# Development of an implicit contact technique for the material point method

González Acosta, José León; Vardon, Philip J.; Hicks, Michael A.

**DOI**
10.1016/j.compgeo.2020.103859

**Publication date**
2021

**Document Version**
Final published version

**Published in**
Computers and Geotechnics

Computers and Geotechnics 130 (2021) 103859

Contents lists available at ScienceDirect

Computers and Geotechnics

journal homepage: www.elsevier.com/locate/compgeo

# Development of an implicit contact technique for the material point method

José León González Acosta, Philip J. Vardon *, Michael A. Hicks

*Section of Geo-Engineering, Faculty of Civil Engineering and Geosciences, Delft University of Technology, the Netherlands*

ARTICLE INFO

*Keywords:*
Contact simulation
Frictional contact
Implicit scheme
Material point method
Soil-structure interaction

ABSTRACT

An implicit contact algorithm for the material point method (MPM) has been developed to simulate contact. This allows recently developed implicit MPM codes to simulate large-scale deformations and interaction with external bodies. The performance of the method has been investigated and compared to an existing explicit method using benchmark and geotechnical examples. In particular, the proposed formulation has been shown to conserve energy in a similar way to the explicit formulation and reach similar results. The method typically converges to the analytical solution when an adequate time step and mesh size are used, with the time step generally being around ten times larger than the explicit method, although during the contact phase this does not always result in faster computation due to the iterative solution procedure.

## 1. Introduction

The material point method (MPM) is an improvement over the classical finite element method (FEM) for problems involving large deformations. In this technique, the material is no longer attached to elements, but to particles or points (i.e. material points) that move freely through a background mesh. MPM has proven to be reliable for simulating mechanical behaviour (Abe et al. 2014, Beuth et al. 2008, Coombs et al. 2018, González Acosta et al. 2020, Jassim et al. 2013, Ma et al. 2006, Ma et al. 2013, Nairn 2003, Wang et al. 2016b, York et al. 1999), as well as large deformations (Coetzee 2014, Huang et al. 2015, Llano-Serna et al. 2016, Wang et al. 2016c, Więckowski 2004, Woo & Salgado 2018). As the fundamental attributes of the method involve material moving substantial distances, for example, as in landslides with an extensive runout (e.g. Wang et al. 2016a,c), the impact of this displaced material on other structures is a logical aspect to be investigated and has great value in many branches of engineering. In a seminal MPM paper, Sulsky et al. (1994) showed that different bodies of material could be simulated together on the same background grid, with a single value velocity field, allowing contact to be automatically calculated. However, the simulated contact was non-slip, (background) mesh dependent and, due to the lack of a specific contact formulation, inaccurate when considering bodies with different properties or initial state conditions.

The first dedicated methodology in MPM to simulate interaction between different bodies, i.e. contact, was proposed by Bardenhagen et al. (2000). In this method, the nodal velocities of distinct bodies are compared with the combined nodal velocities (i.e. nodal velocities which have changed due to the influence of multiple bodies), and contact is defined when these two velocities are not equal. When this happens, constraints on the normal and tangential kinematics are applied, such that the bodies cannot penetrate each other and that the tangential behaviour is defined based on constitutive behaviour, e.g. Coulomb friction. In this way, Bardenhagen et al. (2000) were able to simulate frictional slip behaviour within the MPM framework. Since then, many case studies and improvements have been presented (Bardenhagen et al. 2001, González Acosta et al. 2018, Homel & Herbold 2017, Müller & Vargas 2019, Nairn 2013, Pantev 2016, Zhang et al. 2006). Most of this work has been implemented using an explicit scheme, in line with the method of solving the equations of motion in the respective publications, and this has advantages with respect to the simplicity of the implementation. However, versions of MPM which solve the equations of motion implicitly (e.g. Guilkey & Weiss 2003, Wang et al. 2016b) have been recently developed, which include the advantages typically associated with implicit solutions; in particular, a larger time step. Therefore, the development of an implicit contact scheme is desirable.

Some work regarding implicit contact in MPM has been reported, where, in line with contact methods used in implicit FEM, Lagrange Multipliers (Chen et al. 2017) or the Penalty Method (Liu & Sun 2020) were used. The procedures followed in these techniques to simulate contact are different from the traditional method used in the explicit MPM. In this paper, an alternative form of the implicit contact algorithm is presented. This algorithm adopts the classical equations used in the explicit approach (Bardenhagen et al. 2000) for the implicit scheme and considers the Newton-Raphson iterative procedure. Moreover, to demonstrate the accuracy and performance of the proposed solution,

* Corresponding author at: Section of Geo-Engineering, Delft University of Technology, PO Box 5048, 2600 GA Delft, The Netherlands.
*E-mail address:* P.J.Vardon@tudelft.nl (P.J. Vardon).

https://doi.org/10.1016/j.compgeo.2020.103859
Received 3 June 2020; Received in revised form 11 September 2020; Accepted 30 September 2020
Available online 13 December 2020

three benchmarks and two geotechnical problems are analysed. Finally, comparative remarks are presented relating to the computational performance, followed by some recommendations for increasing the accuracy of the simulations.

## 2. MPM Formulation

The formulation of the explicit and implicit MPM equations is briefly presented and, since MPM shares similarities with the FEM continuum mechanics framework, most of the following equations are the same as in FEM. It should be noted that this formulation refers to a single body. For a detailed elaboration of these equations, the reader is directed to Bathe (1996), Belytschko et al. (2013), Sulsky et al. (1995), and Wang et al. (2016b).

### 2.1. Explicit MPM

The respective equations of mass and momentum conservation are

$$\frac{d\rho}{dt} + \rho\nabla\cdot\mathbf{v} = 0 \tag{1}$$

$$\rho\mathbf{a} = \nabla\cdot\boldsymbol{\sigma} + \rho\mathbf{b} \tag{2}$$

where $\rho$ is the mass density, $\mathbf{v}$ is the velocity vector, $t$ is time, $\mathbf{a}$ is the acceleration vector, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, and $\mathbf{b}$ is the body force. Besides these conservation laws, a stress–strain relationship is also required. The strain increment computed from the velocities, and the resulting stress increment, are respectively given by

$$d\boldsymbol{\varepsilon} = \frac{1}{2}\left(\nabla\mathbf{v} + (\nabla\mathbf{v})^{\mathrm{T}}\right)dt \tag{3}$$

$$d\boldsymbol{\sigma} = \mathbf{D}d\boldsymbol{\varepsilon} \tag{4}$$

where $d\boldsymbol{\varepsilon}$ is the strain increment vector, and $\mathbf{D}$ is the material elastic matrix. Using the principle of virtual power, the weak form of eq. (2), considering traction, is found by taking the product of a test function with the equation of momentum conservation and integrating over the current configuration to give

$$\int_V \rho\mathbf{a}\,\delta\mathbf{v}\,dV = \int_V \boldsymbol{\sigma}\nabla\delta\mathbf{v}\,dV + \int_V \rho\mathbf{b}\,\delta\mathbf{v}\,dV + \int_\Gamma \mathbf{s}^s\delta\mathbf{v}\,d\Gamma \tag{5}$$

where $\delta\mathbf{v}$ is the test function, $\mathbf{s}^s$ is the traction load at the body surface $\Gamma$ (i.e. the boundary condition), and $V$ is the current body volume. Then, since in MPM two discretisations are used (i.e. material points representing the computational body and elements representing the computational space), the integrals in eq. (5) can be expressed as nodal equations via the summation of material point values in the elements. This leads to the (element) equation

$$\delta\overline{\mathbf{v}}\sum_{p=1}^{nmp}\rho_p\mathbf{N}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{N}(\mathbf{x}_p)\overline{\mathbf{a}}|\mathbf{J}|W_p = \delta\overline{\mathbf{v}}\left[-\sum_{p=1}^{nmp}\mathbf{B}^{\mathrm{T}}(\mathbf{x}_p)\boldsymbol{\sigma}_p|\mathbf{J}|W_p + \sum_{p=1}^{nmp}\rho_p\mathbf{N}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{b}|\mathbf{J}|W_p + \sum_{p=1}^{bmp}\mathbf{N}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{s}^s\right] \tag{6}$$

where $\mathbf{N}$ is the element matrix of shape functions (SFs) (see Appendix A for the definition of the matrices used in the paper), $\mathbf{B}$ is the strain–displacement matrix, $\overline{\mathbf{a}}$ is the vector of nodal accelerations, $nmp$ is the number of material points in the element, $bmp$ is the number of boundary material points having a traction load, $\mathbf{x}_p$ is the position of each material point, $\boldsymbol{\sigma}_p$ is the material point stress tensor, $\rho_p$ is the material point density, $\mathbf{J}$ is the Jacobian matrix, and $W_p$ is the material point integration weight. It should be noted that the shape functions are used to interpolate material point variables to the element nodes (i.e. $\mathbf{N}\delta\overline{\mathbf{v}}(\mathbf{x}_p) \approx \delta\mathbf{v}(\mathbf{x}_p)$, where $\delta\overline{\mathbf{v}}$ is the vector of nodal test function velocities), and that the volume of each material point is represented by the Jacobian determinant and the material point weight (i.e. $V_p = |\mathbf{J}|W_p$, where $V_p$ is the material point volume). Finally, since the element test function ($\delta\overline{\mathbf{v}}$) in eq. (6) should hold for any arbitrary virtual velocity that satisfies all the boundary conditions, it vanishes from eq. (6) since $\delta\overline{\mathbf{v}}\mathbf{A} = \delta\overline{\mathbf{v}}\mathbf{C}$, in which $\mathbf{A}$ and $\mathbf{C}$ represent the left and right sides of eq. (6), respectively. Hence, eq. (6) reduces further and may be expressed for an element ($|_{elem}$) using matrix notation as

$$\mathbf{m}|_{elem}\overline{\mathbf{a}}|_{elem} = \mathbf{F}^{ext}|_{elem} - \mathbf{F}^{int}|_{elem} \tag{7}$$

where

$$\mathbf{m}|_{elem} = \sum_{p=1}^{nmp}\rho_p\mathbf{N}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{N}(\mathbf{x}_p)|\mathbf{J}|W_p \tag{8}$$

$$\mathbf{F}^{ext}|_{elem} = \sum_{p=1}^{nmp}\rho_p\mathbf{N}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{b}|\mathbf{J}|W_p + \sum_{p=1}^{bmp}\mathbf{N}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{s}^s \tag{9}$$

$$\mathbf{F}^{int}|_{elem} = \sum_{p=1}^{nmp}\mathbf{B}^{\mathrm{T}}(\mathbf{x}_p)\boldsymbol{\sigma}_p|\mathbf{J}|W_p \tag{10}$$

In eqs. (7) and (8), $\mathbf{m}$ represents the element consistent mass matrix, the use of which can lead to an increase in the solution time. Hence, the element lumped mass matrix is used here, and is computed using the "row-summation" technique as

$$\mathbf{m}_{ii} = \sum_j \mathbf{m}_{ij} \tag{11}$$

Furthermore, the lumped mass matrix can be replaced by an element mass vector ($\overline{\mathbf{m}}_i = \mathbf{m}_{ii}$). This replacement improves the computational time, since matrix operations (i.e. matrix inverse computations) are replaced by linear operations. Finally, note that the use of $\overline{\mathbf{m}}$ is particular to the explicit scheme; in the implicit scheme $\mathbf{m}$ will be used.

#### 2.1.1. Nodal mapping, solution and update phase (explicit)

At the beginning of each solution step, the state variables are mapped to the nodes. For example, the velocity at node $i$ is computed as

$$\mathbf{v}_i^t = \frac{\sum_{p=1}^{nmp}\rho_p\mathbf{v}_p^t N_i(\mathbf{x}_p)\,|\mathbf{J}|W_p}{\sum_{p=1}^{nmp}\rho_p N_i(\mathbf{x}_p)|\mathbf{J}|W_p} \tag{12}$$

where $N_{\mathrm{i}}$ is the nodal SF, and the summation is over all material points surrounding the node. After nodal mapping and integration to form the element equations, an assembly operation over the mesh is undertaken (as in FEM), yielding global vectors and matrices at time $t$, i.e. the global mass matrix $\mathbf{m}^t$, the global external force $\mathbf{F}^{ext,t}$ and the global internal force $\mathbf{F}^{int,t}$, which can be solved to find the global acceleration vector. Since the vector of nodal masses is used, the solution of the equation of equilibrium (in terms of accelerations) is

$$\overline{\mathbf{a}}^t = \frac{\left(\mathbf{F}^{ext,\,t} - \mathbf{F}^{int,\,t}\right)}{\overline{\mathbf{m}}^t} \tag{13}$$

and, using the Euler time integration method, the material point variables are updated at time $t + \Delta t$ as

$$\mathbf{v}_p^{t+\Delta t} = \mathbf{v}_p^t + \mathbf{N}(\mathbf{x}_p)\overline{\mathbf{a}}^t|_{elem}\,\Delta t \tag{14}$$

$$\mathbf{x}_p^{t+\Delta t} = \mathbf{x}_p^t + \mathbf{v}_p^{t+\Delta t}\,\Delta t \tag{15}$$

$$\boldsymbol{\sigma}_p = \mathbf{D}\,\mathbf{B}(\mathbf{x}_p)\;\overline{\mathbf{v}}^{t+\Delta t}|_{elem}\Delta t + \boldsymbol{\sigma}_p^t \tag{16}$$

where $\boldsymbol{\sigma}_p^t$ is the previous material point stress. The vector of (element) nodal velocities at time $t + \Delta t$ in eq. (16) is computed from the material point velocities at time $t + \Delta t$ as

$$\overline{\mathbf{v}}^{t+\Delta t}\big|_{elem} = \frac{\sum_{p=1}^{nmp} m_p \mathbf{v}_p^{t+\Delta t}\mathbf{N}(\mathbf{x}_p)}{\sum_{p=1}^{nmp} m_p \mathbf{N}(\mathbf{x}_p)} \tag{17}$$

where $m_p$ is the material point mass. Note that this vector of nodal velocities is computed from the material point velocities at time $t + \Delta t$ rather than using the Euler time integration method (i.e. $\overline{\mathbf{v}}^{t+\Delta t} = \overline{\mathbf{v}}^t + \overline{\mathbf{a}}^t\Delta t$) to ensure conservation of momentum.

