# 4. Material Point Method: overview and challenges ahead

Wojciech T. **Sołowski**, Aalto University, wojciech.solowski@aalto.fi

*Department of Civil Engineering, School of Engineering, Aalto University*
*P.O. Box 11000 (Otakaari 1B), FI-00076 AALTO, Finland*

Martin **Berzins**, University of Utah, mb@sci.utah.edu

*Scientific Computing and Imaging Institute, University of Utah, Salt Lake City, UT 84112 USA*

William M. **Coombs**, Durham University, w.m.coombs@durham.ac.uk

*Department of Engineering, Durham University, Lower Mountjoy, South Road, Durham, DH1 3LE, UK*

James E. **Guilkey**, University of Utah, James.Guilkey@utah.edu

*Mechanical Engineering, University of Utah, Salt Lake City, UT, USA*

Matthias **Möller**, Delft University of Technology, M.Moller@tudelft.nl

*Delft University of Technology, Department of Applied Mathematics, Van Mourik Broekmanweg 6, 2628 XE Delft, The Netherlands*

Quoc Anh **Tran**, formerly Aalto University, tranquocanh90@gmail.com

*Department of Civil Engineering, School of Engineering, Aalto University*

*P.O. Box 11000 (Otakaari 1B), FI-00076 AALTO, Finland*

Tito **Adibaskoro**, Aalto University, tito.adibaskoro@aalto.fi

*Department of Civil Engineering, School of Engineering, Aalto University*

*P.O. Box 11000 (Otakaari 1B), FI-00076 AALTO, Finland*

Seyedmohammadjavad **Seyedan**, Aalto University, seyedmohammadjavad.seyedan@aalto.fi

*Department of Civil Engineering, School of Engineering, Aalto University*

*P.O. Box 11000 (Otakaari 1B), FI-00076 AALTO, Finland*

Roel **Tielen** , Delft University of Technology, R.P.W.M.Tielen@tudelft.nl

*Delft University of Technology, Department of Applied Mathematics, Van Mourik Broekmanweg 6, 2628 XE Delft, The Netherlands*

Kenichi **Soga**, University of California, Berkeley, soga@berkeley.edu

*Department of Civil and Environmental Engineering, University of California Berkeley, Berkeley, USA*

**Abstract**

The paper gives an overview of Material Point Method and show its evolution over the last 25 years. The Material Point Method developments followed a logical order. The article aims at identifying this order and show not only the current state of the art, but explain the drivers behind the developments and identify what is currently still missing.

The paper explores modern implementations of both explicit and implicit Material Point Method. It concentrates mainly on uses of the method in engineering, but also gives a short overview of Material Point Method application in computer graphics and animation. The article also gives overview of errors in the material point method algorithms, as well as identify gaps in knowledge, filling which would hopefully lead to a much more efficient and accurate Material Point Method. The paper also briefly discusses algorithms related to contact and boundaries, coupling the Material Point Method with other numerical methods and modelling of fractures. It also gives an overview of modelling of multi-phase continua with Material Point Method. The paper closes with numerical examples, aiming at showing the capabilities of Material Point Method in advanced simulations. Those include landslide modelling, multiphysics simulation of shaped charge explosion and simulations of granular material flow out of a silo undergoing changes from continuous to discontinuous and back to continuous behaviour.

The paper uniquely illustrates many of the developments not only with figures, but also with videos, giving the whole extend of simulation instead of just a timestamped image.

**Keywords: Material Point Method, MPM, GIMP, CPDI, numerical modelling**

# 1. Introduction

Material Point Method (MPM) has been developed by Sulsky et al. (1994, 1995) as an extension of FLuid Implicit Particle (FLIP) method (Harlow, 1963, 1988) to solve standard continuum solid mechanics problems in time. MPM is a robust method, especially capable of computing problems with very large deformation, poorly suited for other numerical methods based on continuum mechanics like Finite Element Method (FEM) or Finite Difference Method. On the other hand, unlike other particle based methods, such as Discrete Element Method (DEM) or Smoothed Particles Hydrodynamics that are well suited for problems with large displacements, MPM uses the continuum mechanics framework. Therefore, all the constitutive models implemented in FEM codes can be quite easily transferred into MPM codes, allowing for solving the problems that fail to converge in the FEM simulations. This is currently the main niche for MPM – simulations with large displacements, which are difficult to solve with other methods based on continuum mechanics due to mesh distortion and other algorithmic problems. On the other hand, the accuracy increase with grid density and the number of material points is currently lower than that of FEM with high order elements. Therefore, MPM simulations require more computation time than those made with FEM for equal numerical accuracy, which prevents MPM from competing with FEM in standard calculations.

The current MPM developments aim to improve the method in terms of (i) accuracy, by removing method specific errors and increasing convergence rate, (ii) versatility, by improving the ability to model, among others, cracks and material flows, and by introducing more accurate and versatile boundaries and contact algorithms and (iii) usability, by introducing coupling with FEM and other numerical methods, as well as extending the method to modelling coupled Thermo-Hydro-Mechanical problems. All the developments preserve the ability of the method to give results even in the case of rapidly occurring very large displacements and

strains, often referred to as robustness . That is perhaps the most pronounced advantage of MPM over other numerical methods based on mechanics of continua.

MPM discretises the initial problem with material points, also referred to in the text of this paper as particles. The material points initially did not have any physical dimensions (Sulsky et al., 1994), but in further developments they gained deformable domains (Bardenhagen & Kober, 2004), which later become fully deformable also in shear deformations (Sadeghirad et al., 2011, 2013). The material points are cast on a computational grid. In each step of calculations (in explicit MPM, real time step) the momentum from the material points is transmitted to the grid nodes. That is done with an integral of functions being a conjunction of particle characteristic functions (describing where the particle domain is) and grid shape functions (describing how the properties from given grid cell point should be transmitted to the grid nodes). At the grid nodes, the sum of the momenta is computed. Afterwards, the data from the nodes is moved back to particles with the reversed procedure. The algorithm conserves energy accurately. The inaccuracies in calculations arise due to (i) simple choice of shape functions (current implementations use 3 or 4 node grid cells), (ii) time integration and (iii) other errors which are related, for example, to non-optimal number and positioning of material points, as well as material points crossing cells boundaries.

## 2. Explicit formulations of the Material Point Method

Main contributors: Sołowski, Tran, Berzins, Guilkey, Möller, Tielen

The Material Point Method original formulation solves the variational form of conservation of momentum equation over domain $\Omega$ with boundary $\Omega_\tau$:

$$\int_\Omega \rho \mathbf{a} \cdot \delta \mathbf{v} d\Omega + \int_\Omega \boldsymbol{\sigma} : \nabla \delta \mathbf{v} d\Omega = \int_\Omega \rho \mathbf{b} \cdot \delta \mathbf{v} d\Omega + \int_{\Omega_\tau} \boldsymbol{\tau} \cdot \delta \mathbf{v} d\Omega_\tau \tag{1}$$

where **a** is the acceleration, **b** is the body forces, **σ** is the stress tensor, **τ** is the forces acting on body boundaries $\Omega_t$, $\delta\mathbf{v}$ is the admissible velocity field (variational test function) and $\rho$ is the material density. Equation 1 is satisfied in each time step at each grid node, with all the material points with the domains in the grid cells sharing the node contributing. The MPM algorithms differ in how they compute the integrals at the nodes basing on the data contained in material points. However, the general algorithm of the method remains the same.

MPM stores all the computational quantities, including history variables, at the material points, with grid and grid nodes being just used for computations in each time step. Therefore, at the beginning of the algorithm step, all the data, including mass, velocity, stress and strain, as well as – possibly – domain size and shape, are at the material points.

### 1. 2.1 Initial discretisation

To initially discretise the problem, the domain **Ω** is divided into material points, such that the sum of material point domains covers the whole domain. Each material point has domain described by a non-negative particle characteristic function $\chi_p$. To avoid the errors, the sum of all the particles characteristic functions (from 1 to total number of point *np*) at any point **x** of the discretised domain **Ω** should be 1, i.e. they should be a partition of unity:

$$\sum_{i=1}^{np} \chi_p^i(\mathbf{x}) = 1 \tag{2}$$

In principle, there is no requirement that the particle domains do not overlap, but the weighting of all the particles should preserve the partition of unity. Having such a discretisation, any quantity in domain **Ω** can be assigned to particles. For example, the i-th particle volume is

$$V_p^i = \int_\Omega \chi_p^i(\mathbf{x}) d\Omega \tag{3}$$

The i-th particle mass is

$$m_p^i = \int_\Omega \rho(\mathbf{x}) \chi_p^i(\mathbf{x}) d\Omega \,, \tag{4}$$

where $\rho$ is the density of the material at point **x**. The i-th particle stress $\boldsymbol{\sigma}_p^i$ is

$$\boldsymbol{\sigma}_p^i = \int_\Omega \frac{1}{V_p^i} \boldsymbol{\sigma}(\mathbf{x}) \chi_p^i(\mathbf{x}) d\Omega \tag{5}$$

where $\boldsymbol{\sigma}(\mathbf{x})$ is the initial value of the stress at point **x** of the domain **Ω** and the integral is computed for each of the stress components separately. The values are averaged over the particle domain and even if we initially have the accurate data of the given field at the initial state, it is irrecoverable accurately after discretisation, leading to the initial discretisation error. However, as long as the integrals are accurately computed, the total amount of mass and momentum is preserved. The mass and momentum are also preserved in the whole MPM algorithm.

### 2.2 General algorithm of explicit Material Point Method

MPM solves Eq. 1 on the grid, while the material points contain all the data at the beginning of each time step. Hence, in each step of computations, the algorithm has to transfer the data from the material points to the grid nodes. For this transfer we use grid shape functions $S_v$. Additionally, to avoid errors, the grid shape functions should satisfy the partition of unity, that is at any point **x** of the domain **Ω**, the sum of all the grid shape functions should be one:

$$\sum_{i=1}^{nn} S_v^i(\mathbf{x}) = 1 \tag{6}$$

, where *nn* is the total number of grid nodes. Having chosen the grid shape functions and the particle characteristic functions, there is enough information to transfer the data from the particles to the grid nodes. To calculate the quantities at the nodes, from each material point and each grid node, the algorithm defines the weighting function $S_{vp}$, and the gradient of the weighting function $\nabla S_{vp}$ as:

$$S_{vp} = \frac{1}{V_p} \int_{\Omega} S_v(\mathbf{x}) \chi_p(\mathbf{x}) d\Omega \tag{7}$$

$$\nabla S_{vp} = \frac{1}{V_p} \int_{\Omega} \nabla S_v(\mathbf{x}) \chi_p(\mathbf{x}) d\Omega \tag{8}$$

Note that the integral given in Equations 7 and 8, for each pair (node, material point) gives a single value in case of 1D calculations, two values, corresponding to x and y directions in 2D case, and three values in 3D case. Also, as both $S_v$ and $\chi_p$ are partition of unity, $S_{vp}$ is also partition of unity. Having defined the weighting function and its gradient, the nodal quantities, such as nodal momentum $\mathbf{p}_v$ and nodal forces are computed as the product of weighting function and material point momentum $\mathbf{p}_p$:

$$\mathbf{p}_v = \sum_{i=1}^{np} \mathbf{p}_p S_{vp}^i \tag{9}$$

and the nodal masses are:

$$m_v = \sum_{i=1}^{np} m_p S_{vp}^i \tag{10}$$

The internal forces for given grid node due to stress in the material points are

$$\mathbf{f}_v^{\text{int}} = -\sum_{i=1}^{np} \boldsymbol{\sigma}_p^i \nabla S_{vp}^i V_p^i \tag{11}$$

The body forces **b** are taken into account as

$$\mathbf{f}_v^b = \sum_{i=1}^{np} m_p^i \mathbf{b}^i S_{vp}^i \tag{12}$$

And the boundary conditions satisfied at the nodes are

$$\mathbf{f}_v^\tau = \int_{\partial\Omega_\tau} \tau S_v(\mathbf{x}) dS \tag{13}$$

Equation 13 describing traction boundaries is not trivial to discretize. As the material points can move freely through the grid, establishing the traction at grid nodes at the initial discretisation is not sufficient, because the material points may not be present at given grid nodes after deformation. As such, often, the boundary forces are stored at the material points and in each step moved to the nodes, similarly to body forces. Typically, such a simple solution is of sufficient accuracy, even though it does not take into account rigid bodies rotations and changes in boundary shape. When an accurate representation of boundary tractions via discretization of Equation 13 is required, there is a need for an algorithm involving accurate tracking of boundaries, e.g. as proposed by Cortis et al. (2018).

Having computed the values of nodal forces (Equations 11-13) and nodal mass, the discretised version of Equation 1 is used to balance the rate of change of momentum at each node:

$$\dot{\mathbf{p}}_v = \mathbf{f}_v^{\text{int}} + \mathbf{f}_v^b + \mathbf{f}_v^\tau \tag{14}$$

If there are any Dirichlet boundary conditions (e.g. specified velocity and in particular velocity equal to zero), it is applied here by overwriting the computed grid momentum.

Typically, to reduce the numerical diffusion, the rate of material point position changes is updated using grid nodal velocities, while the rate of material point momentum change is

updated using grid nodal accelerations (computed based on the grid nodal rate of momentum change):

$$\dot{\mathbf{x}}_p = \sum_{i=1}^{nn} \frac{\mathbf{p}_v^i}{m_v^i} S_{vp}^i = \sum_{i=1}^{nn} \mathbf{v}_v^i S_{vp}^i \tag{15}$$

$$\dot{\mathbf{p}}_p = \sum_{i=1}^{nn} \frac{\dot{\mathbf{p}}_v^i m_p}{m_v^i} S_{vp}^i = m_p \sum_{i=1}^{nn} \mathbf{a}_v^i S_{vp}^i \tag{16}$$

The above transfers to and from nodes conserve mass and momentum. The actual positions are updated typically with a Forward Euler scheme, i.e. by multiplying the time derivatives with the given time step value dt.

The nodal velocities gradients can be used to compute the rate of change in the field of strain at any point **x** of the domain **Ω**:

$$\dot{\boldsymbol{\varepsilon}}(\mathbf{x}) = \frac{1}{2}\left(\nabla \mathbf{v}(\mathbf{x}) + \nabla \mathbf{v}(\mathbf{x})^T\right) = \frac{1}{2}\sum_{i=1}^{nn}\left(\nabla S_v^i \mathbf{v}_v^i + \mathbf{v}_v^i \nabla S_v^i\right) \tag{17}$$

This strain field can be associated with strain at material points as:

$$\dot{\boldsymbol{\varepsilon}}_p = \frac{1}{V_p}\int_{\Omega} \chi_p(\mathbf{x})\dot{\boldsymbol{\varepsilon}}(\mathbf{x}) d\mathbf{x} = \frac{1}{2}\sum_{i=1}^{nn}\left(\nabla S_{vp}^i \mathbf{v}_v^i + \mathbf{v}_v^i \nabla S_{vp}^i\right) \tag{18}$$

The strains are further used to compute the new stresses in particles (update stress last). Such an update has slightly dissipative properties (Bardenhagen, 2002), and is implemented in many MPM codes. The alternative is to update stress first, that is update strains and stresses before solving equation 14. More details on the implementation choices and discussion between these alternative formulations are given e.g. in (Bardenhagen, 2002; Buzzi et al., 2008).

Once all the data is mapped back to the particles and position of the material points is updated, the next step of the algorithm begins with the time increased by the time step value dt. The grid in the next step typically stays the same as in the previous step, i.e. nodal positions are not updated.

Almost all versions of the explicit MPM follow this algorithm, with differences related to, for example, choice of the functions and domain tracking. Furthermore, the accuracy of the calculations fundamentally depends on the grid density. The grid also affects other errors in the calculations. For example, material points crossing the boundaries of grid cells may lead to errors. As such, the grid density is essential in calculations, even though the method discretises the initial problem into material points, not grid nodes.

### 2.3 Original Material Point Method

The development of the original Material Point Method (Sulsky et al., 1994) took inspiration from the Particle in Cell and FLIP methods (Harlow, 1963, 1988), extending them into solid mechanics. This first MPM assumed that the particles do not have a domain, but just mass and velocity. Therefore, the particle characteristic function $\chi_p$ is the Dirac delta function. The chosen grid shape functions are linear, and equal to 1 in the given node and zero in all the nodes in the neighbouring cells. The material points, during initial discretisation, assume mass computed as equivalent to the amount of mass of the discretised domain. This mass is constant. The particles also have strain and corresponding stress.

The original Material Point Method is relatively robust. However, as the grid shape functions are linear, the gradient of those functions is – in case of taking the grid size as a unity – plus or minus 1. Therefore, when a particle crosses the grid boundary, the force at the node due to particles stress (eq 11) changes sign, as the shape function gradient changes sign. That leads to error and abrupt changes in solution when the particle crosses grid cell boundary, see Figure 1.

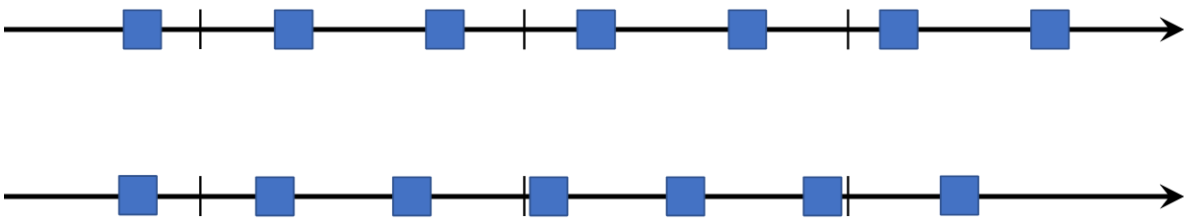


Figure 1. Initial discretisation of 1D bar with particles (blue squares) initially distributed uniformly, two per cell (top). As compression increases, eventually a node will have two particles on one side, and three on the other (bottom), leading to an imbalance in internal force when using linear shape functions, as described in the text.

Astute readers of the early MPM literature recognized a potential source of error arising from the internal force calculation, given by Equation (11). The following simple thought experiment will illuminate a primary source of error, namely, when particles cross from one cell to a neighbouring cell. Suppose we have a one-dimensional elastic bar in quasi-static compression, such that all particles have the same value of stress. Assume a common scenario in which all particles have the same volume, and are distributed uniformly with 2 particles/cell, in cells with width dx (Figure 1, top).

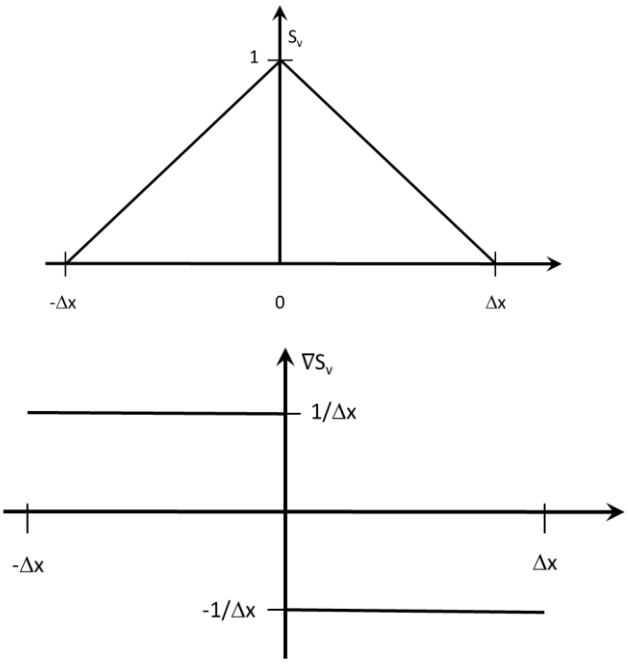



Figure 2. Linear shape function (left) and its gradient, discontinuous at the node.

For the linear shape function, its gradient on either side of a grid node is +/- 1/Δx, as shown in Figure 2. While the load is small enough such that all cells contain two particles, the internal force equation becomes:

$$\varphi_v^{int} = -\left(\sigma_p \frac{1}{\Delta x} V_p + \sigma_p \frac{1}{\Delta x} V_p - \sigma_p \frac{1}{\Delta x} V_p - \sigma_p \frac{1}{\Delta x} V_p\right) = 0 \quad (19)$$

which is the correct result for a bar in static equilibrium. If the load is increased slowly such that a state of near equilibrium is maintained, eventually the particles will become arranged as shown in Figure 1 (bottom), where at least one node will have three particles on one side, and two on the other. The force balance is disrupted in a stepwise manner as soon as this imbalanced particle distribution is reached. This leads to reduction of accuracy in calculation. The force may eventually rebalance, but in such a case, the uniform stress distribution in the particles is lost. The situation may also lead to a total loss of stability in computations.

For highly dynamic simulations such as Taylor impact or rod penetration, these types of force imbalances may not be significant. However, when MPM was a new method, it was natural for those interested in it to try to solve very simple quasi-static problems, and for the reasons described here, those simulations often did not go well. The use of higher order B-splines alleviates cell crossing errors, but also introduces other complications, such as treatment of grid-based boundary conditions and enlarging the distance at which self-contact begins.

Cell crossing error was the primary motivation for the development of finite domain based particle methods, the first of which was the Generalized Interpolation Material Point (GIMP) method of Bardenhagen & Kober (2004), as well as the “Dual Domain” Material Point (DDMP) method of Zhang et al. (2011). It seems unlikely that MPM would have found anywhere near the level of adoption that it currently has were it not for these authors’ contributions to the method, as shown in further sections.

### 2.4 Generalized Interpolation Material Point Method (GIMP)

The Generalized Interpolation Material Point Method (Bardenhagen & Kober, 2004) is a general class of Material Point Method algorithms , developed to improve accuracy and address instabilities of the original MPM formulation. The general GIMP algorithm is given in section 2.2. However, typically GIMP is understood as Contiguous Particles GIMP Method. In this case, the particle characteristic function $\chi_p(\mathbf{x})$ is given as:

$$\chi_p(\mathbf{x}) = \begin{cases} 1 - \text{inside particle domain, i.e. when } \mathbf{x} \in \Omega_p \\ 0 \ - \text{ outside of particle domain, i.e. when } \mathbf{x} \notin \Omega_p \end{cases} \tag{20}$$

The particle domain is chosen as a line segment in 1D, a square (or a rectangle) in 2D and a cube in 3D. GIMP combines this function with a linear grid shape function being 1 at the given grid node and 0 at the neighbouring nodes. In this case, the integration (eq. 7 and 8) can be done analytically, leading to the exact weighting function $S_{vp}$, and the gradient of the weighting function $\nabla S_{vp}$. Two main implementations of GIMP differ in how they treat the domain of the particle $\Omega_p$. When this domain is unchanged during simulation, the variant of the method is called UGIMP. On the other hand, when the domain volume is updated in line with strains, the method is typically called cpGIMP. However, to preserve the analytically computed weighting functions, the domain is retaining the rectangular shape in 2D (cube in 3D) and hence the method does not take into account any shear deformations and rigid body rotations. Also, GIMP does not ensure continuity, i.e. the domains of the neighbouring points are not connected, which means that during a simulation the particle characteristic functions of two particles may be 1 in a given particular point of space (particle may overlap, $\sum \chi_p = 2$ ). On the other hand, there may also be discontinuities with some parts of the initial domain not

covered by any particle characteristic function ($\sum \chi_p = 0$). These problems would be reduced in the Fuzzy Particle GIMP Method, also proposed by Bardenhagen & Kober (2004), where the particle domains overlap in the initial discretisation. In this case, assuming, e.g. overlap of 2 particles in each point of the discretised domain, if after deformations the domains become misaligned, the total value of the particle characteristic function would be $\sum \chi_p = 1.5$ or $\sum \chi_p = 0.5$. However, the Fuzzy Particle GIMP Method has not been implemented into major MPM codes, and therefore there is not much experience in its robustness and accuracy.

Wallstedt & Guilkey (2008) performed a Method of Manufactured Solutions (MMS) analysis comparing the accuracy of UGIMP and cpGIMP. They have demonstrated the superior accuracy of both GIMP methods over the original MPM. While cpGIMP is shown to be considerably more accurate compared to UGIMP for a special case where the manufactured solution is designed to keep the particles aligned to the axis and rectangular, for more general loading, the advantage is nearly negligible, as even cpGIMP imposes the constraint that particles remain rectangular (or cuboid in three-dimensions) and aligned to the principal axes. These limitations prevent cpGIMP from providing significantly improved accuracy for all but a few special problems.

### 2.5 Convected Particle Domain Interpolation (CPDI) Material Point Method

GIMP was a tremendous step forward in allowing MPM to be considered a method for practical engineering calculations, but still includes a few deficiencies. First, curved geometry is described by a ragged “stair-step” edge. Next, the effect of shear deformation on the particle domain is not accounted for, and finally, at large deformation, a numerical fracture is still possible as a gap between particles develops in tensile loading.

The improvement to GIMP, which allows for shear deformation of the particles domain is the Contiguous Particle Domain Interpolation Material Point Method (Sadeghirad et al., 2011). The method proposes an alternative evaluation of Eq. 7 and 8 that does not require division of the particle domains along cell boundaries. To that end, it constructs the alternative grid basis functions at a given point $\mathbf{x}$ as an interpolation of standard grid basis function $S_v^i$ at four corners in 2D ($\alpha = 1..4$) of each particle domain multiplied by the Finite Element shape function related to a given corner of the particle domain $Q_\alpha^p(\mathbf{x})$:

$$S_v^{i,app} = \sum_{\alpha=1}^{4} Q_\alpha^p(\mathbf{x}) S_v^i(\mathbf{x}_\alpha^p) \tag{21}$$

The original $S_v^i$ and alternative function $S_v^{i,app}$ differ inside the domain, but not in particles corners. Having the approximate shape function, the weighting function and its gradient are defined as before, but with the $S_v^{i,app}$:

$$S_{vp} = \frac{1}{V_p}\int_\Omega S_{vp}^{app}(\mathbf{x})\chi_p(\mathbf{x})d\mathbf{x} = \frac{1}{V_p}\sum_{\alpha=1}^{4} S_v^i(\mathbf{x}_\alpha^p)\int_\Omega Q_\alpha^p(\mathbf{x})d\mathbf{x} \tag{22}$$

$$\nabla S_{vp} = \frac{1}{V_p}\int_\Omega \nabla S_{vp}^{app}(\mathbf{x})\chi_p(\mathbf{x})d\mathbf{x} = \frac{1}{V_p}\sum_{\alpha=1}^{4} S_v^i(\mathbf{x}_\alpha^p)\int_\Omega \nabla Q_\alpha^p(\mathbf{x})d\mathbf{x} \tag{23}$$

where it was assumed that the particle characteristic function $\chi_p(\mathbf{x})$ is equal to 1 in its domain. The weighting functions for the i-th particle are computed in 2D as:

$$S_{vp}^i = \frac{1}{4}\left(S_v^i(\mathbf{x}_1^p) + S_v^i(\mathbf{x}_2^p) + S_v^i(\mathbf{x}_3^p) + S_v^i(\mathbf{x}_4^p)\right) \tag{24}$$

while the gradient is dependent on the domain size, see (Sadeghirad et al., 2011). In CPDI, the particles domains are not connected, i.e. each domain is computed separately with the stretch tensor, the areas of the initial domain where the particle characteristic function breach the partition of unity, i.e. $\sum \chi_p \neq 1$ are still possible. Nonetheless, as the domains of the particles are now better tracked, the magnitude of such errors should be much smaller than in cpGIMP.

