

## Development of an implicit material point method for geotechnical applications

Wang, B; Vardon, PJ; Hicks, MA; Chen, Z

**DOI**

[10.1016/j.compgeo.2015.08.008](https://doi.org/10.1016/j.compgeo.2015.08.008)

**Publication date**

2016

**Document Version**

Final published version

**Published in**

Computers and Geotechnics

**Citation (APA)**

Wang, B., Vardon, P.J., Hicks, M.A., & Chen, Z. (2016). Development of an implicit material point method for geotechnical applications. *Computers and Geotechnics*, 71(January), 159-167.  
<https://doi.org/10.1016/j.compgeo.2015.08.008>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.



## Research Paper

## Development of an implicit material point method for geotechnical applications

Bin Wang<sup>a</sup>, Philip J. Vardon<sup>a,\*</sup>, Michael A. Hicks<sup>a</sup>, Zhen Chen<sup>b,c</sup><sup>a</sup> Geo-Engineering Section, Faculty of Civil Engineering and Geosciences, Delft University of Technology, P.O. Box 5048, 2600 GA Delft, The Netherlands<sup>b</sup> Department of Engineering Mechanics, Faculty of Vehicle Engineering and Mechanics, Dalian University of Technology, Dalian 116024, PR China<sup>c</sup> Department of Civil and Environmental Engineering, University of Missouri, Columbia, MO 65211-2200, USA

## ARTICLE INFO

## Article history:

Received 9 April 2015

Received in revised form 6 July 2015

Accepted 29 August 2015

Available online 23 October 2015

## Keywords:

Dynamics

Quasi-statics

Finite element method

Implicit procedures

Material point method

Slope collapse

## ABSTRACT

An implicit material point method (MPM), a variant of the finite element method (FEM), is presented in this paper. The key feature of MPM is that the spatial discretisation uses a set of material points, which are allowed to move freely through the background mesh. All history-dependent variables are tracked on the material points and these material points are used as integration points similar to the Gaussian points. A mapping and re-mapping algorithm is employed, to allow the state variables and other information to be mapped back and forth between the material points and background mesh nodes during an analysis. In contrast to an explicit time integration scheme utilised by most researchers, an implicit time integration scheme has been utilised here. The advantages of such an approach are twofold: firstly, it addresses the limitation of the time step size inherent in explicit integration schemes, thereby potentially saving significant computational costs for certain types of problems; secondly, it enables an improved algorithm accuracy, which is important for some constitutive behaviours, such as elastoplasticity. The main purpose of this paper is to provide a unified MPM framework, in which both quasi-static and dynamic analyses can be solved, and to demonstrate the model behaviour. The implementation closely follows standard FEM approaches, where possible, to allow easy conversion of other FEM codes. Newton's method is used to solve the equation of motion for both cases, while the formation of the mass matrix and the required updating of the kinematic variables are unique to the dynamic analysis. Comparisons with an Updated Lagrangian FEM and an explicit MPM code are made with respect to the algorithmic accuracy and time step size in a couple of representative examples, which helps to illustrate the relative performance and advantages of the implicit MPM. A geotechnical application is then considered, illustrating the capabilities of the proposed method when applied in the geotechnical field.

© 2015 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The material point method (MPM) has been shown to be a robust spatial discretisation method for simulating multi-phase interactions involving large deformations and failure evolution. During 1994–96, Sulsky et al. [1–3] first developed and applied the method for modelling solid materials. This led to researchers, from different fields, recognising the potential of the method and adapting it to various applications, e.g. silo discharge and plastic forming [4,5], explosion problems, exploiting its ability to represent an arbitrary geometry [6,7], large-scale response of cellular constructs in biomechanics [8], and, more recently, for

geotechnical analysis, including the modelling of retaining wall failure [4], anchor pull-out [9], soil column collapse [10,11], landslides and debris flows [12], landslide induced interactions with structures [13], and quasi-static analyses of slope stability [14,15].

MPM uses two spatial discretisations, the first one that discretises a continuum body with a set of material points carrying all the state variables, and the second one that discretises the background grid (a computational mesh) to solve the equations of motion. The computational mesh may be maintained in its original position, or it can be adjusted in an appropriate way to avoid mesh distortion after each time/loading step, thereby removing the disadvantage of the finite element method (FEM) for which extreme mesh distortion may occur due to large deformations. As with FEM, the time integration of MPM can be either implicit or explicit, in which the latter has been employed for most of the MPM developments so far. This paper is concerned with the implementation

\* Corresponding author.

E-mail addresses: [B.Wang@tudelft.nl](mailto:B.Wang@tudelft.nl) (B. Wang), [P.J.Vardon@tudelft.nl](mailto:P.J.Vardon@tudelft.nl) (P.J. Vardon), [M.A.Hicks@tudelft.nl](mailto:M.A.Hicks@tudelft.nl) (M.A. Hicks), [chenzh@missouri.edu](mailto:chenzh@missouri.edu) (Z. Chen).

of an implicit MPM framework, and the validation of the resulting implicit solver, as well as with comparisons with an explicit MPM with respect to the time step size and accuracy of the results. In this paper, the term implicit MPM refers to a framework where both dynamic and quasi-static problems (with inertial terms neglected) can be solved effectively. Although implicit dynamic MPM formulations [16,17] have been reported, this paper aims to provide a clear and straightforward description of all the necessary techniques for adapting an existing FEM implementation into one based on the implicit MPM.

In the remaining sections of the paper, the theoretical formulation is first presented. This closely follows the standard FEM procedure, thereby clearly demonstrating the similarities between MPM and FEM. Implementation details are then discussed, where a special treatment for MPM is needed. The subsequent section focuses on a series of representative examples to investigate and validate the presented framework for quasi-static and dynamic analyses, respectively, with the results being compared with those obtained from an explicit code, in order to gain a thorough understanding of how the implicit algorithm behaves.

## 2. Theoretical formulation of the implicit material point method

### 2.1. General framework

To describe the implicit MPM, Fig. 1 demonstrates the standard mapping and remapping procedure between the material points and background computational mesh. In the first phase (Fig. 1(a)) the state variables are mapped from the material points to the nodes of the background mesh; in the second phase (Fig. 1(b)), the equation of motion is solved over the background mesh to find the current acceleration, with the element integration being based on the material points (rather than on the information mapped to Gauss points); and, in the third phase (Fig. 1(c)), the state variables on the material points are updated via remapping from the deformed background mesh, and the mesh is then reset, leaving the material points at their updated locations. These phases are repeated until the end of the time/loading steps.

Connectivity can be set up between the material points and background grid nodes, and thus information can be mapped back and forth between them. Due to the different ways that may be adopted for solving the equation of motion in time in the second phase, the formulation can yield either implicit or explicit MPM approaches.

### 2.2. Continuum equations

At the continuum scale, the governing differential equations under purely mechanical loading can be derived from the respective conservation equations for mass and momentum,

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

$$\rho \frac{d\mathbf{v}}{dt} = \nabla \cdot \boldsymbol{\sigma} + \mathbf{b} \tag{2}$$

supplemented with a suitable constitutive equation to describe the stress–strain relation. In Eqs. (1) and (2),  $\rho$  is the mass density,  $\mathbf{v}$  is the velocity,  $\boldsymbol{\sigma}$  is the Cauchy stress, and  $\mathbf{b}$  is the body force due to, for example, gravity.

The mass of a given material point is independent of time, and hence Eq. (1) is automatically satisfied. For Eq. (2), a derivation based on the static equilibrium between the internal force, represented by  $\boldsymbol{\sigma}$ , and external force, represented by  $\mathbf{b}$ , is introduced first for simplicity, i.e.