## 2.2. *Implicit MPM*

To derive the equation for implicit equilibrium, the principle of virtual work is used. This states that the difference between the internal and external work should be equal to zero $\left(\text{i.e. } \Pi = W^{int} - W^{ext} = 0\right)$, and is written as

$$\Pi = \frac{1}{2}\int_V \mathbf{D}\boldsymbol{\varepsilon}^2\, dV - \left(\int_V \mathbf{u}^{\mathrm{T}}\rho(\mathbf{b}-\mathbf{a})\, dV + \int_\Gamma \mathbf{u}^{\mathrm{T}}\,\mathbf{s}^s\, d\Gamma\right) = 0 \tag{18}$$

where $\mathbf{u}$ represents the continuous displacement field. In eq. (18), the inertia term (i.e. $\mathbf{a}$) is included as part of the body forces using D'Alambert's principle and, if this term is excluded, a quasi-static scheme is obtained. Next, by evaluating $\delta\Pi = 0$ with respect to the displacements and knowing that $\delta\varepsilon^2 = 2\varepsilon\delta\varepsilon$ (Davies 2011; Bathe 1996), eq. (18) becomes

$$\int_V \delta\boldsymbol{\varepsilon}^{\mathrm{T}}\mathbf{D}\boldsymbol{\varepsilon}\, dV = \int_V \delta\mathbf{u}^{\mathrm{T}}\rho(\mathbf{b}-\mathbf{a})\, dV + \int_\Gamma \delta\mathbf{u}^{\mathrm{T}}\mathbf{s}^s d\Gamma \tag{19}$$

where $\delta\mathbf{u}$ and $\delta\boldsymbol{\varepsilon}$ represent the test function in terms of displacements and strains. Then, following similar procedures as in eq. (6) and splitting the total stresses into the sum of the previous stress and the incremental stress (i.e. $\boldsymbol{\sigma} = \boldsymbol{\sigma}^0 + \Delta\boldsymbol{\sigma}$), eq. (19) is expressed as

$$\begin{aligned}\left[\sum_{p=1}^{nmp}\mathbf{B}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{D}\mathbf{B}(\mathbf{x}_p)|\mathbf{J}|W_p\right]\overline{\mathbf{u}} - \sum_{p=1}^{nmp}\rho_p\mathbf{N}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{N}(\mathbf{x}_p)|\mathbf{J}|W_p\overline{\mathbf{a}} = \\ -\sum_{p=1}^{nmp}\mathbf{B}^{\mathrm{T}}(\mathbf{x}_p)\boldsymbol{\sigma}_p^0|\mathbf{J}|W_p + \sum_{p=1}^{nmp}\rho_p\mathbf{N}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{b}|\mathbf{J}|W_p + \sum_{p=1}^{bmp}\mathbf{N}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{s}^s\end{aligned} \tag{20}$$

where $\mathbf{B}$ is used to interpolate the virtual strains using virtual nodal displacements, i.e. $\mathbf{B}(\mathbf{x}_p)\delta\overline{\mathbf{u}} \approx \delta\boldsymbol{\varepsilon}(\mathbf{x}_p)$, where $\delta\overline{\mathbf{u}}$ are the nodal virtual displacements, $\boldsymbol{\sigma}^0$ represents the vector of initial stresses, and $\overline{\mathbf{u}}$ represents the vector of (element) nodal displacements which is obtained from the interpolation of strains, i.e. $\mathbf{B}\overline{\mathbf{u}} = \boldsymbol{\varepsilon}$. As for eq. (7), the previous equations represent equilibrium and can be expressed in matrix form as

$$(\mathbf{K}\Delta\overline{\mathbf{u}})\,|_{elem} + (\mathbf{m}\overline{\mathbf{a}})\,|_{elem} = \left(\mathbf{F}^{ext} - \mathbf{F}^{int}\right)|_{elem} \tag{21}$$

where the element stiffness matrix is

$$\mathbf{K}|_{elem} = \sum_{p=1}^{nmp}\mathbf{B}^{\mathrm{T}}(\mathbf{x}_p)\mathbf{D}\,\mathbf{B}(\mathbf{x}_p)|\mathbf{J}|W_p \tag{22}$$

and $\mathbf{F}^{ext}|_{elem}$ and $\mathbf{F}^{int}|_{elem}$ are similar to eq. (9) and eq. (10), respectively. If the inertia term is ignored, the equation of equilibrium reduces to

$$(\mathbf{K}\Delta\overline{\mathbf{u}})\,|_{elem} = \left(\mathbf{F}^{ext} - \mathbf{F}^{int}\right)|_{elem} \tag{23}$$

which represents quasi-static equilibrium. Again, an assembly operation over the mesh is undertaken (as in FEM), yielding global vectors and matrices.

By using Newmark's (1959) time integration technique, which is recognised to be highly effective (Bathe 2007), eq. (21) can be solved. The velocities and displacements at time $t + \Delta t$ are computed as

$$\overline{\mathbf{v}}^{t+\Delta t} = \overline{\mathbf{v}}^t + \left[(1-\gamma)\overline{\mathbf{a}}^t + \gamma\overline{\mathbf{a}}^{t+\Delta t}\right]\Delta t \tag{24}$$

$$\overline{\mathbf{u}}^{t+\Delta t} = \overline{\mathbf{u}}^t + \overline{\mathbf{v}}^t\Delta t + \left[\left(\frac{1}{2}-\alpha\right)\overline{\mathbf{a}}^t + \alpha\overline{\mathbf{a}}^{t+\Delta t}\right]\Delta t^2 \tag{25}$$

where $\alpha$ and $\gamma$ are time stepping parameters that are here chosen to be $\alpha = 0.25$ and $\gamma = 0.5$ (giving a constant-average-acceleration approach). Then, isolating $\overline{\mathbf{a}}^{t+\Delta t}$ from eq. (25) leads to

$$\overline{\mathbf{a}}^{t+\Delta t} = \left(\frac{4\overline{\mathbf{u}}^t}{\Delta t^2} - \frac{4\overline{\mathbf{v}}^t}{\Delta t} - \overline{\mathbf{a}}^t\right) \tag{26}$$

and substituting eq. (26) into eq. (24) results in

$$\overline{\mathbf{v}}^{t+\Delta t} = \frac{2\overline{\mathbf{u}}^{t+\Delta t}}{\Delta t} - \overline{\mathbf{v}}^t \tag{27}$$

Finally, by substituting eq. (26) into eq. (21) and adding the Newton-Raphson iteration procedure, the equation of equilibrium is written as

$$\left(\mathbf{K}^t + \frac{4\mathbf{m}^t}{\Delta t^2}\right)\ {}^{(k)}\Delta\overline{\mathbf{u}} = \ {}^{(k-1)}\left[\mathbf{F}^{ext,\,t+\Delta t} - \mathbf{m}^t\left(\frac{4^{(k-1)}\overline{\mathbf{u}}^t}{\Delta t^2} - \frac{4\overline{\mathbf{v}}^t}{\Delta t} - \overline{\mathbf{a}}^t\right) - \mathbf{F}^{int,\,t+\Delta t}\right] \tag{28}$$

or

$$\overline{\mathbf{K}}^t\ {}^{(k)}\Delta\overline{\mathbf{u}} = \ {}^{(k-1)}\left(\mathbf{F}^{ext} - \mathbf{F}^{kin} - \mathbf{F}^{int}\right)^{t+\Delta t} \tag{29}$$

where $\Delta\overline{\mathbf{u}} = \overline{\mathbf{u}}^{t+\Delta t} – \overline{\mathbf{u}}^t$ is the vector of incremental displacements, and

$$\overline{\mathbf{K}}^t = \left(\mathbf{K}^t + \frac{4\mathbf{m}^t}{\Delta t^2}\right) \tag{30}$$

$$\mathbf{F}^{kin,\,t+\Delta t} = \mathbf{m}^t\left(\frac{4^{(k-1)}\overline{\mathbf{u}}^t}{\Delta t^2} - \frac{4\overline{\mathbf{v}}^t}{\Delta t} - \overline{\mathbf{a}}^t\right) \tag{31}$$

In these equations, $\overline{\mathbf{K}}$ represents a modified stiffness matrix, $\mathbf{F}^{kin}$ are the kinetic forces, and the left superscript $k$ refers to the Newton-Raphson iteration step. The iteration is stopped after the desired convergence criterion has been reached, which in this paper is computed as (Guilkey & Weiss 2003)

$$\frac{\|{}^{(k)}\Delta\mathbf{u}\|}{\|\mathbf{u}\|} < tol \tag{32}$$

where *tol* is the tolerance value. There are several alternative convergence criteria which could be used. However, this convergence criterion has been selected due to its simplicity, although it is noted to be susceptible to false convergence due to iteration stagnation.

### 2.2.1. *Nodal integration and update phase (implicit)*

The integration of nodal external loads, mass, velocity and stiffness has already been given in eqs. (9), (11), (12), and (22), respectively. In contrast to the explicit scheme, nodal accelerations are also needed and are computed as

$$\mathbf{a}_i^t = \frac{\sum_{p=1}^{nmp}\rho_p\mathbf{a}_p^t N_i(\mathbf{x}_p)\ |\mathbf{J}|W_p}{\sum_{p=1}^{nmp}\rho_p N_i(\mathbf{x}_p)\ |\mathbf{J}|W_p} \tag{33}$$

After assembling the stiffness and mass matrices (eq. (30)), the equilibrium in terms of displacements is computed (eq. (29)) and the vectors of internal forces, nodal accelerations and kinetic forces are updated every iteration step using eqs. (10), (26), and (31), respectively. Finally, after the desired iteration tolerance is reached, a material point's acceleration, velocity, position and stress are updated as

$$\mathbf{a}_p^{t+\Delta t} = \mathbf{N}(\mathbf{x}_p)\overline{\mathbf{a}}^{t+\Delta t}|_{elem} \tag{34}$$

$$\mathbf{v}_p^{t+\Delta t} = \mathbf{v}_p^t + \frac{\left[\mathbf{N}(\mathbf{x}_p)\overline{\mathbf{a}}^{t+\Delta t}|_{elem} + \mathbf{a}_p^t\right]}{2}\Delta t \tag{35}$$

$$\mathbf{x}_p^{t+\Delta t} = \mathbf{x}_p^t + \mathbf{N}(\mathbf{x}_p)\overline{\mathbf{u}}|_{elem} \tag{36}$$

$$\boldsymbol{\sigma}_p^{t+\Delta t} = \mathbf{D}\,\mathbf{B}(\mathbf{x}_p)\ \overline{\mathbf{u}}|_{elem} + \boldsymbol{\sigma}_p^t \tag{37}$$

### 2.3. MPM mapping, integration and stress recovery enhancement

Since standard MPM suffers from oscillation problems, the procedures elaborated in González Acosta et al. (2020) are used to mitigate such inaccuracies. In this paper, the double mapping DM-GC technique is used, in which the (element) vectors of external and internal nodal forces are integrated using standard GIMP SFs (Bardenhagen & Kober 2004) as

$$\mathbf{F}^{ext}\big|_{elem} = \sum_{p=1}^{nmp} \rho_p \mathbf{S}_{ip}(\mathbf{x}_p)\mathbf{b}|\mathbf{J}|W_p + \sum_{p=1}^{bmp} \mathbf{S}_{ip}(\mathbf{x}_p)\mathbf{s}^s \tag{38}$$

$$\mathbf{F}^{int}\big|_{elem} = \sum_{p=1}^{nmp} \nabla\mathbf{S}_{ip}^{\mathrm{T}}(\mathbf{x}_p)\boldsymbol{\sigma}_p|\mathbf{J}|W_p \tag{39}$$

where $\mathbf{S}_{ip}$ is the element matrix of GIMP SFs, and $\nabla\mathbf{S}_{ip}$ is the strain–displacement matrix for the GIMP SFs. Since the implicit solution scheme is used, the (element) stiffness matrix is computed using double mapping (DM) procedures as

$$\mathbf{K}\big|_{elem} = \sum_{g=1}^{ngauss} \mathbf{B}^{\mathrm{T}}(\mathbf{x}_p)\left[\sum_{i=1}^{nn}\left(N_i(\mathbf{x}_g)\sum_{p=1}^{smp} S_{ip^*}(\mathbf{x}_p)\mathbf{D}W_p\right)\right]\mathbf{B}(\mathbf{x}_p)|\mathbf{J}|W^{FE} \tag{40}$$

where $N_i$ is the nodal SF, $S_{ip^*}$ is the local GIMP SF of node *i*, *ngauss* is the number of Gauss integration points in the element, *smp* is the number of material points with a support domain inside the element, and $W^{FE}$ is the finite element (FE) weight associated with the Gauss point *g*. In González Acosta et al. (2020), the DM implementation and its performance are presented. The incremental stress of a material point is then updated for the explicit and implicit schemes using CMPM (González Acosta et al. 2017) as

$$\Delta\boldsymbol{\sigma}_p = \mathbf{D}\mathbf{B}_C(\mathbf{x}_p)\overline{\mathbf{v}}^{ext}\,\Delta t \tag{41}$$

and

$$\Delta\boldsymbol{\sigma}_p = \mathbf{D}\mathbf{B}_C(\mathbf{x}_p)\Delta\overline{\mathbf{u}}^{ext} \tag{42}$$

where $\mathbf{B}_C$ is the (element) matrix of the CMPM SF gradients (i.e. the strain–displacement matrix for CMPM), and $\overline{\mathbf{u}}^{ext}$ and $\overline{\mathbf{v}}^{ext}$ are the vectors of nodal displacements and velocities, respectively, in the extended CMPM domain. Note that since GIMP and local GIMP SFs are used to integrate both the forces and stiffness matrix, the stencil (i.e. support domain of each material point) is the same in both cases, including when using double mapping. CMPM uses an extended domain ($\overline{\mathbf{u}}^{ext}$ or $\overline{\mathbf{v}}^{ext}$) to recover stresses, which is not necessarily equal to the GIMP stencil. An explanation of the stencil is given in Appendix B, and the accuracy of using DM-GC is demonstrated in Appendix C. Finally, it is important to remark that the state variables (i.e. mass, velocity, acceleration) are mapped using GIMP SFs.