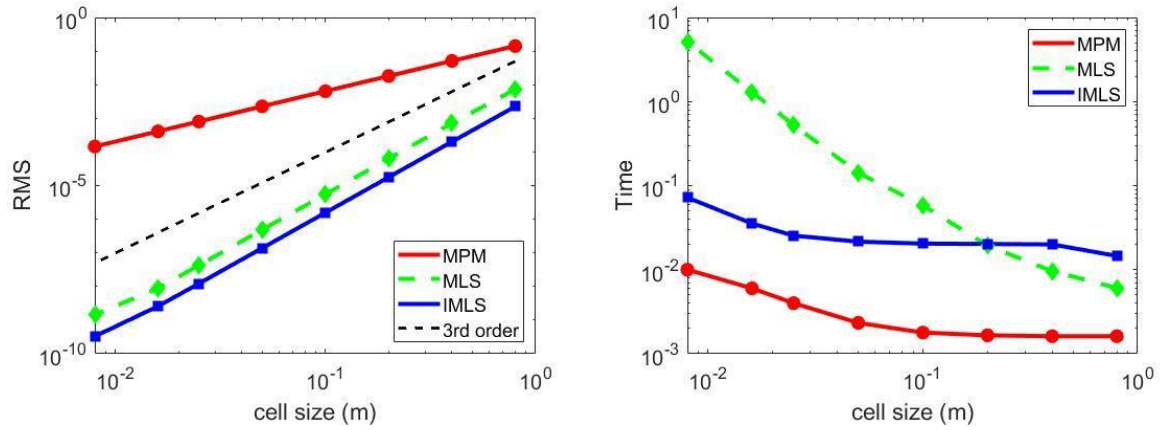
Second-order convected particle domain interpolation (CPDI2) (Sadeghirad et al., 2013) improved the domain tracking further. The particle domains are quadrilaterals in 2D and hexahedra in 3D. The improved method ensures that the particle domains are tightly covering the domain and are similar to a finite element mesh constructed with four-node elements. Therefore, CPDI2 can be seen as a bridge between Finite Element Method and Material Point Method. CPDI2 can also be enriched to model weak discontinuities (Sadeghirad et al., 2013).

In the original CPDI paper, the manufactured solutions used by Wallstedt & Guilkey (2008) were used again. These showed that, for the special case in which cpGIMP has second-order accuracy, CPDI gave identical levels of accuracy, but for more general loading, the latter method is considerably more accurate. Also exposed in that work was the fact that most any effort made to approximate a contiguous domain for cpGIMP will break down when certain deformations are applied. More recently, Kamojjala et al. (2015), constructed additional manufactured solutions that further demonstrate the superior accuracy of CPDI relative to GIMP. This accuracy does come at the cost of some degree of robustness. Because CPDI particle domains do change, it is possible to deform them to the point that they invert. Also, particle domains can become so extended as to break parallel implementations that rely on a fixed amount of ghost data. Both of these issues are addressed to a large degree through the use of so-called “domain scaling” as described by Homel et al. (2016). Here, a method is described in which particle domains are scaled so that the particle corners stay within a sphere

of user-defined radius. This approach provides a reasonable compromise between the accuracy offered by CPDI and the robustness of GIMP.

### 2.6 Convected Particle Least Squares Material Point Method (CPLS)

In a search for higher convergence variants of MPM, it became apparent that, in certain cases, CPDI solutions are not converging at second order. That led to developments of shape functions based on splines moving least squares (MLS) reconstruction technique (Sulsky & Gong, 2016). CPLS uses improved moving least squares (IMLS) reconstruction and applies it to the CPDI Material Point Method (Tran et al., 2020). This leads to a method that obtains higher convergence rate and retains the advantages of the CPDI Material Point Method.




(a) Numerical accuracy comparison (b) Numerical efficiency comparison

**Figure 3:** Comparison of the algorithm performance between MPM, Moving Least Squares (MLS) and Improved Moving Least Squares (IMLS) ( Tran et al., 2020)

In general, the MLS method has shown to retain the second order of MPM but suffers from certain limitations related to the singular or ill-conditioned coefficient matrix of the linear systems. On the other hand, the IMLS method uses the orthogonal polynomials as the alternative of the monomials in the original MLS. As a result, the IMLS method can obtain

high order of accuracy like the MLS method (see Figure 3.a) but is (i) more stable in the ill-conditioned of the linear systems and (ii) more efficient than the MLS method in terms of computational cost (see Figure 3.b). The proposed IMLS method was demonstrated to obtain more accurate solutions compared to existing MPM methods in literature in large deformation benchmarks (using Methods of Manufactured Solutions).

## 2.7 Other examples of explicit formulations of Material Point Method

### 2.7.1 Dual Domain Material Point Method

Another approach which reduces the cell-crossing errors is introduced in the Dual Domain technique (DDMP) developed by Zhang et al. (2011). DDMP keeps the linear shape function but modifies the gradient form to be continuous between the grid cells. DDMP also enlarges the influence domain of material points through dual domains - a cell contains the material points (grey cell in *Figure 4*) and neighbour-node cells (blue dash cell in *Figure 4*), without using the characteristic functions. The value of the shape function gradient in the cell which contains the material points $\nabla S_i$ is computed from the original MPM, while the value of the shape function gradient in the neighbor-node cells is calculated from the node-based function $\widetilde{\nabla S_\iota}$. In DDMP, the gradient of the shape function is modified from $\nabla S_i$ to $\underline{\nabla S_i}$:

$$\underline{\nabla S_i}(x) = (x).\nabla S_i + [1-(x)].\widetilde{\nabla S_\iota} \tag{25}$$

with the node-based function $\widetilde{\nabla S_\iota}$ being:

$$\widetilde{\nabla S_\iota}(x) = \sum_{j=1}^{nn} \frac{S_j(x)}{V_j} \int S_j \nabla S_i dv \tag{26}$$

where $V_j = \int S_j dv$ is the nodal volume at node *j*. In eq. (26), the integral $\int S_j \nabla S_i dv$ is composed of the linear basis function $S_j$ and their gradient $\nabla S_i$. Therefore, it can be solved analytically, as the background grid remains unchanged during the calculation.

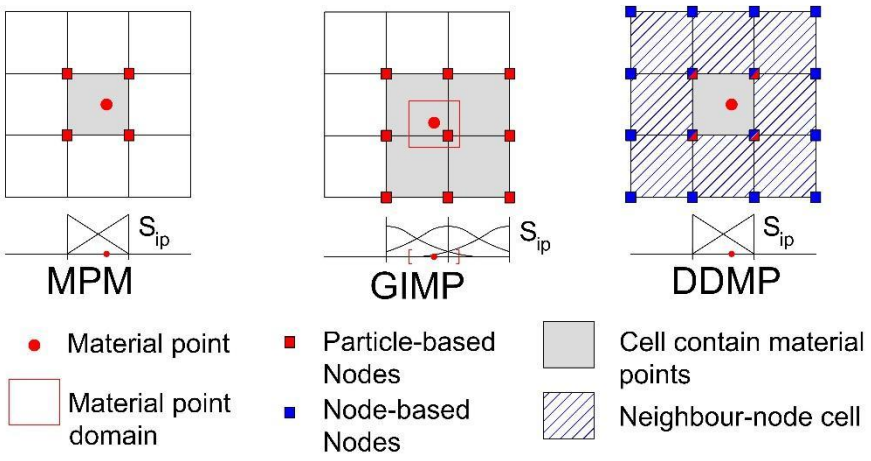



Figure 4 Schematic of different mapping scheme in MPM

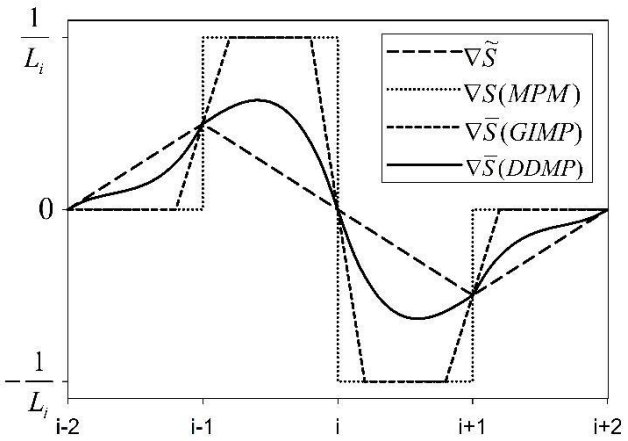



Figure 5 Gradient of the shape function in 1D

*Figure 5* shows the gradient of the shape function in a one-dimensional case. On the right-hand side in Equation (25), the first term represents the mapping of material points to particle-based nodes (defined as particle-based mapping) and the second term represents the mapping of material points to node-based nodes (defined as node-based mapping). Both mappings are weighted by , a weighting function. The gradient of the shape function is continuous only if is continuous and = 0 at cell boundaries. The choice of can retrieve the gradient of the shape function of other versions of MPM as follows:

$$\nabla S_i = \begin{cases} \nabla S_i \text{ for MPM }, & \alpha = 1 \\ \overline{\nabla S_i} \text{ for DDMP}, & \alpha = \alpha_1 \\ \overline{\nabla S_i} \text{ for GIMP }, & \alpha = \alpha_2 \end{cases} \quad (27)$$

### 2.7.2 2. B-spline Material Point Method

The B-spline Material Point Method (BS-MPM) (Steffen et al., 2008; Steffen et al., 2010) replaces the piecewise-linear shape functions used on the original MPM method and most of the aforementioned modifications by higher-order B-spline basis functions (grid shape functions, $S_v^i$). Typically, quadratic and cubic B-splines are adopted as basis functions $S_v^i$ which are, as a consequence, once and twice continuously differentiable, respectively. As a consequence, the gradients of the B-spline basis functions are at least continuous, and hence, the error related to grid-crossings is eliminated completely. Furthermore, the use of B-spline basis functions has the potential to yield high-order spatial convergence if all other sources of errors like mass lumping and inaccurate MPM-type integration are eliminated. Figure 6 (c) demonstrates that the use of quadratic (p=2) BS-MPM yields optimal order of convergence p+1 in the displacement (u) and the velocity (v) and p in the stress **σ** when used in concert with the consistent mass matrix and an accurate numerical integration based on function reconstruction. Use of B-spline basis functions alone with standard MPM-type integration leads to suboptimal convergence in the stress (p-1) and displacement (p) variable; see Figure 6(c) blue symbols. It should be noted that research on high-order BS-MPM is ongoing to improve the method's robustness in practical applications. Different approaches exist to construct B-spline basis functions. In the first approach, the basis functions are defined for each node separately (Steffen et al., 2010).

Alternatively, univariate B-spline basis functions can be defined for the whole domain at once based on a single knot vector. The latter approach follows the idea of Isogeometric Analysis (Hughes et al., 2005), a computational approach that unifies finite element analysis and

conventional computer-aided design. The extension to multiple dimensions is straightforward, by adopting the tensor product of univariate B-spline basis functions. Furthermore, B-spline basis functions have been applied within the implicit formulations of MPM (Motlagh & Coombs, 2017).

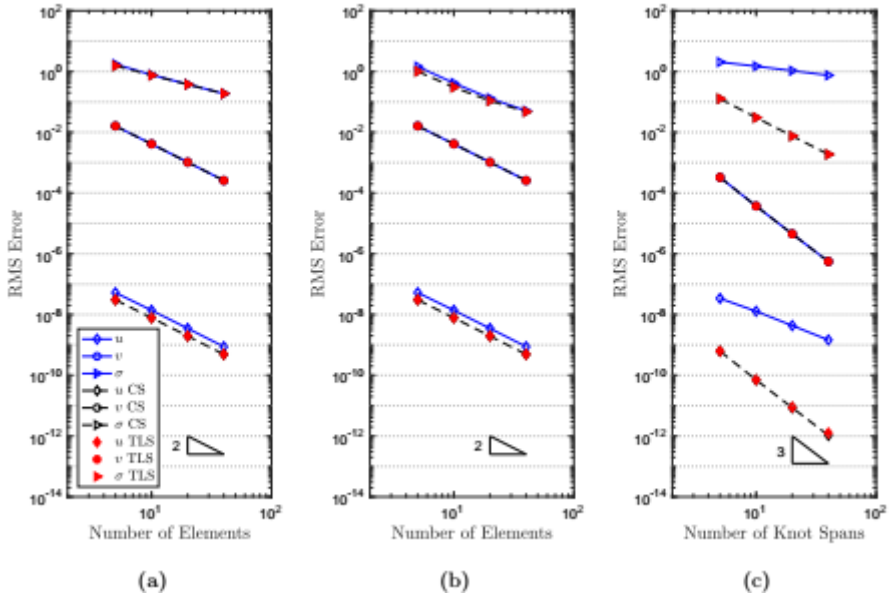



Figure 6 Spatial convergence for a one-dimensional vibrating bar (no grid crossing) with: (a) the original MPM (b) DDMPM and (c) quadratic BS-MPM without function reconstruction techniques and when adopting cubic spline interpolation (CS) and TLS reconstruction.

In recent years, BS-MPM has been combined with function reconstruction techniques (Tielen et al., 2017; Wobbes et al., 2019) to improve the particle based numerical integration typically adopted in MPM. A common problem of many reconstruction techniques is that they are either inaccurate or fail to conserve the density or momentum distribution of the original particle data. Taylor-Least Squares reconstruction (Wobbes et al., 2019) is one of the few approaches that enables both accurate and conservative function reconstruction by using Taylor basis functions within a discontinuous Galerkin framework. It should be noted, that the Taylor basis functions are only used for the reconstruction of a piecewise continuous function from the discrete particle data, which is used as surrogate within the numerical integration procedure and not as

the shape functions within the MPM. *Figure 6* illustrates the spatial convergence observed for a one-dimensional vibrating bar problem (no grid crossing) with the original MPM, DDMPM and BS-MPM using cubic spline interpolation and TLS as reconstruction technique, respectively. For the original MPM and DDMPM, the use of function reconstruction techniques slightly lowers the observed RMS error, but does not change the order of spatial convergence. When applied to BS-MPM, the use of function reconstruction techniques improves the observed spatial convergence for both displacement and stress.

Multivariate B-spline basis functions are defined on a (topologically) structured mesh. Extensions to locally refined meshes exist, e.g., THB splines (Giannelli et al., 2012) and LR-splines (Dokken et al., 2013), but the overall approach still assumes a structured background mesh. A possible extension of spline functions to truly unstructured triangulations consists of using Powell-Sabin splines as basis functions (Powell & Sabin, 1977), leading to the recently proposed Powell-Sabin Material Point Method (PS-MPM) (de Koster et al., 2020).

Table 1 contains the comparison of the explicit Material Point Method algorithms.

Table 1. Comparison of the explicit MPM algorithms

| Formulation | Particle Characteristic Function $\chi_p$ | Shape function $S_v$ | Convergence rate | Robustness | Domain tracking 2D | Publication |
|---|---|---|---|---|---|---|
| Original MPM | Dirac delta | linear | 2nd order, irregular (affected by errors) | High | no | Sulsky et al. (1994) |

| | | | | | | |
|---|---|---|---|---|---|---|
| GIMP | Typically top-hat (0/1) | Typically linear | Improved over original MPM | High / Very high | cpGIMP, rectangular domains | Bardenhagen & Kober (2004) |
| Fuzzy GIMP | Overlapping particles | Linear | Improved over original MPM | High / Very high | Rectangular domains | Bardenhagen & Kober (2004) |
| DDMP | Top-hat (0/1) | Linear, gradient chosen independently | Improved over original MPM | High / Very high | rectangular domains possible | Zhang et al. (2011) |
| CPDI | Top-hat (0/1) | Constructed for particles | Typically $2^{nd}$ order | Medium / High | Parallelogram | Sadeghirad et al. (2011) |
| CPDI 2 | Top-hat (0/1) | Constructed for particles | Typically $2^{nd}$ order | Medium / High | Quadrilateral, corners tracked | Sadeghirad et al. (2013) |
| CPLS | Top-hat (0/1) | IMLS (Tran et al., 2019) | Typically $2^{nd}$ order | Medium / High | Parallelogram | Tran et al. (2020) |
| BS-MPM | Dirac delta | Quadratic/cubic B-splines | Typically $2^{nd}$ order (limited by other errors) | Medium / Ongoing research | no | Steffen et al. (2008) |
| PS-MPM | Dirac delta | Quadratic/cubic Powell-Sabin splines | Typically $2^{nd}$ order (limited by other errors) | Medium / Ongoing research | no | de Koster et al. (2020). |

## 2.8 Material Point Method in simulations for animation

The ability of the MPM method to provide solutions for challenging physical problems involving large deformations and tearing has not gone unnoticed by those modellers working with the animation industry. The needs of this industry are somewhat different from the simulation for prediction in standard engineering. The key aspects of methods for use in entertainment is that they should be physically realistic as perceived by the viewers and that the methods should be fast and robust enough to be used in a production environment.

This difference in methodology not-withstanding, the need to use MPM for some very challenging 3D applications has led to a number of developments that are likely to be useful in other settings if and when they are analysed and verified.

The key starting point is the simulations of snow in the movie Frozen based on work by Stomakhin et al. (2013, 2014). There are a number of innovations developed in this context starting with a 3D extension of the work of Steffen et al. (2008) and more significantly using the idea of rigid and affine particles in cell approaches to ensure the conservation of momentum and angular velocity. The details of these methods are given in the Appendices of the thesis by Jiang (2015). The central idea is that mappings from particles to grids and back again should conserve affine velocities $v$ of the form of

$$v_i^{n+1} = v + Cx_i^n \tag{28}$$

where C is an d x d matrix and d is the dimension. Jiang (2015) shows that this also helps conserve momentum and angular momentum. In a sense, this is a generalization of the linearity preservation approaches applied to much simpler MPM models by many, including Gritton et al. (2015). In both cases this is a way of reducing the impact of the null space problem described in Section 4.3. The

affine velocities idea is extended still further by the polynomial particle in cell method of Fu et al. (2017), which has a still richer and more complex representation of velocities at particles.

The need to generate production solutions quickly has also led to the development of time integration approaches that allow large stable timesteps to be taken without being always overconcerned with accuracy providing that conservation holds. One approach that is used in this context is to formulate an optimization problem to be solved at each large time step (Gast et al., 2015). This approach is used by others in animation and involves noting that the equations to be solved when an implicit method is used may be written as

$$H(x^{n+1}) = \mathbf{M}\frac{x^{n+1} - x^n - \Delta t v^n}{\Delta t^2} - \mathbf{f}\left(x^{n+1}, \frac{x^{n+1} - x^n}{\Delta t}\right) = \mathbf{M}\frac{x^{n+1} - x^n - \Delta t v^n}{\Delta t^2} v + \frac{\partial \Phi(x^{n+1})}{\partial x} = 0 \tag{29}$$

, where M is the lumped mass matrix, and forces **f** are linked to a function $\Phi$ as: $\mathbf{f} = -\frac{\partial \Phi}{\partial x}$. The problem may be solved as an optimisation problem, to improve robustness over the typical Newton scheme. If $\tilde{x} = x^n + \Delta t v^n$ then $H(x^{n+1}) = 0$ occurs at the minimum of $E(x^{n+1})$ as $\frac{\partial E}{\partial x} = H(x)$ for

$$E(x) = \frac{1}{2\Delta t^2}(x - \tilde{x})^T M (x - \tilde{x}) + \Phi(x) \tag{30}$$

Other approaches involve semi-implicit methods such as Daviet & Bertails-Descoubes (2016).

Another approach is to use a kind of multilevel approach either to apply different time steps in different regions (Fang et al., 2018) as well as optimization approaches on GPUs (Gao, Wang, et al., 2018). Finally very recent work on a Hierarchically Optimized Time integration method (HOT) by Wang et al. (2019) uses a multigrid method that is especially designed for MPM in conjunction with Quasi Newton methods as opposed to either Newton's method or the ADMM alternating

direction method of multipliers optimization algorithm of Overby et al. (2017). A Multigrid V-cycle preconditioner is used with a symmetric Gauss Seidel smoother and a conjugate gradient method at the coarsest level.

The final area in which MPM has been extended lies in improved applications of MPM with improved physical models. There are many such papers in the online survey of Jiang (2020). Examples are on modelling of: fluids foams and sponges (Ram et al., 2015), phase change (Stomakhin et al., 2014), sand (Klár et al., 2016), baking and cooking (Ding et al., 2019), cloth models (Montazeri et al., 2019) and Fracture (Wang et al., 2019; Wolper et al. 2019), hair rubber and soft tissues (Han et al., 2019) and landslides (J. Zhao et al., 2019).

While many of these innovations are driven by the needs of computer graphics and animation, they are clearly applicable in the broader engineering context too, once verified for that application space.

### 2.9 Future developments

Material Point Method has been significantly developed since its beginning in 1994. Yet, the available algorithms, despite developments (e.g. Sulsky, 2017; Tielen et al., 2017; Tran et al., 2020; Wobbes et al., 2019), still do not offer high order convergence. This seems to be an area where the method is trailing behind FEM, where high order elements offer much faster convergence. The question of optimal choice of time stepping algorithm and time step still seem to be not fully solved (with some attention given to it e.g. in (Berzins, 2018; Guilkey & Weiss, 2003) and likely will improve in the future. The further optimisation of the algorithm such that its robustness is not reduced, while reducing or eliminating various errors is also a worthy direction of investigations. Furthermore, the presented algorithms needs enhancements to simulate contact more accurately – without them, the contact occurs when a grid node is shared leading to an early contact (Figure 7/ Video 1).

Figure 7 / Video 1. With special contact algorithm, contact occurs when the domains of material points starts interacting (bottom), and is not dependent on the grid as in GIMP (top)

## 3. Implicit formulations of the Material Point Method

Main contributor: Coombs

The vast majority of the MPM published in the literature adopt an explicit formulation. There are several reasons for this, some of which include that the original MPM paper used an explicit formulation and that explicit formulations are conceptually simpler than implicit methods. The preference for explicit approaches can be highlighted by the difference in the number of citations, even when normalising the number of years since publication, for the original explicit (Sulsky et al., 1994), average 43 citations/year) and implicit (Guilkey & Weiss, 2003), average 9 citations/year) material point papers. This preference for explicit methods is not restricted to MPM, for example the original implicit particle-in-cell paper of Cummins and Brackbill (Cummins & Brackbill, 2002) has received even fewer citations, at 3.5/year, whereas the original explicit fluid mechanics particle-in-cell paper (Harlow, 1964) currently sits at 13.3/year. However, in geotechnical engineering, the majority of numerical analyses adopt implicit approaches, especially in the finite element literature. This can be attributed to several reasons, such as researchers/engineers being often interested in the quasi-static response of the problem under consideration, the complex nature of the constitutive models adopted and that the bulk motion of the soil response often being of primary interest.

The first implicit MPM approach was published in Guilkey & Weiss (2003) and will be discussed in more detail below. Another important early MPM implementation within the context of implicit algorithms was the semi-implicit approach of Sulsky & Kaul (2004). Since

these papers, a number of other research groups have adopted an implicit approach to the MPM, including [add references], amongst others.

For implicit formulations of MPM, it is important to distinguish between quasi-static and dynamic equilibrium equations. For quasi-static analyses the inertia term in Equation (1) is set to zero, leading the following weak statement of equilibrium

$$\int_{\Omega} \sigma : \nabla \delta v \, d\Omega = \int_{\Omega} \rho b \cdot \delta v \, d\Omega + \int_{\Omega_\tau} \tau \cdot \delta v \, d\Omega_\tau \tag{31}$$

Both dynamic and quasi-static implementations of the material point method will be considered in the following subsections, with Section 3.1 focusing on the original implicit formulation of Guilkey and Weiss (2003) and Section 3.2 looking at the quasi-static implementation of Charlton *et al.* (2017). Following these two sections, other approaches to implicit material point analysis will be considered and a comparison with explicit methods presented.

### 3.1 Original implicit formulation

The original implicit formulation of the MPM by Guilkey and Weiss (2003) was based on the observation that

*"Calculations on the computational grid are carried out in a manner that is exactly the same as those performed for finite element calculations. The material points essentially function as integration points for the assembly."*

This observation has been used by several other authors to express MPM simply as the *finite element method where the integration points are allowed to move*. Although this is a gross simplification of a method that actually includes some key algorithmic differences when compared to FEM (e.g. Moresi et al., 2003), it is a useful starting point for conceptual understanding of implicit MPMs.

All implicit material point methods attempt to satisfy the following discretised equilibrium equation on the background grid

$$f_g^{ext} - f_g^{int} - M_g \cdot a_g = 0 \tag{32}$$

at the end of the current time step. In the above equation $f_g^{ext}$ and $f_g^{int}$ are the external and internal force vectors, respectively, $M_g$ is the global mass matrix and $a_g$ are the accelerations of the background grid nodes. In order to solve Equation (28), Guilkey and Weiss (2003) adopted a fully-implicit Newton-Raphson procedure and combined it with a fairly-standard unconditionally stable trapezoidal rule to update the grid kinematics (displacement and velocity of the background grid nodes). One of the key differences between MPM and FEM is that it is not possible to transfer velocity/acceleration grid information directly between time steps. This is due to the fact that it is possible that material points will move into grid cells that were empty on the previous step. Instead, grid velocity/acceleration values at the start of the time step are mapped from material point values using a mass-normalised projection, for example

$$v_i = \frac{\sum_p \quad S_{ip} m_p v_p}{M_i}, \tag{33}$$

where $v_i$ is the background grid velocity of a specific grid node, $S_{ip}$ is the basis function linking the material point, $p$, and the grid node, $i$, $m_p$ is the mass at a material point, $v_p$is the velocity of a material point and $M_i$is the mass associated with the background grid node.

