi}}^T - \left(n_x x'' + n_y y''\right)\boldsymbol{D}\boldsymbol{D}^T\right] \tag{B.2}$$

$$\boldsymbol{K_{c3}} = k A_s \boldsymbol{B_n}\boldsymbol{B_n^T} \tag{B.3}$$

$$\boldsymbol{K_{c4}} = -0.5 T_c \boldsymbol{B_n}\left(\boldsymbol{A_0} + \boldsymbol{A_1}\right)^T \tag{B.4}$$

with,

$$\boldsymbol{D} = \frac{\boldsymbol{B_t} + g\boldsymbol{B_{n,\xi}}}{x'^2 + y'^2 - g n_x x'' - g n_y y''} \tag{B.5}$$

$$\boldsymbol{B_n} = -\begin{bmatrix} 0 & 0 & -n_x & -n_y & 0 & 0 & n_x N_0 & n_y N_0 & n_x N_1 & n_y N_1 & n_x N_2 & n_y N_2 & n_x N_3 & n_y N_3 \end{bmatrix}^T \tag{B.6}$$

$$\boldsymbol{B_t} = -\begin{bmatrix} 0 & 0 & -x' & -y' & 0 & 0 & x' N_0 & y' N_0 & x' N_1 & y' N_1 & x' N_2 & y' N_2 & x' N_3 & y' N_3 \end{bmatrix}^T \tag{B.7}$$

$$\boldsymbol{B_{n,\xi}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & n_x N'_0 & n_y N'_0 & n_x N'_1 & n_y N'_1 & n_x N'_2 & n_y N'_2 & n_x N'_3 & n_y N'_3 \end{bmatrix}^T \tag{B.8}$$

$$\boldsymbol{A_0} = \begin{bmatrix} t_{0x} & t_{0y} & -t_{0x} & -t_{0y} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{B.9}$$

$$\boldsymbol{A_1} = \begin{bmatrix} 0 & 0 & -t_{1x} & -t_{1y} & t_{1x} & t_{1y} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{B.10}$$

The terms (n_x, n_y), (t_{0x}, t_{0y}) and (t_{1x}, t_{1y}) are, respectively, the coordinates of the outward normal vector of the master body $\overrightarrow{n_c}$ and of the unit tangent vectors $\overrightarrow{t_{s0}}$ and $\overrightarrow{t_{s1}}$ linking the slave nodes (Figure 4(b)). The contact shape functions N_i and their first derivatives N'_i are evaluated at the contact point P_c, using the local coordinate ξ_c. This is also the case of the first and second derivatives of the coordinates of P_c, namely, $\left(x', y'\right)$ and $\left(x'', y''\right)$, using Equations (19) and (20). g is the signed gap, k is the contact stiffness and T_C is the contact force given by Equation (21). Finally, A_s is the surface onto which the contact force T_C is applied and is given by

$$A_s = \left(L_{s0} + L_{s1}\right)/2, \tag{B.11}$$

where L_{s0} and L_{s1} are the distances between the relevant slave nodes (Figure 4(b)).

APPENDIX C: WALL CONTACT ELEMENT

The wall contact element introduced in this appendix is basically a simplified version of the element described in Appendix B. It is based on the use of Equations (22) and (23), with

$$\boldsymbol{K_{c1}} = \boldsymbol{K_{c2}} = [\boldsymbol{0}]_{6\times 6} \tag{C.1}$$

$$\boldsymbol{K_{c3}} = k A_s \boldsymbol{B_n} \boldsymbol{B_n^T} \tag{C.2}$$

$$\boldsymbol{K_{c4}} = -0.5 T_c \boldsymbol{B_n} \left(\boldsymbol{A_0} + \boldsymbol{A_1}\right)^T \tag{C.3}$$

and

$$\boldsymbol{B_n} = -\begin{bmatrix} 0 & 0 & -n_x & -n_y & 0 & 0 \end{bmatrix}^T \tag{C.4}$$

$$\boldsymbol{A_0} = \begin{bmatrix} t_{0x} & t_{0y} & -t_{0x} & -t_{0y} & 0 & 0 \end{bmatrix}^T \tag{C.5}$$

$$\boldsymbol{A_1} = \begin{bmatrix} 0 & 0 & -t_{1x} & -t_{1y} & t_{1x} & t_{1y} \end{bmatrix}^T . \tag{C.6}$$

The six degrees of freedom implied in this element are those of the three slave nodes presented in Figure 4. The elementary tangent matrix and force vector such obtained are added to the global tangent system accordingly.