## 3. Contact formulation

The contact formulation relies on the equations of motion presented in the previous section, which are solved for a series of separately defined bodies, with the interactions (contact) between them being included in the equations of equilibrium as additional external forces. The bodies may be pre-defined (as in the examples here), or identified by the program to allow for the separation or joining of bodies. The original contact procedures in MPM (Bardenhagen et al. 2000) use two steps to simulate contact: (i) the velocity field sharing step, and (ii) the contact force evaluation step. In this work, an extra (proximity detection) step to ensure mesh independency has been included, using a distance limit to activate the contact. In the velocity sharing step, the velocities are interpolated to the nodes to detect if any of the bodies may be interacting (which is possible if two or more bodies interpolate velocities to the same node). Using the individual nodal velocities of each body and the combined domain velocities (i.e. the nodal velocities accumulated from all material points in the domain), velocity sharing is detected at node *i* if

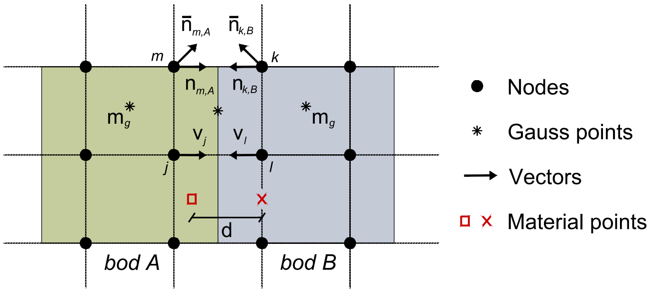

**Fig. 1.** Contact variables.

$$\mathbf{v}_{i,\,C} - \mathbf{v}_{i,bod} \neq \mathbf{0} \tag{43}$$

where

$$\mathbf{v}_{i,\,C} = \sum_{bod=1}^{nb} \frac{\overline{\mathbf{m}}_{i,\,bod}\mathbf{v}_{i,\,bod}}{\overline{\mathbf{m}}_{i,\,C}} \tag{44}$$

$$\overline{\mathbf{m}}_{i,\,C} = \sum_{bod=1}^{nb} \overline{\mathbf{m}}_{i,\,bod} \tag{45}$$

where $\mathbf{v}_{i,C}$ is the nodal combined velocity accounting for all bodies in the domain, $\mathbf{v}_{i,bod}$ is the nodal velocity of each independent body, $\overline{\mathbf{m}}_{i,C}$ is the combined nodal mass, *bod* denotes the body, and *nb* is the number of bodies in the domain. If eq. (43) is true for any node *i*, the proximity detection rule is then evaluated. This proximity condition is

$$d\left(\mathbf{x}_{p,\,bod1}^{i},\,\mathbf{x}_{p,\,bod2}^{i}\right) \leqslant d_{min} \tag{46}$$

where $\mathbf{x}_{p,\,bod1}^{i}$ and $\mathbf{x}_{p,\,bod2}^{i}$ are the coordinates of the closest material points *p* from each body (*bod1* and *bod2*) to the possible contact node *i*, *d* is the distance between $\mathbf{x}_{p,\,bod1}^{i}$ and $\mathbf{x}_{p,\,bod2}^{i}$, and $d_{min}$ is the minimum distance required to activate the contact. Note that $d_{min}$ should have a maximum value of the cell size to ensure mesh independence, but it has no minimum value. It should be noted that, in eq. (46), only the contact between two bodies (*bod1* and *bod2*) is considered, and the formulation to account for contact at a single node by more bodies is not included. If the velocity and proximity conditions are satisfied (i.e. eq. (43) and eq. (46)), it is then necessary to evaluate if the bodies are approaching each other (contact evaluation step). This last condition is relevant because, if the bodies are separating (i.e. the first two conditions are true but the bodies are moving in opposite directions), interaction is considered not to be taking place; otherwise it would add energy into the system. This step is formulated as

$$\left(\mathbf{v}_{i,\,bod} - \mathbf{v}_{i,\,C}\right)\cdot\mathbf{n}_{i,\,bod} > 0 \tag{47}$$

where $\mathbf{n}_{i,bod}$ is the unit normal vector of the body at the contact node. The normal vector can be computed using different approaches (Nairn 2013), but in this paper the approach described in Huang et al. (2011) is used, which is

$$\mathbf{n}_{i,\,1} = -\mathbf{n}_{i,\,2} = \frac{\left(\overline{\mathbf{n}}_{i,\,1} - \overline{\mathbf{n}}_{i,\,2}\right)}{\left|\overline{\mathbf{n}}_{i,\,1} - \overline{\mathbf{n}}_{i,\,2}\right|} \tag{48}$$

$$\overline{\mathbf{n}}_{i,\,bod} = \frac{\sum_{g=1}^{ngp}\nabla\mathbf{N}m_g}{\left|\sum_{g=1}^{ngp}\nabla\mathbf{N}m_g\right|} \tag{49}$$

where the normal $\overline{\mathbf{n}}$ is the individual normal of the body *bod* around node *i*, in which the direction is governed by the mass $m_g$ concentrated at the (central) Gauss position, and *ngp* is the number of (central) Gauss positions with a concentrated mass $m_g$ around node *i*. Note that the normal $\overline{\mathbf{n}}$ can be inconsistent (i.e. may not be perpendicular) to the

```
1. [  Evaluate nodal force, mass and velocity (eqs. 9 or 38, 10 or 39, 11 and 12): $\mathbf{F}^{ext,\,t}_{i,\,bod}$, $\mathbf{F}^{int,\,t}_{i,\,bod}$, $\mathbf{m}^{t}_{i,\,bod}$ and $\bar{\mathbf{v}}^{t}_{i,\,bod}$
2.    Evaluate combined velocity (eq. 44): $\mathbf{v}^{t}_{i,\,C}$
3. [  3-1 Estimate new contact velocities (eq. 50): $^{*}\mathbf{v}^{t}_{ic,\,bod}$
      3-2 Estimate contact and frictional forces (eqs. 51, 53): $\mathbf{F}^{nc,\,t}_{ic,\,bod}$ and $\mathbf{F}^{fric,\,t}_{ic,\,bod}$
4. [  4-1 Estimate nodal accelerations (eqs. 13 or 54): $\bar{\mathbf{a}}^{t}_{i,\,bod}$
      4-2 Update material point velocity at time $t + \Delta t$ (eq. 14): $\mathbf{v}^{t+\Delta t}_{p,\,bod}$
      4-3 Evaluate nodal velocity (eq. 17): $\bar{\mathbf{v}}^{t+\Delta t}_{i,\,bod}$
      4-4 Update material point trial stress (eq. 16 or 41): $\boldsymbol{\sigma}^{trial}_{p,\,bod}$
      4-5 Loop 1: $Y(\boldsymbol{\sigma}^{trial}_{p,\,bod}) > 0$: return stress to the yield surface Y
              Plastic iteration loop
              IF $Y(\boldsymbol{\sigma}^{trial}_{p,\,bod}) = 0$: EXIT Loop 1
          END Loop 1
      4-6 Update material point stress after plasticity: $\boldsymbol{\sigma}^{t+\Delta t}_{p,\,bod} = \boldsymbol{\sigma}^{trial}_{p,\,bod}$
      4-7 Update material point position (eq. 15): $\mathbf{x}^{t+\Delta t}_{p,\,bod}$
```

**Fig. 2.** Steps followed in the explicit solution scheme, in which the square brackets indicate a loop over the bodies.

surface of the body due to the irregular distribution of $m_g$ around node $i$. This inconsistency can be lessened by using $\mathbf{n}$, the normal direction at node $i$ in which the influence of the neighbouring body is considered. Furthermore, to compute $\bar{\mathbf{n}}$, bi-linear or GIMP SF gradients can be used, as the results are identical due to the centrally located Gauss point. Fig. 1 illustrates the different variables during contact between body A and body B, and highlights the difference between $\bar{\mathbf{n}}$ and $\mathbf{n}$.

If eqs. (43), (46), and (47) are true, contact is deemed to occur, and contact conditions must be applied by assigning contact forces. To do so, first the corrected nodal velocities of each body are calculated to avoid interpenetration. Hence,

$$^{*}\mathbf{v}_{ic,\,bod} = \mathbf{v}_{ic,\,bod} - \left[\left(\mathbf{v}_{ic,\,bod} - \mathbf{v}_{ic,\,C}\right)\cdot\mathbf{n}_{ic,\,bod}\right]\mathbf{n}_{ic,\,bod} \tag{50}$$

where $^{*}\mathbf{v}_{ic,\,bod}$ is the corrected nodal velocity, and $ic$ represents the contact node. Then, the normal contact force is computed as

$$\mathbf{F}^{nc}_{ic,\,bod} = \frac{\bar{\mathbf{m}}_{ic,\,bod}\left[\left({}^{*}\mathbf{v}_{ic,\,bod} - \mathbf{v}_{ic,\,bod}\right)\cdot\mathbf{n}_{ic,\,bod}\right]\mathbf{n}_{ic,\,bod}}{\Delta t} \tag{51}$$

where $\mathbf{F}^{nc}_{ic,\,bod}$ is the nodal normal contact force. The tangential force required to ensure non-slip (i.e. stick) conditions is calculated as

$$\mathbf{F}^{stick}_{ic,\,bod} = -\frac{\bar{\mathbf{m}}_{ic,\,bod}\mathbf{n}_{ic,\,bod} \times \left[\left(\mathbf{v}_{ic,\,bod} - \mathbf{v}_{ic,\,C}\right) \times \mathbf{n}_{ic,\,bod}\right]}{\Delta t} \tag{52}$$

where $\mathbf{F}^{stick}_{ic,\,bod}$ is the frictional force required to ensure non-slip conditions. The frictional force, obeying Coulomb's theory, is then

$$\mathbf{F}^{fric}_{ic,\,bod}\Big|_{elem} = \frac{\mathbf{F}^{stick}_{ic,\,bod}}{\left|\mathbf{F}^{stick}_{ic,\,bod}\right|}\min\left(\mu\left|\mathbf{F}^{nc}_{ic,\,bod}\right|, \left|\mathbf{F}^{stick}_{ic,\,bod}\right|\right) \tag{53}$$

where $\mathbf{F}^{fric}_{ic,bod}$ is the final frictional force between the bodies, μ is the friction coefficient which depends on the material characteristics, the quotient ensures that the direction of the force opposes the relative motion, and the min function determines the magnitude of the force. Finally, if contact occurs, nodal accelerations are no longer computed using eq. (13), since the contact loads should be included as

$$\bar{\mathbf{a}}^{t+\Delta t}_{bod} = \frac{\left(\mathbf{F}^{ext,\,t}_{i,\,bod} - \mathbf{F}^{int,\,t}_{i,\,bod} + \mathbf{F}^{nc,\,t}_{ic,\,bod} - \mathbf{F}^{fric,\,t}_{ic,\,bod}\right)}{\bar{\mathbf{m}}^{t}_{i,\,bod}} \tag{54}$$

## 4. Explicit and implicit contact algorithms

### 4.1. Explicit contact algorithm

The implementation of contact using the explicit scheme is straightforward. Fig. 2 shows the algorithm of the explicit scheme including the contact steps. Several minor steps are omitted, such as the activation of the boundary conditions and the computation of the material point local coordinates, in order to maintain the algorithm simplicity. The square brackets located at the left side of the figure indicate a loop for each body; i.e. the combination of steps should be repeated for each body before continuing with the following steps. The plastic iteration loop is indicated, but the procedures are not elaborated; the reader is directed to Smith et al. (2013) and Sloan et al. (2001) where the plastic procedures are described in detail.

Note that the steps shown in Fig. 2 indicate that the modified update stress last (MUSL) method (Sulsky et al. 1995) is used. The standard update stress last (USL) method consists of computing the increment of stresses after the update of nodal velocities, but, since the nodal velocities are computed from the updated material point velocities, the MUSL is obtained. The MUSL algorithm has been shown to conserve energy better than the USL approach (Nairn 2003, Pantev 2016).

### 4.2. Implicit contact algorithm

To simulate contact using the implicit scheme, the same approach as in the explicit scheme is used. Newmark's time integration technique and the Newton-Raphson method are used, so the equations used to estimate contact must be updated in each iteration step. Moreover, if contact occurs, it is assumed that it persists throughout the Newton-Raphson iterative process, i.e. between times $t$ and $t + \Delta t$. Considering these assumptions, the equations used to update the nodal velocities during the iterative procedure are then

$$^{(k)}\mathbf{v}^{t+\Delta t}_{ic,\,C} = \sum_{bod=1}^{nb}\frac{\bar{\mathbf{m}}_{ic,\,bod}\mathbf{v}^{t+\Delta t}_{ic,\,bod}}{\bar{\mathbf{m}}_{ic,\,C}} \tag{55}$$

and

$$^{(k)*}\mathbf{v}^{t+\Delta t}_{ic,\,b} = \mathbf{v}^{t+\Delta t}_{ic,\,b} - \left[\left(\mathbf{v}^{t+\Delta t}_{ic,\,bod} - \mathbf{v}^{t+\Delta t}_{ic,\,C}\right)\cdot\mathbf{n}_{ic,\,bod}\right]\mathbf{n}_{ic,\,bod} \tag{56}$$

The contact loads are computed as

```
1.  1-1 Evaluate nodal mass, velocities and accelerations (eqs. 11, 12 and 33): $\mathbf{m}^{t}_{i,\,bod}$, $\overline{\mathbf{v}}^{t}_{i,\,bod}$ and $\overline{\mathbf{a}}^{t}_{i,\,bod}$
    1-2 Evaluate nodal external loads and stiffness (eqs. 9 or 38 and 30): $\mathbf{F}^{ext,\,t}_{i,\,bod}$ and $\overline{\mathbf{K}}^{t}_{bod}$
2.      Evaluate combined velocity (eq. 44): $\mathbf{v}^{t}_{i,\,C}$
3.      Initialise $\overline{\mathbf{u}}^{t+\Delta t}_{ic,\,bod} = 0$, $\mathbf{v}^{t+\Delta t}_{ic,\,bod} = \overline{\mathbf{v}}^{t}_{i}$, $\mathbf{F}^{nc,\,t+\Delta t}_{ic,\,bod} = 0$ and $\mathbf{F}^{fric,\,t+\Delta t}_{ic,\,bod} = 0$
        Loop 1: over the number of iteration steps
4.          4-1 Evaluate nodal internal forces (eq. 10 or 39): ${}^{(k)}\mathbf{F}^{int,\,t+\Delta t}_{i,\,bod}$
            4-2 Evaluate kinetic forces (eq. 31): ${}^{(k)}\mathbf{F}^{kin,\,t+\Delta t}_{i,\,bod}$
            4-3 Estimate contact forces (eqs. 51 and 53): ${}^{(k)}\mathbf{F}^{nc,\,t+\Delta t}_{ic,\,bod}$ and ${}^{(k)}\mathbf{F}^{fric,\,t+\Delta t}_{ic,\,bod}$
5.          5-1 Estimate incremental displacements (eq. 29 or 60): ${}^{(k)}\overline{\mathbf{u}}_{i,\,bod}$
            5-2 Update nodal velocities (eq. 27): ${}^{(k)}\overline{\mathbf{v}}^{t+\Delta t}_{i,\,bod}$
6.          Estimate combined velocity (eq. 55): ${}^{(k)}\mathbf{v}^{t+\Delta t}_{ic,\,C}$
7.          7-1 Estimate new contact velocities (eq. 56): ${}^{(k)*}\mathbf{v}^{t+\Delta t}_{ic,\,bod}$
            7-2 Update material point trial stress (eq. 37 or 42): $\boldsymbol{\sigma}^{trial}_{p,\,bod}$
            7-3 Loop 2: $Y(\boldsymbol{\sigma}^{trial}_{p,\,bod}) > 0$: return stress to the yield surface Y
                    Plastic iteration loop
                    IF $Y(\boldsymbol{\sigma}^{trial}_{p,\,bod}) = 0$: EXIT Loop 2
                END Loop 2
            Update material point stress after plasticity: $\boldsymbol{\sigma}^{t+\Delta t}_{p,\,bod} = \boldsymbol{\sigma}^{trial}_{p,\,bod}$
            IF convergence is true or the maximum number of iterations is reached: EXIT Loop 1
        END Loop 1
8.      Update material point variables (eqs. 34, 35 and 36): $\mathbf{a}^{t+\Delta t}_{p,\,bod}$, $\mathbf{v}^{t+\Delta t}_{p,\,bod}$ and $\mathbf{x}^{t+\Delta t}_{p,\,bod}$
```