Linearsing Equation (28) with respect to the unknown background grid displacements, $u_g$, results in the following equation

$$K_g^{k-1} \cdot \delta u_g^k = f_g^{ext} - \left(f_g^{int}\right)^{k-1} - M_g a_g^{k-1}, \tag{34}$$

where $k$ is the Newton-Raphson iteration counter and $\delta$ denotes an iterative change. Note that in the above equation the external forces at the nodes, $f_g^{ext}$, and the global mass matrix, $M_g$, are constant over the time step as the material points are assumed to deform consistently with

the background grid according to the basis functions; that is the value of the basis function linking a specific node with a specific material point will not change throughout the time step. All of the other terms will, in general, vary over the time step and must be recalculated at each iteration of the algorithm. The Newton-Raphson iterations continue until Equation (28) is satisfied to within a given tolerance and once convergence has been achieved the positions, velocities and accelerations of the material points are updated. Guilkey and Weiss (2003) mentioned that, as the method has adopted an exact linearization of the discretised equilibrium equation (including use of the appropriate algorithmically consistent tangent for the stiffness of the underlying material), asymptotic quadratic convergence should be expected from the Netwon-Raphson process. However, Guilkey & Weiss (2003) also state that

*"…the nature of MPM is such that the material points will not in general correspond to the optimal sampling locations within elements of the computational grid. This may affect convergence via the accuracy of the integral evaluations."*

This point is not supported by this review as the integration of the residual and the linearized stiffness adopt the same quadrature points. It is true that the location of points within the background grid can impact on the conditioning of the global system of equations and this can lead to issues in terms of the accuracy of the linear solver, but this is a consequence of the difference in magnitude of entries in the global stiffness matrix rather than the use of material points as quadrature points – the same issues can be experienced in non-linear finite element analysis.

Guilkey & Weiss (2003) contribution was limited to elastic materials (compressible neo-Hookean model) undergoing finite deformations and only considered problems where FEM is able to provide more efficient and accurate solutions – elastic compression of a billet and stress

waves in connected circular elastic rods. The quasi-static implicit implementation of MPM presented in the next section will also consider elasto-plastic material behaviour.

### 3.2 Implicit Generalized Interpolation Material Point Methods

Charlton *et al.* (2017) was the first paper to present a quasi-static implicit material point formation for large deformation elasto-plasticity for the Generalized Interpolation Material Point Method (GIMPM). An interesting point about this work was the focus on maintaining local integration of the stiffness of each background element to keep the method as similar to the finite element method as possible. Before this paper a more material point-focused approach, where the contribution of a material point to all of the nodes that it influences being computed at the same time, tended to be adopted for generalized interpolation, and other domain-based, formulations. The approach of Charlton *et al.* (2017) requires the contribution of a material point to a node, or rather the basis functions and the associated spatial derivatives linking the material point and the node, to be split element by element. This point is illustrated in Figure 1, which shows the one-dimensional basis functions for three nodes (a, b, c) based on the contributions from two elements. The solid dark line shows the basis functions for node *b* and the dashed lines show how this basis function can be split into contributions from its two associated elements. Splitting the basis function in this way was achieved by expressing the basis functions in terms of local element coordinates, rather than global coordinates as in the original GIMP approach (Bardenhagen & Kober, 2004).

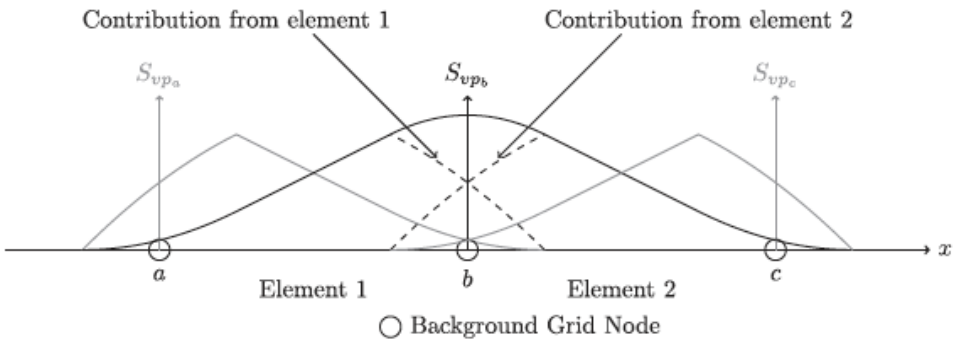



Figure 8: Element-specific generalized interpolation basis functions (reproduced from [7]), where the solid dark line is the basis function associated with node b and the dashed lines show the contribution of each element.

Charlton *et al.* (2017) also highlighted the importance of mapping the spatial derivatives of the basis functions into the current configuration when implementing the MPM method with an updated Lagrangian formulation. This mapping can be expressed as

$$\frac{\partial S_{vp}}{\partial x_j} = \frac{\partial S_{vp}}{\partial \tilde{X}_i}\left(\Delta F_{ij}\right)^{-1}, \tag{35}$$

where $\Delta F_{ij}$ is the incremental deformation gradient over the current time step, $x_i$ are the current (or deformed) coordinates and $\tilde{X}_i$ are the coordinates at the start of the time step. This is especially important for the GIMPM as the basis functions for this method rely on a regular background grid. The deformation of this background grid over a time step will not change the basis function values but it will change the spatial derivatives of these basis functions, which are required to determine the nodal internal force contribution of a material point. If this mapping is not included, or the spatial basis function derivatives not obtained in an alternative way, then the method is not actually solving (19). This point is discussed in detail by Coombs et al. (2020), who proposed a *previously converged* Lagrangian material point formation that avoids this mapping by expressing equilibrium at the start of the time step.

The issue of appropriate domain updating was also highlighted by Charlton *et al.* (2017), especially for problems involving large rotational components of the deformation gradient. In this paper, the use of the right symmetric stretch matrix, $U_{ij} = \sqrt{F_{ki}F_{kj}}$, was advocated rather

than the normal components of the deformation gradient when updating the domain lengths - domains can artificially collapse if they are updated according to the normal components of the deformation gradient. This point was explored in more depth by Coombs *et al.* (2020), who proposed a CPDI-inspired domain updated method as the right symmetric stretch update method induces spurious volume changes when analysing problems involving shear deformation.

Although the paper of Charlton *et al.* (2017) was focused on large deformation elasto-plastic analysis, all of the investigated problems can be simulated using the more efficient and accurate finite element method. Also, the paper only included a linear elastic-perfectly plastic von Mises constitutive model and there may be other issues when analysing problems with more complex constitutive models in terms of the stability of a quasi-static implicit material point formulation. In the authors' experience, quasi-static implicit implementations of the material point method can suffer from issues relating to the conditioning of the global stiffness matrix. This issue can become more severe when considering domain-based material point methods due to the small overlaps between the material point domains and the background mesh. It is also, in general, more of an issue with quasi-static analyses than for dynamic problems as the mass matrix often provides a stabilisation effect to the conditioning of the linear system of equations. For quasi-static analysis, one method to reduce this issue is to include a degree of *soft stiffness*, where a small amount of stiffness is added to the global stiffness matrix based on the integration of the background grid using conventional Gauss quadrature (Wang et al., 2014). The additional stiffness is normally several orders of magnitude less than the stiffness of the true material[1] and is chosen to balance: (i) the conditioning of the global system of equations and (ii) the speed of convergence of the global equilibrium equations. The second

[1] The authors have found that $1\times10\text{-}6$ to $1\times10\text{-}9$ of the Young's modulus of the physical material is sufficient, however [10] suggested the much larger value of $1\times10\text{-}2$, which will significantly influence the convergence rather of the global equilibrium iterations.

point refers to the fact that, by adding extra stiffness to the linear system, the global stiffness matrix is no longer a true linearization of the residual out-of-balance force of the non-linear problem. Adding this *soft stiffness* will not change the converged solution, at least not within *engineering accuracy*, but it may change the number of iterations required to achieve this solution. This point is in conflict to the results shown in (Wang et al., 2014), where the displacement of a cantilever beam was shown to be dependent on the soft stiffness factor (values between $1 \times 10^{-1}$ and $1 \times 10^{-4}$ were investigated). However, the paper did not specify the number of iterations that have been used or the convergence tolerance adopted so it is possible that the analyses with larger amounts of soft stiffness had not converged to the same tolerance as those adopting a smaller value.

### 3.3 Other implicit approaches to the MPM

A number of other researchers have adopted an implicit approach to the material point method and investigated various aspects of the method, including: the application of boundary conditions (Bing et al., 2019; Cortis et al., 2018), the issue of volumetric locking with isochoric plasticity (Coombs et al., 2018), the suitability of different domain-based material point formulations to analysing problems involving large distortion (Wang et al., 2019), energy consistent time stepping approaches for elastic and elasto-plastic materials undergoing large deformations (Love & Sulsky, 2006a, 2006b) and the use of matrix free Newton-Krylov conjugate gradient solvers for hyperelastic materials (Nair & Roy, 2012). Implicit material point methods have also been applied to the modelling of: three-dimensional multicellular constructs (Guilkey et al., 2006), geotechnical engineering problems (Beuth et al., 2008; Wang et al., 2016), in addition to a number of general benchmarking problems such as the collision of discs and simple beams.

AMPLE (A Material Point Learning Environment) (Coombs & Augarde, 2020) is specifically focused on implicit material point formulations and has been designed as an environment for researchers to learn about, and try out new ideas in, the MPM. However, being MATLAB-based it will not offer the computational efficiency obtained by compiled codes. Another alternative may be Kratos Multiphysics, where simple MPM implicit implementation has been recently introduced (Iaconeta et al. 2017, 2019).

### 3.4 Implicit versus explicit

A number of papers have presented comparisons between implicit and explicit methods. However, the problems selected normally are based on the methodical preference of the authors. That is, the authors select problems that they *a priori* know favour implicit or explicit approaches based on the thrust of the article and the points that the authors are trying to make rather than providing a range of problems covering the appropriate application areas of the two approaches. For example, it is well known that explicit methods are suited to impact and short-duration dynamic analyses, and implicit methods are efficient at solving low strain rate/quasi static problems where the *bulk motion* of the body is the main interest (Sulsky & Kaul, 2004). This has led to conflicting conclusions on the relative performance of the two approaches in the literature and a comprehensive comparison has yet to be published. One difficulty here is the number of different explicit, semi-implicit and implicit time stepping approaches that are available and the lack of a common framework for comparison.

In general explicit schemes determine the new *state* of the material points based on information from the previous timestep (or, expressed in a different way from information at the start of the current step) whereas implicit method use information from both the previous and current time steps. For linear problems, implicit methods should be unconditionally stable for any timestep

size, but for explicit methods the size of the allowable timestep is bounded by the CFL condition (or some factor of the CFL condition). In practice, we would still bound the size of the timestep in implicit methods for other reasons, such as visualisation or to minimise the excessive motion of a material point within a timestep. In addition to this, as we have already seen, MPM is inherently non-linear and there will be a limit on the allowable timestep size in implicit methods due to this (the same is true of PIC methods, see Cummins & Brackbill, 2002). Even with these limitations in place, implicit methods have been shown to allow for larger timestep sizes compared to explicit methods, see for example (Guilkey & Weiss, 2003; Love & Sulsky, 2006a; Sulsky & Kaul, 2004; Wang et al., 2016). However, allowing for larger timestep sizes does not mean that implicit schemes are more computationally efficient[2] as the *cost* of a timestep into the two methods are not equal; explicit steps are usually orders of magnitude less *expensive* than implicit steps due to the removal of the need to solve a linear system by adopting a diagonal global mass matrix.

The original implicit MPM paper (Guilkey & Weiss, 2003) included a comparison between implicit and explicit approaches and the finite element method for the analysis of the compression of an elastic billet (a problem where implicit methods are well suited), finding that the proposed implicit approach offered benefits of around 10% in terms of accuracy. However, the impact of the time step size on the performance of the methods was not investigated. Similar findings were presented by Sulsky and Kaul (2004) and in one example they found that their proposed implicit approach required 1.2% of the computational effort of the explicit solver – once again, the influence of timestep size was neglected. This point was investigated by Wang *et al.* (2016) who analysed the collision of two elastic discs and concluded, unsurprisingly, that larger timesteps were possible in the implicit formulation.

[2] Here the term efficient refers to the balance between run time and accuracy, with the different in efficiency between two methods being expressed in terms of the accuracy for a given level of computational effort or the computational effort required to achieve a given accuracy.

However the respective run times were not presented, so it is not possible to judge *efficiency*. As far as the authors are aware, there are no comparisons of explicit and implicit methods for problems involving non-linear material behaviour, such as elasto-plasticity. This is a curious oversight as MPM is normally applied to, and indeed was developed for, materials undergoing extreme deformations and non-linear behaviour.

Another problem that should be considered is the compatibility of different time stepping method with advanced constitutive formulations. This is especially important in one of the key application areas of MPM – geotechnics – where the formulations are very sensitive to incremental inputs of stress/strain. The improved accuracy of the implicit method may provide advantages here, especially in terms of avoiding spurious unloading within the stress-strain constitutive algorithm where the material artificially oscillates between elasto-plastic and elastic behaviour.

# 4. Convergence and approximation errors in MPM

Main contributors: Sołowski, Berzins, Guilkey, Möller, Tielen

## 4.1 Introduction

In engineering, the given numerical method needs to be accurate enough to be useful, while providing results in reasonable time. The required accuracy is usually in the range of 1% for most engineering applications, as beyond that, the uncertainties of material parameters values, as well as errors due to assumed geometry and materials models are usually order of magnitude higher. However, to ensure the numerical errors related to discretisation and algorithms are negligible, typically the simulations aim is the accuracy of 0.1%. This is typically taken as an accuracy for a given geometry, set of constitutive models and parameters and is compared to the accurate analytical solution of the governing equations the numerical method solves in a

discrete fashion. In this section, we deal with convergence of MPM to such an ideal analytical solution and discuss the MPM solution errors. The convergence is affected by the order of MPM solution, spatial discretisation and discretisation in time, with the first two factors discussed in this section in more detail. The errors can be reduced with the increase of grid density and number of material points, but also by modifying the MPM algorithm, so it is more accurate for given discretised geometry.

The other problem is how to approximate the given engineering problem and solve it with MPM. As MPM is evolving rapidly, the best practices change. However, it is always a good idea to check the spatial convergence of the solution, by solving the same problem with double grid density (while maintaining the amount of material points per grid cell as in the initial analysis) and comparing the results. Any discrepancy between the two results beyond the accuracy we aim for (typically 1%) should lead to in-depth analysis of errors and further simulations with denser grids, until the accuracy of the results is sufficient for the problem. In principle, the influence of the time step could also be checked. However, the errors arising from time steps are typically much smaller than those related to discretisation, as long as the time step is kept well below the Courant-Friedrichs-Lewy stability condition. Additionally, the allowed time step size by the stability condition, automatically computed by the MPM software with a suitable safety margin, is affected by the grid size, hence with the spatial convergence study, we also reduce the errors related to time stepping.

## 4.2 Background

While there has been much analysis of particle in cell methods such as the book by Grigoriev et al. (2012), one of the first rigorous attempts to provide a solid theoretical basis for particle methods was due to Raviart (1985) who provide a mathematical introduction to the vortex numerical method, by starting with the numerical solution of linear hyperbolic equation of the

first order. Typical error estimates for solutions to a linear advection equation are of the form given by his Theorem 4.2:

$$\| u - u_{h,\sigma} \| = O\left( \sigma^n + \left( \frac{h}{\sigma} \right)^m \right) \tag{36}$$

where σ describes a smoothing length, n is the order of a regularization scheme, m is the order of a quadrature rule, and h is some form of particle spacing.

For fixed values of n, m, and σ, the optimal choice of h thus is a compromise that balances both contributions. In the case n = m, this results in $h \sim \sigma^2$. In other words, the particle spacing h needs to be negligible compared to the smoothing length. This result has recently been improved by (Kirchhart & Rieger, 2019), but results of this type for the original MPM method are conspicuous by their absence. The move to more general methods that make use of cubic splines for example somewhat eases this problem by not employing the more problematic and less reliable delta particle basis of the original MPM method (Steffen et al., 2008).

Any analysis of MPM has to take into account two uncommon features of the method, both of which give rise to unusual computational and theoretical challenges. The first challenge is that particles cross cell boundaries giving rise to a grid crossing error and the second is that there are almost always many more particles than grid points, thus giving rise to a loss of information as particle values are mapped to nodal values and back again.

### 4.3 Cell Crossing

The observation that errors in both particle in cell methods and MPM increased when particles jumped over cell boundaries helped lead to the creation of methods such as the GIMP method (Bardenhagen & Kober, 2004). This increase in error was quantified by Tran et al. (2010) as coming from time integration discontinuities in higher derivatives of the nodal basis functions.

The comparative lack of smoothness of the spatial basis grid functions used in MPM translates into a lack of smoothness in time when particles cross grid points and then have properties that are redefined in terms of the basis functions in the next interval. The definition of particle velocity updates in terms of nodal velocity values means that the higher time derivatives of the particle velocity are discontinuous when a particle crosses a grid point.

For example, consider a simple 1D case (see Figure 1) and update the velocity $v_p^n$ at point $x_p^n$ using nodal accelerations $a_i^n, a_{i-1}^n$ with a value of *dt* that crosses the cell boundary to get

$$v_p^{n+1} = v_p^n + dt\left[ a_{i-1}^n + \frac{x_p^n - x_{i-1}}{x_i - x_{i-1}}(a_i^n - a_{i-1}^n) \right] \tag{37}$$

Suppose that *dt* is broken down into two steps $dt = dt_1 + dt_2$ to get

$$\bar{v}_p^{n+1} = v_p^n + dt_1\left[ a_{i-1}^n + \frac{x_p^n - x_{i-1}}{x_i - x_{i-1}}(a_i^n - a_{i-1}^n) \right] + dt_2\left[ a_i^n \right] \tag{38}$$

Then the difference between the two approaches is a first order error given by

$$v_p^{n+1} - \bar{v}_p^{n+1} = dt_2\left[ \frac{x_p^n - x_i}{x_i - x_{i-1}}(a_i^n - a_{i-1}^n) \right] = O(dt) \tag{39}$$

This is one order of accuracy lower than the standard local error of a timestep. The root cause is the lack of smoothness of the linear interpolation associated with the linear basis functions. There is still a grid crossing error from the use of say cubic splines, but it corresponds to jumps in higher spatial derivatives of the underlying spline basis. In the same context, Steffen et al. (2008) shows that, in particular, the acceleration calculated using the MPM algorithm may have significant quadrature errors in space and discontinuities in time, both of which make second-order temporal convergence unrealizable to the MPM practitioner. Figure 2 in Steffen et al.

(2010) shows a sample of a typical grid acceleration field encountered in standard MPM when piecewise-linear basis functions are used‡. The jump in acceleration occurs when a particle's position in space crosses a grid cell boundary. The calculated acceleration is obviously not smooth in this case, the impact of which has repercussions on the updated velocity and displacement of the particle (see also section 2.4).

### 4.4 The Null Space Problem

The second major computational and theoretical issue relates to the loss of information due to the mapping from particles to grid points and back again. The PIC method is well-known, e.g. (Brackbill (1988), to have an aliasing error due to the difference between the degrees of freedom at the grid points of the spatial mesh cell compared to the degrees of freedom at the particles. This error may result in oscillatory solution values. Brackbill (1988) states that, because the number of particles is finite, the number of Fourier modes is also finite. Thus, when there are n particles in each cell, there are n times as many Fourier modes as there are grid points. When values are mapped from nodes to particles, the lack of resolution at the nodes compared to resolution at the particles can cause an aliasing error. Again, to quote Brackbill, "Aliases occur because all Fourier modes with wavelengths shorter than the grid spacing are indistinguishable at the grid points." Early attempts to address this instability often started from the PIC jiggling work e.g. Brackbill & Lapenta (1994).

While Fourier analysis is a useful way to illustrate this error, it does not provide a general description. This aliasing is exactly the null space error addressed in this work. Brackbill showed that the user of better interpolants associated with the nodes to particles mapping helped reduce these errors and conducted a Fourier analysis of the problem. Gritton et al. (2015) have shown that the mapping matrix from particles to grids as denoted by $S_{vp}$ is rectangular and may have a nontrivial nullspace. For example, let *c* be a vector in $R^n$. If

$$S_{vp}c = 0 \tag{40}$$

then we say that c is in the nullspace of $S_{vp}$. For model problems, the nullspace is found by making use of its singular value decomposition, SVD. To get

$$S_{vp} = U \sum V^T \tag{41}$$

where $U$ has dimension m by m, $\Sigma$ is m by n, and V is n by n. The matrices $U$ and $V$ are unitary, meaning that the columns are orthonormal. Any vector a ∈ R m $a \in R^m$ can be expressed as a linear combination of the columns of $U$. The number of zero diagonal entries of $\Sigma$ corresponds to the lost information in mapping from particles to nodes. This lost acceleration corresponds to an acceleration error. This error is then mapped onto the particles for the next timestep. While this null space error only shows up as missing acceleration with this matrix $S_{vp}$, when the particles move and the matrix changes, the null space terms will then give a sudden non-zero acceleration. The solution to this problem is to use mappings with smaller null space errors such as those introduced in Gritton & Berzins, (2017) for the original MPM and Tran & Sołowski (2019) for GIMP.

### 4.5 Locking and zero energy modes

In the Finite Element Method, it has been long known that due to the inexact approximation of the displacement and strain fields by the elements, the elements may predict overly stiff behaviour of the material, dubbed as shear and volumetric locking, related to shear and volume deformations, respectively. The underlying theory and causes in the case of Finite Element Method are well given in Babuška & Suri (1992a, 1992b), with extra details related to common Finite Elements in MacNeal (1994). As mentioned by MacNeal (1994), "*Element designers are more interested in finding fixes for locking than in understanding why locking occurs*".

This statement, sadly, extends to most of the errors in Material Point Method, and in particular to the available anti-locking algorithms. The Material Point Method yet awaits a comprehensive overview of reasons for locking and estimation of related errors, similar to those in Babuška & Suri (1992) and MacNeal (1994) for common Finite Elements. However, due to similarities of Material Point Method calculations to those in Finite Element Method with Q4 element, locking is a significant issue in MPM. Furthermore, the Finite Element Q4 exhibits significant locking issues, which reduces only very slowly with the increase in element numbers. This may be the reason why an increase of grid density is not a reasonable solution for locking also in MPM. Therefore, the issue of locking needs to be fixed by other means.

The Finite Element Method offers a wealth of solutions to reduce the locking of the elements. The most typical one, suitable for the reduction of volumetric locking in hydro-mechanically coupled simulations, is reducing the integration order for the water phase. That is not yet adopted in MPM. However, first Mast et al. (2012) proposed an anti-locking scheme based on Hu–Washizu multifield variational principle. Coombs et al. (2018b) adopted another scheme from FEM based on the $\bar{F}$ approach of de Souza Neto et al. (1996). For more technical description of anti-locking algorithms, as applied to hydro-mechanical coupling, see section 8.

As volumetric locking is mainly important in nearly incompressible materials, the developments of many anti-locking schemes link to developments for hydro-mechanically coupled MPM for fully saturated porous materials as well as for MPM modelling of fluids. The developments include, among others Iaconeta et al. (2019), Kularathna & Soga (2017a,b), Li et al. (2014), Yang et al. (2018), Zhang et al. (2017, 2018), Zheng et al. (2013). Yet, it seems that the development of a robust and efficient Material Point Method algorithm which would remove problems related to locking, without introducing other errors and inaccuracies, is still a subject worthy of investigation.

### 4.6 Stability and accuracy in MPM

The stability of MPM is challenging to address, given its nonlinear nature. Currently either a Fourier-based analysis, e.g.(Brackbill 1988, 2015) or energy-conservation approach is taken. However, Wallstedt & Guilkey (2008) and Steffen et al (2010) rightly point out that the nonlinear nature of the MPM scheme makes classic linear stability analysis inappropriate. Berzins started to address this by noting that the standard time integration methods used in MPM correspond to the use of the semi-implicit Euler method, or symplectic Euler-A (Berzins 2018). The nonlinear stability of MPM is addressed by considering a one-dimensional model problem as an ordinary differential equations system in the values at particles and nodes. While this does not address the well-known issues to do with ringing that is described above, the aim is to consider how to bound the timestep when nonlinearity is taken into account.

This result reflects the form of the original equations in that Young's modulus constant E appears in its original form in the equation rises to the following equation:

$$dt \leq \frac{h\sqrt{m}}{4m(E + F_{\max p} + \| v \|) + 9h(F_{\max p} + \sigma_{\max p} + \frac{2}{l\sqrt{m}}\sigma F_n)}$$

where the mesh spacing is h , E is the Youngs modulus, m is the number of particles per interval and the maximum terms are maximum values of forces and stresses. In contrast, the usual speed of sound equation is given by $dt_{sound} = \frac{h}{\sqrt{E}}$. The full value of E in the equation above may be replaced by $\sqrt{E}$ if the equations are scaled so that this term appears in two equations , see Berzins (2018). This justifies the form if not the detail of the speed of sound approach. There is much still to be done regarding timestep choice for MPM based upon error control.

In the area of accuracy  Steffen et al. (2010) look carefully at space and time errors and what degree of accuracy is seen on simple example problems, thus paving the way for many of the developments that followed such as the use of higher order methods in both space and time.

Another of these recent developments is the use of functions reconstruction techniques. As material points move through the domain over time, the obtained accuracy and observed (spatial) convergence are limited. Therefore, different function reconstruction techniques have been applied in MPM to overcome this. The key idea of these techniques is to (globally) reconstruct a quantity of interest, based on the known values at the material point positions, and evaluate the reconstructed function at 'optimal' integration points. Typically, the Gaussian quadrature is adopted as a quadrature rule of choice to determine these positions. Moving Least Squares and cubic spline interpolation have been popular choices to use within MPM (Gan et al., 2018; Deborah Sulsky & Gong, 2016; Tielen et al., 2017; Tran et al., 2020). Alternatively, Taylor Least Squares (TLS) function reconstruction can be adopted. In contrast to other reconstruction techniques, TLS reconstruction in MPM allows for the preservation of total mass and linear momentum of the system and has been applied in combination with standard MPM, DDMPM and BS-MPM (Wobbes et al., 2019).