APPENDIX D: DYNAMIC TERMS

As explained in Appendix A, the consistent tangent matrix $\boldsymbol{K}$ and the residual vector $\boldsymbol{R}$ used in the tangent system of Equation (14) are composed of several terms:

$$\boldsymbol{K} = \boldsymbol{K_i} + \boldsymbol{K_c} + \boldsymbol{K_{dyn}} \tag{D.1}$$

$$\boldsymbol{R} = \boldsymbol{R_i} + \boldsymbol{R_f} + \boldsymbol{R_n} + \boldsymbol{R_c} + \boldsymbol{R_{iner}} + \boldsymbol{R_{damp}}. \tag{D.2}$$

The terms $\boldsymbol{K_i}$, $\boldsymbol{R_i}$, $\boldsymbol{R_f}$ and $\boldsymbol{R_n}$ are detailed in Appendix A, and the contact terms $\boldsymbol{K_c}$ and $\boldsymbol{R_c}$ are given in Appendices B and C. The dynamic terms $\boldsymbol{K_{dyn}}$, $\boldsymbol{R_{iner}}$ and $\boldsymbol{R_{damp}}$ are scheme dependent. Hence, their expressions are different in each of the two sub-steps that are used in the composite scheme proposed by Bathe [35]. In the first sub-step, when a Newmark trapezoidal rule is used on a $\Delta t/2$ interval, the following formulas apply:

$$\boldsymbol{K_{dyn}} = \frac{16}{\Delta t^2}\boldsymbol{M} + \frac{4}{\Delta t}\boldsymbol{C} \tag{D.3}$$

$$\boldsymbol{R_{iner}} = -\boldsymbol{M}\left[\frac{16}{\Delta t^2}\left({}^{t+\Delta t/2}\boldsymbol{U}^{(i-1)} - {}^{t}\boldsymbol{U}\right) - \frac{8}{\Delta t}{}^{t}\dot{\boldsymbol{U}} - {}^{t}\ddot{\boldsymbol{U}}\right] \tag{D.4}$$

$$\boldsymbol{R_{damp}} = -\boldsymbol{C}\left[\frac{4}{\Delta t}\left({}^{t+\Delta t/2}\boldsymbol{U}^{(i-1)} - {}^{t}\boldsymbol{U}\right) - {}^{t}\dot{\boldsymbol{U}}\right], \tag{D.5}$$

where ${}^{t+\Delta t/2}\boldsymbol{U}^{(i-1)}$ is the displacement obtained from the previous NR iteration. However, in the second sub-step, a Euler backward rule is applied, and the following formulas are used:

$$\boldsymbol{K_{dyn}} = \frac{9}{\Delta t^2}\boldsymbol{M} + \frac{3}{\Delta t}\boldsymbol{C} \tag{D.6}$$

$$\boldsymbol{R_{iner}} = -\boldsymbol{M}\left[\frac{9}{\Delta t^2}{}^{t+\Delta t}\boldsymbol{U}^{(i-1)} - \frac{12}{\Delta t^2}{}^{t+\Delta t/2}\boldsymbol{U} + \frac{3}{\Delta t^2}{}^{t}\boldsymbol{U} - \frac{4}{\Delta t}{}^{t+\Delta t/2}\dot{\boldsymbol{U}} + \frac{1}{\Delta t}{}^{t}\dot{\boldsymbol{U}}\right] \tag{D.7}$$

$$\boldsymbol{R_{damp}} = -\boldsymbol{C}\left[\frac{3}{\Delta t}{}^{t+\Delta t}\boldsymbol{U}^{(i-1)} - \frac{4}{\Delta t}{}^{t+\Delta t/2}\boldsymbol{U} + \frac{1}{\Delta t}{}^{t}\boldsymbol{U}\right]. \tag{D.8}$$