**Fig. 3.** Steps followed in the implicit solution scheme, in which the square brackets indicate a loop over the bodies.

$$
{}^{(k)}\mathbf{F}^{nc,\,t+\Delta t}_{ic,\,bod}\Big|_{elem} = \frac{\overline{\mathbf{m}}_{ic,\,bod}\left[\left({}^{*}\mathbf{v}^{t+\Delta t}_{ic,\,bod} - \mathbf{v}^{t+\Delta t}_{ic,\,bod}\right)\cdot\mathbf{n}_{ic,\,bod}\right]\mathbf{n}_{ic,\,bod}}{\Delta t} \tag{57}
$$

with the tangential force to ensure sticking conditions calculated as

$$
{}^{(k)}\mathbf{F}^{stick,\ t+\Delta t}_{ic,\,bod} = -\frac{\overline{\mathbf{m}}_{ic,\,bod}\mathbf{n}_{ic,\,bod}\times\left[\left(\mathbf{v}^{t+\Delta t}_{ic,\,bod} - \mathbf{v}^{t+\Delta t}_{ic,\,C}\right)\times\mathbf{n}_{ic,\,bod}\right]}{\Delta t} \tag{58}
$$

and the frictional force

$$
{}^{(k)}\mathbf{F}^{fric,\ t+\Delta t}_{ic,\,bod}\Big|_{elem} = \frac{{}^{(k)}\mathbf{F}^{stick,\ t+\Delta t}_{ic,\,bod}}{\left|{}^{(k)}\mathbf{F}^{stick,\ t+\Delta t}_{ic,\,bod}\right|}\min\left(\mu\left|{}^{(k)}\mathbf{F}^{nc,\,t+\Delta t}_{ic,\,bod}\right|,\left|{}^{(k)}\mathbf{F}^{stick,\ t+\Delta t}_{ic,\,bod}\right|\right) \tag{59}
$$

Note that these equations are similar to those used in the explicit scheme, with the exception that the nodal combined velocities ($\mathbf{v}^{t+\Delta t}_{ic,\ bod}$) are updated every iteration, which leads to a new set of corrected velocities and nodal forces. Finally, if contact occurs, the incremental displacements should be computed using

$$
\left(\overline{\mathbf{K}}_{bod}\ {}^{(k)}\overline{\mathbf{u}}_{i,\ bod}\right) = {}^{(k-1)}\left(\mathbf{F}^{ext,\,t+\Delta t}_{i,\,bod} - \mathbf{F}^{kin,\,t+\Delta t}_{i,\,bod} - \mathbf{F}^{int,\,t+\Delta t}_{i,\,bod} + \mathbf{F}^{nc,\,t+\Delta t}_{ic,\,bod} - \mathbf{F}^{fric,\,t+\Delta t}_{ic,\,bod}\right) \tag{60}
$$

Fig. 3 shows the steps followed to simulate contact using the implicit scheme. It should be noted that the vector of external forces ($\mathbf{F}^{ext}$) and the stiffness matrix ($\overline{\mathbf{K}}$) are evaluated at time $t + \Delta t$ outside the iteration loop, as both remain constant throughout the iterative procedure, i.e. the modified Newton-Raphson method is used. The small time steps used to capture the kinematics ensure that this results in a reasonable solution.

## 5. Application and evaluation of contact methods

### 5.1. Benchmark problems

To evaluate the performance of the implicit contact method, three benchmarks are analysed and the solution of each compared against the explicit solution, as well as against the analytical and FEM solutions where available. The first two benchmarks (i.e. a collision and a 1D vibrating bar) are used to study energy conservation. The first benchmark considers a contact occurring instantaneously between two blocks, and the second considers contact which persists throughout the whole simulation. The third benchmark is the simulation of a block sliding on a rigid surface, and is used to analyse the interaction between bodies when considering frictional forces. Each benchmark was simulated using four equally spaced material points per background element, which were

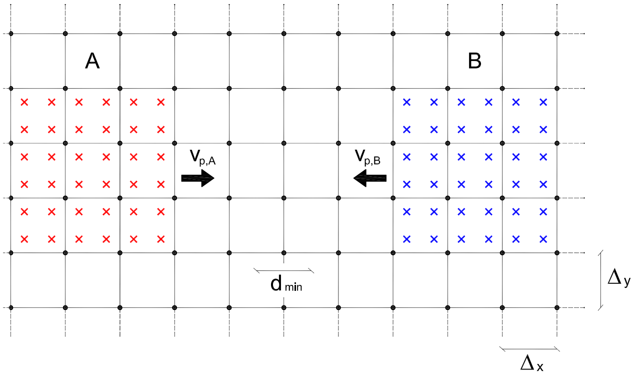

**Fig. 4.** Collision benchmark.

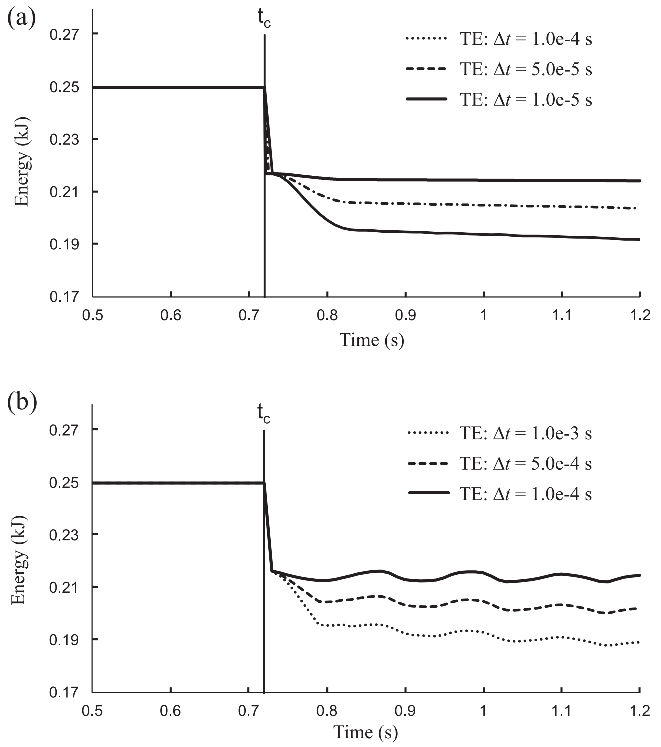

**Fig. 5.** Energy conservation after collision using a) the explicit scheme, and b) the implicit scheme.

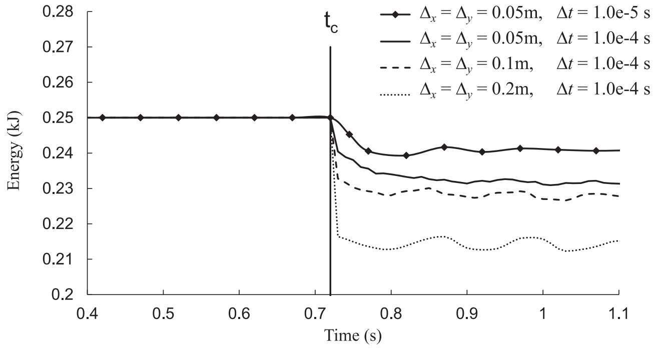

**Fig. 6.** Energy conservation with implicit scheme using different mesh sizes.

initially positioned at the local coordinates $\xi = \pm 0.5$ and $\eta = \pm 0.5$. All the benchmarks and geotechnical simulations introduced later are simulated using plane strain conditions.

### 5.1.1. *Collision benchmark*

In Fig. 4, the initial configuration of the collision benchmark is shown. The benchmark consists of two square blocks (A and B) of size 0.6 m, moving freely towards each other through a background mesh with a mesh spacing of $\Delta_x = \Delta_y = 0.20$ m. Each block has an initial horizontal velocity of 0.5 m/s. The elastic properties of both blocks are Young's modulus, $E = 500$ kPa, and Poisson's ratio, $\nu = 0.30$. The tolerance values used in the Newton-Raphson iterative procedure, and to establish contact, are $tol = 1.0 \times 10^{-8}$ and $d_{min} = 0.2$ m, respectively. Gravity forces have not been considered.

Fig. 5 shows the results of the collision in terms of total energy (TE), for both the explicit and the implicit schemes. In Fig. 5a, for the explicit scheme, it is observed that, at the moment of the collision ($t_c$), the energy decreases significantly (by 17%) and, depending on the time step used, the total energy either continues decreasing or stays almost constant after the collision. Fig. 5b shows that, using the implicit scheme, the total energy behaves in a similar way to the explicit solution, except that some oscillation of the energy is observed after the collision. These oscillations are attributed to not capturing accurately the internal stress waves due to the large time step used, which is 10 times greater than that used with the explicit scheme for almost the same performance.

**Table 1**
Collision benchmark computational time.

| Mesh size (m) | Computational time (min) | | Implicit / Explicit |
|---|---|---|---|
| | Implicit | Explicit | |
| 0.2 | 1 | 3 | 0.33 |
| 0.1 | 5 | 7 | 0.71 |
| 0.05 | 21 | 31 | 0.68 |
| 0.02 | 224 | 330 | 0.68 |

Additionally, Fig. 6 and Table 1 demonstrate the energy conservation and computational time required to complete each simulation with respect to the mesh (element) size for the implicit scheme. The study of energy conservation is performed using a time step of $\Delta t = 1.0 \times 10^{-4}$ s, and mesh sizes of 0.20, 0.10 and 0.05 m. Also, an additional curve (solid line with markers) computed using a mesh size of 0.05 m and a time step of $\Delta t = 1.0 \times 10^{-5}$ s is added to Fig. 6, to study the effects of reducing the time step and the mesh size simultaneously. The contact tolerance values used are the maximum values for the mesh, i.e. $d_{min} = 0.20$, 0.10 and 0.05 m, respectively. It is observed that the drop of energy at the point of contact decreases with the reduction of the mesh size, improving the conservation of energy. Furthermore, by reducing the mesh size and time step together, the drop of energy and the oscillations after contact reduce drastically, reinforcing the theory that the oscillations in Fig. 5b were caused by the large time step. However, despite the improvement in the results, some reduction in energy is still observed.

An analysis of the computational time has been performed, considering the time steps $\Delta t = 1.0 \times 10^{-5}$ s and $\Delta t = 1.0 \times 10^{-4}$ s, for the explicit and the implicit schemes, respectively. It is observed that the computational time needed to complete a simulation grows substantially with the reduction of the mesh size. Nevertheless, the time relationship is almost constant, with the implicit solution being on average 30% faster than the explicit solution.

The reason for the unrealistic decrease of energy observed in Figs. 5 and 6 is the inconsistency between the contact loads and the internal loads developed at the contact ($\mathbf{F}^{nc} \neq \mathbf{F}^{int}$). One method to prevent this inconsistency is to compute the contact loads as a function of the internal loads developed by each body at the contact (Xiao-Fei et al. 2008). At the contact interface, the normal acceleration of each body can be assumed to be equal (i.e. $\mathbf{a}^t_{i,A}\cdot\mathbf{n}_{i,A} = -\mathbf{a}^t_{i,B}\cdot\mathbf{n}_{i,B}$). Hence, from Newton's second law, it is established that

$$m_{i,A}\left(\mathbf{a}_{i,A}\cdot\mathbf{n}_{i,A}\right) = \mathbf{F}^{int}_{i,A}\cdot\mathbf{n}_{i,A} - \mathbf{F}^{nc}_{i} \tag{61}$$

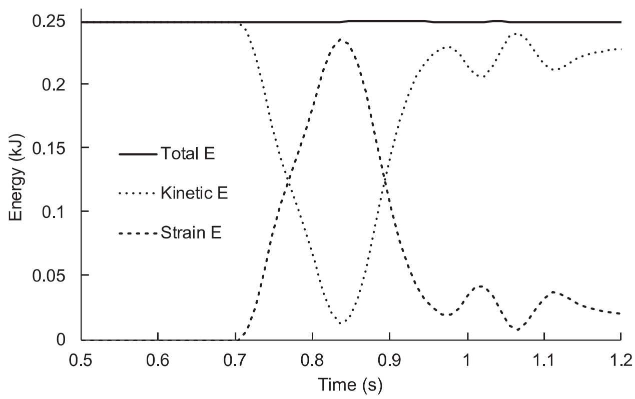

**Fig. 7.** Energy conservation during block collision with the explicit scheme using eq. (63).

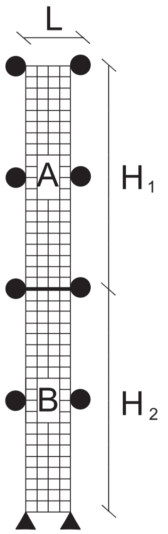

**Fig. 8.** 1D bar benchmark background mesh, where each element was initially filled with 4 material points.

$$m_{i,B}\left(\mathbf{a}_{i,B}\cdot\mathbf{n}_{i,B}\right) = \mathbf{F}^{int}_{i,B}\cdot\mathbf{n}_{i,B} + \mathbf{F}^{nc}_{i} \tag{62}$$

which reduces to

$$\mathbf{F}^{nc}_{i,A} = \frac{\left(m_{i,B}\,\mathbf{F}^{int}_{i,A} - m_{i,A}\,\mathbf{F}^{int}_{i,B}\right)\cdot\mathbf{n}_{i,A}}{\left(m_{i,A} + m_{i,B}\right)} = -\mathbf{F}^{nc}_{i,B} \tag{63}$$

where $\mathbf{F}^{nc}_{i,A}$ and $\mathbf{F}^{nc}_{i,B}$ are the contact forces between the bodies as a function of the internal forces developed. In Fig. 7, the energy conservation of the collision benchmark is plotted using eq. (63) to compute the contact loads, a step size of $\Delta t = 1.0 \times 10^{-5}$ s, a mesh of size of $\Delta_x = \Delta_y = 0.20$ m, and the explicit scheme. It is seen that the energy conservation improves considerably because the contact loads and the internal loads are equivalent. However, this approach should be implemented with caution, since the contact forces are dependent on the material point stresses which are known to suffer from numerical oscillations, leading to oscillations of the contact loads. Since stress oscillation problems are typical in MPM, in order to implement eq. (63) in any problem more complicated than this example, further improvements are needed. Additionally, the implementation of this contact technique using the implicit scheme is not straightforward, due to the interdependence of the internal and contact loads and the iterative nature of the implicit scheme. Since the improvements needed to implement this technique in both solution schemes are beyond the scope of this paper, all remaining examples have been computed using the initial approach.