Typically, MPM adopts the row-sum lumped version of the consistent mass matrix. As the resulting matrix is a diagonal matrix, it allows for an efficient solution of the equation of motion at every time step. However, lumping the mass matrix is known to limit the spatial convergence to $O(h^2)$. Note that this only limits the spatial convergence in practice when high-order basis functions (e.g. B-splines or Powell-Sabin splines) are adopted combined with function reconstruction techniques. Furthermore, it might lead to spatial oscillations when applied with PS-MPM (de Koster et al., 2020). Finally, lumping the mass matrix hinders the conservation of energy and angular momentum (Love & Sulsky, 2006b).

### 2.3 General Local Space and Time Errors in MPM

As many of the equations are similar in the different formulations of MPM it is helpful to identify some of the local errors in these equations for future use. For simplicity only a one dimensional case with the original form of the MPM method is considered, but without the delta functions used originally with that method at the particles. The approach described builds on that of (Tran et al. 2010), (Steffen et al. 2010) for time local errors of (Berzins, 2019). Furthermore the errors accumulated over just one step starting with zero error are considered. The extension to the more general case requires taking into account that every solution variable now carries an error from the previous step. For simple cases this analysis is undertaken by (Steffen et al. 2010).

The first errors are those due to the mappings of momentum and mass from particles to grids as defined by equations (9) and (10) . These errors are caused by whichever coefficients $S_{vp}^{i}$ are used and may be denoted by $Ep_v^n$ and $Em_v^n$ respectively for momentum and mass. For example if linear interpolation is used we would expect that

$$Ev_p^n = Ch^2 \tag{0.1}$$

Where C is a generic constant that may be different each time it is used and generally depends on higher derivatives of the quantity being mapped. When cubic splines are used we would expect

$$Ev_v^n = Ch^4 \tag{0.2}$$

Similar expressions apply in the case of $Em_v^n$. The second main source of error is that in the nodal velocity as denoted by $EV_v^n$ at time $t_n$. There are two components to this error. The first is the approximation error, $Ea_v^n$ due to the use of the mapping coefficients $\nabla S_{vp}^{in}$ and depends on the difference between the approximate internal force $f_v^{\text{int}}$ and the true internal force $\tilde{f}_v^{\text{int}}$.

The other forces that may be presented are neglected for simplicity.

Steffen et al. consider particular instances of such errors in different cases. Hence given the acceleration formula in equation (14) ,

$$a_v^n = \frac{\mathbf{f}_v^{\text{int}}}{m_v} = -\frac{1}{m_v}\sum_{i=1}^{np} \boldsymbol{\sigma}_p^i \nabla S_{vp}^i V_p^i$$

$$Ea_v^n = \frac{\mathbf{f}_v^{\text{int}}}{m_v} - \frac{\tilde{\mathbf{f}}_v^{\text{int}}}{\tilde{m}_v}$$

$$LEV_v^n = \frac{dt^2}{2}\frac{da_v^n}{dt} + h.o.t.$$

An approximation to this is possible by using finite differences in time on the acceleration.

$$LEV_v^n = \frac{dt}{2}\left[ -\frac{1}{m_v^n}\sum_{i=1}^{np} \boldsymbol{\sigma}_p^{i,n} \nabla S_{vp}^{i,n} V_p^{i,n} + \frac{1}{m_v^{n-1}}\sum_{i=1}^{np} \boldsymbol{\sigma}_p^{i,n-1} \nabla S_{vp}^{i,n-1} V_p^{i,n-1} \right] + h.o.t.$$

The errors $Ea_v^n$ that arise from the use of the mapping coefficients $\nabla S_{vp}^i$ are one order less of the nodal spacing, h,. In other words for linear basis functions the error is

$$Ea_v^n = Ch \tag{0.3}$$

In the case of quadratic splines $Ea_v^n = Ch^2$, while for cubic splines $Ea_v^n = Ch^3$. The total local error from both space and time at the node over one timestep then given by

$$EV_v^n = dtEa_v^n + LEV_v^n.$$

The errors in the position, $EX_p^n$ , and the velocity $EV_p^n$ of the particles then depend on both the fundamental mapping error associated with the coefficients $S_{vp}^i$ and the nodal velocity error $EV_v^n$ that is mapped onto the equations (15) and (16) for updating both the position of the particles and their velocity.

$$EX_p^n = dt\sum_{i=1}^{nn} S_{vp}^n EV_v^n + EXsv_v^n + LEX_p^n$$

$$EV_p^n = m_p dt\sum_{i=1}^{nn} S_{vp}^n Ea_v^n + EVsv_v^n + LEV_p^n .$$

In these expressions the errors $EXsv_p^n$ and $EVsv_p^n$ are dependent on the choice of mapping coefficients $S_{vp}^i$ as above to give powers of the nodal spacing h. The time local errors $LEX_v^p$ and $LEV_v^p$ are $O(dt^2)$ from the application of the Forward Euler method to equations (15) and (16), respectively.

Finally the errors in the strain field as defined by equation (18) may be derived in much the same way as above. The strain field error at a particle is given by

$$E\dot{\varepsilon}_p = \frac{1}{2}\sum_{i=1}^{nn} \nabla S_{vp}^i EV_i^n + E\nabla S_{vp}^i V_i^n \tag{0.4}$$

In this case the error introduced by the gradient basis $\nabla S_{vp}^{in}$ is of the same order of accuracy as above, in other words one power of h less than the mapping associated with the coefficients $S_{vp}^i$.

In this first step we have assumed no errors in existing solution variables, it is straightforward if complex in terms of notation to extend the approach to the c as in which every solution variable carries and existing error.

While of course the challenge in any such analysis relies on a precise quantification of the different terms defined above and in the extension to very complex problems, the above framework shows that error growth in MPM is first of all nonlinear and secondly combines spatial and temporal errors in a complex and intimate way. This explains to some extent the comment in (Wallstedt and Guilkey,2008) that there was no advantage to be seen in applying better time integration techniques. The explanation is that unless more accurate time integration is combined with more accurate spatial discretization the benefits of better time integration will be lost. The analysis above confirms this as do the examples and experiments of (Steffen et al., 2010). It is of course inevitable that error analysis seldom applies to the complex problems solved in reality and for this we need to use the method of manufactured solutions. Nevertheless the framework presented here illustrates some of the challenges that exist and is sufficiently general to apply to other MPM methods.

### 4.7 Energy Conservation Errors in MPM

A key property of MPM calculations is energy conservations as considered first by Bardenhagen (2002) and it is shown that the standard MPM method gives second order energy conservation over a timestep or first order overall. The analysis of Love & Sulsky (2006a,b) extends these results and shows that energy conservation is possible if a full mass matrix is used. The same authors also show that using a lumped mass matrix gives second order locally energy conservation. Other improved time integration methods based upon a central difference approach are considered by Wallstedt & Guilkey (2010). The relationship between MPM time integration and symplectic time integration methods is considered by Berzins (2019). Such symplectic methods have good conservation properties and the well known Stormer-Verlet (Berzins, 2019) method has third order accuracy locally. This method is symplectic and very widely used in many applications such as molecular dynamics and planetary orbits and even dates back to Newton as was demonstrated by Feynman. Berzins applies this method to an MPM model problem and proves third order energy conservation locally. Numerical experiments comparing the approaches on a model problem show that the Stormer Verlet method has better accuracy and conservation properties than the standard symplectic Euler (Euler Cromer) method generally used in many MPM calculations. There is clearly scope for applying even higher order time integration methods.

### 4.8 Discussion

Material Point Method is still a relatively new numerical method, and the optimisation of the method algorithm for robustness and accuracy is still ongoing. There are quite some developments related to error corrections and convergence in recent years, including (González Acosta et al., 2020; Hammerquist & Nairn, 2017; Nairn et al., 2020; Steffen et al., 2010; Sulsky & Gong, 2016; Tran et al., 2020; Wobbes et al., 2019). Yet, it seems that still there is much to uncover related to method accuracy and that creation of a robust arbitrary high order algorithm

without any numerical issues and inaccuracies is something the MPM community will be working on in the coming years.

## 5. Contact and boundaries in the Material Point Method

Main contributors: Sołowski, Seyedan

One of the main advantages of the Material Point Method is that the contact between bodies is automatically detected, without any need for extra algorithms (Sulsky et al., 1994; Sulsky et al., 1995). This original non-slip contact stems from the fact that the material points data in each step is moved to the grid nodes, and the materials points within the grid shape function support area will be in contact (equations 7-14). This standard contact behaviour is little affected by interpolation used. However, when the particles have domains, the contact is initiated a bit earlier, as it is enough that part of the domain of the material point share the same grid node . Yet, the results are almost the same for original MPM and e.g. CPDI, see attached Video 2 /3.

Video 2 /3. Contact occurs when material points in two bodies share the same grid node, before the bodies starts actually being in contact (original MPM, left, CPDI, right).

In the original MPM algorithm, the boundary conditions are imposed directly on the grid nodes (by overwriting the values of e.g. velocity resulting from the algorithm and projecting back the set velocity), leading to a complete method. The original contact and boundary conditions formulations are still in use. Yet, in great many applications, the non-slip contact is insufficient. The advanced algorithms will lead to a more physical contact, as in Video 4/5.

See attached video 4/5. With special contact algorithm, contact occurs when the domains of material points starts touching. In the figure / video, algorithm of Seyedan & Sołowski (2019, 2021) is used, but any other advanced algorithm will lead to similar results.

Similarly, also the boundaries often need a more accurate and sophisticated implementation than just the imposition of the conditions on the grid nodes.

The needs drove the development of a number of contact algorithms. Bardenhagen et al. (2000, 2001) have perhaps first widely followed contact algorithm for frictional contact, allowing for non-slip condition, frictional slippage and zero tensile strength. This algorithm, though very successful, was improved to allow very large stiffness differences between materials and a more elaborate cohesive-frictional shearing condition by Ma et al. (2014). This contribution also includes penalty function reducing oscillations significantly in the contact computations, and thus reducing the need of damping, necessary for example in Andersen & Andersen (2010) and Nairn (2003). The contact algorithms above are still limited to enforcing the contact on the grid nodes and use a single grid, similarly to e.g. contact algorithm defined by Huang et al. (2011) and Ma et al (2010).

A different implementation of frictional contact, in which no damping is needed, is a multi-grid contact algorithm by Hu & Chen (2003), where contact condition is checked not by the centre of mass velocity field as in Bardenhagen et al. (2000, 2001), but by a sweep over the particles defined as contact particles, being on the outside of the modelled bodies. This leads to an efficient algorithm, especially in case of separation, which is achieved simply by moving the particles previously in contact back to separate grids. The algorithm proposed by Hu & Chen (2003), however, is meant mainly for large rotation and large displacement simulations, but had not been tested for very deformable bodies, where the outside layer of contact particles may be broken, leading to unphysical contact.

There are many more modifications of the contact algorithms, including recent work of Nairn et al. (2020), building on the earlier work of Hammerquist & Nairn (2018). This new contact algorithm improves the contact, decoupling it from grid, but instead relying on logistic regression methods. This contact, compared to previous grid based formulations, deteriorate less at the edges of the domains in contact, where normal may be difficult to find, while the logistic regression methods can find normal vector at any point of the domain. The contact algorithm given in (Nairn et al., 2020) also allows for a contact of multiple materials in a given node, something requiring special extensions in previous algorithms.

Somewhat similar idea, but based on B-splines instead of logistic regression methods has been presented by Bing et al. (2019). The paper highlights that imposing boundary conditions strictly on the grid nodes is sometimes inconvenient, as the boundaries may be complex. The proposed method allows for imposing boundary conditions on arbitrary shaped boundaries described by B-splines. The method is general for the case of Neumann boundary conditions, i.e. traction boundaries, and confined to implicit MPM in case of displacements boundaries. One downside of the approach is that, when imposing inhomogeneous Dirichlet boundary conditions the method requires a node-boundary stencil least squares minimisation problem to be solved to determine the nodal values that *best fit* the required solution on the boundary based on the basis functions of the background grid. This adds additional computational cost, but it allows for complex boundary conditions to be included within implicit MPMs. It seems that the extension of the method to contact is quite possible, as well as employing the technique for the creation of the material boundaries, replacing the logistic regression methods.

## 6. Fracture modelling in the Material Point Method

Main contributors: Adibaskoro, Seyedan, Soga

MPM offers great application potential in modelling dynamic fractures. Although MPM has been used for calculating parameters of fracture parameters (Tan & Nairn, 2002) and fracture formation (Zabala & Alonso, 2011), these contributions employ implicit crack modelling using symmetry condition or plasticity. A true explicit fracture modelling in MPM requires solving three separate problems. First, enhancing MPM for modelling explicit cracks. Second, developing methods for calculation of cracks parameters in MPM. Third, adding the capability of crack propagation in MPM. This section reviews the available methods and solutions for each of these problems.

### 6.1 MPM modelling with explicit crack

MPM requires modification to perform dynamic fracture analysis. Any numerical method for dynamic fracture mechanics should model the discontinuity of displacement and stress around the cracks. However , MPM is based on continuum mechanics and therefore inherently incapable of considering an explicit crack in the middle of the material or modelling discontinuity.

In common MPM algorithms, all material points send data to the nodes of one background grid. If there is a crack close to a node, material points from the opposite side of the crack send data to this node. MPM algorithm then combines the data from opposing side material points, updates the nodal momentum and returns the data back to the material points. As such, the material points from the opposite side of the crack affect each other and interact as if there is no crack. These interactions are unphysical and prevent MPM from modelling explicit cracks.

Nairn (2003) introduced CRAMP to solve a MPM problem with explicit crack. CRAMP is an enhancement to MPM and can model explicit cracks and their discontinuities. CRAMP uses separate and parallel grids around the explicit cracks. Each of these parallel grids receives information from material points on one side of cracks. Therefore, CRAMP does not combine

the data from opposing side material points, material points from the opposite side of crack do not affect each other and do not interact allowing the replication of crack discontinuity.

CRAMP requires two separate sub-algorithms. First, a sub-algorithm for determining which grid a material point close to the crack should sends data to. Such a sub-algorithm is computationally expensive and should be written efficiently. Therefore, Nairn (2003) provides an optimal line crossing sub-algorithm for this purpose. Second, a contact sub-algorithm is necessary for material points on the opposite side of cracks. In CRAMP, the opposite sides of cracks have separate grids and displacements. As such, the two sides might unphysically cross-over and a contact sub-algorithm is necessary to prevent such cross-over. Nairn (2003) modifies the contact algorithm of Bardenhagen et al. (2000, 2001) to provide a reliable contact method for crack surfaces.

CRAMP is a capable and efficient tool for fracture dynamic modelling. Simulations of CRAMP show good results when compared to experiments and Finite Element Method solutions (Nairn, 2003). Nairn (2007) shows the capability of CRAMP in radial crack propagation of wood, while Nairn (2016) uses CRAMP for simulation of chip curling as part of wood cutting process. Furthermore, CRAMP is computationally efficient and a well-written algorithm for it has around 10% higher computational cost than MPM (Nairn, 2003).

### 6.2 Fracture parameters and propagation

From establishing the capability of simulating discontinuity, the progression naturally went towards introducing crack propagation. J-integral was made possible with MPM and the discrete crack formulation (Guo & Nairn, 2004); a sensible choice to avoid calculations involving singularity around the crack tip. The analytical formulation for the J-integral was proposed by Cherepanov (1979), which maintains path independence despite the presence of

dynamic effects. The recommended integration path is to encircle the crack tip within a 4x4 grids, although any other arbitrary closed paths are, analytically speaking, equally correct.

With J-integral implemented, crack propagation becomes possible (Nairn & Guo, 2005). Although the energy release rate (G) from J-integral is compared to the critical value ($G_c$), no actual energy dissipation takes place due to the separation of crack surfaces. Instead, the stored elastic energy before propagation is released during propagation into the surrounding material in the form of mechanical stress waves. The loading process needs to be slow enough that crack would not propagate too quickly, allowing the resulting stress waves to dissipate via e.g. material damping or energy-absorbing boundaries. This phenomenon limits the application of the method to quasi-static cases.

In order to properly dissipate energy representing $G_c$, Nairn (2009) and Bardenhagen et al. (2011) introduce cohesive force immediately behind the crack tip. Consequently, physical phenomena such as energy dissipation, cohesive/process zone, and rising R-curve (at the start of propagation) are represented. Furthermore, without the need to wait for dissipation of mechanical stress waves , simulation of dynamic cases become possible.

### 6.3 Other Methods

Besides the techniques revolving around the concept of multiple background grids, several studies have shown successful implementation of various methods to simulate fracture in MPM. On the side of the discontinuous approach, Schreyer et al. (2002) proposed a method of fracture simulation by defining fracture surface and evolving traction as decohesion progresses, which paper was published before the first paper of CRAMP. The method requires the crack path to be known a priori, therefore the discussion is limited to simulating delamination problems. Liang et al. (2017) modified the particle displacement approximation by introducing the discontinuous enriched function to represent the crack and simulated the crack propagation

with J-integral and stress intensity factors. Another more recent discontinuous approach to fracture in MPM is published by Moutsanidis et al. (2019), which identifies separation via damage gradient partitioning similar to the method by Homel et al. (2017) but maintaining the use of only one background grid. Most of other developments in the discontinuity approach are related to the concept of CRAMP, i.e., multiple background grids.

Besides the discontinuous approach, the introduction of phase field, damage, and softening can also introduce a “smeared” discontinuity resembling a crack. Unlike the discrete approach, this continuous approach, in principle, retains continuity in the numerical discretization technique. However, a continuous approach may become more expensive than a discrete approach, as a sudden enough jump in the displacement field requires a significantly finer mesh. For example, the implementation of phase field for fracture simulation generally requires mesh size smaller than the process zone of the crack propagation. Figures 11 and 12 show examples of discrete and continuous approaches and how they differ.

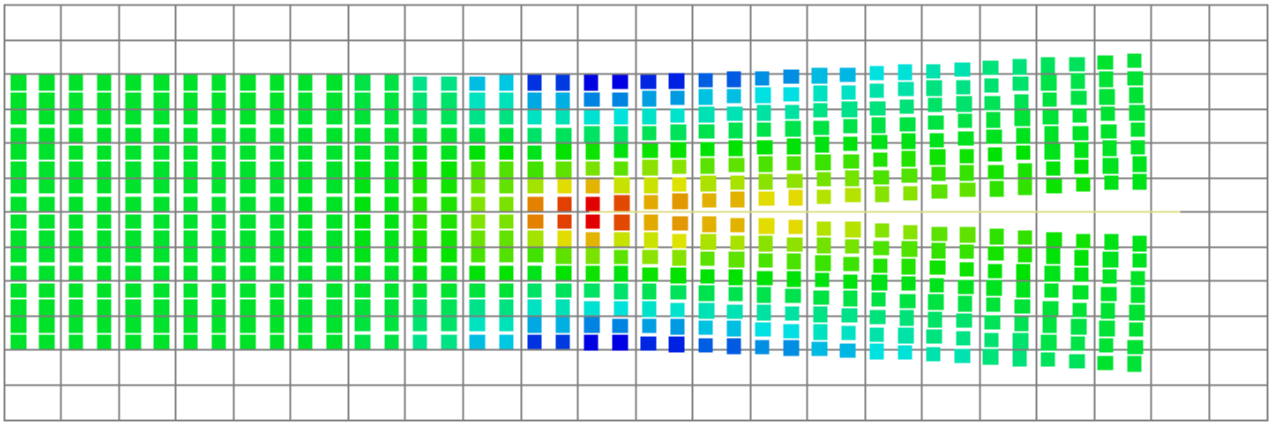


Figure 11 An example of discrete approach (CRAMP), fracture breaks continuity of the discretization method *Nairn (2003)*

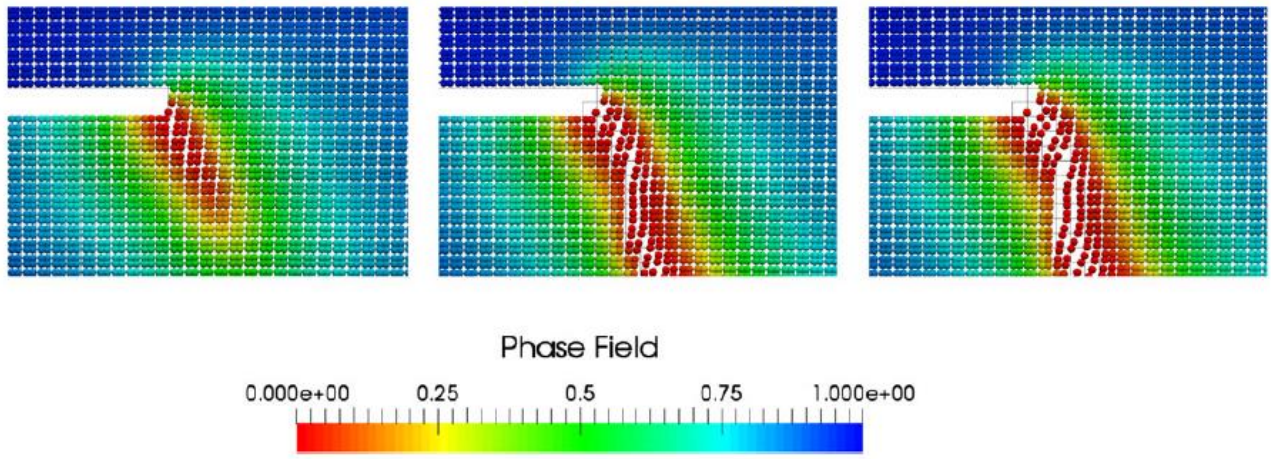



Figure 12 An example of continuous approach (Phase Field Fracture), discretization method still maintains continuity *Kakouris & Triantafyllou (2017)*

Kakouris & Triantafyllou (2017) introduced phase field with damage to MPM for the first time to model fracture behaviour, showing e.g. convincing pure mode I and pure mode II simulation results on a single-edge-notched specimen. In addition, the paper also introduced a method to identify an explicit crack path throughout the development of the phase field. Further advancement in this method allows for application in anisotropic medium (Kakouris & Triantafyllou, 2018). Kakouris and Triantafyllou (2019) then introduced the concept of surface energy, thus keeping the simulation consistent with Griffith's theory of fracture (Griffith, 1921). Wolper et al. (2019) also implemented phase field and damage for simulating dynamic fracture in MPM for the purpose of animation, resulting in more convincing destruction of jelly-like and other organic materials than traditional MPM. Cheon & Kim (2019) proposed an adaptive refinement process for the MPM coupled with a phase-field fracture model to simulate crack propagation in brittle materials.

Plasticity and softening are another approach for simulating material separation to some extent. Many simulations of landslides show shear bands resembling fractures and separation of chunks of material with plasticity, e.g. Mohr-Coulomb, Cam-Clay, and their derivatives. For

example, see e.g. (Alonso et al. 2015; Andersen & Andersen, 2010; Soga et al., 2016; Yerro et al., 2016). In addition, Zabala & Alonso (2011) simulated the fracture formation of Aznalcollar dam progressive failure employing the Mohr-Coulomb plasticity, while Shen (2009) used a Drucker-Prager model, a rate-dependent damage model and a bifurcation-based decohesion model to describe the behaviour of compression, tensile and localization respectively, and simulated the glass fragmentation under impact.

### 6.4 Benefits of MPM in fracture modelling

One of the advantages of MPM is the ability to simulate large deformation and displacement. Therefore, MPM inherently offers the capability of simulating large crack opening and closing, difficult to replicate in other numerical methods. Such capability is shown in the following simulation of a ring with radially-arranged initial cracks, which schematic shown in Figure 13. The simulation is part of a work in progress by Adibaskoro et al. (2020).

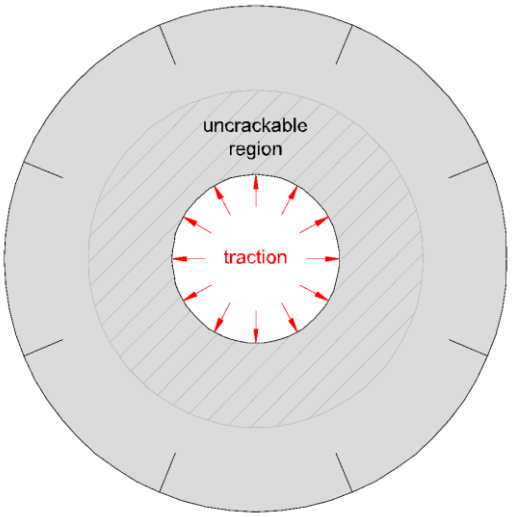
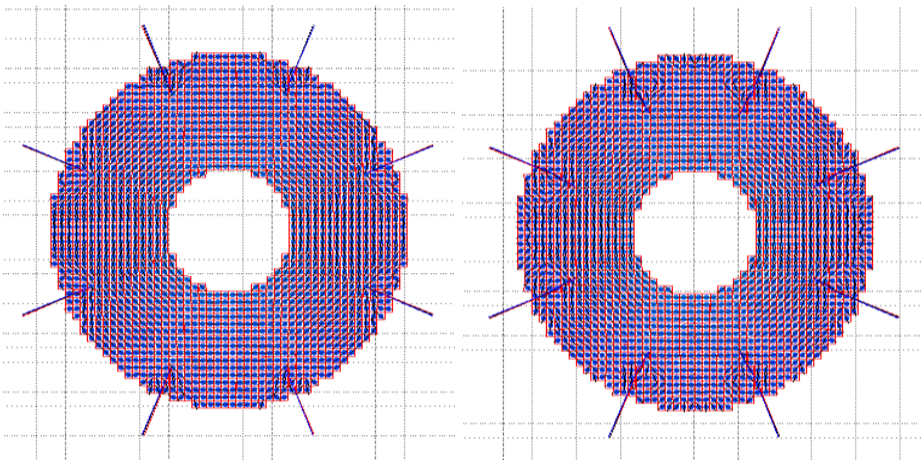



Figure 13 Ring simulation with radially-arranged initial cracks

Fracture is modelled by employing CRAMP, modified for efficient multiple crack path simulation. Rankine criterion (Rankine, 1857) determines whether propagation occurs, while

the direction of maximum principal stress dictates the direction of propagation. The simulation utilises CPDI MPM.

In the beginning, tension at the crack tips results in crack propagation up to the uncrackable region (see Figure 14). Further loading opens the crack up to the maximum displacement shown in Figure 15. Note the separating blue and red lines, indicating the crack surfaces. After the load is completely released, the ring goes back to roughly its initial size with the cracks closed (see Figure 16). Whole simulation is visible in video 6.