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = \mathbf{0} \tag{3}$$

By applying the principle of virtual displacement, followed by the use of the divergence theorem, the equilibrium equation expressed in the weak form [18] with respect to the current configuration, at time  $t + \Delta t$ , is given by

$$\int_{V^{t+\Delta t}} \mathbf{S}^{t+\Delta t} \cdot \delta \boldsymbol{\varepsilon}^{t+\Delta t} dV = \int_{V^{t+\Delta t}} \mathbf{b}^{t+\Delta t} \cdot \delta \mathbf{u}^{t+\Delta t} dV + \int_{S^{t+\Delta t}} \boldsymbol{\tau}^{t+\Delta t} \cdot \delta \mathbf{u}^{t+\Delta t} dS \tag{4}$$

where  $\mathbf{S}$  is the second Piola–Kirchhoff stress tensor,  $\delta \boldsymbol{\varepsilon}$  is the Green–Lagrange strain tensor,  $\delta \mathbf{u}$  represents the virtual displacement,  $\boldsymbol{\tau}$  denotes the prescribed part of the traction on the surface  $S$  and the volume of the body is represented by  $V$ .

Using the last known configuration at time  $t$  as a reference, the stress can be decomposed into the incremental form,

$$\mathbf{S}^{t+\Delta t} = \mathbf{S}^t + \Delta \boldsymbol{\sigma} = \boldsymbol{\sigma}^t + \Delta \boldsymbol{\sigma} \tag{5}$$

whereas the strain at time  $t + \Delta t$ , with respect to the time  $t$ , is actually the incremental strain  $\boldsymbol{\varepsilon}^{t+\Delta t} = \Delta \boldsymbol{\varepsilon}$ . The incremental strain is then divided into two parts; a linear part as commonly used in small strain analysis, plus a high order term, i.e.  $\Delta \boldsymbol{\varepsilon} = \Delta \boldsymbol{\varepsilon} + \Delta \boldsymbol{\eta}$ , in which

$$\Delta \boldsymbol{\varepsilon} = \frac{1}{2} (\nabla \bar{\mathbf{u}} + (\nabla \bar{\mathbf{u}})^T); \quad \Delta \boldsymbol{\eta} = \frac{1}{2} ((\nabla \bar{\mathbf{u}})^T \cdot \nabla \bar{\mathbf{u}}) \tag{6}$$

where  $\bar{\mathbf{u}}$  is the incremental displacement.

By substituting Eqs. (5) and (6) into the weak formulation (4) and assuming, for the moment, that the loading is deformation independent, then, by expressing the right hand side of Eq. (4) as  $\mathbf{F}_{ext}^{t+\Delta t}$ , which is the external loading accounting for the effects of both body loads and tractions, and neglecting the high order term  $\int_{V^t} \Delta \boldsymbol{\sigma} \cdot \delta \Delta \boldsymbol{\eta}$ , the small strain equation of motion in the Updated Lagrangian (UL) formulation is obtained as,

$$\int_{V^t} \Delta \boldsymbol{\sigma} \cdot \delta \Delta \boldsymbol{\varepsilon} dV + \int_{V^t} \boldsymbol{\sigma}^t \cdot \delta \Delta \boldsymbol{\eta} dV = \mathbf{F}_{ext}^{t+\Delta t} - \int_{V^t} \boldsymbol{\sigma}^t \cdot \delta \Delta \boldsymbol{\varepsilon} dV \tag{7}$$

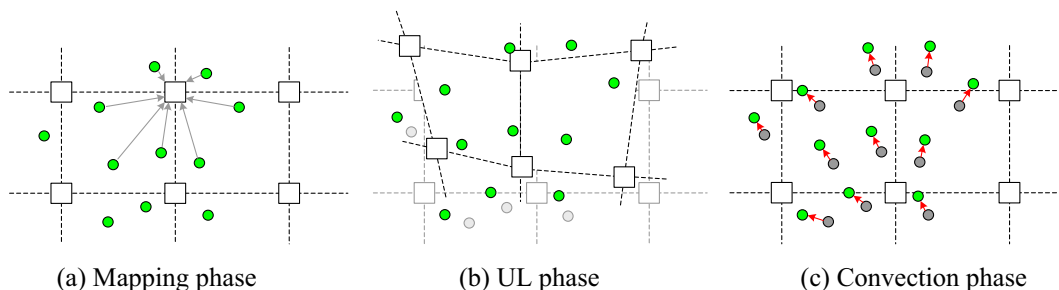


Fig. 1. Computational cycle of MPM (after Sulsky and Schreyer [3]).

To handle a large strain increment in implicit MPM, an appropriate stress measure should be used. Generally, the incremental stress can be calculated either from the second Piola–Kirchhoff stress and Green–Lagrange strain tensors (see [19]), or, as in this paper, via a rate dependant formulation using the Jaumann stress rate and velocity strain tensors [19]. For consistency, the stress and strain rates are here shown in incremental form, with the Jaumann stress rate being given by,

$$\Delta \boldsymbol{\sigma}^j = \mathbf{C} \Delta \boldsymbol{\epsilon} \quad (8)$$

where  $\Delta \boldsymbol{\sigma}^j$  is the Jaumann stress increment and  $\mathbf{C}$  is the incremental stress–strain tensor.

Following Bathe [19], the Cauchy stress increment can then be written as

$$\Delta \boldsymbol{\sigma} = \Delta \boldsymbol{\sigma}^j - \boldsymbol{\sigma}^t \cdot \Delta \boldsymbol{\omega} - \Delta \boldsymbol{\omega} \cdot \boldsymbol{\sigma}^t \quad (9)$$

where  $\Delta \boldsymbol{\omega}$  is the spin tensor (also called vorticity tensor) increment at time  $t$ , in which

$$\Delta \boldsymbol{\omega} = \frac{1}{2} [(\nabla \mathbf{u}) - (\nabla \mathbf{u})^T] \quad (10)$$

Substituting Eqs. (8) and (9) into the equation of motion for small strain, Eq. (7), the final equilibrium equation for large deformation analysis is obtained as

$$\begin{aligned} & \int_{V^t} \mathbf{C} \Delta \boldsymbol{\epsilon} \cdot \delta \Delta \boldsymbol{\epsilon} dV - \int_{V^t} (\boldsymbol{\sigma}^t \Delta \boldsymbol{\omega} + \Delta \boldsymbol{\omega} \boldsymbol{\sigma}^t) \cdot \delta \Delta \boldsymbol{\epsilon} dV + \int_{V^t} \boldsymbol{\sigma}^t \cdot \delta \Delta \boldsymbol{\eta} dV \\ & = \mathbf{F}_{ext}^{t+\Delta t} - \int_{V^t} \boldsymbol{\sigma}^t \cdot \delta \Delta \boldsymbol{\epsilon} dV \end{aligned} \quad (11)$$

### 2.3. Spatial discretisation

To solve Eq. (11) it must be spatially discretised. MPM discretises a continuum body in the original configuration into a finite set of  $N_p$  material points that are tracked throughout the deformation process. The points are selected to represent a material sub-domain and do not have a defined shape. The method solves the equations on the background mesh; hence this spatial discretisation is undertaken utilising typical finite element methodology, with the major difference being that the integration uses the material points directly.