REFERENCES

1. Mollon G, Dias D, Soubra A-H. Probabilistic analysis of tunnelling-induced ground movements. *Acta Geotechnica* 2013; **8**(2):181–199.
2. Mollon G, Dias D, Soubra A-H. Continuous velocity fields for collapse and blow-out of a pressurized tunnel face in purely cohesive soil. *International Journal for Numerical and Analytical Methods in Geomechanics* 2013; **37**(13):2061–2083.
3. Cundall PA, Strack ODL. A discrete numerical model for granular assemblies. *Géotechnique* 1979; **29**(1):47–65.
4. Jean M. The non-smooth contact dynamics method. *Computer Methods in Applied Mechanics and Engineering* 1999; **177**(3-4):235–257.
5. Mollon G, Richefeu V, Villard P, Daudon D. Discrete modelling of rock avalanches: sensitivity to block and slope geometries. *Granular Matter* 2015; **17**(5):645–666.
6. Mollon G, Zhao J. Characterization of fluctuations in granular hopper flow. *Granular Matter* 2013; **15**(6):827–840.
7. Mollon G. A numerical framework for discrete modelling of friction and wear using Voronoi polyhedrons. *Tribology International* 2015; **90**:343–355.
8. Azema E, Radjai F, Saussine G. Quasistatic rheology, force transmission and fabric properties of a packing of irregular polyhedral particles. *Mechanics of Materials* 2009; **41**:729–741.
9. Jiang MJ, Yu H-S, Harris D. A novel discrete model for granular material incorporating rolling resistance. *Computers and Geotechnics* 2005; **32**(5):340–357.
10. Salot C, Gotteland P, Villard P. Influence of relative density on granular materials behavior: DEM simulation of triaxial tests. *Granular Matter* 2009; **11**:221–236.
11. Li H, McDowell G, Lowndes I. Discrete element modelling of a rock cone crusher. *Powder Technology* 2014; **263**:151–158.
12. Godet M. The third-body approach: a mechanical view of wear. *Wear* 1984; **100**:437–452.
13. Mirea DA, Trunfio-Sfarghiu A-M, Matei CI, Munteanu B, Piednoir A, Rieu JP, Blanchin MG, Berthier Y. Role of the biomolecular interactions in the structure and tribological properties of synovial fluid. *Tribology International* 2013; **59**:302–311.
14. Gethin DT, Lewis RW, Ransing RS. A discrete deformable element approach for the compaction of powder systems. *Modelling and Simulation in Materials Science and Engineering* 2003; **11**:101–114.
15. Procopio AT, Zavaliangos A. Simulation of multi-axial compaction of granular media from loose to high relative densities. *Journal of Mechanics and Physics of Solids* 2005; **53**:1523–1551.
16. Zhang J. A study of composite particles by multi-particle finite element method. *Composite Science and Technology* 2009; **69**:2048–2053.
17. Harthong B, Jerier J-F, Richefeu V, Chareyre B, Doremus P, Imbault D, Donzé F-V. Contact impingement in packings of elastic-plastic spheres, application to powder compaction. *International Journal of Mechanical Science* 2012; **61**:32–43.
18. Gustafsson G, Haggblad H-A, Jonsen P. Multi-particle finite element modelling of the compression of iron pellets with statistically distributed geometric and material data. *Powder Technology* 2013; **239**:231–238.