(a) (b)

Figure 14 In the beginning of simulation, before (a) and after (b) propagation occurs.

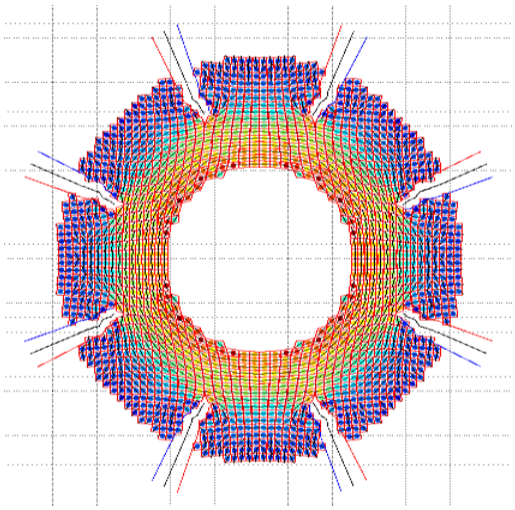


Figure 15 Maximum crack opening.

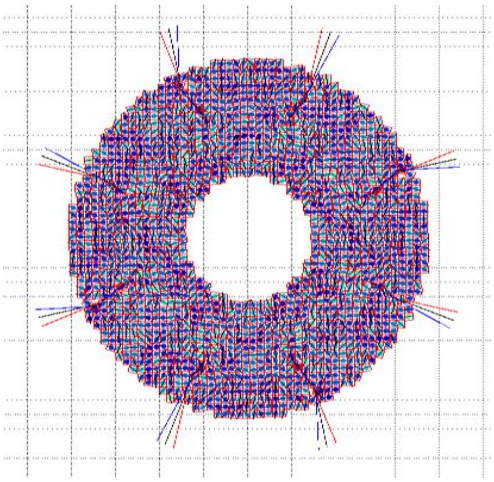


Figure 16 Cracks closing after load is released.

See attached video 6. Ring simulation with radially-arranged initial cracks

# 7. Two and three phase continua modelling in MPM

Main contributors: Tran, Soga

## 7.1 Overview of literature

MPM can be used to model coupled solid-fluid interaction problems. Recent development on the coupled hydro-mechanical MPM formulations can be categorized into two approaches: one-layer material-point formulation and double-layer material-point formulation. The one-layer material point formulation uses only a single set of material points to represent both the solid and liquid phases. For example, Zhang et al. (2008) proposed a velocity - pressure ($v_s$-$p_w$) formulation in the one-layer MPM framework and used it to solve a problem involving the impact of solid bodies in a saturated porous media. Similarly, Zabala & Alonso (2011) applied

the $v_s$-$p_w$ MPM formulation with strain softening Mohr-Coulomb to model the progressive failure of Aznalcollar dam. Zheng et al. (2013) implemented the $v_s$-$p_w$ MPM formulation for the Convected Particle Domain Interpolation. Pinyol et al. (2018) extended velocity - pressure ($v_s$-$p_w$) formulation to the velocity – pressure - temperature ($v_s$-$p_w$-T) formulation to study the thermal effects on landslide mobility. Apart from the velocity - pressure ($v_s$-$p_w$) formulation, the velocity - velocity $v_s$-$v_w$ scheme brings several advantages such as (i) simpler imposing of impervious boundary condition, (ii) no need for zero pressure detection and (iii) being suitable for large deformation at failure when the separations between liquid and solid phases may occur. Later, the $v_s$-$v_w$ formulation was extended into the three-phase formulation for modelling unsaturated soils by using the three momentum balance equations ($v_s$-$v_w$-$v_a$) of Higo et al (2010) with the coupled MPM–FDM, and by Yerro et al. (2015). An alternative approach is to adopt the degree of saturation into the effective stress tensor with only two momentum balance equations ($v_s$-$v_w$) to model the rainfall-induced slope failure by Abe et al. (2014), Bandara & Soga (2015) and Wang et al. (2018).

Apart from the one-layer formulation for multi-phase coupling, another alternative way to model the soil-water interaction is to use a double-point MPM formulation. It utilises two sets of Lagrangian material point layers to consider solid and water layers. Mackenzie-Helnwein et al. (2010) consider the momentum balance equation for each phase, along with drag interaction models to solve the acceleration of each phase in the Eulerian formulation. This approach requires drag interaction models to capture the interaction between phases through drag forces that occur due to the relative motion of the phases. Abe et al. (2013) and Bandara & Soga (2015) use MPM to model saturated soil with a solid skeleton and pore water material point layers while using the effective stress approach to capture the solid skeleton behaviour. Abe et al. (2013) considered the Biot's mixture theory formulation while neglecting the relative acceleration of water with respect to the solid skeleton. This requires the velocity of the water

phase to be computed at each time step using the generalised Darcy's equation, which may lead to limitations in modelling rapid motions. Bandara & Soga (2015) derived a coupled MPM formulation based on the mixture theory approach considering the relative acceleration of water with respect to the solid skeleton. This method allows modelling of extremely rapid flows while also conserving the mass of both solid skeleton and pore water.

Use of two material point layers is extremely helpful when modelling soil–water interaction problems such as submarine landslides, dredging and erosion modelling. The two-layer formulation provides opportunities to solve dynamic flow problems in geomechanics such as internal and external erosions as well as fluid flow failures (seepage failures). The two-layer approach was adopted to study different strategies of modelling dragging interactions; for solid-fluid interaction in animation (Gao et al., 2017; Tampubolon et al., 2017); combine with the DDMP ( Tran et al. 2017; Tran & Sołowski 2019) to model the large consolidation; combine with the GIMP (Liu et al., 2017) and to develop the two-layer approach with a thermodynamic constitutive model (Baumgarten & Kamrin, 2019). For instance, it allows modelling of fluidisation of soil particles at the water interface and simulations of internal erosion by transferring a portion of soil particle mass to fluid particle. Baumgarten & Kamrin (2019) use this model to show complex fluid–grain interactions under dilation and flow such as viscous inertial rheology of submerged wet grains under steady shearing flows and the change in the effective viscosity of the fluid due to the presence of suspended grains. The existence of two layers of material points results in greater computation time, and careful modelling of interface regions that separate saturated soil from dry soil and free water is required. Table **Error! Reference source not found.**2.

Table 2. Summary of coupled hydro-mechanical MPM formulation

| Type | Method | Interpolation based on | Coupled formulation | Application |
|---|---|---|---|---|
| single-point for saturated soils | Zhang et al. (2009) | Linear basis | $v_s$-$p_w$ | solid bodies impact to saturated porous media |
| | Zabala & Alonso (2011) | Linear basis + one-point quadrature | $v_s$-$p_w$ | Aznalcollar dam failure |
| | Zheng et al. (2013) | CPDI | $v_s$-$p_w$ | solid bodies impact to saturated porous media |
| | Jassim et al. (2013) | Linear basis + one-point quadrature | $v_s$-$v_w$ | Wave attack on sea dike |
| | Bandara et al. (2016) | Linear basis + one-point quadrature | $v_s$-$v_w$ | Rainfall-induced slope failure |
| | Wang et al. (2018) | Linear basis + B-bar | $v_s$-$v_w$ | Rainfall-induced slope failure |
| | Pinyol et al. 2018 [27] | Linear basis + one-point quadrature | $v_s$-$p_w$-T | Thermal effects in landslide |

| | | | | |
|---|---|---|---|---|
| single-point for unsaturated soils | Higo et al. (2010) | Linear basis | $v_s$-$v_w$-$v_a$ | Seepage failure embankment |
| | Yerro et al. (2015) | Linear basis + one-point quadrature | $v_s$-$v_w$-$v_a$ | Rainfall-induced slope failure |
| double-point for saturated soils | Mackenzie-Helnwein et al. (2010) | Linear basis | $v_s$-$v_w$ | Study different phase interactions |
| | Abe et al. (2014) | GIMP | $v_s$-$v_w$ | Seepage failures |
| | Bandara & Soga (2015) | Linear basis + one-point quadrature | $v_s$-$v_w$ | Seepage failures |
| | Tran et al (2017), Tran & Sołowski (2017, 2019) | DDMP | $v_s$-$v_w$ | Large strain consolidation |
| | Liu et al. (2017) | GIMP | $v_s$-$v_w$ | Seepage failures |
| | Tampubolon et al. (2017) | APIC + B-spline | $v_s$-$v_w$ | Dam breach animation |
| | Gao et al. (2018) | APIC + B-spline | $v_s$-$v_w$ | Particle-Laden flow animation |
| | Müller & Vargas (2019) | GIMP | $v_s$-$v_w$ | Slope under a rock block impact |

| | | | | |
|---|---|---|---|---|
| | Baumgarten & Kamrin (2019) | Cubic Spline | $v_s$-$v_w$ | Saturated mixture in many flow regimes |
| | Yamaguchi et al. (2020) | B-Spline | $v_s$-$v_w$ | Wave collision to sandpile |

## 7.2 Governing Equations for the hydro-mechanical coupling

This section summarizes two different coupled hydro-mechanical formulations including the velocity - pressure ($v_s$-$p_w$) formulation and the velocity - velocity ($v_s$-$v_w$) formulation.

### 7.2.1 **Notation**

| Solid phase | | Liquid phase | |
|---|---|---|---|
| $\rho_s$ | Solid density | $\rho_w$ | Liquid density |
| $a_s$ | Solid acceleration | $a_w$ | Liquid acceleration |
| $v_s$ | Solid velocity | $v_w$ | True liquid velocity |
| $b$ | body acceleration | $q_w$ | Darcy velocity |
| $n$ | porosity | $p_w$ | pore pressure |
| $k$ | permeability | $K_w$ | bulk modulus of liquid phase |
| $g$ | gravitational acceleration | $K_s$ | bulk modulus of solid phase |

#### 7.2.2 Velocity - pressure ($v_s$-$p_w$) balance equations

Considering the relative acceleration of liquid with respect to solid skeleton is negligible ($a = a_w = a_s$). The momentum balance equation for the mixture (solid and liquid phase) is:

$$\rho \mathbf{a} = \nabla . \boldsymbol{\sigma} + \rho \mathbf{b} \tag{43}$$

where $\boldsymbol{\sigma}$ is the total stress, $\rho = (1 - n)\rho_s + n\rho_w$. The liquid momentum balance equation is:

$$\mathbf{q_w} = -\frac{k}{\rho_w}\left(\nabla p_w - \rho_w \mathbf{b} + \rho_w \mathbf{a}\right) \tag{44}$$

The gradient of pore water pressure is derived from the fluid mass balance equation leading to:

$$\frac{p_w}{t} = -\frac{K_w}{n}[\nabla . q_w + \nabla . v_s] \tag{45}$$

#### 7.2.3 Assumptions

Several assumptions are made for the hydro-mechanical coupling MPM formulation including:

- Solid and Liquid are described in a Lagrangian formulation,
- Solid grains are incompressible while the fluid is compressible,
- Isothermal condition,
- No mass exchanged between solid and liquid phase,
- Terzaghi's effective stress is applied,
- Darcy law is applied.

#### 7.2.4 Velocity - velocity ($v_s$-$v_w$) balance equations

8. The momentum balance equation for the liquid phase is:

$$n\rho_w \mathbf{a_w} = -n\nabla p_w + n\rho_w \mathbf{b_w} - \mathbf{f_d} \tag{46}$$

9. where $\mathbf{f_d}$ is the dragging force. The momentum balance equation for the solid phase is:

$$(1 - n)\rho_s \mathbf{a_s} = \nabla . \boldsymbol{\sigma}' - (1 - n)\nabla p_w + (1 - n)\rho_s \mathbf{b_s} + \mathbf{f_d} \tag{47}$$

The solid mass balance equation is:

$$\frac{n}{t} = (1 - n)\nabla . v_s \tag{48}$$

The gradient of pore water pressure is derived from the fluid mass balance equation leading to:

$$\frac{p_w}{t} = -\frac{K_w}{n}\nabla.[nv_w + (1-n)v_s] \tag{49}$$

Decomposing the right-hand slide results to:

$$\frac{p_w}{t} = -\frac{K_w}{n}[n\nabla.v_w + (1-n)\nabla.v_s + (v_w - v_s)\nabla n] \tag{50}$$

#### 9.1.1 Terzaghi effective stress concept

Reviewing the saturated porous media, the total stress in the saturated mixture can be defined with the Terzaghi's effective stress as:

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}' - \mathbf{p_w} \tag{51}$$

where $\boldsymbol{\sigma}'$ is the effective stress and $\mathbf{p_w} = \mathrm{p_w}\mathbf{I}$ with $\mathbf{I}$ is the identity matrix and $\boldsymbol{p_w}$ is the value of the pore water pressure. The momentum balance equations are written separately for each phase.

#### 9.1.2 Darcy dragging force

The dragging force determined from the Darcy law with a permeability $k$ as follows:

$$\mathbf{f_d} = \frac{\mathrm{n}^2\rho_\mathrm{w}\mathrm{g}}{\mathrm{k}}(\mathbf{v_w} - \mathbf{v_s}) \tag{52}$$

#### 9.1.3 Stability of numerical solution

In the explicit MPM formulation, the Courant-Freidrichs-Levy condition (CFL) is commonly used to determine an appropriate time increment to obtain a stable numerical solution. The time increment is selected regarding the stiffness of the materials as (Bandara & Soga, 2015):

$$t = L/c \tag{53}$$

where c is calculated as:

$$c = \sqrt{\frac{\left(\frac{n}{K_w} + \frac{1-n}{K_s}\right)^{-1} + \frac{4}{3}G}{(1-n)_{\ s} + n_{\ w}}} \tag{54}$$

However, the CFL condition, which was used for the linear stability analysis, does not always ensure a stable non-linear solution for the MPM. Therefore, Jiang et al. (2017) and Berzins (2018) proposed formulations allowing to calculate the critical time step depending on the maximum velocity of the material points. Furthermore, for the low permeability materials, the permeability time dependent criteria should be satisfied as:

$$\Delta t = \frac{2\tilde{\rho} k}{\rho_w} \tag{55}$$

where $\tilde{\rho} = (1-n)\rho_s + n\rho_w + \left(\frac{1}{n} - 2\right)\rho_w$ (Mieremet et al., 2016).

### 10.1 Volumetric locking

Some instabilities in the Finite Element Method, also occur in the MPM. Mast et al. (2012) demonstrated a kinematic locking, including a volumetric locking for incompressible materials. For the explicit hydro-mechanical coupling MPM formulation, the water bulk modulus is extremely high, leading to volumetric locking behaviour in a similar way to one-phase MPM formulation with nearly compressible materials. To overcome the volumetric locking in the explicit hydro-mechanical coupling MPM formulation, Yang et al. (2018) proposed the cell-based algorithm, which averages the strain rate of the material points in a cell. Then, the filtered strain rate $\dot{\varepsilon}^{t}_{p,filter}$ will be calculated as:

$$\delta\varepsilon^{t+1}_{p,filter} = \delta\varepsilon^{t+1}_{c} \quad \text{and} \quad \delta\varepsilon^{t+1}_{c} = \frac{\sum_p m_p \delta\varepsilon^{t+1}_{p}}{\sum_p m_p} \tag{56}$$

Alternatively, the node-based algorithm, on the other hand, calculates the filtered strain rate by a nodal interpolation as follows:

$$\delta\varepsilon^{t+1}_{p,filter} = \sum_i S_{ip}\delta\varepsilon^{t+1}_{i} \quad \text{and} \quad \delta\varepsilon^{t+1}_{i} = \frac{1}{m^{t+1}_{i}}\sum_p S_{ip} m_p \delta\varepsilon^{t+1}_{p} \tag{57}$$

Bandara & Soga, (2015) and Wang et al. (2018) have been adopted the B-bar interpolation by replacing the gradient mapping matrix B-matrix

$$\mathbf{B}_{sp} = \begin{bmatrix} \frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{x}} & \mathbf{0} \\ \mathbf{0} & \frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{y}} \\ \frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{y}} & \frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{x}} \end{bmatrix} \tag{58}$$

with the B-bar matrix, for example in the plane strain configuration) as:

$$\mathbf{B}_{sp} = \begin{bmatrix} \frac{\mathbf{1}}{\mathbf{2}}\left(\frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{x}} + \frac{\delta \mathbf{S_i}(\mathbf{x_c})}{\delta \mathbf{x}}\right) & \frac{\mathbf{1}}{\mathbf{2}}\left(\frac{\delta \mathbf{S_i}(\mathbf{x_c})}{\delta \mathbf{y}} - \frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{y}}\right) \\ \frac{\mathbf{1}}{\mathbf{2}}\left(\frac{\delta \mathbf{S_i}(\mathbf{x_c})}{\delta \mathbf{x}} - \frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{x}}\right) & \frac{\mathbf{1}}{\mathbf{2}}\left(\frac{\delta \mathbf{S_i}(\mathbf{x_c})}{\delta \mathbf{y}} + \frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{y}}\right) \\ \frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{y}} & \frac{\delta \mathbf{S_i}(\mathbf{x_{sp}})}{\delta \mathbf{x}} \end{bmatrix} \tag{59}$$

On the other hand, Coombs et al. (2018) adopted the F-bar method (de Souza Neto et al., 1996) developed for the Finite Element Method. If the liquid phase is modelled by using the weakly compressible material, the filtered deformation gradient $\mathbf{F}_{\mathrm{p,filter}}^{t+1}$ is calculated as follows:

$$\mathbf{F}_{\mathrm{p,filter}}^{t+1} = \left(\frac{\mathrm{J_c}}{\mathrm{J_p}}\right)^{1/3} \mathbf{F}_{\mathrm{p}}^{t+1} \tag{60}$$

where $\mathrm{J_p} = \det\left(\mathbf{F}_{\mathrm{p}}^{t+1}\right)$ is the determinant of the updated deformation gradient and $\mathrm{J_c} = \det(\mathbf{F}_{\mathrm{c}}^{t+1})$ is the determinant of the updated deformation gradient at the centroid of the cell, which is approximated by the ratio of the current volume $\mathrm{V_p^{t+1}}$ and initial volume $\mathrm{V_p^o}$ of all material points $\mathrm{n_p}$ in that cell as:

$$\mathrm{J_c} = \frac{\sum_{\mathrm{p=1}}^{\mathrm{n_p}} \mathrm{V_p^{t+1}}}{\sum_{\mathrm{p=1}}^{\mathrm{n_p}} \mathrm{V_p^o}} \tag{61}$$

Then, the velocity gradient is modified to be consistent with the modified deformation gradient as follows:

$$\nabla \mathbf{v}_{\mathrm{p,filter}}^{t+1} = \nabla \mathbf{v}_{\mathrm{p}}^{t+1} + \mathbf{I}\left(\frac{\log\left(\frac{\mathrm{J_c}}{\mathrm{J_p}}\right)}{3\mathrm{dt}}\right) \tag{62}$$

where **I** is the identity matrix and dt is the time step.

### 10.2 Treatment of pore pressure oscillation

The common approach to deal with the pore water pressure oscillation in the Material Point Method is to average pore water pressure of the cell (Bandara & Soga, 2015) instead of the pore pressure of the material points as follows:

$$\mathrm{p_{w_c}^{t+1}} = \frac{\sum_{\mathrm{p}} \mathrm{p_{w_p}^{t+1}}}{\mathrm{N_p}} \tag{63}$$

where $N_p$ is the total number of the material points in a cell. Then, the one-point quadrature rule will be used to calculate the internal forces of the liquid phase. This approach is simple to use but the pore water pressure in a cell is constant leading to the first-order accuracy. In general, the pore-water pressure comes from the null-space instability of the Material Point Method as demonstrated by Tran et al (2017) and Tran & Sołowski (2019) to remove the null space errors, a null space filter is proposed to filter the pore pressure oscillations by using the dual-domain technique and a QR decomposition. In this method, the liquid internal forces are updated as follows:

$$\mathbf{f}_{\mathrm{wi}}^{\mathrm{int,t}} = \sum_{\mathrm{wp}=1}^{\mathrm{N_{wp}}} \nabla \mathrm{S_i}\left(\mathbf{x}_{\mathrm{wp}}^{\mathrm{t}}\right) \alpha\left(\mathbf{x}_{\mathrm{wp}}^{\mathrm{t}}\right) \mathbf{p}_{\mathrm{wp}}^{\mathrm{t}} \mathrm{V}_{\mathrm{wp}}^{\mathrm{t}} - \sum_{\mathrm{j}=1}^{\mathrm{N_n}} \mathbf{p}_{\mathrm{wj}}^{\mathrm{t}} \cdot \int_{\Omega_w} \mathrm{S_j} \nabla \mathrm{S_i} \mathrm{dv} \tag{64}$$

where $\left(x_{\alpha p}^{t}\right)$ is a weighted function evaluated at the material point $x_{\alpha p}^{t}$ to switch different version of MPM including MPM, GIMP and DDMP. The nodal pore water pressures for the liquid phase $p_{wj}^{t}$ at the neighbour nodes j are calculated as:

$$\mathbf{p}_{\mathrm{wj}}^{\mathrm{t}} = \frac{1}{\mathrm{V_{wj}}} \sum_{\mathrm{wp}=1}^{\mathrm{N_{wp}}} \mathrm{S_j}\left(\mathbf{x}_{\mathrm{wp}}^{\mathrm{t}}\right) \left(1 - \alpha\left(\mathbf{x}_{\mathrm{wp}}^{\mathrm{t}}\right)\right) \mathbf{p}_{\mathrm{wp}}^{\mathrm{t}} \mathrm{V}_{\mathrm{wp}}^{\mathrm{t}} \tag{65}$$

The liquid strain increments are updated using the dual-domain, the first is the particle-based liquid strain increments are calculated as follows:

$$\Delta\boldsymbol{\varepsilon}_{\mathrm{wp},1}^{\mathrm{t+1}} = \sum_{\mathrm{i=1}}^{\mathrm{N_n}} \left( \nabla \mathrm{S_i}\left(\mathbf{x}_{\mathrm{wp}}^{\mathrm{t}}\right)\mathbf{v}_{\mathrm{wi}}^{\mathrm{L}} + \left(\nabla \mathrm{S_i}\left(\mathbf{x}_{\mathrm{wp}}^{\mathrm{t}}\right)\mathbf{v}_{\mathrm{wi}}^{\mathrm{L}}\right)^{\mathrm{T}} \right) \tag{66}$$

The second is the grid-based liquid strain increments are calculated as follows:

$$\Delta\boldsymbol{\varepsilon}_{\mathrm{wp},2}^{\mathrm{t+1}} = \sum_{\mathrm{i=1}}^{\mathrm{N_n}} \mathrm{S_i}\left(\mathbf{x}_{\mathrm{wp}}^{\mathrm{t}}\right)\Delta\boldsymbol{\varepsilon}_{\mathrm{wj}}^{\mathrm{t}} \tag{67}$$

where nodal strain increments $\Delta \overset{t}{_{wj}}$ calculated as:

$$\Delta\boldsymbol{\varepsilon}_{\mathrm{wj}}^{\mathrm{t}} = \frac{1}{\mathrm{V_{wj}}} \sum_{\mathrm{i=1}}^{\mathrm{N_n}} \mathbf{v}_{\mathrm{wi}}^{\mathrm{L}} \int_{\Omega_{\mathrm{w}}} \mathrm{S_j} \nabla \mathrm{S_i} \mathrm{dv} \tag{68}$$

The final strain increment vectors $\Delta \overset{k+1}{_{\alpha p}}$ is calculated with the weighted function $(x)$:

$$\Delta\boldsymbol{\varepsilon}_{\alpha\mathrm{p}}^{\mathrm{t+1}} = \alpha\left(\mathbf{x}_{\alpha\mathrm{p}}^{\mathrm{t}}\right)\Delta\boldsymbol{\varepsilon}_{\alpha\mathrm{p},1}^{*\,\mathrm{t+1}} + \left(1 - \alpha\left(\mathbf{x}_{\alpha\mathrm{p}}^{\mathrm{t}}\right)\right)\Delta\boldsymbol{\varepsilon}_{\alpha\mathrm{p},2}^{\mathrm{t+1}} \tag{69}$$

where $\Delta\boldsymbol{\varepsilon}_{\alpha\mathrm{p},1}^{*\,\mathrm{t+1}}$ is the null-space filtered value of the particle-based liquid strain increments. Details of the null-space filter is described in detail by Tran & Sołowski (2019).

Jassim et al. (2013) first applied a fractional time stepping method originally developed for mixed finite elements for dynamic poromechanical problems. This method performs well for dynamic problems in the undrained limit, but it is conditionally stable in time and does not cover problems in drained conditions. More recently, Yamaguchi et al. (2020) implemented this fractional step projection method for the time discretization of the momentum equation of the water phase showing the robustness and efficiency can be improved. Zhao & Choo (2020) instead used the polynomial pressure projection (PPP) method, which was initially developed for mixed finite elements for coupled poromechanics. Stabilization terms are augmented to the balance of mass to make equal-order mixed discretization stable even in the undrained limit. They implemented it in a fully-implicit MPM code and showed the use of standard low-order interpolation functions coupled with this stabilization method for undrained and drained poromechanical problems. Both techniques are used to reduce the pressure oscillations and strict restriction of the time step size due to the high bulk modulus of water and low permeability.

# 11. Coupling Material Point Method with other methods

Main contributors: Sołowski, Berzins, Guilkey

## 7.1 Coupling with the Finite Element Method

The coupling of Material Point Method with Finite Element Method in a single simulation is quite common. Such coupling allows to use the benefits of Finite Element Method at small strains and displacements, while using MPM in the part of domain where displacements and deformations are very large. There are many formulations for the coupling, but perhaps the formulation of (X. Zhang et al., 2006) with further developments (Y. P. Lian et al., 2011; Y.P. Lian et al., 2011; Yan Ping Lian et al., 2014) , culminating in (Chen et al., 2015), is the one most worthy of investigation. The coupling uses the multi-mesh concept, with the coupling done essentially via contact algorithm. The contact between material points and finite elements is treated as particle to surface Coulomb friction contact developed by (Z. T. Ma et al., 2010). This means that the consistent meshing between FEM and MPM domains is not needed, while the coupling efficiency is much higher than that in previously developed methods. Further evolution of this algorithm include (Y. Lian & Zhang, 2018), while an alternative algorithm is given e.g. by (Y.-J. Cheon & Kim, 2018).