Taking the first term in Eq. (11) as an example, which is an incrementally linear stiffness term multiplied by the unknown displacement, and using the same method as in FEM, the shape functions and nodal values of displacement are used to approximate the continuum field, i.e.  $\bar{\mathbf{u}}$ . Using the strain–displacement transformation matrix,  $\Delta \boldsymbol{\epsilon} = \mathbf{B}_L \bar{\mathbf{u}}$ , and the method of weighted residuals, the stiffness part of the term can be easily transformed into matrix form, i.e.

$$\mathbf{K}_L^t = \int_{V^t} \mathbf{B}_L^T \mathbf{C} \mathbf{B}_L dV \quad (12)$$

where  $\mathbf{B}_L$  is the linear strain–displacement transformation matrix.

The integrals of the weak form are then converted to the sums of quantities evaluated at the material points, which yields

$$\mathbf{K}_L^t = \sum_p (\mathbf{B}_L^T(\mathbf{x}_p) \mathbf{C}_p \mathbf{B}_L(\mathbf{x}_p)) V_p^t \quad (13)$$

where, for a single element,  $p$  is the number of material points within the element and the shape function differential  $\mathbf{B}_L$  is a function of the material point positions  $\mathbf{x}_p$ , which are moved after each time step. Similarly, the second and third terms in Eq. (11), referring to the non-linear contributions due to the geometry change and Jaumann stress, are written as (with the displacement part omitted),

$$\mathbf{K}_{NL}^t = \sum_p (\mathbf{B}_{NL}^T(\mathbf{x}_p) \widehat{\boldsymbol{\sigma}}_p \mathbf{B}_{NL}(\mathbf{x}_p) - \mathbf{B}_L^T(\mathbf{x}_p) \tilde{\boldsymbol{\sigma}}_p \mathbf{B}_L(\mathbf{x}_p)) V_p^t \quad (14)$$

where the nonlinear strain–displacement transformation matrix is expressed as:

$$\mathbf{B}_{NL} = \begin{bmatrix} \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \dots & \frac{\partial N_p}{\partial x} & 0 \\ \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & 0 & \dots & \frac{\partial N_p}{\partial y} & 0 \\ 0 & \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & \dots & 0 & \frac{\partial N_p}{\partial x} \\ 0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & \dots & 0 & \frac{\partial N_p}{\partial y} \end{bmatrix} \quad (15)$$

where  $\partial N_p / \partial x$  and  $\partial N_p / \partial y$  are the shape function differentials with respect to the Cartesian coordinates,  $x$  and  $y$ , respectively, for the configuration at time  $t$ .  $\widehat{\boldsymbol{\sigma}}_p$  is the Cauchy stress matrix, while the matrix  $\tilde{\boldsymbol{\sigma}}_p$ , used in the second term of the integrand of  $\mathbf{K}_{NL}$ , is defined (for plane strain problems) as:

$$\tilde{\boldsymbol{\sigma}}_p = \begin{bmatrix} 2\sigma_{11} & 0 & \sigma_{12} & 0 \\ 0 & 2\sigma_{22} & \sigma_{12} & 0 \\ \sigma_{12} & \sigma_{12} & (\sigma_{11} + \sigma_{22})/2 & 0 \\ 0 & 0 & 0 & 2\sigma_{33} \end{bmatrix} \quad (16)$$

Note that the component  $\mathbf{K}_{NL}^t$  becomes highly significant when the stresses are of the same order as the material stiffness moduli.

The internal force, i.e. the last term in the Eq. (11), at the reference time  $t$  and the external force in Eq. (4) are respectively expressed as,

$$\mathbf{F}_{int}^t = \sum_p (\mathbf{B}_L^T(\mathbf{x}_p^t) \boldsymbol{\sigma}_p) \cdot V_p^t \quad (17)$$

$$\mathbf{F}_{ext}^{t+\Delta t} = \sum_p (\mathbf{N}(\mathbf{x}_p^t) \mathbf{b}_p + \mathbf{N}(\mathbf{x}_p^t) \boldsymbol{\tau}^s) \cdot V_p^t \quad (18)$$

where  $\mathbf{N}(\mathbf{x}_p^t)$  are the shape functions at location  $\mathbf{x}_p^t$  at time  $t$ . After substituting Eqs. (13), (14), (17) and (18) into Eq. (11), the equilibrium equation in matrix form can then be formulated as:

$$\mathbf{K}^t \bar{\mathbf{u}} = \mathbf{F}_{ext}^{t+\Delta t} - \mathbf{F}_{int}^t \quad (19)$$

where  $\mathbf{K}^t = \mathbf{K}_L^t + \mathbf{K}_{NL}^t$ , and is comprised of both linear and nonlinear terms.

### 2.4. Dynamic form

A dynamic governing equation is obtained by adding an inertial term in Eq. (19), satisfying Eq. (2). Hence

$$\mathbf{K}^t \bar{\mathbf{u}} + \mathbf{M}^t \mathbf{a}^{t+\Delta t} = \mathbf{F}_{ext}^{t+\Delta t} - \mathbf{F}_{int}^t \quad (20)$$

where  $\mathbf{M}^t$  is the mass matrix at time  $t$ , which, in lumped form, is given by,

$$\mathbf{M}^t = \sum_p \mathbf{M}_p \mathbf{N}(\mathbf{x}_p^t) \quad (21)$$

where  $\mathbf{M}_p$  is the material point mass matrix, of size  $N_{dim} \times N_{dim}$  (where  $N_{dim}$  is the number of dimensions). Following Newmark's time integration scheme [20],

$$\mathbf{v}^{t+\Delta t} = \mathbf{v}^t + [(1 - \delta) \mathbf{a}^t + \delta \mathbf{a}^{t+\Delta t}] \Delta t \quad (22)$$

$$\mathbf{u}^{t+\Delta t} = \mathbf{u}^t + \mathbf{v}^t \Delta t + \left[ \left( \frac{1}{2} - \alpha \right) \mathbf{a}^t + \alpha \mathbf{a}^{t+\Delta t} \right] \Delta t^2 \quad (23)$$

where  $\Delta t$  is the time step,  $\mathbf{a}^{t+\Delta t}$ ,  $\mathbf{v}^{t+\Delta t}$  and  $\mathbf{u}^{t+\Delta t}$  are the accelerations, velocities and displacements at time  $t + \Delta t$ , and  $\alpha$  and  $\delta$  are time stepping parameters which influence the integration accuracy and

stability, and are chosen as  $\delta = 0.5$  and  $\alpha = 0.25$  in the following analyses.

The incremental displacement is given by  $\bar{\mathbf{u}} = \mathbf{u}^{t+\Delta t} - \mathbf{u}^t$ . Hence, substituting Eq. (23) into Eq. (20), and rearranging the resulting equation in the form of Eq. (19), leads to,

$$\left( \mathbf{K}^t + \frac{\mathbf{M}^t}{\alpha \Delta t^2} \right) \bar{\mathbf{u}} = \mathbf{F}_{ext}^{t+\Delta t} + \mathbf{M}^t \left( \frac{1}{\alpha \Delta t} \mathbf{v}^t + \left( \frac{1}{2\alpha} - 1 \right) \mathbf{a}^t \right) - \mathbf{F}_{int}^t \quad (24)$$

Let  $\bar{\mathbf{K}}^t$  represent a modified stiffness matrix, which takes the form  $\bar{\mathbf{K}}^t = \mathbf{K}^t + \frac{\mathbf{M}^t}{\alpha \Delta t^2}$ , and assume the new external force contains the kinetic terms from the previous time step, i.e.  $\bar{\mathbf{F}}_{ext}^{t+\Delta t} = \mathbf{F}_{ext}^{t+\Delta t} + \mathbf{M}^t \left( \frac{1}{\alpha \Delta t} \mathbf{v}^t + \left( \frac{1}{2\alpha} - 1 \right) \mathbf{a}^t \right)$ . Hence the rewritten governing equation takes the form,

$$\bar{\mathbf{K}}^t \bar{\mathbf{u}} = \bar{\mathbf{F}}_{ext}^{t+\Delta t} - \mathbf{F}_{int}^t \quad (25)$$

so that both the dynamic and quasi-static formulations can be solved in the same manner using the Newton–Raphson method.