### 5.1.2. 1D bar benchmark

This benchmark is similar to other 1D vibration benchmarks, except that this 1D bar is made up of two bars (bodies A and B). Initially, the bodies have zero internal stresses, but, due to the influence of gravity, they compress as soon as they are allowed to move. The equations of motion are calculated separately for the two bodies and the interaction between them is calculated via the contact forces.

Fig. 8 shows a sketch of the benchmark, including the dimensions of the bars and the boundary conditions. The background mesh is shown, where each element was initially filled with 4 material points. Each bar has a height of $H_1 = H_2 = 5$ m and a width of $L = 1$ m. The size of the mesh is $\Delta_x = \Delta_y = 0.25$ m. The properties of both bars are Young's modulus, $E = 1000$ kPa, Poisson's ratio, $\nu = 0.35$, and unit weight, $\gamma = 15$ kN/m$^3$. The tolerance values used in the Newton-Raphson iterative procedure and the contact distance are $tol = 1.0 \times 10^{-8}$ and $d_{min} = 0.25$ m,respectively.

Figs. 9 and 10 show the sum of the total energy of each 1D bar and the displacement of the top material point over 3 s using various time steps, for the explicit and implicit solution schemes, respectively. An FEM result is also included to provide a reference solution. As can be seen in Fig. 9a, by reducing the time step the energy conservation is improved. The largest drop of energy is during compression, when most of the penetration occurs. Using the smallest time step of $\Delta t = 1.0 \times 10^{-5}$ s, the loss of energy is the lowest (4.7% after 3 s). On the other hand, using the largest time step, $\Delta t = 1.0 \times 10^{-4}$ s, the loss of energy is considerable (22.5% after 3 s). Consistent with the calculated loss of energy, Fig. 9b shows that the

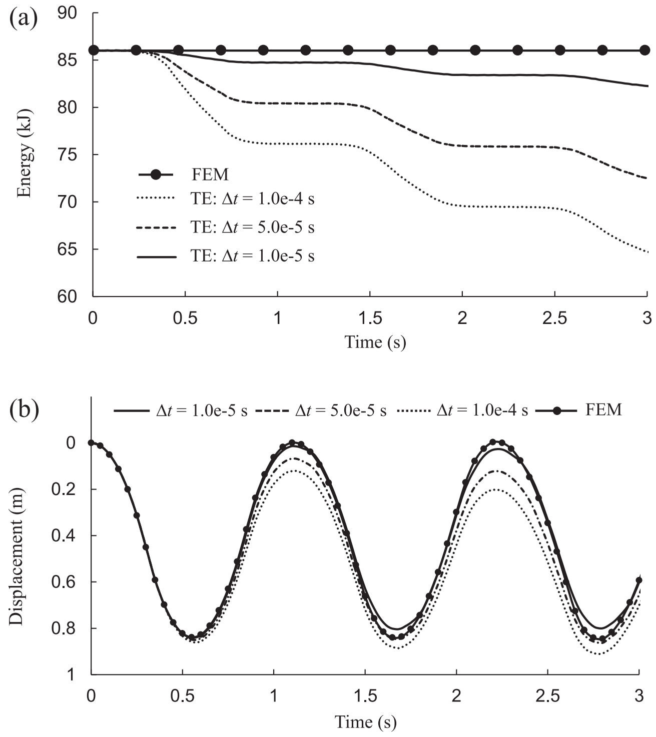

**Fig. 9.** a) Sum of the total energy of each 1D bar using the explicit scheme, and b) displacement of the top material point.

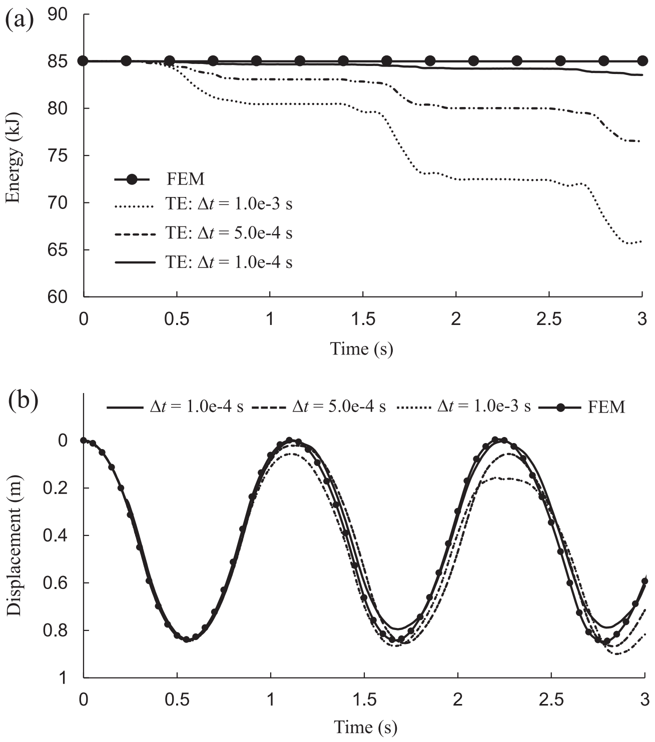

**Fig. 10.** a) Sum of the total energy of each 1D bar using the implicit scheme, and b) displacement of the top material point.

top material point is unable to reach its original position when a large time step is used. In contrast, when the smallest time step is used, the result is nearly equal to the FEM result (i.e. accurate). This is because the computed contact loads can adjust satisfactorily to the change of internal forces and velocities of each body during contact when smaller time steps are used, restricting the penetration of the bodies.

When the implicit scheme is used (Fig. 10), the results for energy conservation and material point position are similar to those using the explicit scheme, based on each implicit simulation using a time step 10 times larger than in the explicit simulation.

### *5.1.3. Sliding benchmark*

In this benchmark, a block under gravity loading is located on a flat surface and slides. Fig. 11 is a sketch of the problem, showing that part of the background mesh where each element was initially filled with 4 material points. Body A is a rigid surface and a body B slides over body A due to the gravity load (**g**) which is rotated by θ = 15° to the vertical. The lengths of the two bodies are $L_1 = 1.5$ m and $L_2 = 6.0$ m, respectively, and the height of each body is $H_1 = H_2 = 0.75$ m. The mesh size is $\Delta_x = \Delta_y = 0.25$ m, and body A is fixed at the bottom boundary and at the vertical boundaries. The elastic properties of the bodies are Young's modulus, $E_\mathrm{A} = 1000$ kPa and $E_\mathrm{B} = 5000$ kPa, in which $E_\mathrm{A}$ and $E_\mathrm{B}$ refer to body A and B, respectively. The Poisson's ratio and unit weight for both bodies are $\nu = 0.35$ and $\gamma = 20$ kN/m$^3$, respectively. Two simulations using two different friction factors, $\mu = 0$ and $\mu = 0.15$, have been performed, and the results are compared with the analytical solution of Jiang & Yeung (2004). Time steps of $\Delta t = 5.0 \times 10^{-5}$ s and $\Delta t = 5.0 \times 10^{-4}$ s were used for the explicit and the implicit simulations, respectively. The tolerance values used in the Newton-Raphson iterative procedure and the contact distance are *tol* $= 1.0 \times 10^{-8}$ and $d_{min} = 0.25$ m, respectively.

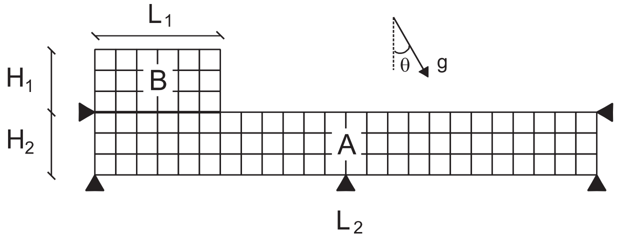

**Fig. 11.** Schematic of the sliding benchmark background mesh, where each element was initially filled with 4 material points.

Fig. 12 shows the results of the benchmark over two seconds of sliding. It is seen that the simulated results using the explicit and the implicit schemes are close to the analytical solution. Nevertheless, small deviations can be seen, which are attributed to the non-smooth distribution of the contact loads at the interface between bodies A and B. Fig. 13 shows this distribution at the beginning of the simulation (i.e. no slip had yet occurred) and after some sliding considering $\mu = 0.15$. Fig. 13a shows that the normal load $F^{nc}$ is constant at the centre of the block and drops close to the corners. In addition, it shows that the frictional load $F^{fric}$ is developing from right ($F^{fric} = -0.43$ kN at 1 m) to left ($F^{fric} = 0.29$ kN at 0.25 m) due to the weight of body B and the inclined gravity force. The frictional force has both positive and negative components. After slip occurs, Fig. 13b shows that the frictional loads are constant below body B, indicating that the frictional force is fully developed. Additionally, it is observed that the results obtained using the explicit and the implicit schemes are similar. The reader is directed to the work of Oden & Pires (1984) in which a similar simulation was performed, although under static conditions with loading applied via a traction on a single edge. The similarities in the

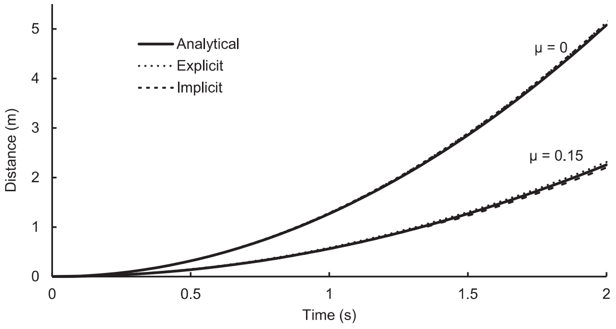

**Fig. 12.** Sliding simulation using the explicit and the implicit schemes.

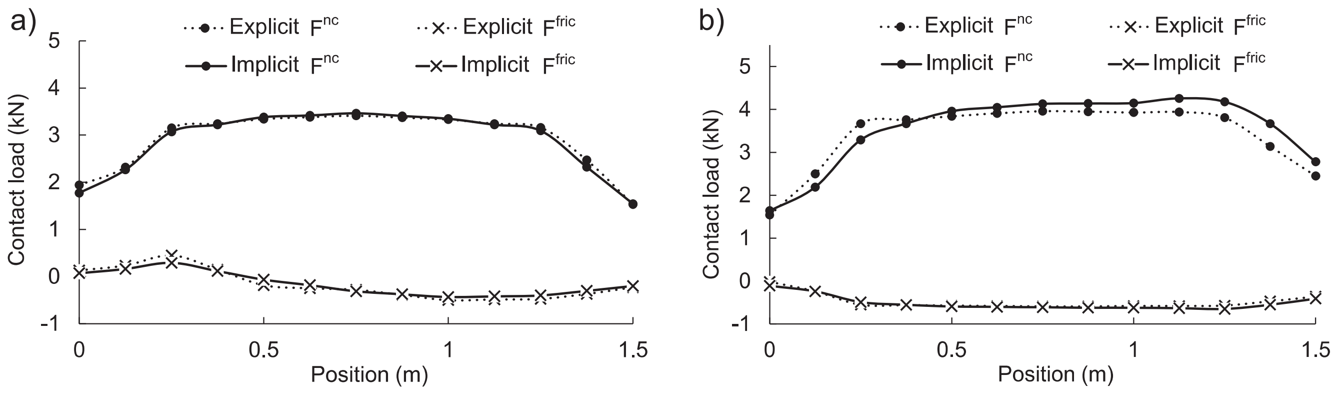

**Fig. 13.** Normal ($F^{nc}$) and tangent ($F^{fric}$) contact loads below body B, a) before displacement and b) after displacement.

qualitative distributions of the normal and frictional loads before the movement of the block give confidence in the analysis. Figs. 14 and 15 show the average of the normal and tangent loads, respectively, below body B during the sliding. It is observed that both loads are close and oscillate around the analytical solution (which is computed considering static conditions). These oscillations are caused mainly by the vibration of body B during sliding. This is more evident in the explicit results, which exhibit larger oscillations.

## 5.2. Geotechnical applications

Two geotechnical problems are simulated using both the explicit and the implicit contact schemes. The first problem consists of a shallow foundation that penetrates into the soil due to an increase in gravitational loading applied to the foundation. The second problem consists of a vertical cutting that, after failing, slides and collides against a wall. In these problems, the Young's modulus used for the rigid elements (i.e. the foundation and the wall) are lower than realistic values for such elements. The reason for this is to permit a limited amount of bending of the 'rigid' bodies, to demonstrate the capabilities of the method.

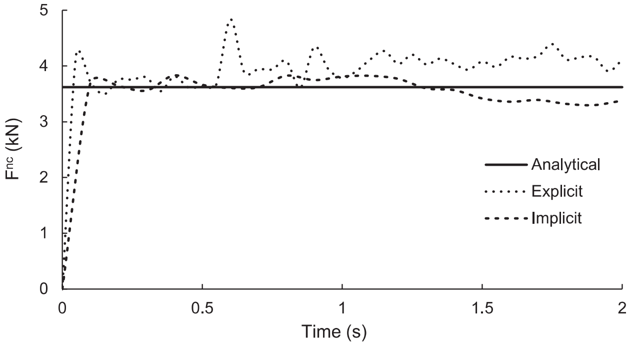

**Fig. 14.** Interface normal force ($F^{nc}$).

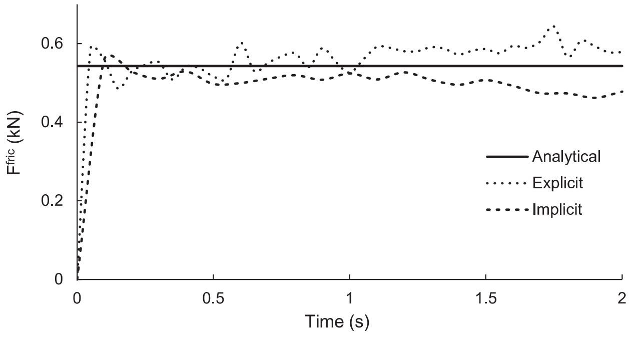

**Fig. 15.** Interface tangent force ($F^{fric}$).

### 5.2.1. Foundation

This problem involves a foundation slab that penetrates into an incompressible, linear elastic, perfectly plastic von Mises soil, with a rough interface, i.e. no tangential slip is allowed. The initial stresses in the soil are computed as $\sigma_y = \gamma_{soil} h_y$ and $\sigma_x = \sigma_z = K_0\sigma_y$, where $\gamma_{soil} = 17.5$ kN/m$^3$ is the soil unit weight, $h_y$ is the depth of the material point below the soil surface, and $K_0 = 0.7$ is the coefficient of earth pressure at rest. The elastic parameters of the soil are Young's modulus, $E_{soil} = 5.0\times10^3$ kPa, and Poisson's ratio, $\nu_{soil} = 0.49$, and its undrained shear strength is $S_u = 10$ kPa. The slab Young's modulus and Poisson's ratio are $E_{slab} = 1.0 \times10^4$ kPa and $\nu_{slab} = 0.45$, respectively.