## 7.2 Coupling with Discrete Element Method

The Material Point Method and Discrete Element Method for some time has been seen as competitors. That is incorrect – the methods base on very different assumptions, as MPM is a continuous method, while Discrete Element Method bases on modelling of number of usually rigid particles. The methods has been compared side to side many times, including, among others, (Ceccato et al., 2018; Coetzee, 2014; Coetzee et al., 2007; Kumar et al., 2013, 2017) Yet, as those methods aim at different description of material, it is sometimes of use to couple

them together in a single simulation. such coupling often takes as basis coupling algorithms of Discrete Element Method with the Finite Element Method, e.g. (N. Guo & Zhao, 2014).

Recent examples of MPM-DEM coupling include (W. Liang & Zhao, 2019; Liu et al., 2018; Y. Yang et al., 2017) Taking the most recent contribution by (W. Liang & Zhao, 2019) which extends and improves the approach of (Liu et al., 2018), coupling is achieved by computing stress based on computing the incremental displacement gradient, transferring it to the representative volume element made of Discrete Element particles, and based on calculations with Discrete Elements, the resulting stress transferred back to the MPM solver. Other option for coupling is related to use of interfaces, and solving a contact problem when the discrete element particles are entering MPM zone, similarly to (Y. Yang et al., 2017).

### 7.3 Coupling with Computational Fluid Dynamics

While MPM alone is capable of some very challenging fluid-structure interaction (FSI) simulations, such as shaped charge jet formation (where the explosive products of reaction are largely contained) general fluid flow calculations are typically best done with Eulerian algorithms. FSI approaches are typically divided into “separated” and “averaged”. In the former, the fluid flow takes place in a domain that is distinct from the solid mechanics portion of the simulation, with interactions only occurring at the interface between the two domains. In the latter, there is a possibility that all material phases can exist in some quantity at any point in a common domain. It is in this approach that MPM has found utility.

Kashiwa and Rauenzahn described a multi-material compressible flow formulation in a series of technical reports and publications (Kashiwa & Rauenzahn, 1994a, 1994b, Kashiwa, 2001). There, a set of equations are developed for the conservation of mass, momentum and energy for an arbitrary number of materials, any or all of which may exist at any point in the domain at any time. Along with these quantities, evolution equations are proposed for temperature,

specific volume, volume fraction, deviatoric stress and pressure. These equations are solved using a "cell-centered", multi-material ICE (for Implicit, Continuous fluid, Eulerian) algorithm that uses operator splitting to divide each simulation timestep into a Lagrangian portion and an advection step. In the Lagrangian phase, the effects of the physical processes that an element of material is subjected to, including interaction with other materials, are computed. In the Eulerian phase, the updated material state is moved according to local velocities.

In FSI simulations, which are a special case of multi-material flow, MPM can fill a few important roles. First, material points provide a place to store history dependent material state data. Second, since MPM advection is free from diffusion, the interface between fluid and solid is maintained. Implementation details of this approach are given by Guilkey et al. (2007), but the basic idea is that, at the beginning of a timestep, solid material data is projected to the nodes of the computational grid in the usual MPM fashion. From there, these data are collected at the cell centers of the grid. Next, the Lagrangian portion of the calculation is carried out for all materials, after which change in state for the solid materials are interpolated back to the nodes, and the particles are updated in the usual MPM manner. Meanwhile, the fluid material is mov ed using an Eulerian advection scheme. The basic steps of the algorithm are:

2. Interpolate Particle state to grid. MPM
3. Compute the equilibrium pressure. ICE
4. Compute face centered velocities for the Eulerian advection. ICE
5. Compute sources of mass, momentum and energy as a result of phase changing chemical reactions. BOTH
6. Compute an estimate of the time advanced pressure. ICE
7. Calculation of face Centered Pressure using a density weighted approach. ICE
8. Material Stresses calculation. MPM

9. Compute Lagrangian phase quantities at cell centers. ICE
10. Calculate Momentum and heat exchange between materials. BOTH
11. Compute the evolution in specific volume due to the changes in temperature and pressure during the foregoing Lagrangian phase of the calculation. ICE
12. Advect Fluids For the fluid phase. ICE
13. Advect Solids For the solid phase, interpolate the time advanced grid velocity and the corresponding velocity increment back to the particles, and use these to advance the particle's position and velocity, respectively. MPM

This technique has been implemented in Uintah (Parker et al., 2006) as MPM-ICE, and has been used to investigate FSI scenarios ranging from explosions of energetic devices to flow over human vocal folds and the dynamic growth and shrinkage of silicon battery anodes (Gritton et al., 2017). Additionally, a variant of this approach is frequently used in which the solid material, represented by MPM particles, is used as an embedded boundary condition, enabling easy insertion of complex geometry into a flow field.

# 8. Numerical Examples

Main contributors: Tran, Guilkey, Seyedan

## 9.1 Landslide simulations

In 1994, a sensitive clay landslide occurred in Sainte-Monique, Quebec. The landslide was characterized as a spread with horsts and grabens observed on site. The spreads could be classified as an upward progressive failure because the trigger mechanism (e.g., erosion, excavation) causes a horizontal failure surface propagating upward. This failure includes two processes: (i) the propagation of a horizontal quasi-static failure surface (shear band) and (ii) the extension and dislocation of the soil mass above the remoulded shear surface, forming hosts and grabens. We use the Uintah software with the GIMP interpolator to replicate the spreads. The soil model and MPM model were coupled and validated with various benchmarks (Tran et al., 2017, Tran et al. 2017), see Video 7.



fallcone.mpg

See attached video 7. Fall cone validation of the MPM for sensitive clays (Tran et al., 2017).

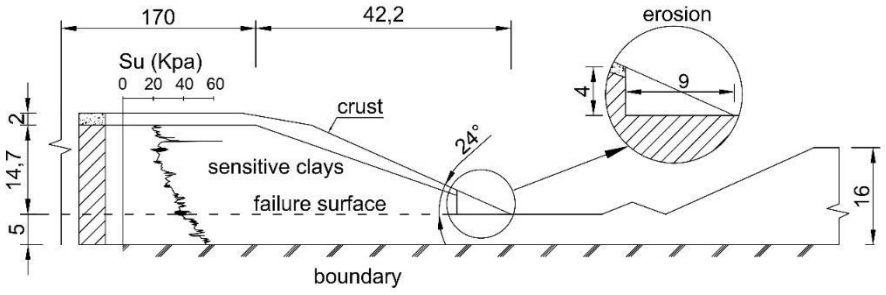



Figure 17 Schematic of the slope

Figure 17 shows the schematic of a 2D plane strain numerical model. The initial height of the slope was about 16.7m with the inclination of approximately 24 degrees. The unit weight of the clay is 16 kN/m3. In the numerical model, the square grid resolution is 0.2 x 0.2m with 4

material points per cell. The left and the right boundary are fixed in the horizontal direction while the bottom boundary is fixed in the vertical direction. The initial stress condition was generated in the drained condition with typical drained shear strength of Canadian sensitive clays (cohesion (c') and friction angle (\') of 10 kPa and 25-40° (Locat et al., 2015). Because the spread failure occurs rapidly, the soils are modelled in the undrained condition with undrained shear strength obtained from soil investigations (Locat et al., 2015).The erosion at the toe of the slope triggered the failure of the slope. The progressive failure simulation represents the erosion as a small amount of soil excavated on the toe of the slope.

### 9.1.1 Soil model

The undrained behaviour of clay is described by an elasto-plastic Tresca material model with a non-associated flow rule. Some important mechanical behaviours of clay are supplemented in the model, including the strain-rate effect, the shear strength degradation effect on the sensitive clays. The effect of the strain rate was captured by using a power law with the strain rate parameter  proposed by Einav and Randolph (Einav & Randolph, 2005):

$$\frac{s_u}{s_{u,ref}} = \left(\frac{\delta\gamma}{\delta\gamma_{ref}}\right)^{\beta} \tag{70}$$

Also, the dynamic shear modulus depends on the shear strain rate (Sorensen et al., 2010) as:

$$G_u(\delta\gamma) = G_{u,ref}\left(\frac{\delta\gamma}{\delta\gamma_{ref}}\right)^{\beta} \tag{71}$$

The same parameter  keeps the elastic shear strain value constant when the shearing rate changes. In the numerical model, the shear strain rate  is computed from the strain rate tensor as:

$$\delta\gamma = \frac{1}{2}\sqrt{2\left(\delta\varepsilon_{ii} - \delta\varepsilon_{jj}\right)^2 + 3\delta\varepsilon_{ij}{}^2} \tag{72}$$

Apart from strain rate effects, clays show a shear strength degradation, for example sensitive clays after the maximum shear strength is mobilized. Therefore, the model degrades the undrained shear strength with the increase of the shear strain (Einav & Randolph, 2005) as follows.

$$s_u(\gamma, S_t) = s_{u,ref}[\frac{1}{S_t} + (1 - \frac{1}{S_t})e^{-3\gamma/\gamma_{95}} \quad (73)$$

where  is the current accumulated shear strain, $\gamma_{95}$ is the accumulated shear strains required to obtain 95% reduction of shear strength and the sensitivity $S_t$ is the ratio of undisturbed to remoulded undrained shear strength $s_{ur,ref}$:

$$S_t = \frac{s_{u,ref}}{s_{ur,ref}} \quad (74)$$

### 9.1.2 Soil investigation and initial condition

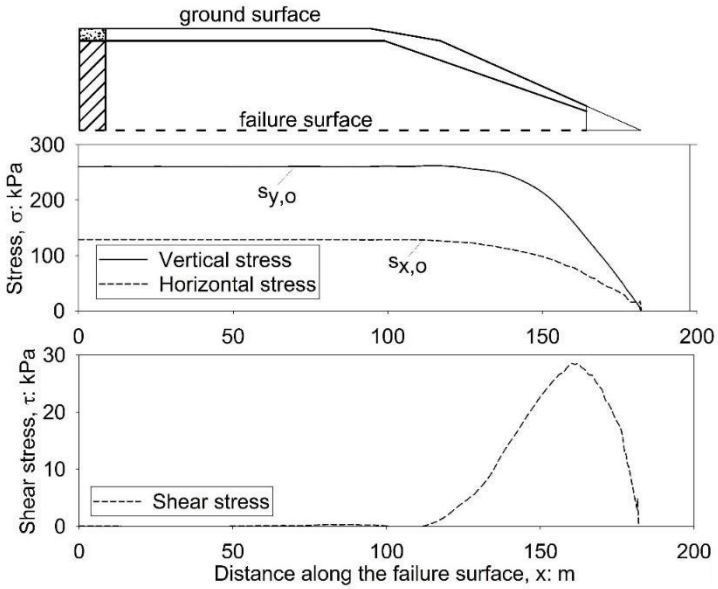



Figure 18 Initial stress condition along the slope

Canadian sensitive clays, in drained condition, have typical value of cohesion (c') and friction angle (') of 10 kPa and 25-40°, respectively (Lefebvre, 2017). These values are obtained based on drained triaxial compression tests with the range of reconsolidation pressure of 5-30 kPa

and pre-consolidation pressure from 100 – 400 kPa. Therefore, the initial condition is generated with Mohr-Coulomb model with cohesion of 10kPa, friction angle of $30^{o}$ and zero dilatancy angle. The initial stress in the pre-failure condition was generated by the gravity loading with the earth pressure coefficient Ko of 0.5 (typical value of normally consolidated clays). Figure 18 present the initial stress of the slope. The soil investigation found a thick brown sand reaching 2m from the ground surface. The undrained shear strength for this layer is assumed to be constant. Below the sand layer, there is a sensitive, normally consolidated clay reaching the depth of 44m. The undrained shear strength of that sensitive clays was determined by the fall cone tests, the vane shear test and the CPTUs. The undrained shear strength profile determined from these in-situ tests are used here to calibrate the numerical parameters of the constitutive model (see Figure 19). Also, stress-strain behaviour is calibrated from the direct shear test, leading to parameter $_{95}$ being 120% to replicate the softening behaviour (see Figure 20). For the strain rate effects, the reference strain rate is 0.05s-1 which corresponds to the shear strain rate of the standard van shear tests at the rotation rate of 0.1deg/(Boukpeti et al., 2012). The strain rate parameters ranging from 0.06 (median value for undisturbed sensitive clays (Yin et al., 2011) to 0.17 (maximum value for disturbed sensitive clays (Jeong et al., 2009)). Therefore, several numerical analyses were performed with clay having the strain rate parameter within the established range, that is between 0 (no strain rate) and 0.17. Similar to the fall cone test, the dynamic shear modulus used was equal to 167 su, ref, while the Poisson's ratio was taken as 0.49 due to undrained conditions. The shear band is embedded in the numerical model and scaled with strain to keep strain energy independent from the mesh size, leading to mesh independency. The shear zone was selected as 0.2m (median value in the literature (Andresen & Jostad, 2007; Leroueil, 2001). Table 3 summarizes the numerical parameters of the model.

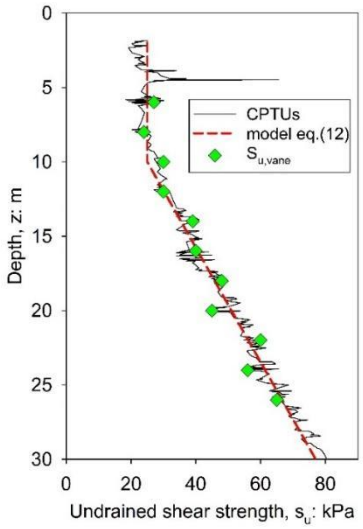



Figure 19 Undrained shear strength with depth

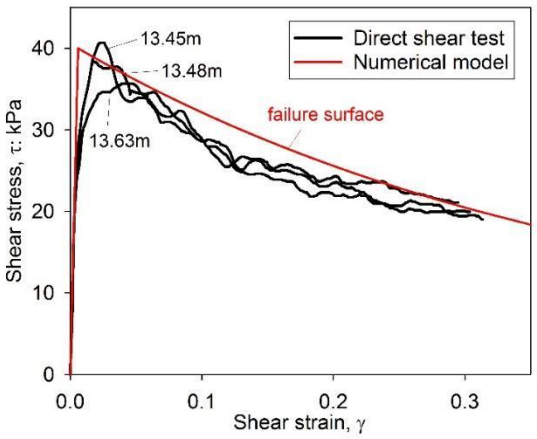



Figure 20 Stress-strain behaviour in direct shear test

Table 3 Parameters used in the Sainte-Monique landslide simulations

| Layer | $s_{u,ref}$ (kPa) | St | $\gamma_{95}$ | $\delta\gamma ref$ (s-1) | β | Gu, ref (kPa) | υu | tshear (m) | $s_{ur}$ (kPa) |
|---|---|---|---|---|---|---|---|---|---|
| Crust | Figure 19 | - | - | - | - | 8350 | 0.49 | - | - |
| Sensitive clay | Figure 19 | 4-55 | 1.2 | 0.05 | 0-0.17 | 167su, ref | 0.49 | 0.2 | $\frac{s_{u,ref}}{S_t}$ |

### 9.1.3 Numerical results

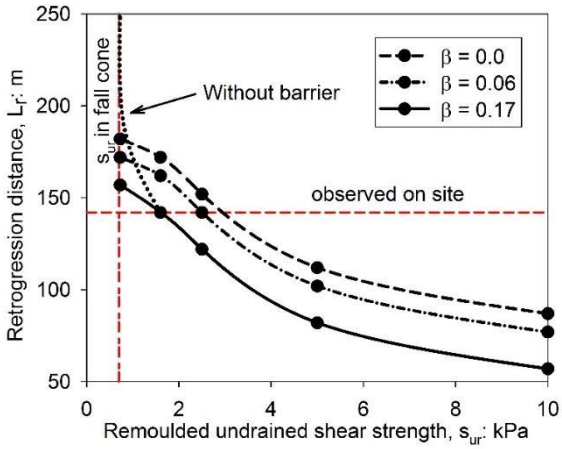



Figure 21 Influence of strain rate parameter ® on the retrogression distances

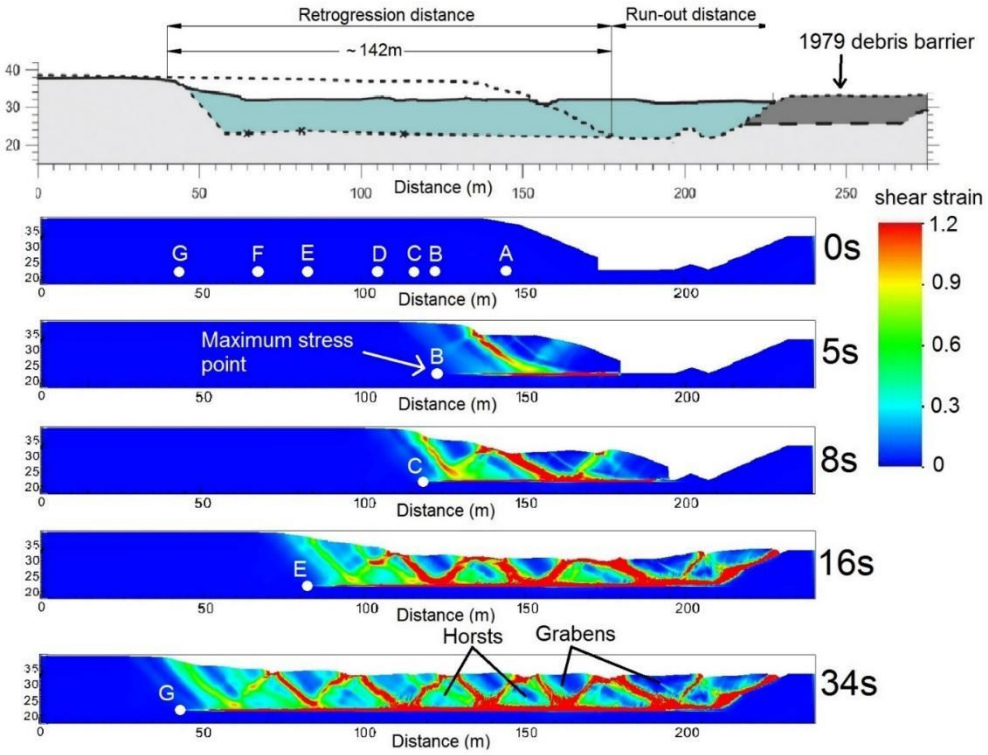



Figure 22 Illustration of the sensitive clay landslides (Q.-A. Tran & Sołowski, 2019)

Figure 21 shows the influence of the strain rate parameter ® on the retrogression distance of the sensitive clay landslide. The retrogression distance could be larger with the remoulded shear strength $s_{ur}$ < 2kPa if there were no debris of the 1979 landslide (dark grey on the right in Figure 22) acting as a barrier to block the debris flow of the 1994 landslide. Without the barrier, the retrogression distance could be very high such as for the case β of 0.17 (dot line

marked without barrier in Figure 21). Because the Canadian sensitive clays typically have a high value of sensitivity (St > 25) and low value of remoulded undrained shear strength ($s_{ur}$ < 2kPa), the reference case with $s_{ur}$ = 1.6kPa, St = 25 and β = 0.17 was selected to analyse the global dynamic behaviour of the landslide.

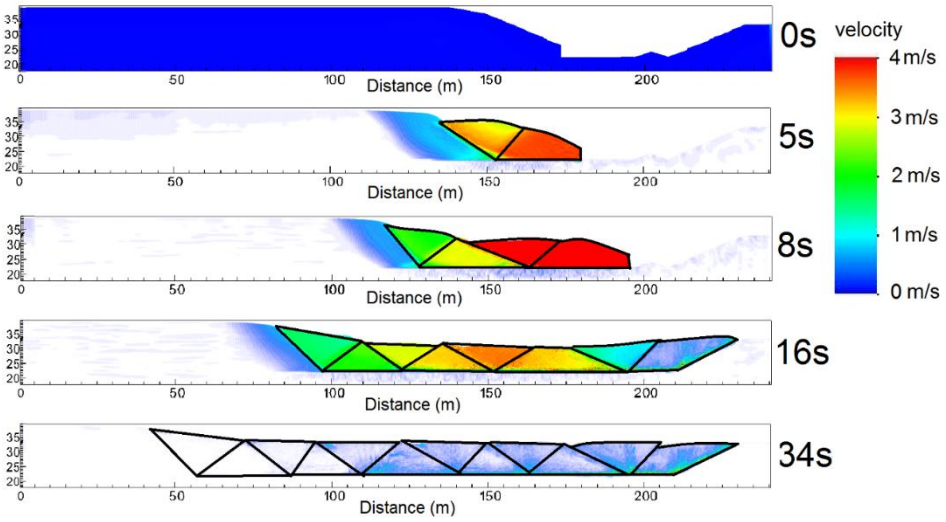



Figure 23 / see attached video 8. Dynamic motion of the sensitive clay landslides (Tran & Sołowski, 2019)

Figure 22 presents the shear strain and Figure 23 presents the dynamic velocity of the Sainte-Monique landslides. Initially, a small amount of soil mass was excavated on the toe of the slope, representing the erosion of the river, to trigger the landslide. After 5s, the erosion induced a thin horizontal shear band propagating from the toe of the slope upward. It was followed by the first failure with a curved shear band propagating from the horizontal failure surface to the ground surface.

This first failure triggered the progressive failure afterwards. The horizontal failure surface kept propagating while above, the neighbouring blocks were moving and dislocating, leading to inclined shear bands between blocks. The propagations of the inclined shear bands formed the horsts and grabens which are made from relatively undisturbed materials characterised by a high undrained shear strength. Regarding the dynamic motion, each horsts and grabens had

a similar lateral displacement rate (same colours of each blocks in Figure 23), with the velocity increasing dramatically to approximately 4m/s and reaching the equilibrium after 34s.

Overall, a full process of the sensitive clay landslide including propagation, formation of shear bands and dislocation of soil blocks is simulated, under assumption that the landslide was triggered by erosion near the toe of the slope. The simulation captures known features of the landslide as well as the final spread well. The numerical analysis showed that the large retrogression landslide can occur with the clays with a high value of sensitivity ($St > 25$) and low value of remoulded undrained shear strength ($sur < 2kPa$). Apart from the remoulded undrained shear strength and the degradation rate of the shear strength, the strain rate dependency of the undrained shear strength is a crucial factor for the prediction of retrogression and run-out distances.

### 9.2 Simulation of shaped charge

Shaped charges are devices in which an explosive load is used to collapse a (usually metal) liner, thereby creating a high velocity jet that is capable of penetrating deep into metal, concrete or other target materials (Walters & Zukas, 1989). Shaped charges are widely used for both military purposes, and in oil field applications, where the perforations they create through steel casing and drilling damage connect the wellbore to the oil-bearing rock.

Simulations of shaped charge detonation, jet formation and target penetration are challenging for any code. Typically, Eulerian hydrocodes are used, but these have their own set of difficulties. Any Lagrangian approach would necessarily need to be particle based due to the large deformations that the materials involved are subjected to. The MPM implementation within Uintah (Berzins et al., 2016; Davison et al., 2000; Parker et al., 2006) has been used to simulate this process with reasonable success. An example of one such simulation is provided here.

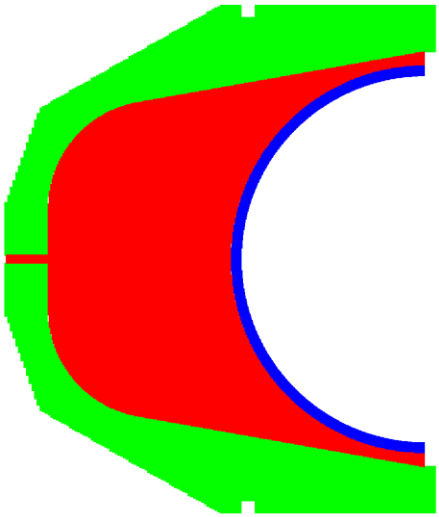


Figure 24. Cross sectional view of the shaped charge geometry (description in text)

Figure 24 depicts a cross sectional view of the shaped charge geometry used in this demonstration calculation. The blue material is the liner, made of tungsten, the green material is a steel case, while the red material is an HMX based high explosive (HE). Shaped charges typically have rotational symmetry, so the simulation is carried out using an axisymmetric implementation (Nairn & Guilkey, 2015). The cylinder has a diameter of 4.7 cm, while the liner is hemispherical with an outer diameter of 3.6 cm and an inner diameter of 3.4 cm.

Detonation of the HE begins at the small orifice on the left side of the figure. In a real charge, some sort of initiation device, such as detonation cord, would be used to initiate the explosive. Here, a so-called "program burn" model is used to simulate the transformation of the HE from a solid at ambient pressure to a gas at the same density, and thus very high pressure. The burn front of the transformation advances in a spherical manner at a rate that corresponds to the known detonation velocity of the HE. The resulting high pressure causes the liner to collapse, and this collapse creates a high-speed jet of material and a slower moving slug.

Both the case and the liner are modeled using a Mie-Gruneisen equation of state to govern the volumetric response, with a Johnson-Cook flow rule and a von Mises yield condition.

Properties are given in Table 4, along with properties for the aluminum target that the jet penetrates.

Table 4. Properties of material used in the shaped charge simulation

| Property | $\rho$ | $C_0$ | $\gamma$ | $S_\alpha$ | G | A | B | C | n | m | $\dot{\varepsilon}_0$ | $T_m$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Material | $kg/m^3$ | cm/μs | | | Mbar | kbar | kbar | | | | 1/s | K |
| Tungsten | 19220 | .400 | 1.54 | 1.24 | 1.24 | 15.07 | 1.766 | 0.016 | 0.12 | 1.0 | 1.0 | 3695 |
| Steel | 7830 | .3574 | 1.69 | 1.92 | 0.80 | 7.92 | 5.10 | 0.014 | 0.26 | 1.03 | 1.0 | 1793 |
| Aluminium | 2780 | .53 | 2.00 | 1.34 | 0.26 | 3.24 | 1.14 | 0.002 | 0.42 | 1.34 | 1.0 | 933.5 |

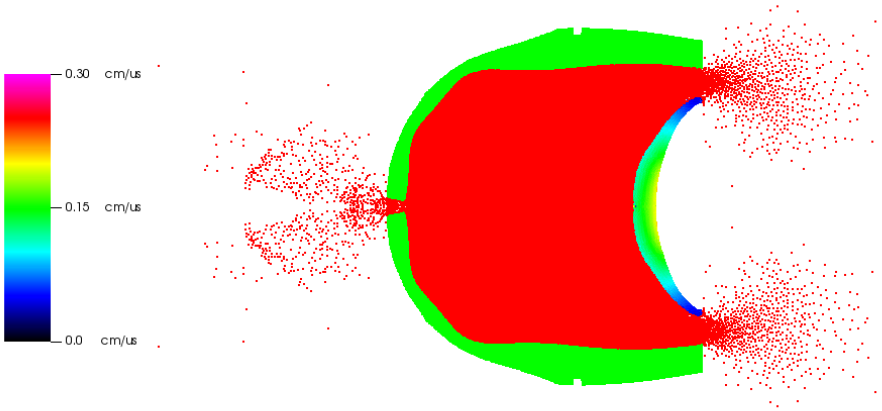
The program burn model uses a Murnaghan equation of state for the unreacted explosive, and a JWL "C-form" equation of state for the reaction products. The form of these equations is given in (Souers et al., 2000). A mixing rule is used to compute pressure for partially reacted particles. Properties for the explosive, including detonation velocity D are provided in Table 5.