To increase the numerical stability, it is common to include a damping force in the governing equation, which is assumed to be proportional to the nodal velocity. ‘‘Rayleigh’’ damping, by assuming the damping matrix to be a linear combination of the mass and stiffness matrices, has been widely proven to be stable and useful [21]; however, it is frequency-dependent, and therefore prior knowledge about the frequency of the system is needed. Cundall [22] described a local non-viscous damping to overcome the issue of frequency dependence, in which the damping force on a node is proportional to the magnitude of the out-of-balance force, with a directional function that ensures that vibrational modes are damped, i.e.

$$\mathbf{F}_i^{damp} = -c |\mathbf{f}_i| \text{sign}(\mathbf{v}_i) \quad (26)$$

where  $c$  is a dimensionless damping factor, here chosen to be 0.75 [23],  $\mathbf{f}_i$  are the nodal resultant forces, and  $\text{sign}(\mathbf{v}_i)$  is the velocity direction.

Therefore, by adding the damping force into Eq. (25), the final governing equation for the analysis can be formulated as,

$$\bar{\mathbf{K}}^t \bar{\mathbf{u}} = \bar{\mathbf{F}}_{ext}^{t+\Delta t} - \mathbf{F}_{int}^t + \mathbf{F}_{damp}^{t+\Delta t} \quad (27)$$

## 2.5. Update of kinematic variables

In a quasi-static analysis, after the incremental displacement is obtained by solving the equation of motion, Eq. (19), the next step is only to move the material point in accordance with the displacement of the computational grid. However, dynamic analysis involves the solution of Eq. (27) to update accelerations and velocities. Details of kinematic variable updating are therefore presented in this sub-section.

Corresponding to the three MPM phases illustrated in Fig. 1, the kinematic variables need to be updated three times within the MPM computational cycle.

### 2.5.1. Mapping phase

Since information, such as velocity and acceleration, is initially stored at the material points and the background mesh is reset regularly after each time step, it is necessary to map (i.e. transfer) the associated kinematic information from the material points to the grid nodes at the start time  $t$  of each time step by utilising the shape functions, i.e.

$$\mathbf{v}_i^t = \frac{\sum_p N_i(\mathbf{x}_p^t) m_p \mathbf{v}_p^t}{m_i^t} \quad (28)$$

in which  $i$  refers to a grid node,  $p$  refers to material points surrounding the grid node,  $m_p$  is the material point mass,  $\mathbf{v}_p$  is the material

point velocity and  $m_i$  is the node mass which is assembled from the material points within the adjacent elements, see Eq. (21).

The acceleration is updated in the same way, with the displacement being initialised to zero at the beginning of each time step, i.e.  $\mathbf{u}_i^t = \mathbf{0}$ .

### 2.5.2. Updated-Lagrangian phase

For this computational stage, Newton’s method is used to calculate the incremental displacement  $\bar{\mathbf{u}}_i$  (Eq. (27)). Solving for the velocity and acceleration vectors, using Eqs. (22) and (23) with  $\delta = 0.5$  and  $\alpha = 0.25$ , the following is obtained:

$$\mathbf{v}_i^{t+\Delta t} = \frac{2}{\Delta t} \bar{\mathbf{u}}_i - \mathbf{v}_i^t \quad (29)$$

$$\mathbf{a}_i^{t+\Delta t} = \frac{4}{\Delta t^2} \bar{\mathbf{u}}_i - \frac{4}{\Delta t} \mathbf{v}_i^t - \mathbf{a}_i^t \quad (30)$$

### 2.5.3. Convection phase

The final stage is to map the information from the grid nodes back to the material points. The acceleration and displacement vectors are directly mapped from the grid nodes using the shape functions, i.e.

$$\mathbf{a}_p^{t+\Delta t} = \sum_{i=1}^{N_n} \mathbf{a}_i^{t+\Delta t} N_i(\mathbf{x}_p^t); \quad \bar{\mathbf{u}}_p^{t+\Delta t} = \sum_{i=1}^{N_n} \Delta \bar{\mathbf{u}}_i^{t+\Delta t} N_i(\mathbf{x}_p^t) \quad (31)$$

where  $N_n$  is the number of grid nodes which provide support to the material point and in this case is the number of nodes in an element (local support), while, in cases of utilising generalised interpolation MPM [24], the nodes of surrounding elements can also be included to provide a non-local support so that the cell-crossing error could be reduced. The trapezoidal rule is used here to update the velocity,

$$\mathbf{v}_p^{t+\Delta t} = \mathbf{v}_p^t + \frac{1}{2} (\mathbf{a}_p^t + \mathbf{a}_p^{t+\Delta t}) \Delta t \quad (32)$$

## 2.6. Explicit MPM

To compare the results and illustrate the performance of the implicit MPM presented in this paper, a version of the code utilising explicit time integration has also been produced, following the approach by Sulsky et al. [1,3], where the discrete form of the momentum equation (Eq. (2)) becomes

$$\mathbf{M}^f \mathbf{a}^{t+\Delta t} = \mathbf{F}_{ext}^{t+\Delta t} - \mathbf{F}_{int}^t \quad (33)$$

Once the solution for the nodal acceleration is obtained, other kinematic variables, i.e., velocities and displacements, are advanced explicitly.

Furthermore, to improve the algorithmic stability of the explicit MPM, the authors follow Sulsky et al. [3], who proposed a ‘‘modified update stress last’’ (MUSL) approach, in which the updated material point momentum vector is used to calculate the nodal velocities and thereby update the material point strain and stress rates. The explicit MPM version is herein applied only as a comparison, and therefore detailed aspects are not discussed.

## 3. Numerical examples

In this section, numerical examples for both quasi-static and dynamic analyses are provided. Firstly, a cantilever beam is considered as a quasi-static case to illustrate how the algorithm behaves. In particular, it is shown that, by adding an extra stiffness to the background mesh, known as a ‘‘soft stiffness’’, the numerical stability of the solution is improved when elements exist that are not fully filled by material points. Moreover, the influence of the num-

ber of material points per element on the numerical results is discussed. Secondly, the collision between disks has been investigated. This is a typical dynamic test case in MPM, and has been investigated by other researchers [17,25]. Here it is considered for a direct comparison between the implicit and explicit MPM codes. Finally, a geotechnical application, using the dynamic MPM, is considered, which investigates the effect of an excavation on slope instability and post-failure behaviour.

### 3.1. Cantilever beam

A 2-D (plane strain) linear elastic cantilever beam has been analysed, and three aspects investigated: (a) the algorithm to initialise the positions of the material points; (b) the influence of the initial number of material points per element; (c) the influence of the extra “soft stiffness” used across the background mesh.

The beam, as shown in Fig. 2, has a length of 1 m, a depth of 0.3 m, a Young’s modulus of 100 kPa and a Poisson’s ratio of 0. It is built-in along its left edge and its initial configuration is shown in grey, representing the originally filled background mesh. The self-weight was increased from 0 kN/m<sup>3</sup> to 4 kN/m<sup>3</sup> in 20 loading steps, with each applied increment being equal to 0.2 kN/m<sup>3</sup>. The analysis used 8-noded quadrilateral elements, with 16 material points being initially located in each element.