19. Belytschko T, Lu YY, Gu L. Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering* 1994; **37**:229–256.
20. Ibáñez DI, Garcia Orden JC. Galerkin meshfree methods applied to the nonlinear dynamics of flexible multibody systems. *Multibody System Dynamics* 2011; **25**(2):203–224.
21. Nayroles B, Touzot G, Villon P. Generalizing the finite element method: diffuse approximation and diffuse elements. *Computational Mechanics* 1992; **10**:307–318.
22. Zhu T, Atluri SN. A modified collocation and penalty formulation for enforcing the essential boundary conditions in the element-free Galerkin method. *Computational Mechanics* 1998; **21**:211–222.
23. Krongauz Y, Belytschko T. Enforcement of essential boundary conditions in meshless approximations using finite element. *Computer Methods in Applied Mechanics and Engineering* 1996; **131**:133–145.
24. Arroyo M, Ortiz M. Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. *International Journal for Numerical Methods in Engineering* 2006; **65**:2167–2202.
25. Sukumar N, Wright W. Overview and construction of meshfree basis functions: from moving least squares to entropy approximants. *International Journal for Numerical Methods in Engineering* 2007; **70**:181–205.
26. Liu GR, Zhang GY, Gu YT, Wang YY. A meshfree radial point interpolation method (RPIM) for three-dimensional solids. *Computational Mechanics* 2005; **36**:421–430.
27. Bathe K-J, Ramm E, Wilson EL. Finite element formulations for large deformation dynamic analysis. *International Journal for Numerical Methods in Engineering* 1975; **9**:353–386.
28. Zavarise G, De Lorenzis L. The node-to-segment algorithm for 2D frictionless contact: classical formulation and special cases. *Computer Methods in Applied Mechanics and Engineering* 2009; **198**:3428–3451.
29. Nguyen D-T, Rauchs G, Ponthot J-P. The impact of surface higher order differentiability in two-dimensional contact elements. *Journal of Computational and Applied Mathematics* 2013; **246**:195–205.
30. Nguyen D-T. Modélisation 2D par éléments finis du contact : effet de l'utilisation de méthodes de représentation des surfaces présentant un ordre de continuité élevé et méthodes permettant de passer le patch-test. *PhD Thesis*, Université de Poitiers-ENSMA, France, in French, 2013.
31. Cohen J, Lin MC, Manocha D, Ponamgi MK. I-COLLIDE: an interactive and exact collision detection system for large scale environments. In *ACM Interactive 3D Graphics Symposium*, Zyda M (ed.). Monterey: USA, 1995; 189.
32. Mirtich B. Impulse-based dynamic simulation of rigid body systems. *PhD Dissertation*, UC Berkeley, USA, 1996.
33. Simo JC, Laursen TA. An augmented Lagrangian treatment of contact problems involving friction. *Computers and Structures* 1992; **42**(1):97–116.
34. Pietrzak G, Curnier A. Large deformation frictional contact mechanics: continuum formulation and augmented Lagrangian treatment. *Computer Methods in Applied Mechanics and Engineering* 1999; **177**:351–381.
35. Bathe KJ. Conserving energy and momentum in nonlinear dynamics: a simple implicit time integration scheme. *Computers and Structures* 2007; **85**:437–445.
36. Hauret P, Le Tallec P. Energy-controlling time integration methods for nonlinear elastodynamics and low-velocity impact. *Computer Methods in Applied Mechanics and Engineering* 2006; **195**(37-40):4890–4916.
37. Bank RE, Coughran WM, Fichter W, Grosse EH, Rose DJ, Smith RK. Transient simulations of silicon devices and circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*; **4**:436–451.
38. *MATLAB and Statistics Toolbox Release*. The Mathworks, Inc.: Natick, Massachusetts, United States, 2012b.
39. Mollon G, Zhao J. Fourier–Voronoi-based generation of realistic samples for discrete modelling of granular materials. *Granular Matter* 2012; **14**:621–638.
40. Persson P-O, Strand G. A simple mesh generator in MATLAB. *SIAM Review* 2004; **46**(2):329–345.
41. Mohamed A-NA, Shabana AA. A nonlinear visco-elastic constitutive model for large rotation finite element formulation. *Multibody System Dynamics* 2011; **26**:57–79.
42. Mollon G, Zhao J. Generating realistic 3D sand particles using Fourier descriptors. *Granular Matter* 2013; **15**(1): 95–108.
43. Mollon G, Zhao J. 3D generation of realistic granular samples based on random fields theory and Fourier shape descriptors. *Computer Methods for Applied Mechanics and Engineering* 2014; **279**:46–65.
44. Ando E, Viggiani G, Hall S, Desrues J. Experimental micro-mechanics of granular media studied by X-ray tomography: recent results and challenges. *Géotechnique Letters* 2013; **3**(3):142–146.