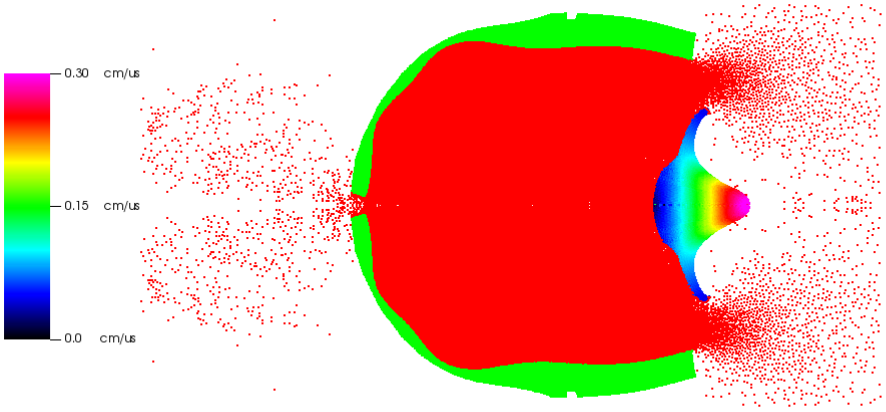
Table 1. Properties of the explosive used in the shaped charge simulation

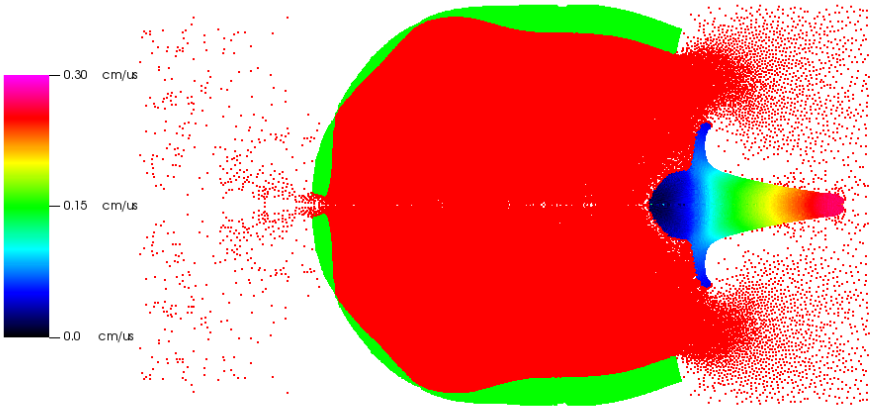
| $\rho$ | n | $\kappa$ | A | B | C | $R_1$ | $R_2$ | $\omega$ | D |
|---|---|---|---|---|---|---|---|---|---|
| $kg/m^3$ | | $Mbar^{-1}$ | Mbar | kbar | kbar | | | | cm/μs |
| 1717.0 | 7.4 | 39.0 | 5.1 | 40.5 | 13.23 | 4.06 | 0.94846 | 0.35 | 0.81 |

The simulations carried out here were done using GIMP, with grid cell size of 0.25 mm in each direction. While CPDI offers better accuracy, in principle, it is not sufficiently robust to for the high levels of deformation experienced by the particles. Because of the challenging nature of these simulations, a few "tricks" were brought to bear to enable getting a reliable solution. First, because the steel case undergoes relatively little deformation, it is represented by 2X2

particles/cell. By contrast, the explosive, which undergoes tremendous expansion, and the liner, which experiences large shear strain, are represented using 4X4 particles/cell. Additionally, even with GIMP, simulation failures occur at times due to reacted explosive particles effectively becoming inverted. This seems to be due, at least in part, to the gaseous reacted explosive having no shear strength. Because of this, in these simulations, all explosive particles are deleted after 30 microseconds, by which time the jet is fully formed. Explosive particles are also deleted before they collide with the target. The other practical reason for deleting these particles is for efficiency. Simulations of jet formation and penetration can last 200 microseconds, and after the jet is formed, neither the case nor the explosive are participatory in the ballistic penetration. By deleting the explosive particles, significant speedup of the simulation can be achieved.


0.30 cm/us
0.15 cm/us
0.0 cm/us


0.30 cm/us
0.15 cm/us
0.0 cm/us


0.30 cm/us
0.15 cm/us
0.0 cm/us

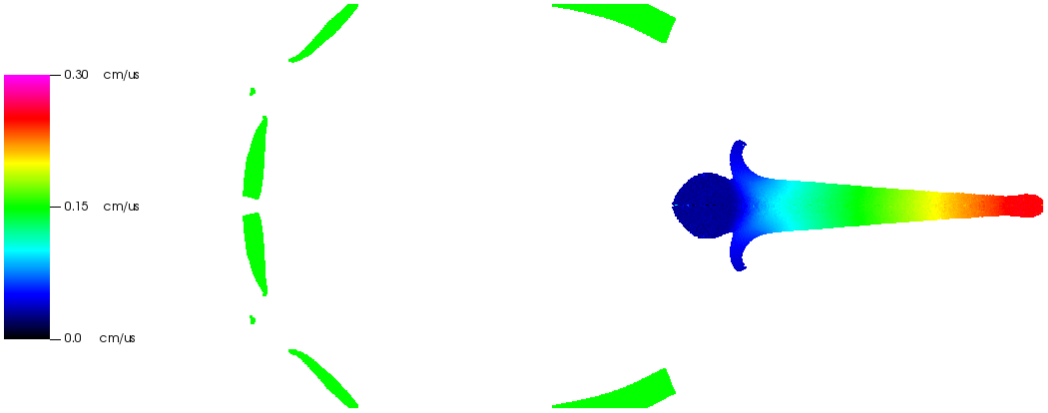
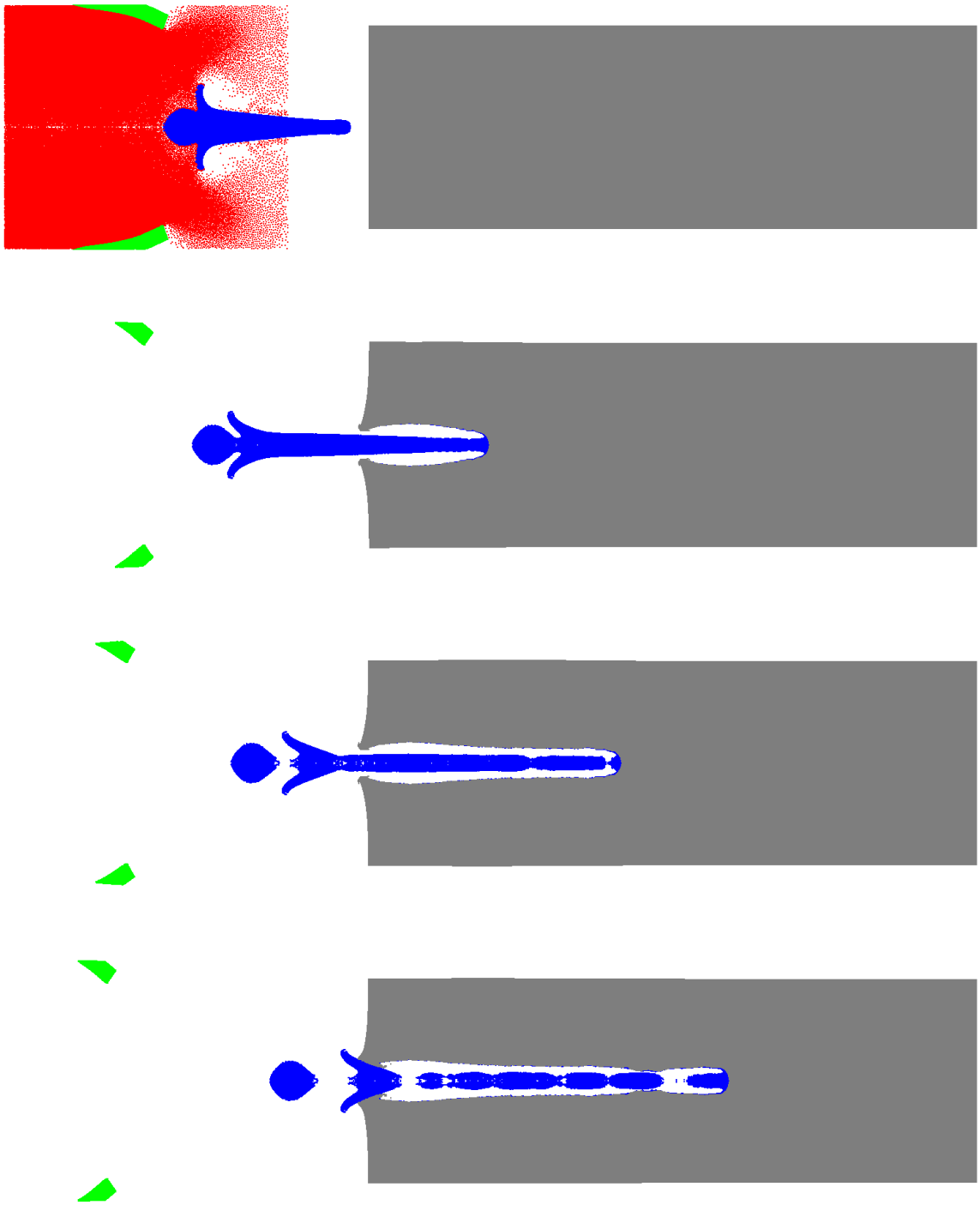



Figure 25. Shaped jet formation at 11μs (a), 14.6μs (b), 22μs (c) and 33μs (d)

See attached video 9. Shaped jet formation

Figure 25 (a)-(d) depict the shaped charge jet formation at 11, 14.6, 22 and 33 microseconds, respectively. In Fig. 25 the liner material particles are colored by velocity, while the explosive and case remain in their original colors. The jet tip velocity reaches approximately 0.3 cm/μs (3 km/s). This is relatively low by shaped charge standards, a higher jet tip velocity could be achieved using a different liner geometry.



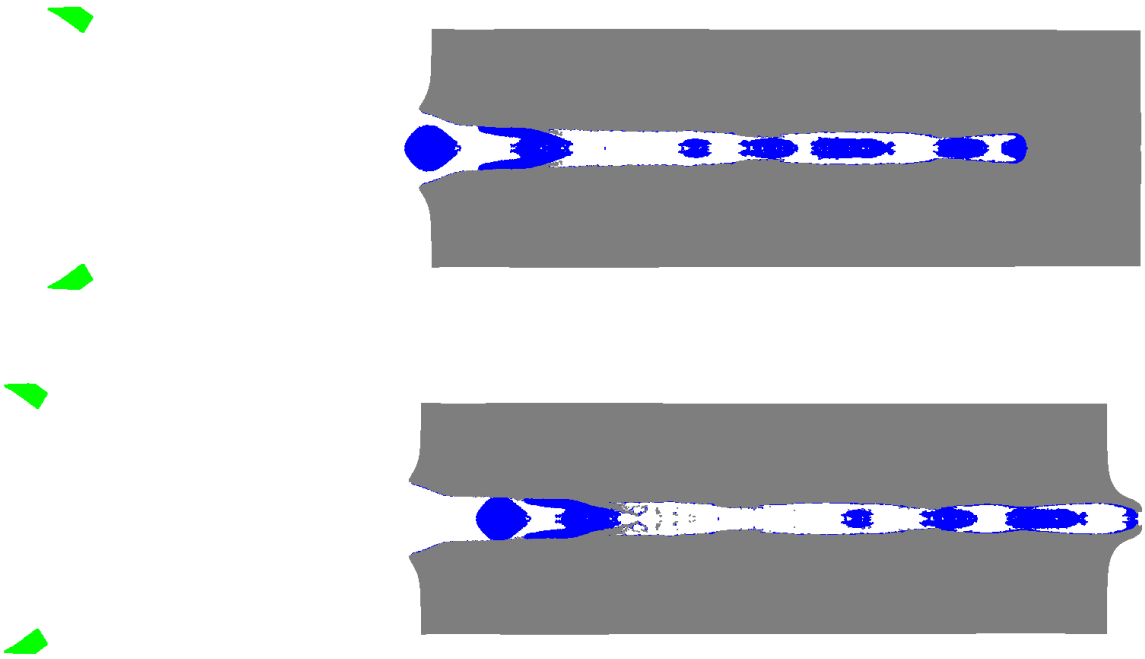


Figure 26. Shaped jet penetrating aluminium target at 10μs (a), 30μs (b), 50μs (c), 75μs (d), 100μs (e), 150μs (f), and 200μs (g)

See attached video 10. Shaped jet penetrating aluminium target

Figure 26 (a) – (g) depict the jet penetrating an aluminum target with diameter of 5 cm and a length of 15 cm at a of 10 cm, at 30, 50, 75, 150 and 200 microseconds, respectively. Figure 26 (d), and those at later times, illustrates a point about which one must be cautious. Namely, the jet has begun breaking up into segments, despite what appears to be sufficient particle density to prevent numerical fracture. This may give misleading penetration results, due to the "segmented penetrator effect". On the other hand, a sufficiently sophisticated material model may be able to accurately predict this type of behavior, and a Lagrangian description of the material is likely needed for this to be possible. Regardless, these simulations demonstrate the ability of MPM to serve as a design tool when seeking to achieve a jet with a particular velocity profile.

## 9.3 Granular flows modelling

Granular flow modelling is an important application for MPM. In flow, granular materials go through significant deformation and behaviour changes from solid-like to low-density state. Therefore, MPM is a suitable method for these flows as it can model large deformations and consider changes in the material constitutive behaviour. Many investigations have used MPM for modelling granular flows, e.g. (Fern & Soga, 2016; Kumar et al., 2017; Mast et al., 2015; Seyedan & Solowski, 2017; Sołowski & Sloan, 2015; Sołowski & Sloan, 2013; Sołowski & Sloan, 2014). However, these investigations were focused on solid to flow like state and did not include low density disconnected states of granular flows.

MPM investigations which include low density granular flows are limited. Dunatunga & Kamrin (2015, 2017) conducted some of the initial MPM studies of low density granular flows. In addition, Redaelli et al. (2017) used a set of novel constitutive models for MPM modelling of granular flows through different stages. Although, these research assumes that MPM is capable of modelling low density stages of flow, Dunatunga & Kamrin (2017) faces problem in using versions of MPM that update the extent of material points (e.g. CPDI). Seyedan & Sołowski (2019) investigated MPM capability in modelling low density granular flows, showed unphysical interactions between material points in this stage and suggested a solution for preventing the unphysical interactions. Their research builds on the framework of Dunatunga & Kamrin (2015) and introduces a modified algorithm for MPM modelling of granular flows that can model all stages of granular flows. The new algorithm can use all different version of MPM and provides better results than available versions of MPM. Video 11 compares capability of different MPM algorithm in simulating filling of a flat bottom silo.

See attached video 11. MPM modelling of silo filling. In this figure / video Granular Original MPM and Granular CPDI are modified Original MPM and modified CPDI based on algorithm Seyedan & Sołowski (2019, 2021).

# 10. Outlook for the future

The Material Point Method has greatly matured since the first publication of Sulsky et al. (1994). The method become more accurate and more stable, proving itself in all sort of applications where dynamics is important. To mention a few: [references] The method is also widely used as basis for computer animation (see section 2.9 for details), confirming further its versatility and ability to realistically simulate multi-physics effects and difficult materials such as cloth.

Yet, the Material Point Method still needs further developments to fully complement Finite Element Method in a toolbox of engineer. The community still needs to work on accuracy and efficiency of the method. Higher order of convergence, as well as mathematical proofs guaranteeing method stability and accuracy. Also, it seems that modelling of multi-physics problems, as well as those involving accurate contact laws, complex boundary conditions are areas which could be developed further.

On the other hand, there is perhaps also a need for a more user friendly input for the software solvers. The community is also fragmented, meaning that there is no single MPM solver incorporating all recent advancements and developments. It is currently also not easy to combine the MPM with other numerical methods in a single simulation, as the developments are fragmented.

Finally, the method waits for a wider adoption by practicing engineers. Even though it is well acknowledged in military applications, and is widely used in advanced applications, such as

needed in Los Alamos and Sandia, the method is not widely known elsewhere. Partially, it is due to lack of easy to use software, but perhaps also due to relative lack of awareness of the unique capabilities of the method. Those could be, e.g. used to simulate explosives in mines and tunnelling, drilling in oil industry and penetration problems in civil engineering. Another application area not well explored would be biomechanics, where large deformations are commonplace and the need of new numerical tools is acute.

# 11. References


Abe, K., Soga, K., & Bandara, S. (2014). Material Point Method for Coupled Hydromechanical Problems. *Journal of Geotechnical and Geoenvironmental Engineering*, *140*(3), 04013033. https://doi.org/10.1061/(ASCE)GT.1943-5606.0001011

Adibaskoro, T., Bordas, S., Sołowski, W., & Hostikka, S. (2020). Multiple Discrete Crack Initiation and Propagation in CPDI Material Point Method. *Unpublished*.

Alonso, EE; Yerro, A; Pinyol, N. (2015). Recent developments of the Material Point Method for the simulation of landslides. *IOP Conference Series: Earth and Environmental Science*, *26*(1), 012003.

Andersen, S., & Andersen, L. (2010). Modelling of landslides with the material-point method. *Computational Geosciences*, *14*(1), 137–147. https://doi.org/10.1007/s10596-009-9137-y

Andresen, L., & Jostad, H. P. (2007). *Numerical Modeling of Failure Mechanisms in Sensitive Soft Clay-Application to Offshore Geohazards*. https://doi.org/10.4043/18650-ms

Babuška, I., & Suri, M. (1992a). Locking effects in the finite element approximation of elasticity problems. *Numerische Mathematik*. https://doi.org/10.1007/BF01396238

Babuška, I., & Suri, M. (1992b). On Locking and Robustness in the Finite Element Method. *SIAM Journal on Numerical Analysis*, *29*(5), 1261–1293.

https://doi.org/10.1137/0729075

Bandara, S., Ferrari, A., & Laloui, L. (2016). *Modelling landslides in unsaturated slopes subjected to rainfall infiltration using material point method*. https://doi.org/10.1002/nag

Bandara, S., & Soga, K. (2015). Coupling of soil deformation and pore fluid flow using material point method. *Computers and Geotechnics*, *63*, 199–214. https://doi.org/10.1016/j.compgeo.2014.09.009

Bardenhagen, S. G. (2002). Energy conservation error in the material point method for solid mechanics. *Journal of Computational Physics*, *180*(1), 383–403. https://doi.org/10.1006/jcph.2002.7103

Bardenhagen, S. G., Brackbill, J. U., & Sulsky, D. (2000). The material-point method for granular materials. *Computer Methods in Applied Mechanics and Engineering*, *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. *CMES - Computer Modeling in Engineering and Sciences*, *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. *CMES - Computer Modeling in Engineering and Sciences*, *5*(6), 477–495. https://doi.org/10.3970/cmes.2004.005.477

Bardenhagen, Scott G., Nairn, J. A., & Lu, H. (2011). Simulation of dynamic fracture with the Material Point Method using a mixed J-integral and cohesive law approach. *International Journal of Fracture*. https://doi.org/10.1007/s10704-011-9602-1

Baumgarten, A. S., & Kamrin, K. (2019). A general fluid-sediment mixture model and

constitutive theory validated in many flow regimes. *Journal of Fluid Mechanics*, *861*, 721–764. https://doi.org/10.1017/jfm.2018.914

Berzins, M. (2018). Nonlinear stability and time step selection for the MPM method. *Computational Particle Mechanics*, *5*(4), 455–466. https://doi.org/10.1007/s40571-018-0182-y

Berzins, M. (2019). TIME INTEGRATION ERRORS AND ENERGY CONSERVATION PROPERTIES OF THE STORMER VERLET METHOD APPLIED TO MPM. In P. W. & T. Z. E. Onate, M. Bischoff, D.R.J. Owen (Ed.), *VI International Conference on Particle-based Methods – Fundamentals and Applications PARTICLES 2019* (pp. 555–566).

Berzins, M., Beckvermit, J., Harman, T., Bezdjian, A., Humphrey, A., Meng, Q., Schmidt, J., & Wight, C. (2016). Extending the Uintah framework through the petascale modeling of detonation in arrays of high explosive devices. *SIAM Journal on Scientific Computing*. https://doi.org/10.1137/15M1023270

Beuth, L., Benz, T., Vermeer, P., & Więckowski, Z. (2008). *Large deformation analysis using a quasi-static material point method. 38*, 45–60.

Bing, Y., Cortis, M., Charlton, T. J., Coombs, W. M., & Augarde, C. E. (2019). B-spline based boundary conditions in the material point method. *Computers and Structures*. https://doi.org/10.1016/j.compstruc.2018.11.003

Boukpeti, N., White, D. J., Randolph, M. F., & Low, H. E. (2012). Strength of fine-grained soils at the solid-fluid transition. *Geotechnique*, *62*(3), 213–226. https://doi.org/10.1680/geot.9.P.069

Brackbill, J. U. (1988). The ringing instability in particle-in-cell calculations of low-speed

flow. *Journal of Computational Physics*, *75*(2), 469–492. https://doi.org/10.1016/0021-9991(88)90123-4

Brackbill, J. U. (2016). On energy and momentum conservation in particle-in-cell plasma simulation. *Journal of Computational Physics*. https://doi.org/10.1016/j.jcp.2016.04.050

Brackbill, J. W., & Lapenta, G. (1994). A Method to Suppress the Finite-Grid Instability in Plasma Simulations. *Journal of Computational Physics*, *114*(1), 77–84. https://doi.org/10.1006/jcph.1994.1150

Buzzi, O., Pedroso, D. M., & Giacomlnl, A. (2008). Caveats on the implementation of the generalized material point method. *CMES - Computer Modeling in Engineering and Sciences*, *31*(2), 85–106.

Ceccato, F., Redaelli, I., di Prisco, C., & Simonini, P. (2018). Impact forces of granular flows on rigid structures: Comparison between discontinuous (DEM) and continuous (MPM) numerical approaches. *Computers and Geotechnics*, *103*(February), 201–217. https://doi.org/10.1016/j.compgeo.2018.07.014

Charlton, T. J., Coombs, W. M., & Augarde, C. E. (2017). iGIMP: An implicit generalised interpolation material point method for large deformations. *Computers and Structures*, *190*, 108–125. https://doi.org/10.1016/j.compstruc.2017.05.004

Chen, Z. P., Qiu, X. M., Zhang, X., & Lian, Y. P. (2015). Improved coupling of finite element method with material point method based on a particle-to-surface contact algorithm. *Computer Methods in Applied Mechanics and Engineering*, *293*, 1–19. https://doi.org/10.1016/j.cma.2015.04.005

Cheon, Y.-J., & Kim, H.-G. (2018). An efficient contact algorithm for the interaction of material particles with finite elements. *Computer Methods in Applied Mechanics and*

*Engineering*, *335*, 631–659. https://doi.org/10.1016/j.cma.2018.02.005

Cheon, Y. J., & Kim, H. G. (2019). An adaptive material point method coupled with a phase-field fracture model for brittle materials. *International Journal for Numerical Methods in Engineering*. https://doi.org/10.1002/nme.6167

Cherepanov, G. P. (1979). *Mechanics of brittle fracture*. McGraw-Hill.