In order to increase the numerical stability, a “soft stiffness”, as described by Lim [26], has been assigned to the background mesh, which has been assembled using conventional Gauss point integration of the mesh (i.e. independently of the material points). The ratio of the soft stiffness to the actual stiffness is represented by  $\eta$ , where  $\eta = 0.01$  is applied in this case.

The final configuration of the beam is also shown in Fig. 2, with the tip displacement (taken from the tracked material point located in the bottom right corner) being  $-0.38$  m and  $-0.61$  m, respectively, in the horizontal and vertical directions. Accordingly, the final activated mesh/elements are shown as a wireframe, which was determined by tracing the material point positions. The colouration of the material points represents the longitudinal stress along the beam, with blue representing compression and red representing tension, ranging from  $-40.8$  kPa (compression) to  $27.6$  kPa (tension) on the material points located nearest the fixed end. By using extrapolation, the stresses at the element nodes

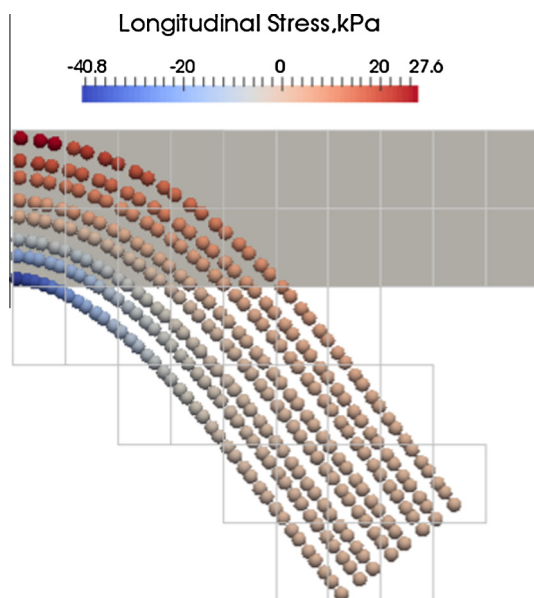


Fig. 2. Initial and final deformed configurations.

can be obtained, ranging from  $-47.5$  kPa to  $32.0$  kPa. UL FEM solutions, which are considered to be capable of providing accurate results for large displacement, but small strain, problems [14], are provided here as a direct comparison, where, for the utilised  $10 \times 2$  element mesh, the stress range of  $-47.4$  kPa to  $32.6$  kPa at the fixed end is in good agreement with the MPM solution.

Due to the use of high order elements in the quasi-static code, the potential problems associated with using linear elements, i.e. locking, cell crossing [24], etc., are avoided. However, by omitting the inertial terms, the code stability decreases, which leads to the need for more material points and a “soft stiffness” being used. After a brief description of the method for initialising the material points, the influences of the number of material points and “soft stiffness” are investigated in the following.

#### 3.1.1. Subdivision algorithm

There are many potential ways to initialise the locations of the material points, such as evenly spaced over the area and using Gauss point positions. Here, the subdivision algorithm of Lim [26] is chosen. For example, an 8-noded quadrilateral parent element with 4 Gauss points is adopted initially, and then, for obtaining more material points, the parent element is subdivided into smaller cells; e.g. into 2 by 2 cells, and then, by placing the material points on the local Gauss point positions of the small cells, to give a total of 16 material points for the original finite element.

#### 3.1.2. Soft stiffness influence

The addition of an extra small stiffness across the background mesh is to increase the numerical stability of the code. For example, as a single material point moves near a finite element boundary, the shape function (at the material point) corresponding to the farthest element node may approach zero, while the gradient of the shape function will not be equal to zero; hence it is very hard to get the resultant force on the farthest node converging to zero, thereby leading to a non-convergent analysis. For implicit MPM, the problem normally occurs during the stiffness matrix formation, or factorisation, and can lead to an extremely large/unrealistic solution being output, as illustrated by the near vertical dotted line in Fig. 3. In the second loading step, the code stops due to an extremely large displacement at the beam tip. In contrast, even with a very “soft”/small background stiffness, the outputs are realistic. Results obtained with four different stiffness ratios, i.e. ratio of the soft stiffness to the actual stiffness, of 0.1, 0.01, 0.001 and 0.0001, have been included here, in order to gain a picture of the influence of the “soft stiffness”. 16 material points are initially placed in each element.

For comparisons with the UL-FEM solution, the vertical tip displacement of the beam (which is taken to be the average vertical displacement of the right-most column of material points near

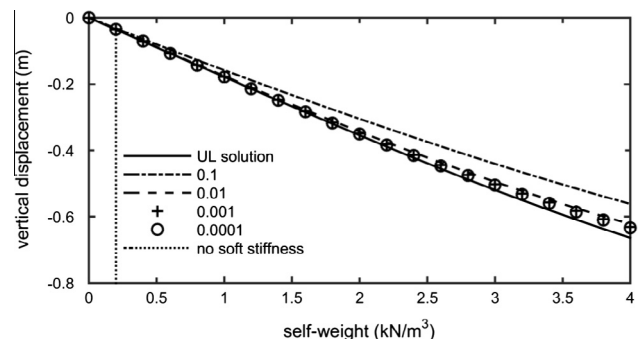


Fig. 3. Applied load versus tip vertical displacement for different ratios of soft stiffness.

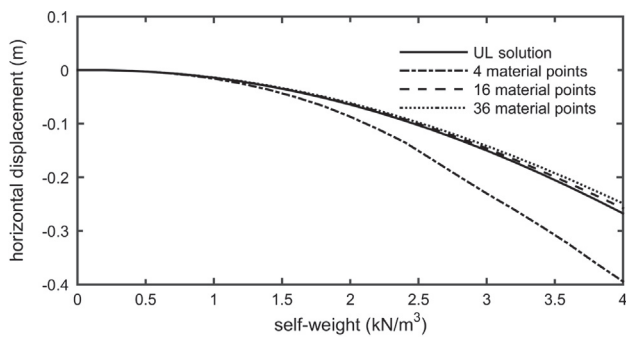
the beam tip) is shown in Fig. 3. It is seen that, the higher the stiffness ratio, the greater the error. When the ratio is increased to 0.1, at the final weight of 4 kN/m<sup>3</sup> the error relative to the UL solution reaches 15%, although, for ratios equal to or smaller than 0.01, there is only a small difference in the results. Meanwhile, as the stiffness ratio increases, the code gets more stable. Therefore, for the beam case considered here, a ratio of 0.01 is adopted for all other simulations.

### 3.1.3. Influence of the number of material points per element

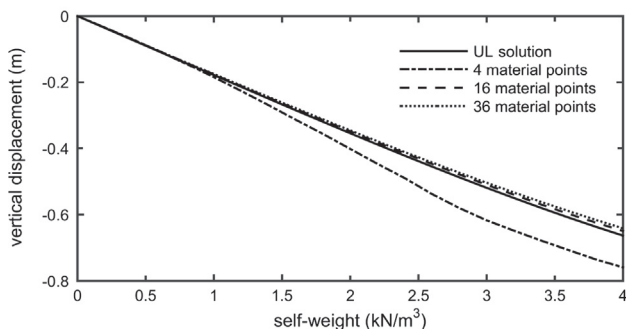
By using the subdivision algorithm introduced above, three different numbers of material points per element (in the initial mesh) have been considered, i.e. 4, 16 and 36. The influence of the number of material points has been investigated by comparing the averaged material point displacements across the beam tip with the UL-FEM results. Fig. 4 shows the relationship between the self-weight and the recorded displacement ((a) horizontal and (b) vertical). It is seen that, with initially 4 material points per element, there is a large divergence from the UL-FEM result, whereas, as the number increases, the differences become negligible, i.e. for 16 and 36 material points per element.