Fig. 16 shows the geometry and boundary conditions of the problem. The slab dimensions are thickness, $H_1 = 0.6$ m, and width, $2L_1 = 4.0$ m (only half of the foundation has been modelled), while the modelled soil domain has a depth of $H_2 = 4.0$ m and a width of $L_2 = 7.0$ m, respectively. The vertical boundaries of the domain are fixed in the horizontal direction, whereas the bottom boundary is fixed in both directions. The mesh comprises four noded square elements of size $\Delta_x = \Delta_y = 0.20$ m, and there are initially four MPs in each element. Based on the results of the benchmark analyses, the time steps selected for the explicit and implicit simulations were $\Delta t_{expl} = 1.0 \times 10^{-5}$ s and $\Delta t_{impl} = 1.0 \times 10^{-4}$ s, respectively. The pressure exerted by the slab on the soil has been applied by increasing the self-weight of the slab by gravitational loading; i.e. by assuming the density of the slab to be $\rho_{slab} = 3000$ kg/m$^3$ and increasing the gravity from zero at a rate of 10 $g$/s. Finally, the tolerance value used in the Newton-Raphson iterative procedure and the contact distance are $tol = 1.0 \times 10^{-8}$ and $d_{min} = 0.20$ m, respectively.

Fig. 17 shows the results for the foundation problem after a settlement of 0.20 m using the implicit scheme (with similar results also being obtained with the explicit scheme). In Fig. 17a, the plastic deviatoric strain contours indicate a failure mechanism resembling the failure described in

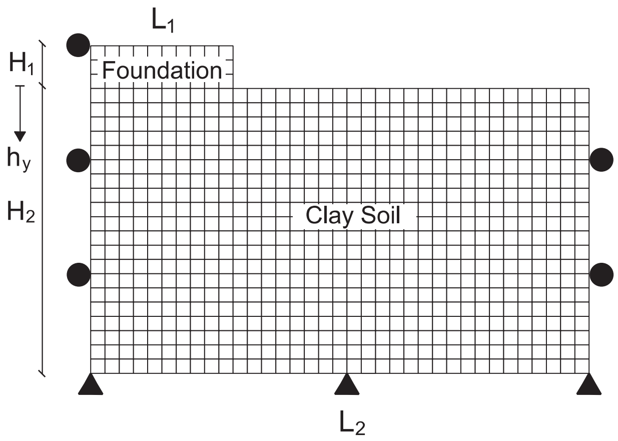

**Fig. 16.** Foundation problem.

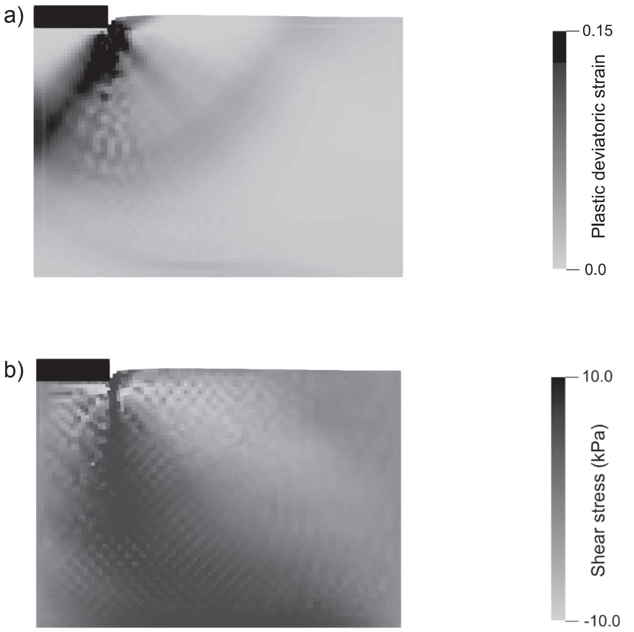

**Fig. 17.** a) Plastic deviatoric strains after failure, and b) shear stresses after failure.

Terzaghi (1943), while Fig. 17b shows the shear stresses developed. By comparing both parts of the figures, it is seen that the larger plastic deviatoric strains develop in zones where the shear stresses shift from positive values to negative values. It is also evident that, due to the soil yielding and the use of a large Poisson's ratio, some checker-board oscillation occurs, as is typical for von Mises incompressible materials using bi-linear elements. The reader is referred to Coombs et al. (2018) and González Acosta et al. (2019), where the volumetric locking problem has been studied.

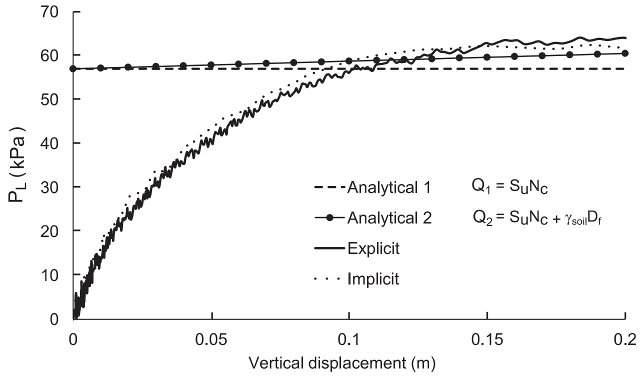

**Fig. 18.** Pressure–displacement curves up to a settlement of 0.2 m using both explicit and implicit schemes.

Fig. 18 shows the pressure ($P_L$)–displacement responses computed using the explicit and implicit schemes, plotted against two analytical solutions of Terzaghi (1943). Both of these solutions for the bearing capacity consider rough contact between the soil and foundation. However, one solution does not include any overburden load ($Q_1 = S_u N_c$), whereas the second solution does ($Q_2 = S_u N_c + \gamma_{soil} D_f$, where $D_f$ is the depth of the base of the foundation). The pressure load between the slab and the soil is computed from the contact forces ($\mathbf{F}^{nc}$) and the size of the foundation. As can be seen, despite the stress oscillations observed in Fig. 17, the maximum simulated bearing capacity of the soil is close to both analytical solutions, especially the second solution in which the influence of the deformation has been included (via the overburden component).

#### 5.2.2. *Vertical cutting*

This problem consists of a vertical cutting that fails due to self-weight and then hits a protection wall. Fig. 19 shows a schematic of the initial state of the problem, including the boundary conditions. As for the previous example, the initial stress states in the vertical cutting and the protection wall are computed using the depth of each material point from the surface of each structure (i.e. $h_{y1}$ and $h_{y2}$, respectively), and the unit weights and earth pressure coefficients of the soil and the wall, $\gamma_{soil} = 18.0$ kN/m$^3$, $K_{0\text{-}soil} = 0.7$ and $\gamma_{wall} = 25.0$ kN/m$^3$, $K_{0\text{-}wall} = 0.5$, respectively. To generate the initial stresses in the vertical cut, a static step is applied in which the movement of the material points is prevented. This step causes large localised shear stresses to develop at the toe of the vertical cut, which cause the triggering of the landslide when the material points are released. The boundary conditions are that both bodies are fully fixed at the base, and the soil is also fixed in the horizontal direction at the left-hand vertical boundary. The height and width of the modelled soil domain and the protection wall are $H_1 = 2.5$ m and $L_1 = 6.0$ m, and $H_2 = 1.0$ m and $L_2 = 0.8$ m, respectively. The distance between the soil and the wall is $L_3 = 1.0$ m, and the element size is $\Delta_x = \Delta_y = 0.10$ m.

The elastic parameters of the soil and the wall are Young's modulus, $E_{soil} = 1.0 \times 10^3$ kPa and $E_{wall} = 3.0 \times 10^3$ kPa, and Poisson's ratio, $\nu_{soil} = 0.38$ and $\nu_{wall} = 0.30$, respectively. The soil is modelled as a linear elastic, linear strain-softening von Mises material, with a peak cohesion, $c_p = 14$ kPa, a residual cohesion, $c_r = 5$ kPa, and a softening modulus, $H_s = $ -18 kPa (see Wang et al. 2016b for details of the constitutive model). The wall is simulated as a linear elastic material. The time steps selected for the explicit and the implicit simulations are $\Delta t_{expl} = 5.0 \times 10^{-5}$ s and $\Delta t_{impl} = 5.0 \times 10^{-4}$ s, respectively. The tolerance value used in the Newton-Raphson iterative procedure and the contact distance are $tol = 1.0 \times 10^{-8}$ and $d_{min} = 0.10$ m, respectively.

Fig. 20 shows the distribution of plastic deviatoric strains in the vertical cut, and the distribution of deviatoric stresses in the wall, after a) 0.8, b) 1.25 and c) 2.4 s in the implicit simulation. Fig. 20a shows a moment after the vertical cut has failed, with the soil mass moving downwards and to the right, and increasing its kinetic energy before the

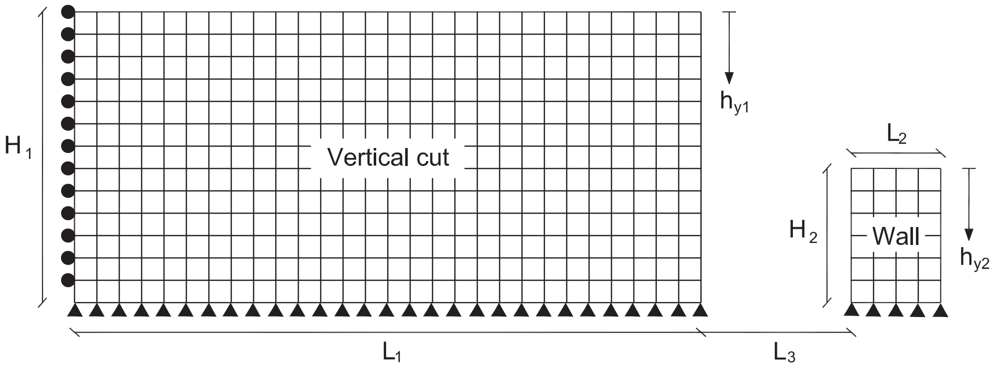

**Fig. 19.** Vertical cutting and protection wall (mesh indicative and not to scale).

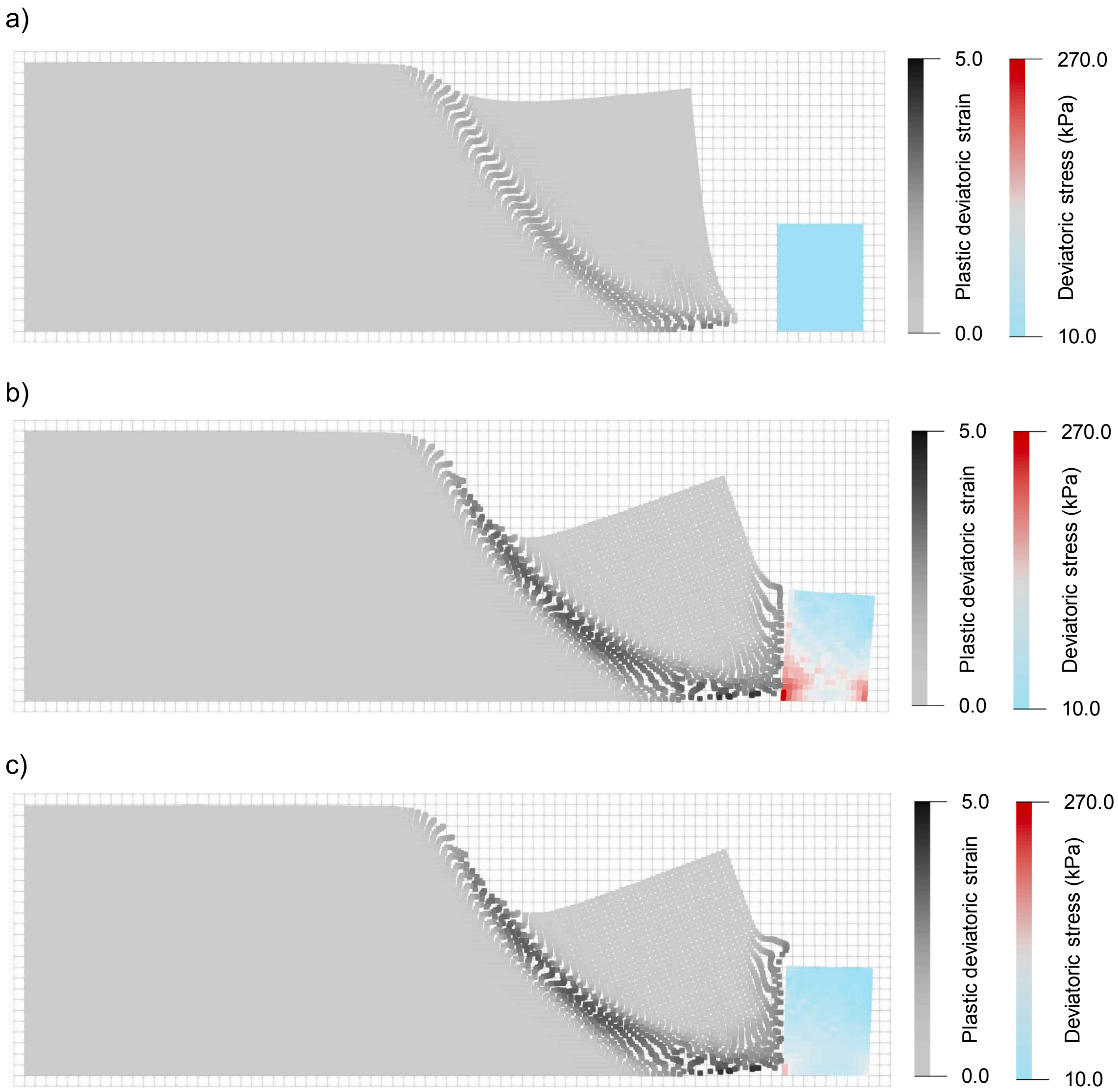

Fig. 20. Collision simulation after a) 0.8, b) 1.25, and c) 2.4 s. Plastic deviatoric strains are shown in the soil, and deviatoric stresses are shown in the wall.