By putting more material points within an element, it increases the possibility of material points being distributed over the mesh more evenly, and thus it increases the code stability. However, as the number increases, the computation cost goes up. Hence, a balance has to be made, and here, initially 16 material points per 8-noded element are considered appropriate to analyse the beam.

In terms of generalising the results in this section, it is difficult to determine a priori the appropriate level of discretisation in terms of both the number of elements and number of material points per element. As with the number of finite elements, good numerical modelling practice is recommended to ensure spatial convergence of the results, i.e. by increasing the level of discretisation and checking the consistency of the results. In the following two examples the presented results are for levels of discretisation that have been checked in this way.



(a) Horizontal displacement curves



(b) Vertical displacement curves

Fig. 4. Influence of particle density on load–displacement response.

### 3.2. Collision between two elastic disks

For analysing the collision between two elastic disks, the implicit dynamic method is utilised. Two things are worthy of note: (a) due to the presence of the mass matrix, in which negative terms may arise for high-order elements, low-order elements are adopted in the dynamic analyses (i.e. a 4-node quadrilateral element in this paper); (b) as the inertial term is included, initially 4 material points per element are found to be enough for ensuring accuracy when carrying out dynamic analyses, and an extra “soft stiffness” is not necessary.

Two disks with radii equal to 0.2 m are simulated, assuming plane strain conditions. Initially, the two disks are located in the lower left and upper right corners of a square background mesh, as shown in Fig. 5(a). The mesh comprises 400 equal-sized square elements of side length equal to 0.05 m. Initial velocities (1.0 m/s, 1.0 m/s) and (−1.0 m/s, −1.0 m/s) are assigned to the two disks, so that they move towards each other along the diagonal of the square. The Young’s modulus is 100.0 kPa, the Poisson’s ratio is 0.3 and the density is 1.0 kg/m<sup>3</sup>. The simulation is run to a final time of  $t = 37.5$  s.

Fig. 5 shows the collision process of the two disks, in which the colouration represents the mean stress distribution within the disks. In Fig. 5(a), the initial positions of the two disks are shown. Before impact, due to the constant velocities assigned to the material points, no velocity gradient will be experienced during this phase, and hence no stress magnitudes are shown on the disks. Fig. 5(b) shows the simulation during the impact, revealing that the material points near the contact region have higher compressive stresses, as would be expected. Finally, Fig. 5(c) shows the disks after impact, where the velocities are now in the opposite direction to those before the impact. Small residual stresses can be observed on the disks, due to the free vibration of the disks after the impact.

The energy conservation errors of the system were tracked, as a function of time, throughout the analysis. At time  $t + \Delta t$ , the total energy of the system can be expressed as

$$E_{tot}^{t+\Delta t} = E_{kin}^{t+\Delta t} + E_{strain}^{t+\Delta t} \quad (34)$$

where  $E_{kin}$  is the kinetic energy and  $E_{strain}$  is the strain energy stored in the material points. These are defined as

$$E_{kin}^{t+\Delta t} = \frac{1}{2} \sum_p m_p (v_p^{t+\Delta t})^2 \quad (35)$$

$$E_{strain}^{t+\Delta t} = E_{strain}^t + \sum_p v_p^{t+\Delta t} \frac{\sigma_p^t + \sigma_p^{t+\Delta t}}{2} : \Delta \varepsilon_p \quad (36)$$

Fig. 6 shows the system energy conservation errors with time for different applied time steps, where the initial velocities were changed to 0.005 m/s in order to simulate a longer time during which the two disks remain in contact. Fig. 6(a) shows the results using the explicit MPM, where the time step was chosen to be  $\Delta t = 0.0025$  s by applying the Courant–Friedrichs–Lewy (CFL) condition (CFL is a necessary condition for stability with explicit time integration schemes). The results obtained with the implicit MPM are provided in Fig. 6(b)–(d), where time steps were chosen to be the same, 10 times and 100 times that of the explicit code, respectively.

For the cases shown in Fig. 6(a)–(c), the energy is well preserved before and after the impact. Before impact, the material points comprising the disks are moving with uniform velocities, and no strains or stresses are generated; hence the energy is purely due to the uniform movement of the disks. During the impact, the kinetic energy decreases to zero, with the strain energy reaching its maximum value at the point of maximum deformation. After

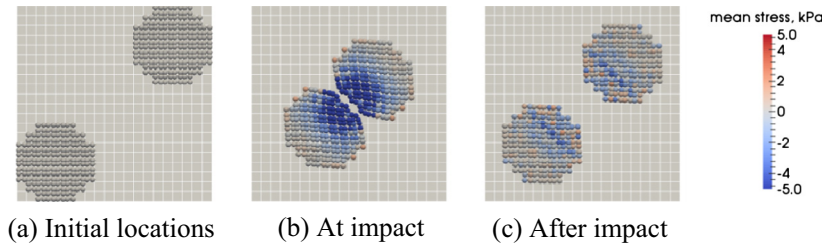
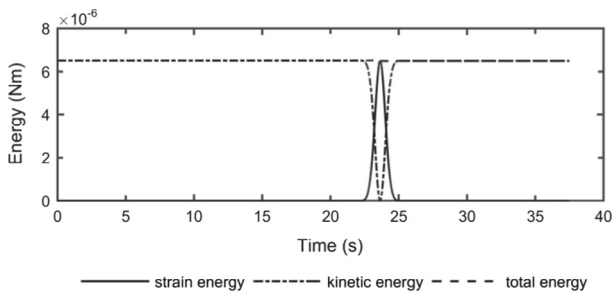
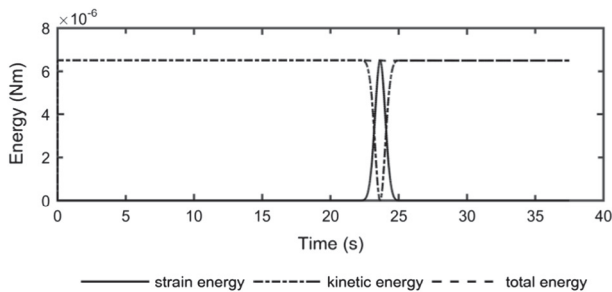


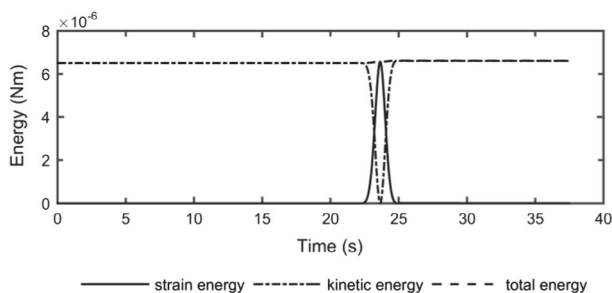
Fig. 5. Collision process of two elastic disks.



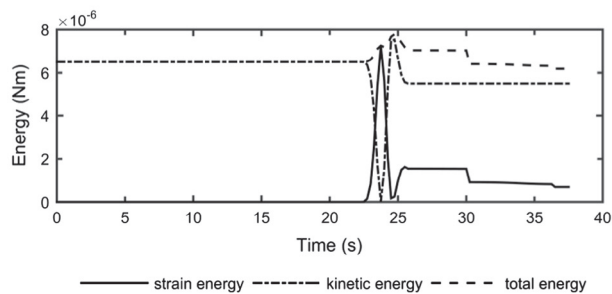
(a) Explicit MPM with  $\Delta t = 0.0025s$



(b)  $\Delta t = 0.0025s$



(c)  $\Delta t = 0.025s$



(d)  $\Delta t = 0.25s$

Fig. 6. Energy conservation errors for elastic disk collision.

that, the disks start to bounce back from each other and, at the same time, the deformation is recovered, leading to a decrease in the strain energy and a resurgence of kinetic energy. After the impact, a small amount of strain energy remains, which is due to the free vibration of the disks after separation.