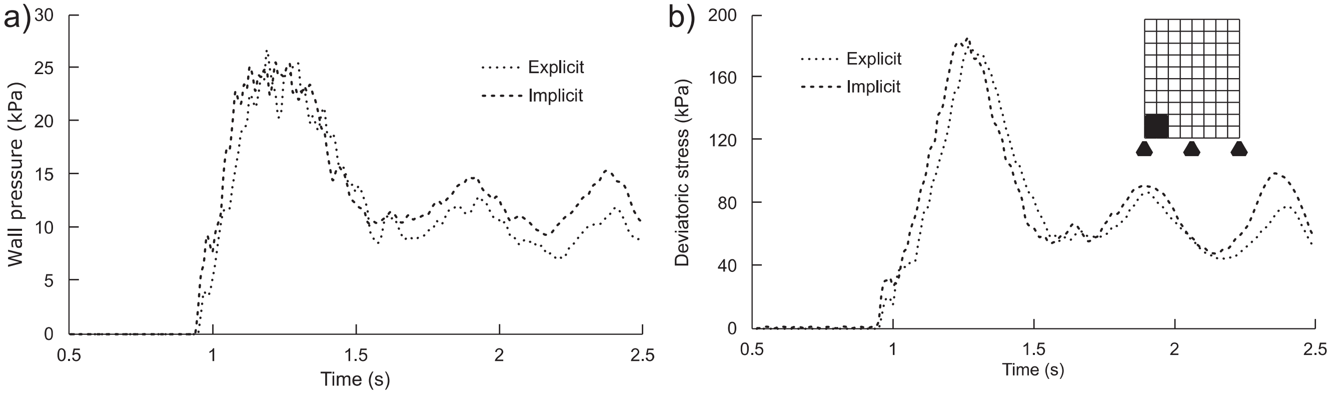

Fig. 21. a) Contact pressure at the wall surface, and b) deviatoric stress at the base of the wall.

collision. Fig. 20b shows the instant at which the contact load and the internal deviatoric stresses in the wall are a maximum. Fig. 20c shows the end of the simulation, when the wall has partially recovered its position. It can be seen that, due to the adopted proximity detection rule, the two bodies are closely connected during contact.

Fig. 21 shows the average pressure at the wall surface and the deviatoric stress at the wall base. The wall pressure in Fig. 21a has been computed by adding the contact force from each contact node and dividing it by the wall height $H_2$. The deviatoric stress $q$ in Fig. 21b has been computed, using the principal stresses, as

$$q = \frac{\sqrt{(\sigma_1 - \sigma_2)^2 + (\sigma_1 - \sigma_3)^2 + (\sigma_2 - \sigma_3)^2}}{\sqrt{2}} \tag{64}$$

**Table 2**
Time step relationship to achieve comparable accuracy, computation time and memory usage for each simulation.

| Problem analysed | Implicit $\Delta t$ / Explicit $\Delta t$ | Explicit CT (min) | Implicit CT (min) | Explicit MU (MB) | Implicit MU (MB) |
|---|---|---|---|---|---|
| Block collision | 5 | 3.3 | 1.1 | 1.9 | 2.1 |
| 1D vibration bar | 10 | 11.6 | 3.6 | 3.0 | 3.5 |
| Sliding block | 10 | 8.4 | 2.3 | 2.3 | 2.5 |
| Foundation settlement | 10 | 305 | 335 | 66.8 | 76.6 |
| Soil-wall collision | 10 | 94.3 | 68.3 | 80.9 | 86.6 |

The material points selected to compute the deviatoric stress are those inside four elements at the bottom left corner of the wall (Fig. 21b). These points were chosen because, in this section of the wall, the deviatoric stress is a maximum. It is observed that the results in Fig. 21 are consistent with each other, in that the maximum deviatoric stress occurs at the same time as the maximum contact pressure. Also, due to the elastic behaviour of the wall, an oscillatory behaviour is observed.

### 5.3. Computational time

The time steps used and the computational times recorded for the simulations from Sections 5.1 and 5.2 are presented, to further assess the capability of the implicit solution. Note that the time steps chosen for the geotechnical simulations were based on the results obtained from the benchmarks (i.e. the largest time steps which returned accurate results). The codes were compiled using Intel Parallel Studio XE, and run on a computer with an Intel Xeon E5-1620 processor and 16 GB RAM. However, it should be noted that no special effort was made to optimise the code (e.g. no parallel computing), which means that the installed RAM was not exploited to its maximum capacity.

Table 2 summarises the implicit/explicit time step relationship, computation time (CT), and memory usage (MU) between the simulations. It is observed that, in most of the simulations (i.e. except the block collision simulation), the time step used for the implicit simulations is 10 times larger than the time step used for the explicit simulations. Due to the use of a larger time step, the computational time is smaller (in most of the simulations) when using the implicit solution. The reason for the larger computational time in the foundation simulation is the incompressibility of the soil, which leads to more iteration steps in order to reach equilibrium. However, in more realistic simulations, where contact only occurs for part of the simulation time, the implicit algorithm should lead to substantial computational time savings. Finally, it is observed that the memory usage is almost the same in both cases. This is because the implicit scheme stores the global matrix in the form of a vector using the skyline strategy, and the adopted solver preserves the skyline during the solution step. Nevertheless, when the problems simulated contain a large number of points (e.g. the foundation and the soil/wall collision problems), the differences in memory usage are more evident.

## 6. Conclusion

MPM is able to simulate large-strain mechanical behaviour and, with the development of the contact solution, the interaction between multiple bodies is feasible. In this paper, an implicit contact solution was developed for MPM, based on existing explicit procedures and using the Newton-Raphson iterative procedure and Newmark's time integration scheme. The relative performance of the formulation with respect to an explicit scheme was investigated through analysing three benchmark problems. It was shown that the results obtained using the explicit and implicit methods were almost identical. It was observed that both methods dissipate energy in a similar manner due to an inconsistency between the contact loads and the increment of material point stresses, which is proportional to both the time step and mesh size discretisation. Moreover, it was observed that by reducing the time step and mesh size to realistically small sizes, a reasonable energy conservation can be achieved. The ability of the method for simulating geotechnical problems was demonstrated through the analysis of foundation slab settlement in a cohesive soil, in which the failure mechanism and the bearing capacity obtained were similar to Terzaghi's analytical solution, and through the analysis of the collision of a mass of soil against a protection wall. Finally, it was shown that the simulation using the implicit approach is faster when the contact loads in the problem are moderate; otherwise, the computational time can be similar to the explicit approach.

## CRediT authorship contribution statement

**José León González Acosta:** Conceptualization, Methodology, Software, Writing - original draft. **Philip J. Vardon:** Conceptualization, Writing - review & editing, Supervision. **Michael A. Hicks:** Conceptualization, Writing - review & editing, Supervision.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

The authors of this work express their gratitude to the Mexican National Council for Science and Technology (CONACYT) for financing this work through the Grant Number 409778.

## Appendix A

The following matrices are formulated considering square four noded elements, FE shape functions and plane strain conditions (i.e. $\varepsilon_z = 0$ and $\sigma_{xz} = \sigma_{yx} = 0$). The respective matrices of the SFs and the SF derivatives are

$$\overline{\mathbf{H}} = \begin{bmatrix} \overline{H}_1 & 0 & \overline{H}_2 & \cdots & 0 & \overline{H}_n & 0 \\ 0 & \overline{H}_1 & 0 & \cdots & \overline{H}_{n-1} & 0 & \overline{H}_n \end{bmatrix} \tag{A1}$$

and

$$\overline{\mathbf{B}} = \begin{bmatrix} \dfrac{\partial \overline{H}_1}{\partial x} & 0 & \cdots & \dfrac{\partial \overline{H}_n}{\partial x} & 0 \\ 0 & \dfrac{\partial \overline{H}_1}{\partial y} & \cdots & 0 & \dfrac{\partial \overline{H}_n}{\partial y} \\ \dfrac{\partial \overline{H}_1}{\partial y} & \dfrac{\partial \overline{H}_1}{\partial x} & \cdots & \dfrac{\partial \overline{H}_n}{\partial y} & \dfrac{\partial \overline{H}_n}{\partial x} \end{bmatrix} \tag{A2}$$

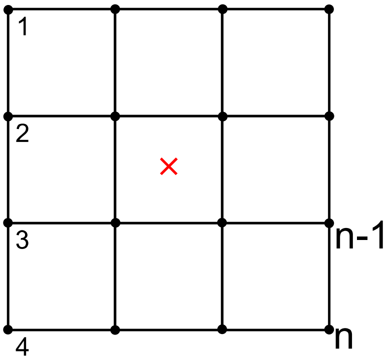

**Fig. A1.** Extended domain for a material point using an array of $3^2$ elements.

where $\overline{\mathbf{H}}$ represents the matrix of FE bi-linear SFs ($\mathbf{N}$) or GIMP SFs ($\mathbf{S}_{ip}$), $\overline{\mathbf{B}}$ represents the strain–displacement matrix $\mathbf{B}$, $\nabla\mathbf{S}_{ip}$ or $\mathbf{B}_C$, the subscript indicates the node number, and $n = 4$ when bi-linear SFs are used or $n$ is the number of nodes inside the material point support domain when GIMP is used. On the other hand, the relationship between stresses and strains ($\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}$) considering a plane strain condition is

$$\begin{Bmatrix}\sigma_x\\ \sigma_y\\ \sigma_{xy}\end{Bmatrix} = \frac{E}{(1+\nu)(1-2\nu)}\begin{bmatrix}1-\nu & \nu & 0\\ \nu & 1-\nu & 0\\ 0 & 0 & \dfrac{1-2\nu}{2}\end{bmatrix}\begin{Bmatrix}\varepsilon_x\\ \varepsilon_y\\ \varepsilon_{xy}\end{Bmatrix} \tag{A3}$$

where the vector of strains ($\boldsymbol{\varepsilon} = \mathbf{B}\overline{\mathbf{u}}$) is

$$\begin{Bmatrix}\varepsilon_x\\ \varepsilon_y\\ \varepsilon_{xy}\end{Bmatrix} = \begin{bmatrix}\dfrac{\partial N_1}{\partial x} & 0 & \dfrac{\partial N_2}{\partial x} & 0 & \dfrac{\partial N_3}{\partial x} & 0 & \dfrac{\partial N_4}{\partial x} & 0\\ 0 & \dfrac{\partial N_1}{\partial y} & 0 & \dfrac{\partial N_2}{\partial y} & 0 & \dfrac{\partial N_3}{\partial y} & 0 & \dfrac{\partial N_4}{\partial y}\\ \dfrac{\partial N_1}{\partial y} & \dfrac{\partial N_1}{\partial x} & \dfrac{\partial N_2}{\partial y} & \dfrac{\partial N_2}{\partial x} & \dfrac{\partial N_3}{\partial y} & \dfrac{\partial N_3}{\partial x} & \dfrac{\partial N_4}{\partial y} & \dfrac{\partial N_4}{\partial x}\end{bmatrix}\begin{Bmatrix}\overline{u}_{x1}\\ \overline{u}_{y1}\\ \overline{u}_{x2}\\ \overline{u}_{y2}\\ \overline{u}_{x3}\\ \overline{u}_{y3}\\ \overline{u}_{x4}\\ \overline{u}_{y4}\end{Bmatrix} \tag{A4}$$

Finally, material point strains are computed using the CMPM extended domain (i.e. $\boldsymbol{\varepsilon} = \mathbf{B}^2\overline{\mathbf{u}}^{ext}$) as

$$\begin{Bmatrix}\varepsilon_x\\ \varepsilon_y\\ \varepsilon_{xy}\end{Bmatrix} = \begin{bmatrix}\dfrac{\partial N_1}{\partial x} & 0 & \cdots & \dfrac{\partial N_n}{\partial x} & 0\\ 0 & \dfrac{\partial N_1}{\partial y} & \cdots & 0 & \dfrac{\partial N_n}{\partial y}\\ \dfrac{\partial N_1}{\partial y} & \dfrac{\partial N_1}{\partial x} & \cdots & \dfrac{\partial N_n}{\partial y} & \dfrac{\partial N_n}{\partial x}\end{bmatrix}\begin{Bmatrix}\overline{u}_{x1}\\ \overline{u}_{y1}\\ \overline{u}_{x2}\\ \vdots\\ \overline{u}_{y(n-1)}\\ \overline{u}_{xn}\\ \overline{u}_{yn}\end{Bmatrix} \tag{A5}$$

where $n$ is the number of nodes in the extended domain as shown in Fig. A1. Note that the extended domain can be constructed using square arrays of $\mathrm{C}^2$ elements, where $\mathrm{C} > 1$.

## Appendix B

In this appendix, the combination of using GIMP SFs and the CMPM extended domain is explained. Fig. B1 shows a domain composed of 9 elements and 16 nodes (both numbered in the $y$ direction, from left to right). The domain is filled with several material points, and a particular material point (indicated by the larger red cross) is used to demonstrate the GIMP and CMPM stencils. Since GIMP SFs and local GIMP SFs are used to integrate the nodal forces and stiffness, the material point support domain is used to select the elements. In this case, four elements are selected (4,5,7,8). On the other hand, CMPM has a different "extended" domain which, in this case, uses the full domain to enhance the accuracy of the stresses computed using Lagrange interpolation functions. It should be noted that the use of a different domain to perform nodal and stiffness integration and to recover stresses does not represent a problem, since the equilibrium (Eqs. (13), (21), (23), (28), (54) and (60)) is computed using only GIMP and, after equilibrium, the stresses are computed using CMPM. The reader is directed to González Acosta et al. (2020) for further information on the implementation of DM-GC, where the case of boundary material points is presented.

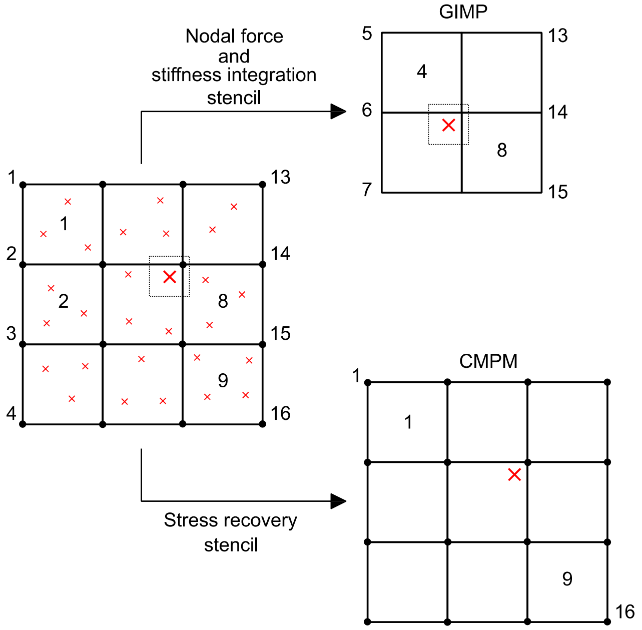

**Fig. B1.** Material point stencil (domain) used to perform nodal force and stiffness integration, and to recover (compute) stresses.

## Appendix C

To demonstrate the accuracy of DM-GC, a 1D benchmark problem is analysed. This problem is similar to the benchmark analysed in Coombs et al. (2018), in which a 1D column of material points is subjected to an increasing gravity force, and the error as a function of the number of square background elements (*nel*) is investigated. In this benchmark, the initial gravity load is zero, and the gravity is increased in increments of $\Delta g = 1.0 \times 10^{-4}\, g$ until maximum gravity loads of $g_{SD} = 0.1g$ and $g_{LD} = 20g$, in which the first maximum gravity load (i.e. $g_{SD}$) is used to study small deformations, and the second (i.e. $g_{LD}$) is used to study large deformations. The 1D bar has an initial height and width of $H_0 = 10$ m and $L = 1$ m. The elastic parameters of the bar are Young's modulus, $E = 1.0 \times 10^3$ kPa, and Poisson's ratio, $\nu = 0.0$. The density of the material points is $\rho = 1500$ kg/m$^3$. The bar is fully fixed at the bottom, and fixed in the horizontal direction along both vertical sides. Each background element is filled (initially) with four equally-spaced material points, and the simulations have been performed using the static scheme to avoid kinematic effects. The error during the calculations is measured as

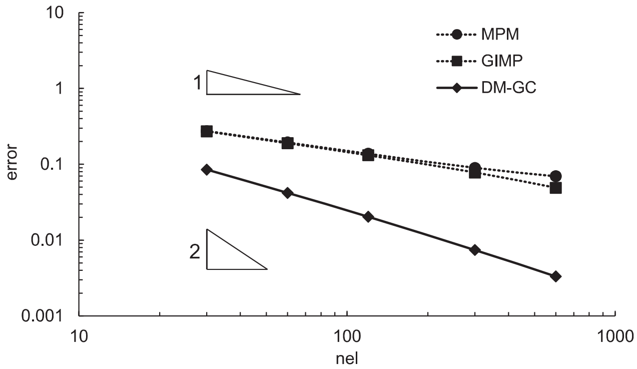

**Fig. C1.** Convergence of MPM, GIMP, and DM-GC after a gravity load of 0.1$g$.

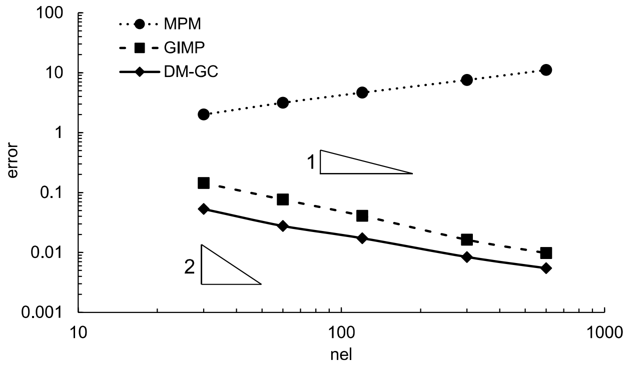

**Fig. C2.** Convergence of MPM, GIMP, and DM-GC after a gravity load of 20$g$.

$$error = \sum_{p=1}^{nmp} \frac{\left\| \sigma_{p,z} - \sigma_{a,z}(z) \right\| V_p^0}{(\rho g H_0) V_{BAR}^0} \tag{C1}$$

in which $\sigma_{p,z}$ is the material point vertical stress, $\sigma_{a,z}(z)$ is the analytical vertical stress in the bar at the height $z$ (i.e. $\sigma_{a,z} = \rho g(H\text{-}z)$, in which $z$ is measured from the bottom), $\|\bullet\|$ is the $L_2$ norm of ($\bullet$), $V_p^0$ is the initial material point volume, $V_{BAR}^0$ is the initial bar volume, and $\rho g H_0$ is the vertical stress at the bottom of the bar used to normalise the difference in the stresses.

Fig. C1 shows the convergence of MPM, GIMP and DM-CG considering small deformations. It is observed that MPM and GIMP have the same convergence ($\alpha \approx 1$, in which $\alpha$ is the order of convergence). Nevertheless, it is observed that the MPM error cannot reduce linearly at a certain point, which is attributed to the element crossing problem. On the other hand, using DM-GC the order of convergence is $1 < \alpha < 2$. Fig. C2 shows the results considering large deformations. In this case, it is observed that the MPM error grows due to the large number of material points crossing elements, which causes highly inaccurate internal stresses. On the other hand, the GIMP and DM-GC errors reduce at a similar rate (with DM-GC having a smaller error).

## References

Abe, K., Soga, K., Bandara, S., 2014. Material point method for coupled hydromechanical problems. J Geotech Geoenviron Eng 140 (3), 04013033. https://doi.org/10.1061/(ASCE)GT.1943-5606.0001011.

Bardenhagen, S.G., Brackbill, J.U., Sulsky, D., 2000. The material-point method for granular materials. Comput Methods Appl Mech Eng 187 (3–4), 529–541. https://doi.org/10.1016/S0045-7825(99)00338-2.

Bardenhagen, S.G., Guilkey, J.E., Roessig, K.M., Brackbill, J.U., Witzel, W.M., Foster, J.C., 2001. An improved contact algorithm for the material point method and application to stress propagation in granular material. Comput Model Eng Sci 2 (4), 509–522. https://doi.org/10.3970/cmes.2001.002.509.

Bardenhagen, S.G., Kober, E.M., 2004. The generalized interpolation material point method. Comp Model Eng Sci 5 (6), 477–495. https://doi.org/10.3970/cmes.2004.005.477.

Bathe, K.J., 2007. Conserving energy and momentum in nonlinear dynamics: A simple implicit time integration scheme. Comput Struct 85 (7–8), 437–445. https://doi.org/10.1016/j.compstruc.2006.09.004.

Bathe, K.J., 1996. Finite element procedures. Prentice Hall, New Jersey.

Belytschko, T., Liu, W.K., Moran, B., Elkhodary, K.I., 2013. Nonlinear finite elements for continua and structures, Second edition, John Wiley & Sons, New Jersey.

Beuth, L., Benz, T., Vermeer, P.A., 2008. Large deformation analysis using a quasi-static material point method. J Theor Appl Mech 38 (1–2), 45–60.

Chen, Z.P., Zhang, X., Qiu, X.M., Liu, Y., 2017. A frictional contact algorithm for implicit material point method. Comput Methods Appl Mech Eng 231, 124–144. https://doi.org/10.1016/j.cma.2017.04.006.

Coetzee, C.J., 2014. Discrete and continuum modelling of soil cutting. Comp Part Mech 1 (4), 409–423. https://doi.org/10.1007/s40571-014-0014-7.

Coombs, W.M., Charlton, T.J., Cortis, M.C., Augarde, C.E., 2018. Overcoming volumetric locking in material point methods. Comput Methods Appl Mech Eng 333, 1–21. https://doi.org/10.1016/j.cma.2018.01.010.

Davies, A.J., 2011. The finite element method: An introduction with partial differential equations. Oxford University Press, Oxford.

González Acosta, J.L., Pantev, I., Vardon, P.J., Hicks, M.A., 2018. The use of MPM to estimate the behaviour of rigid structures during landslides. In: Proceedings of the 9th European Conference on Numerical Methods in Geotechnical Engineering (NUMGE 2018), pp. 651–656.

González Acosta, J.L., Vardon, P.J., Hicks, M.A., 2017. Composite material point method (CMPM) to improve stress recovery for quasi-static problems. Procedia Eng 175, 324–331.

González Acosta, J.L., Vardon, P.J., Remmerswaal, G., Hicks, M.A., 2020. An investigation of stress inaccuracies and proposed solution in the material point method. Comput Mech 65 (2), 555–581. https://doi.org/10.1007/s00466-019-01783-3.

González Acosta, J.L., Zheng, X., Vardon, P.J., Remmerswaal, G., Hicks, M.A., Pisanò, F., 2019. On stress oscillation in MPM simulations involving one or two phases. In: Proceedings of the 2nd International Conference on the Material Point Method for Modelling Soil-Water-Structure Interaction, Cambridge.

Guilkey, J.E., Weiss, J.A., 2003. Implicit time integration for the material point method: Quantitative and algorithmic comparisons with the finite element method. Int J Numer Meth Eng 57 (9), 1323–1338. https://doi.org/10.1002/nme.729.

Homel, M.A., Herbold, E.B., 2017. Field-gradient partitioning for fracture and frictional contact in the material point method. Int J Numer Meth Eng 109 (7), 1013–1044. https://doi.org/10.1002/nme.5317.

Huang, P., Li, S.L., Guo, H., Hao, Z.M., 2015. Large deformation failure analysis of the soil slope based on the material point method. Comput Geosci 19 (4), 951–963. https://doi.org/10.1007/s10596-015-9512-9.

Huang, P., Zhang, X., Ma, S., Huang, X., 2011. Contact algorithms for the material point method in impact and penetration simulation. Int J Numer Meth Eng 85 (4), 498–517. https://doi.org/10.1002/nme.2981.

Jassim, I., Stolle, D., Vermeer, P., 2013. Two-phase dynamic analysis by material point method. Int J Numer Anal Meth Geomech 37 (15), 2502–2522. https://doi.org/10.1002/nag.2146.

Jiang, Q.H., Yeung, M.R., 2004. A model of point-to-face contact for three-dimensional discontinuous deformation analysis. Rock Mech Rock Eng 37, 95–116. https://doi.org/10.1007/s00603-003-0008-x.

Llano-Serna, M.A., Farias, M.M., Pedroso, D.M., 2016. An assessment of the material point method for modelling large scale run-out processes in landslides. Landslides 13 (5), 1057–1066. https://doi.org/10.1007/s10346-015-0664-4.

Liu, C., Sun, W.C., 2020. ILS-MPM: An implicit level-set-based material point method for frictional particulate contact mechanics of deformable particles. Comput Methods Appl Mech Eng 369. https://doi.org/10.1016/j.cma.2020.113168.

Ma, J., Lu, H., Komanduri, R., 2006. Structured mesh refinement in generalized interpolation material point (GIMP) method for simulation of dynamic problems. Comput Model Eng Sci 12 (3), 213–227. https://doi.org/10.3970/cmes.2006.012.213.

Ma, X., Zhang, D.Z., Giguere, P.T., Liu, C., 2013. Axisymmetric computation of Taylor cylinder impacts of ductile and brittle materials using original and dual domain material point methods. Int J Impact Eng 54, 96–104. https://doi.org/10.1016/j.ijimpeng.2012.11.001.

Müller, A., Vargas, E.A., 2019. Stability analysis of a slope under impact of a rock block using the generalized interpolation material point method (GIMP). Landslides 16 (4), 751–764. https://doi.org/10.1007/s10346-018-01131-1.

Nairn, J.A., 2003. Material point method calculations with explicit cracks. Comput Model Eng Sci 4 (6), 649–663.

Nairn, J.A., 2013. Modeling imperfect interfaces in the material point method using multimaterial methods. Comput Model Eng Sci 92 (3), 271–299. https://doi.org/10.3970/cmes.2013.092.271.

Newmark, N.M., 1959. A method of computation for structural dynamics. J Eng Mech Div-ASCE 85 (3), 37–94.

Oden, J.T., Pires, E.B., 1984. Algorithms and numerical results for finite element approximations of contact problems with non-classical friction laws. Comput Struct 19 (1–2), 137–147. https://doi.org/10.1016/0045-7949(84)90212-8.

Pantev, I., 2016. Contact modelling in the material point method, MSc Thesis. Delft University of Technology.

Sloan, S., Abbo, A., Sheng, D., 2001. Refined explicit integration of elastoplastic models with automatic error control. Eng Comput 18 (1/2), 121–194. https://doi.org/10.1108/02644400110365842.

Smith, I.M., Griffiths, D.V., Margetts, L., 2013. Programming the finite element method. John Wiley & Sons.

Sulsky, D., Chen, Z., Schreyer, H.L., 1994. A particle method for history dependent materials. Comput Methods Appl Mech Eng 118 (1–2), 179–196. https://doi.org/10.1016/0045-7825(94)90112-0.

Sulsky, D., Zhou, S., Schreyer, H.L., 1995. Application of a particle-in-cell method to solid mechanics. Comput Phys Commun 87 (1–2), 236–252. https://doi.org/10.1016/0010-4655(94)00170-7.

Terzaghi, K., 1943. Theoretical soil mechanics. John Wiley & Sons, New Jersey.

Wang, B., Hicks, M.A., Vardon, P.J., 2016a. Slope failure analysis using the random material point method. Géotechnique Letters 6 (2), 113–118. https://doi.org/10.1680/jgele.16.00019.

Wang, B., Vardon, P.J., Hicks, M.A., 2016b. Development of an implicit material point method for geotechnical applications. Comput Geotech 71, 159–167. https://doi.org/10.1016/j.compgeo.2015.08.008.

Wang, B., Vardon, P.J., Hicks, M.A., 2016c. Investigation of retrogressive and progressive slope failure mechanisms using the material point method. Comput Geotech 78, 88–98. https://doi.org/10.1016/j.compgeo.2016.04.016.

Więckowski, Z., 2004. The material point method in large strain engineering problems. Comput Methods Appl Mech Eng 193 (39–41), 4417–4438. https://doi.org/10.1016/j.cma.2004.01.035.

Woo, S.I., Salgado, R., 2018. Simulation of penetration of a foundation element in Tresca soil using the generalized interpolation material point method (GIMP). Comput Geotech 94, 106–117. https://doi.org/10.1016/j.compgeo.2017.08.007.

York II, A.R., Sulsky, D., Schreyer, H.L., 1999. The material point method for simulation of thin membranes. Int J Numer Meth Eng 44 (10), 1429–1456. https://doi.org/10.1002/(SICI)1097-0207(19990410)44:10%3C1429::AID-NME536%3E3.0. CO;2-4.

Xiao-Fei, P., Ai-Guo, X., Guang-Cai, Z., Ping, Z., Jian-Shi, Z., Shang, M., Xiong, Z., 2008. Three-dimensional multi-mesh material point method for solving collision problems. Commun Theor Phys 49 (5), 1129–1138. https://doi.org/10.1088/0253-6102/49/5/09.

Zhang, X., Sze, K.Y., Ma, S., 2006. An explicit material point finite element method for hyper-velocity impact. Int J Numer Meth Eng 66 (4), 689–706. https://doi.org/10.1002/nme.1579.