Fig. 6(d) shows the energy conservation errors corresponding to a time step size of  $\Delta t = 0.25$  s, i.e. 100 times the explicit  $\Delta t$ . It is seen that the energy not conserved is severe, although the material point trajectories were not much affected. As pointed out by others, (e.g. [21]), implicit integration schemes such as the trapezoidal rule are unconditionally stable in linear analyses, which enables the code to work at extremely large time steps. However, for the problem presented herein, there is a bound on the implicit time step imposed by the accurate resolution of the collision, known as the characteristic collision time, i.e. the time that it takes for a wave to traverse the disk [17]. In this example, the wave speed is about 10.0 m/s, and the disk diameter is approximately 0.4 m, thus giving the characteristic collision time to be about 0.04 s, i.e. 16 times the explicit time step size. It is worth noting that a reduction in the required number of time steps would typically result in a reduction in the computational time. However, as the solution scheme is different in the two approaches, that is, using an implicit solver (of which a number of types are available) in implicit integration schemes compared to explicit solution schemes where the new result can be simply calculated, it means that a unique direct relationship cannot be established. Moreover, as demonstrated in the above example, the specific characteristics of the analysis (in this case the wave speed and disk diameter) affect possible reductions in the number of time steps required.

### 3.3. Slope collapse under the influence of excavation at the slope toe

Excavation at the toe of a slope can trigger the collapse of the slope. As well as changing the slope geometry, it may also expose significant geologic features such as shear zones, faults and folds in some circumstances, thereby causing slope instability.

To investigate the effect of excavation on a slope, a naturally stable slope was first considered, as shown in Fig. 7, and then a vertical cut of height 0.2 m was excavated at the slope toe, as indicated. The problem is assumed to be plane strain, and the

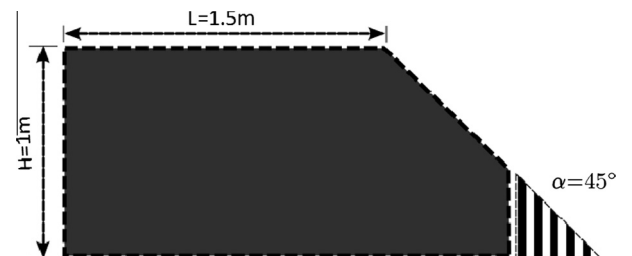


Fig. 7. Sketch of an excavation undertaken at the toe of a slope.



computational grid is made up of 4-noded square elements of 0.05 m side length, with each element initially containing 4 material points. The height of the slope is 1 m and the slope angle is 45°. Roller boundaries are prescribed at the left side of the domain to allow only vertical displacements. The interaction between the slope base and the ground is modelled using a Coulomb frictional boundary, which allows the slope base to move horizontally when friction is overcome.

A linear elastic and cohesion softening Von Mises model was used to describe the soil behaviour, as shown in Fig. 8. The initial yield stress  $\tau_p$  is defined by a peak shear strength of  $c_p = 5.0$  kPa, whereas the residual strength  $\tau_r$  is given by  $c_r = 1.0$  kPa. The softening modulus is  $H = dc/d\bar{\epsilon}_p = -50.0$  kPa, in which the plastic strain invariant  $\bar{\epsilon}_p$ , in its incremental form, is defined as  $d\bar{\epsilon}_p = \sqrt{\frac{2}{3}} d\epsilon_p : d\epsilon_p = \sqrt{\frac{2}{3}} \|d\epsilon_p\|$ , and  $d\epsilon_p$  is the plastic strain increment, normal to the plastic potential surface. The Young's modulus and Poisson's ratio of the soil are  $E = 200$  kPa and  $\nu = 0.33$ , respectively. Since a local softening model is used here for demonstrating the proposed solution procedure, the numerical results are mesh-dependent. However, as the proposed method is based on a FEM formulation, the same regularisation techniques as used in FEM [27] can be used to address this issue.

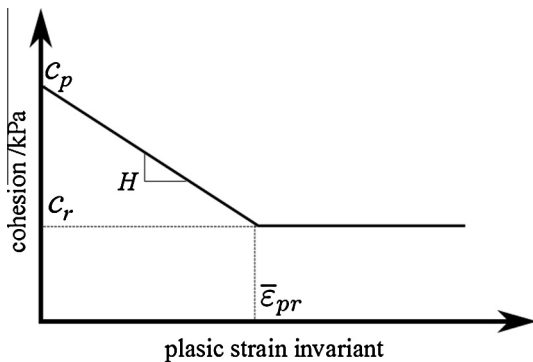


Fig. 8. Sketch of the implemented cohesion softening model.

The dynamic implicit MPM was utilised in this case. The computation was divided into two stages. Initially, the self-weight of  $\gamma = 20$  kN/m<sup>3</sup> is applied to each material point in the slope in one step to generate the initial total stresses. Quasi-static equilibrium is detected by using the criterion based on the energy and force ratio [23], with the out of balance force ratio expressed as,

$$\varepsilon = \frac{\|F^{ext} - F^{int}\|}{\|F^{ext}\|} \tag{37}$$

and a dimensionless energy ratio as,

$$\delta = \frac{E_{kin}}{W^{ext}} \tag{38}$$

where  $E_{kin}$  denotes the kinetic energy of the system,  $W^{ext}$  is the work induced by the external force, and a tolerance of 0.01 was used for both criteria. To obtain faster convergence in the quasi-static equilibrium stage, a local damping of  $c = 0.75$  was used. Secondly, the triangular soil block was “excavated” instantly to trigger the slope collapse. The collapse process is illustrated in Fig. 9, with the colouration representing the accumulated plastic strain  $\bar{\epsilon}_p$ .

Due to the toe excavation, the slope became unstable and a slope slide was triggered. In Fig. 9(a), a complete shear band starting from the new slope toe was formed. Fig. 9(b) shows that the soil above the shear band moves as a block along the failure surface, and the final quasi-static configuration is displayed.

This analysis illustrates the ability of the proposed method to simulate geotechnical behaviour at large strains, which generally occur in arbitrary directions, i.e. not aligned with the mesh. The consequence of a geotechnical instability can then be better quantified, as can further potential unstable situations be observed, for example, in retrogressive and progressive failures, which will be investigated in future publications.

#### 4. Conclusions

An implicit material point method (IMPM) framework has been developed for both quasi-static and dynamic analyses. The improved characteristics in solving large-deformation elasto-

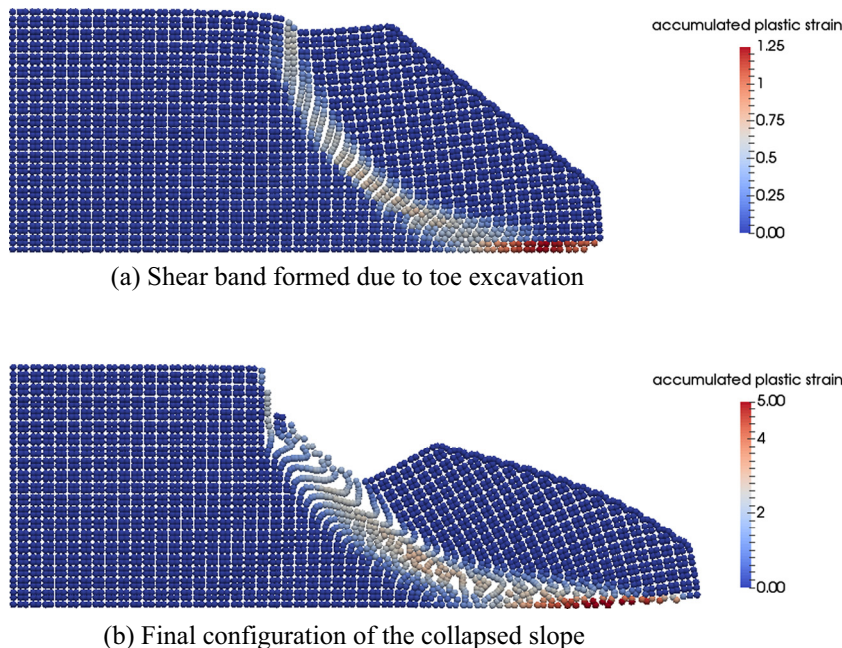


Fig. 9. Evolution of slope failure due to excavation at the toe.

plastic problems as compared to the finite element method (FEM) are highlighted. The paper provides a detailed description of the IMPM formulation and numerical implementation, following, where possible, standard FEM methods, thereby enabling the adaptation of an FEM code to an MPM code in a relatively straightforward manner.

Three numerical examples of increasing complexity are provided so as to clarify and illustrate the implementation procedures, performance and behaviour in a step by step manner, including features particular to MPM, e.g. the number of material points per element and a background “soft stiffness”. The cantilever beam problem represents a quasi-static case, and is used to demonstrate various details of the MPM, including spatial discretisation, the influence of using a “soft stiffness” for the background mesh, and the influence of the number of material points per element. The dynamic case is illustrated via the disk collision example, where comparisons with an explicit MPM are provided to identify the advantage of using IMPM over explicit MPM in terms of the chosen time step size. Finally, an analysis of slope collapse under the influence of toe excavation shows that IMPM is a useful tool to capture post-failure behaviour in geotechnical applications. For large deformation analyses, however, cell-crossing errors are an important issue to be addressed. Recent efforts have been made by other research teams, to reduce the cell-crossing errors and to better combine MPM with FEM for modern engineering applications [28–30]. Future work will be conducted to improve IMPM for general applications.

### Acknowledgments

The authors wish to acknowledge the China Scholarship Council and the Geo-Engineering Section of Delft University of Technology for financial support of the first author, the partial support by a Marie Curie Career Integration Grant, No. 333177, for the second author, and partial support by the Natural Science Foundation of China under Grant No. 11232003 for the fourth author.

### References

- [1] Sulsky D, Chen Z, Schreyer HL. A particle method for history-dependent materials. *Comput Methods Appl Mech Eng* 1994;118:179–96.
- [2] Sulsky D, Zhou SJ, Schreyer HL. Application of a particle-in-cell method to solid mechanics. *Comput Phys Commun* 1995;87:236–52.
- [3] Sulsky D, Schreyer HL. Axisymmetric form of the material point method with applications to upsetting and Taylor impact problems. *Comput Methods Appl Mech Eng* 1996;139:409–29.
- [4] Wieckowski Z. A particle-in-cell solution to the silo discharging problem. *Int J Numer Meth Eng* 1999;45:1203–25.
- [5] Wieckowski Z. The Material Point Method in large strain engineering problems. *Comput Methods Appl Mech Eng* 2004;193:4417–38.
- [6] Henderson TC, McMurtry PA, Smith PJ, Voth GA, Wight CA, Pershing DW. Simulating accidental fires and explosions. *Comput Sci Eng* 2000;2:64–76.
- [7] Hu W, Chen Z. Model-based simulation of the synergistic effects of blast and fragmentation on a concrete wall using the MPM. *Int J Impact Eng* 2006;32:2066–96.
- [8] Guilkey JE, Hoying JB, Weiss JA. Computational modelling of multicellular constructs with the material point method. *J Biomech* 2006;39:2074–86.
- [9] Coetzee CJ, Vermeer PA, Basson AH. The modeling of anchors using the material point method. *Int J Numer Anal Meth Geomech* 2005;29:879–95.
- [10] Anderson S, Anderson L. Analysis of stress updates in the material point method. In: *Proceedings of the 22nd Nordic seminar on computational mechanics*, Denmark; 2009. p. 129–34.
- [11] Kumar K, Soga K, Delenne JY. Multi-scale modelling of granular avalanches. In: *AIP conf proc*; 2013. p. 1250–3.
- [12] Shin WK. Numerical simulation of landslides and debris flows using an enhanced material point method. PhD thesis. University of Washington; 2009.
- [13] Mast CM. Modelling landslide-induced flow interactions with structures using the material point method. PhD thesis. University of Washington; 2013.
- [14] Beuth L, Wieckowski Z, Vermeer PA. Solution of quasi-static large-strain problems by the material point method. *Int J Numer Anal Meth Geomech* 2011;35:1451–65.
- [15] Wang B, Vardon PJ, Hicks MA. Implementation of a quasi-static material point method for geotechnical applications. In: *Proceedings of 3rd international symposium on computational geomechanics*, Krakow, Poland; 2013. p. 305–13.
- [16] Guilkey JE, Weiss JA. Implicit time integration for the material point method: quantitative and algorithmic comparisons with the finite element method. *Int J Numer Meth Eng* 2003;57:1323–38.
- [17] Sulsky D, Kaul A. Implicit dynamics in the material point method. *Comput Methods Appl Mech Eng* 2004;193:1137–70.
- [18] Beuth L, Benz T, Vermeer PA, Wieckowski Z. Large deformation analysis using a quasi-static Material Point Method. *J Theor Appl Mech* 2007;38:45–60.
- [19] Bathe KJ. *Finite element procedures*. 2nd ed. New Jersey: Prentice Hall; 1996.
- [20] Newmark NM. A method of computation for structural dynamics. *J Eng Mech, ASCE* 1959;85:67–94.
- [21] Smith IM, Griffiths DV, Margetts L. *Programming the finite element method*. Wiley; 2013.
- [22] Cundall PA. Distinct element models of rock and soil structure. In: Brown, ET, editor. *Analytical and computational methods in engineering rock mechanics*, Allen and Unwin, London; 1987. p. 129–63.
- [23] Al-Kafaji IKJ. Formulation of a dynamic material point method (MPM) for geomechanical problems. PhD thesis. Stuttgart University; 2013.
- [24] Bardenhagen SG, Kober EM. The generalised interpolation material point method. *Comput Model Eng Sci* 2004;5(6):477–96.
- [25] Buzzi O, Pedroso DM, Giacomini A. Caveats on the implementation of the generalized material point method. *Comput Model Eng Sci* 2008;1:1–21.
- [26] Lim LJ. Pile penetration simulation with material point method. Msc Thesis. Delft University of Technology; 2012.
- [27] De Borst R, Sluys LJ, Mühlhaus H-B, Pamin J. Fundamental issues in finite element analyses of localisation of deformation. *Eng Comput* 1993;10:99–121.
- [28] Lian YP, Zhang X, Liu Y. Coupling of finite element method with material point method by local multi-mesh contact method. *Comput Methods Appl Mech Eng* 2011;200(47):3482–94.
- [29] Sadeghirad A, Brannon RM, Burghardt J. A convected particle domain interpolation technique to extend applicability of the material point method for problems involving massive deformations. *Int J Numer Meth Eng* 2011;86:1435–56.
- [30] Zhang DZ, Ma X, Giguere PT. Material point method enhanced by modified gradient of shape function. *J Comput Phys* 2011;230:6379–98.