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

Coetzee, C. J., Basson, A. H., & Vermeer, P. A. (2007). Discrete and continuum modelling of excavator bucket filling. *Journal of Terramechanics*, *44*(2), 177–186. https://doi.org/10.1016/j.jterra.2006.07.001

Coombs, W. M., & Augarde, C. E. (2020). AMPLE: A Material Point Learning Environment. *Advances in Engineering Software*. https://doi.org/10.1016/j.advengsoft.2019.102748

Coombs, W. M., Augarde, C. E., Brennan, A. J., Brown, M. J., Charlton, T. J., Knappett, J. A., Ghaffari Motlagh, Y., & Wang, L. (2020). On Lagrangian mechanics and the implicit material point method for large deformation elasto-plasticity. *Computer Methods in Applied Mechanics and Engineering*. https://doi.org/10.1016/j.cma.2019.112622

Coombs, W. M., Charlton, T. J., Cortis, M., & Augarde, C. E. (2018). Overcoming volumetric locking in material point methods. *Computer Methods in Applied Mechanics and Engineering*, *333*, 1–21. https://doi.org/10.1016/j.cma.2018.01.010

Cortis, M., Coombs, W., Augarde, C., Brown, M., Brennan, A., & Robinson, S. (2018). Imposition of essential boundary conditions in the material point method. *International Journal for Numerical Methods in Engineering*, *113*(1), 130–152. https://doi.org/10.1002/nme.5606

Cummins, S. J., & Brackbill, J. U. (2002). An implicit particle-in-cell method for granular materials. *Journal of Computational Physics*, *180*(2), 506–548. https://doi.org/10.1006/jcph.2002.7101

Daviet, G., & Bertails-Descoubes, F. (2016). A semi-implicit material point method for the continuum simulation of granular materials. *ACM Transactions on Graphics*. https://doi.org/10.1145/2897824.2925877

Davison De, J., McCorquodale, J., Parker, S. G., & Johnson, C. R. (2000). Uintah: A massively parallel problem solving environment. *Proceedings of the IEEE International Symposium on High Performance Distributed Computing*, *2000-Janua*, 33–41. https://doi.org/10.1109/HPDC.2000.868632

de Koster, P., Tielen, R., Wobbes, E., & Möller, M. (2020). Extension of B-spline Material Point Method for unstructured triangular grids using Powell–Sabin splines. *Computational Particle Mechanics*. https://doi.org/10.1007/s40571-020-00328-3

De Souza Neto, E. A., Perić, D., Dutko, M., & Owen, D. R. J. (1996). Design of simple low order finite elements for large strain analysis of nearly incompressible solids. *International Journal of Solids and Structures*. https://doi.org/10.1016/0020-7683(95)00259-6

Ding, M., Han, X., Wang, S., Gast, T. F., & Teran, J. M. (2019). A thermomechanical material point method for baking and cooking. *ACM Transactions on Graphics*. https://doi.org/10.1145/3355089.3356537

Dokken, T., Lyche, T., & Pettersen, K. F. (2013). Polynomial splines over locally refined box-partitions. *Computer Aided Geometric Design*. https://doi.org/10.1016/j.cagd.2012.12.005

Dunatunga, S., & Kamrin, K. (2015). Continuum modelling and simulation of granular flows through their many phases. *Journal of Fluid Mechanics*, *779*, 483–513. https://doi.org/10.1017/jfm.2015.383

Dunatunga, S., & Kamrin, K. (2017). Continuum modeling of projectile impact and penetration in dry granular media. *Journal of the Mechanics and Physics of Solids*, *100*(December 2016), 45–60. https://doi.org/10.1016/j.jmps.2016.12.002

Einav, I., & Randolph, M. F. (2005). Combining upper bound and strain path methods for evaluating penetration resistance. *International Journal for Numerical Methods in Engineering*, *63*(14), 1991–2016. https://doi.org/10.1002/nme.1350

Fang, Y., Hu, Y., Hu, S. M., & Jiang, C. (2018). A Temporally Adaptive Material Point Method with Regional Time Stepping. *Computer Graphics Forum*. https://doi.org/10.1111/cgf.13524

Fern, E. J., & Soga, K. (2016). The role of constitutive models in MPM simulations of granular column collapses. *Acta Geotechnica*, *11*(3), 659–678. https://doi.org/10.1007/s11440-016-0436-x

Fu, C., Guo, Q., Gast, T., Jiang, C., & Teran, J. (2017). A polynomial particle-in-cell method. *ACM Transactions on Graphics*. https://doi.org/10.1145/3130800.3130878

Gan, Y., Sun, Z., Chen, Z., Zhang, X., & Liu, Y. (2018). Enhancement of the material point method using B-spline basis functions. *International Journal for Numerical Methods in Engineering*, *113*(3), 411–431. https://doi.org/10.1002/nme.5620

Gao, M., Pradhana, A., Han, X., Guo, Q., Kot, G., Sifakis, E., & Jiang, C. (2018). Animating fluid sediment mixture in particle-laden flows. *ACM Transactions on Graphics*. https://doi.org/10.1145/3197517.3201309

Gao, M., Tampubolon, A. P., Jiang, C., & Sifakis, E. (2017). An adaptive generalized interpolation material point method for simulating elastoplastic materials. *ACM Transactions on Graphics*, *36*(6). https://doi.org/10.1145/3130800.3130879

Gao, M., Wang, X., Wu, K., Pradhana, A., Sifakis, E., Yuksel, C., & Jiang, C. (2018). GPU optimization of material point methods. *SIGGRAPH Asia 2018 Technical Papers, SIGGRAPH Asia 2018*. https://doi.org/10.1145/3272127.3275044

Gast, T. F., Schroeder, C., Stomakhin, A., Jiang, C., & Teran, J. M. (2015). Optimization Integrator for Large Time Steps. *IEEE Transactions on Visualization and Computer Graphics*, *21*(10), 1103–1115. https://doi.org/10.1109/TVCG.2015.2459687

Giannelli, C., Jüttler, B., & Speleers, H. (2012). THB-splines: The truncated basis for hierarchical splines. *Computer Aided Geometric Design*. https://doi.org/10.1016/j.cagd.2012.03.025

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. *Computational Mechanics*. https://doi.org/10.1007/s00466-019-01783-3

Griffith, A. A. (1921). VI. The phenomena of rupture and flow in solids. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, *221*(582–593), 163–198.

Grigoryev, Y. N., Vshivkov, V. A., & Fedoruk, M. P. (2012). Numerical "Particle-in-cell" methods: Theory and applications. In *Numerical "Particle-in-Cell" Methods: Theory and Applications*. https://doi.org/10.1515/9783110916706

Gritton, C., & Berzins, M. (2017). Improving accuracy in the MPM method using a null space filter. *Computational Particle Mechanics*, *4*(1), 131–142. https://doi.org/10.1007/s40571-

016-0134-3

Gritton, C., Berzins, M., & Kirby, R. M. (2015). Improving accuracy in particle methods using null spaces and filters. *Proceedings of the 4th International Conference on Particle-Based Methods - Fundamentals and Applications, PARTICLES 2015*.

Gritton, C., Guilkey, J., Hooper, J., Bedrov, D., Kirby, R. M., & Berzins, M. (2017). Using the material point method to model chemical/mechanical coupling in the deformation of a silicon anode. *Modelling and Simulation in Materials Science and Engineering*, *25*(4), 045005. https://doi.org/10.1088/1361-651X/aa6830

Guilkey, J. E., & Weiss, J. A. (2003). Implicit time integration for the material point method: Quantitative and algorithmic comparisons with the finite element method. *International Journal for Numerical Methods in Engineering*, *57*(9), 1323–1338. https://doi.org/10.1002/nme.729

Guilkey, J.E., Harman, T. B., & Banerjee, B. (2007). An Eulerian–Lagrangian approach for simulating explosions of energetic devices. *Computers & Structures*, *85*(11–14), 660–674. https://doi.org/10.1016/j.compstruc.2007.01.031

Guilkey, James E., Hoying, J. B., & Weiss, J. A. (2006). Computational modeling of multicellular constructs with the material point method. *Journal of Biomechanics*. https://doi.org/10.1016/j.jbiomech.2005.06.017

Guo, N., & Zhao, J. (2014). A coupled FEM/DEM approach for hierarchical multiscale modelling of granular media. *International Journal for Numerical Methods in Engineering*. https://doi.org/10.1002/nme.4702

Guo, Y., & Nairn, J. A. (2004). Calculation of J-integral and stress intensity factors using the material point method. *CMES - Computer Modeling in Engineering and Sciences*, *6*(3),

295–308. https://doi.org/10.3970/cmes.2004.006.295

Hammerquist, C. C., & Nairn, J. A. (2017). A new method for material point method particle updates that reduces noise and enhances stability. *Computer Methods in Applied Mechanics and Engineering*, *318*, 724–738. https://doi.org/10.1016/j.cma.2017.01.035

Hammerquist, C. C., & Nairn, J. A. (2018). Modeling nanoindentation using the Material Point Method. *Journal of Materials Research*. https://doi.org/10.1557/jmr.2018.75

Han, X., Gast, T. F., Guo, Q., Wang, S., Jiang, C., & Teran, J. (2019). A Hybrid Material Point Method for Frictional Contact with Diverse Materials. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, *2*(2), 1–24. https://doi.org/10.1145/3340258

Harlow, F. H. (1963). *The particle-in-cell method for numerical solution of problems in fluid dynamics* (pp. 269–288). https://doi.org/10.1090/psapm/015/9942

Harlow, F. H. (1964). The particle-in-cell computing method for fluid dynamics. *Methods in Computational Physics*, *3*, 319–343.

Harlow, F. H. (1988). PIC and its progeny. *Computer Physics Communications*, *48*(1), 1–10. https://doi.org/10.1016/0010-4655(88)90017-3

Higo, Y., Oka, F., Kimoto, S., Morinaka, Y., Goto, Y., & Chen, Z. (2010). A Coupled MPM-FDM Analysis Method for Multi-Phase Elasto-Plastic Soils. *Soils and Foundations*, *50*(4), 515–532. https://doi.org/10.3208/sandf.50.515

Homel, Michael A; Herbold, E. B. (2017). Field-gradient partitioning for fracture and frictional contact in the material point method. *International Journal for Numerical Methods in Engineering*, *109*(7), 1013–1044.

Homel, M. A., Brannon, R. M., & Guilkey, J. (2016). Controlling the onset of numerical

fracture in parallelized implementations of the material point method (MPM) with convective particle domain interpolation (CPDI) domain scaling. *International Journal for Numerical Methods in Engineering*. https://doi.org/10.1002/nme.5151

Hu, W., & Chen, Z. (2003). A multi-mesh MPM for simulating the meshing process of spur gears. *Computers & Structures*, *81*(20), 1991–2002. https://doi.org/10.1016/S0045-7949(03)00260-8

Huang, P., Zhang, X., Ma, S., & Huang, X. (2011). *Contact algorithms for the material point method in impact and penetration simulation*. *July 2010*, 498–517. https://doi.org/10.1002/nme

Hughes, T. J. R., Cottrell, J. A., & Bazilevs, Y. (2005). Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, *194*(39–41), 4135–4195. https://doi.org/10.1016/j.cma.2004.10.008

Iaconeta, I., Larese, A., Rossi, R., & Oñate, E. (2019). A stabilized mixed implicit Material Point Method for non-linear incompressible solid mechanics. *Computational Mechanics*, *63*(6), 1243–1260. https://doi.org/10.1007/s00466-018-1647-9

Iaconeta, Ilaria, Larese, A., Rossi, R., & Oñate, E. (2017). An Implicit Material Point Method Applied to Granular Flows. *Procedia Engineering*, *175*(2016), 226–232. https://doi.org/10.1016/j.proeng.2017.01.017

Jassim, I., Stolle, D., & Vermeer, P. (2013). *Two-phase dynamic analysis by material point method*. *October 2012*, 2502–2522. https://doi.org/10.1002/nag

Jeong, S. W., Leroueil, S., & Locat, J. (2009). Applicability of power law for describing the rheology of soils of different origins and characteristics. *Canadian Geotechnical Journal*,

*46*(9), 1011–1023. https://doi.org/10.1139/T09-031

Jiang, C. (2015). The Material Point Method for the Physics-Based Simulation of Solids and Fluids. In *University of CALIFORNIA LOS ANGELES*. https://www.math.ucla.edu/~jteran/student_thesis/jiang.pdf

Jiang, C. (2020). *MPM IN COMPUTER GRAPHICS: UP TO MARCH 2020*. https://www.seas.upenn.edu/~cffjiang/mpm.html

Jiang, C., Schroeder, C., & Teran, J. (2017). An angular momentum conserving affine-particle-in-cell method. *Journal of Computational Physics*. https://doi.org/10.1016/j.jcp.2017.02.050

Kakouris, E. G., & Triantafyllou, S. P. (2017). Phase-field material point method for brittle fracture. *International Journal for Numerical Methods in Engineering*, *112*(12), 1750–1776. https://doi.org/10.1002/nme.5580

Kakouris, E. G., & Triantafyllou, S. P. (2018). Material point method for crack propagation in anisotropic media: a phase field approach. *Archive of Applied Mechanics*, *88*(1–2), 287–316. https://doi.org/10.1007/s00419-017-1272-7

Kakouris, Emmanouil G., & Triantafyllou, S. P. (2019). Phase-Field Material Point Method for dynamic brittle fracture with isotropic and anisotropic surface energy. *Computer Methods in Applied Mechanics and Engineering*, *357*, 112503. https://doi.org/10.1016/j.cma.2019.06.014

Kamojjala, K., Brannon, R., Sadeghirad, A., & Guilkey, J. (2015). Verification tests in solid mechanics. *Engineering with Computers*, *31*(2), 193–213. https://doi.org/10.1007/s00366-013-0342-x

Kashiwa, B. A. (2001). *A Multifield Model and Method for Fluid-Structure Interaction*

*Dynamics. Technical Report LA-UR-01-1136*.

Kashiwa, B., & Rauenzahn, R. (1994a). *A cell-centered ICE method for multiphase flow simulations. Technical Report LA-UR-93-3922*.

Kashiwa, B., & Rauenzahn, R. (1994b). *A multimaterial formalism. Technical Report LA-UR-94-771*.

Kirchhart, M., & Rieger, C. (2019). *A Particle Method without Remeshing*. http://arxiv.org/abs/1909.07449

Klár, G., Gast, T., Pradhana, A., Fu, C., Schroeder, C., Jiang, C., & Teran, J. (2016). Drucker-prager elastoplasticity for sand animation. *ACM Transactions on Graphics*, *35*(4), 1–12. https://doi.org/10.1145/2897824.2925906

Kularathna, S., & Soga, K. (2017a). Implicit formulation of material point method for analysis of incompressible materials. *Computer Methods in Applied Mechanics and Engineering*. https://doi.org/10.1016/j.cma.2016.10.013

Kularathna, S., & Soga, K. (2017b). Comparison of two projection methods for modeling incompressible flows in MPM. *Journal of Hydrodynamics*, *29*(3), 405–412. https://doi.org/10.1016/S1001-6058(16)60750-3

Kumar, K., Soga, K., & Delenne, J. Y. (2013). Multi-scale modelling of granular avalanches. *AIP Conference Proceedings*. https://doi.org/10.1063/1.4812165

Kumar, K., Soga, K., Delenne, J. Y., & Radjai, F. (2017). Modelling Transient Dynamics of Granular Slopes: MPM and DEM. *Procedia Engineering*, *175*(2016), 94–101. https://doi.org/10.1016/j.proeng.2017.01.032

Lefebvre, G. (2017). Sensitive clays of Eastern Canada: From geology to slope stability. In *Advances in Natural and Technological Hazards Research*. https://doi.org/10.1007/978-

3-319-56487-6_2

Leroueil, S. (2001). Natural slopes and cuts: Movement and failure mechanisms. *Geotechnique*, *51*(3), 197–243. https://doi.org/10.1680/geot.2001.51.3.197

Li, J. G., Hamamoto, Y., Liu, Y., & Zhang, X. (2014). Sloshing impact simulation with material point method and its experimental validations. *Computers and Fluids*, *103*, 86–99. https://doi.org/10.1016/j.compfluid.2014.07.025

Lian, Y. P., Zhang, X., Zhou, X., & Ma, Z. T. (2011). A FEMP method and its application in modeling dynamic response of reinforced concrete subjected to impact loading. *Computer Methods in Applied Mechanics and Engineering*. https://doi.org/10.1016/j.cma.2011.01.019

Lian, Y., & Zhang, X. (2018). A Coupled Finite Element Material Point Method for Large Deformation Problems. *Computational and Experimental Methods in Structures*. https://doi.org/10.1142/9781786344786_0007

Lian, Y.P., Zhang, X., & Liu, Y. (2011). Coupling of finite element method with material point method by local multi-mesh contact method. *Computer Methods in Applied Mechanics and Engineering*, *200*(47–48), 3482–3494. https://doi.org/10.1016/j.cma.2011.07.014

Lian, Yan Ping, Liu, Y., & Zhang, X. (2014). Coupling of membrane element with material point method for fluid-membrane interaction problems. *International Journal of Mechanics and Materials in Design*. https://doi.org/10.1007/s10999-014-9241-6

Liang, W., & Zhao, J. (2019). Multiscale modeling of large deformation in geomechanics. *International Journal for Numerical and Analytical Methods in Geomechanics*. https://doi.org/10.1002/nag.2921

Liang, Y., Benedek, T., Zhang, X., & Liu, Y. (2017). Material point method with enriched

shape function for crack problems. *Computer Methods in Applied Mechanics and Engineering*. https://doi.org/10.1016/j.cma.2017.05.012

Liu, C., Sun, Q., Jin, F., & Zhou, G. G. D. (2017). A fully coupled hydro-mechanical material point method for saturated dense granular materials. *Powder Technology*, *314*, 110–120. https://doi.org/10.1016/j.powtec.2017.02.022

Liu, C., Sun, Q., & Zhou, G. G. D. (2018). Coupling of material point method and discrete element method for granular flows impacting simulations. *International Journal for Numerical Methods in Engineering*, *115*(2), 172–188. https://doi.org/10.1002/nme.5800

Locat, A., Leroueil, S., Fortin, A., Demers, D., & Jostad, H. P. (2015). The 1994 landslide at Sainte-Monique, Quebec: geotechnical investigation and application of progressive failure analysis. *Canadian Geotechnical Journal*, *52*(4), 490–504. https://doi.org/10.1139/cgj-2013-0344

Love, E., & Sulsky, D. L. (2006a). An energy-consistent material-point method for dynamic finite deformation plasticity. *International Journal for Numerical Methods in Engineering*, *65*(10), 1608–1638. https://doi.org/10.1002/nme.1512

Love, E., & Sulsky, D. L. (2006b). An unconditionally stable, energy-momentum consistent implementation of the material-point method. *Computer Methods in Applied Mechanics and Engineering*, *195*(33–36), 3903–3925. https://doi.org/10.1016/j.cma.2005.06.027

Ma, J., Wang, D., & Randolph, M. F. (2014). A new contact algorithm in the material point method for geotechnical simulations. *International Journal for Numerical and Analytical Methods in Geomechanics*. https://doi.org/10.1002/nag.2266

Ma, Z. T., Zhang, X., & Huang, P. (2010). An object-oriented MPM framework for simulation of large deformation and contact of numerous grains. *CMES - Computer Modeling in*

*Engineering and Sciences*, *55*(1), 61–87. https://doi.org/10.3970/cmes.2010.055.061

Mackenzie-Helnwein, P., Arduino, P., Shin, W., Moore, J. A., & Miller, G. R. (2010). Modeling strategies for multiphase drag interactions using the material point method. *International Journal for Numerical Methods in Engineering*, *February*, n/a-n/a. https://doi.org/10.1002/nme.2823

Mast, C.M., Mackenzie-Helnwein, P., Arduino, P., Miller, G. R., & Shin, W. (2012). Mitigating kinematic locking in the material point method. *Journal of Computational Physics*, *231*(16), 5351–5373. https://doi.org/10.1016/j.jcp.2012.04.032

Mast, Carter M., Arduino, P., Mackenzie-Helnwein, P., & Miller, G. R. (2015). Simulating granular column collapse using the Material Point Method. *Acta Geotechnica*, *10*(1), 101–116. https://doi.org/10.1007/s11440-014-0309-0

Mieremet, M. M. J., Stolle, D. F., Ceccato, F., & Vuik, C. (2016). Numerical stability for modelling of dynamic two-phase interaction. *International Journal for Numerical and Analytical Methods in Geomechanics*. https://doi.org/10.1002/nag.2483

Montazeri, Z., Xiao, C., Fei, Y. R., Zheng, C., & Zhao, S. (2019). Mechanics-Aware Modeling of Cloth Appearance. *IEEE Transactions on Visualization and Computer Graphics*, 1–1. https://doi.org/10.1109/TVCG.2019.2937301

Moresi, L., Dufour, F., & Mühlhaus, H. B. (2003). A Lagrangian integration point finite element method for large deformation modeling of viscoelastic geomaterials. *Journal of Computational Physics*. https://doi.org/10.1016/S0021-9991(02)00031-1

Motlagh, Y. G., & Coombs, W. M. (2017). An Implicit High-order Material Point Method. *Procedia Engineering*, *175*(2016), 8–13. https://doi.org/10.1016/j.proeng.2017.01.003

Moutsanidis, Georgios; Kamensky, David; Zhang, Duan Z; Bazilevs, Yuri; Long, C. C. (2019).

Modeling strong discontinuities in the material point method using a single velocity field. *Computer Methods in Applied Mechanics and Engineering*, *345*, 584–601.

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*. https://doi.org/10.1007/s10346-018-01131-1

Nair, A., & Roy, S. (2012). Implicit time integration in the generalized interpolation material point method for finite deformation hyperelasticity. *Mechanics of Advanced Materials and Structures*. https://doi.org/10.1080/15376494.2010.550082

Nairn, JA; Guo, Y. (2005). Material point method calculations with explicit cracks, fracture parameters, and crack propagation. *11th International Conference on Fracture, Turin, Italy*.

Nairn, J. A. (2003). Material point method calculations with explicit cracks. *CMES - Computer Modeling in Engineering and Sciences*, *4*(6), 649–663.

Nairn, John A. (2007). Material point method simulations of transverse fracture in wood with realistic morphologies. *Holzforschung*, *61*(4), 375–381. https://doi.org/10.1515/HF.2007.057

Nairn, John A. (2016). Numerical modelling of orthogonal cutting: application to woodworking with a bench plane. *Interface Focus*, *6*(3), 20150110. https://doi.org/10.1098/rsfs.2015.0110

Nairn, John A., & Guilkey, J. E. (2015). Axisymmetric form of the generalized interpolation material point method. *International Journal for Numerical Methods in Engineering*, *101*(2), 127–147. https://doi.org/10.1002/nme.4792

Nairn, John A., Hammerquist, C. C., & Smith, G. D. (2020). New material point method

contact algorithms for improved accuracy, large-deformation problems, and proper null-space filtering. *Computer Methods in Applied Mechanics and Engineering*, *362*, 112859. https://doi.org/10.1016/j.cma.2020.112859

Nairn, John A. (2009). Analytical and numerical modeling of R curves for cracks with bridging zones. *International Journal of Fracture*, *155*(2), 167.

Overby, M., Brown, G. E., Li, J., & Narain, R. (2017). ADMM ⊇ Projective Dynamics: Fast Simulation of Hyperelastic Models with Dynamic Constraints. IEEE Transactions on Visualization and Computer Graphics. https://doi.org/10.1109/TVCG.2017.2730875

Parker, S. G., Guilkey, J., & Harman, T. (2006). A component-based parallel infrastructure for the simulation of fluid–structure interaction. *Engineering with Computers*. https://doi.org/10.1007/s00366-006-0047-5

Pinyol, N. M., Alvarado, M., Alonso, E. E., & Zabala, F. (2018). Thermal effects in landslide mobility. *Geotechnique*, *68*(6), 528–545. https://doi.org/10.1680/jgeot.17.P.054

Powell, M. J. D., & Sabin, M. A. (1977). Piecewise Quadratic Approximations on Triangles. *ACM Transactions on Mathematical Software (TOMS)*. https://doi.org/10.1145/355759.355761

Ram, D., Gast, T., Jiang, C., Schroeder, C., Stomakhin, A., Teran, J., & Kavehpour, P. (2015). A material point method for viscoelastic fluids, foams and sponges. *Proceedings of the 14th ACM SIGGRAPH / Eurographics Symposium on Computer Animation - SCA '15*, 157–163. https://doi.org/10.1145/2786784.2786798

Rankine, W. J. M. (1857). II. On the stability of loose earth. *Philosophical Transactions of the Royal Society of London*, *27*(147), 9.

Raviart, P. A. (1985). An analysis of particle methods. In F. Brezzi (Ed.), *Numerical Methods*

*in Fluid Dynamics. Lecture Notes in Mathematics, vol 1127.* (pp. 243–324). Springer Berlin Heidelberg. https://doi.org/10.1007/BFb0074532

Redaelli, I., Ceccato, F., Di Prisco, C., & Simonini, P. (2017). Solid-fluid Transition in Granular Flows: MPM Simulations with a New Constitutive Approach. *Procedia Engineering*, *175*(2016), 80–85. https://doi.org/10.1016/j.proeng.2017.01.028

Richard H. MacNeal. (1994). *Finite elements: Their design and performance*. Marcel Dekker, Inc.

Sadeghirad, A., Brannon, R. M., & Burghardt, J. (2011). A convected particle domain interpolation technique to extend applicability of the material point method for problems involving massive deformations. *International Journal for Numerical Methods in Engineering*. https://doi.org/10.1002/nme.3110

Sadeghirad, A., Brannon, R. M., & Guilkey, J. (2011). Enriched convected particle domain interpolation (CPDI) method for analyzing weak discontinuities. *Particle-Based Methods II - Fundamentals and Applications*, *252*(1995), 673–684.

Sadeghirad, A., Brannon, R. M., & Guilkey, J. E. (2013). Second-order convected particle domain interpolation (CPDI2) with enrichment for weak discontinuities at material interfaces. *International Journal for Numerical Methods in Engineering*. https://doi.org/10.1002/nme.4526

Schreyer, H. L., Sulsky, D. L., & Zhou, S. J. (2002). Modeling delamination as a strong discontinuity with the material point method. *Computer Methods in Applied Mechanics and Engineering*, *191*(23–24), 2483–2507. https://doi.org/10.1016/S0045-7825(01)00409-1

Seyedan, S., & Solowski, W. T. (2017). Estimation of granular flow impact force on rigid wall

using material point method. *5th International Conference on Particle-Based Methods - Fundamentals and Applications, PARTICLES 2017*.

Seyedan, S., & Sołowski, W. T. (2019). Continuum modelling of the granular flows in gaseous states using material point method. *2nd International Conference on the Material Point Method for Modelling Soil-Water-Structure Interaction Anura3D*.

S. Seyedan and W. T. Sołowski (2021), "From solid to disconnected state and back: Continuum modelling of granular flows using material point method," Comput. Struct., vol. 251, p. 106545, Jul. 2021, doi: 10.1016/j.compstruc.2021.106545.

Shen, L. (2009). A rate-dependent damage/decohesion model for simulating glass fragmentation under impact using the material point method. *CMES - Computer Modeling in Engineering and Sciences*. https://doi.org/10.3970/cmes.2009.049.023

Soga, K., Alonso, E., Yerro, A., Kumar, K., & Bandara, S. (2016). Trends in large-deformation analysis of landslide mass movements with particular emphasis on the material point method. *Géotechnique*, *66*(3), 248–273. https://doi.org/10.1680/jgeot.15.LM.005

Sołowski, W. T., & Sloan, S. W. (2015). Evaluation of material point method for use in geotechnics. *International Journal for Numerical and Analytical Methods in Geomechanics*, *39*(7), 685–701. https://doi.org/10.1002/nag.2321

Sołowski, Wojciech T., & Sloan, S. W. (2013). Modelling of sand column collapse with material point method. *Computational Geomechanics, COMGEO III - Proceedings of the 3nd International Symposium on Computational Geomechanics*, 698–705.

Sołowski, Wojciech Tomasz, & Sloan, S. W. (2014). Material Point Method Modelling of Granular Flow in Inclined Channels. *Applied Mechanics and Materials*, *553*, 501–506. https://doi.org/10.4028/www.scientific.net/AMM.553.501

Sorensen, K. K., Baudet, B. A., & Simpson, B. (2010). Influence of strain rate and acceleration on the behaviour of reconstituted clays at small strains. *Geotechnique*. https://doi.org/10.1680/geot.07.D.147

Souers, P. C., Anderson, S., Mercer, J., McGuire, E., & Vitello, P. (2000). JWL++: A Simple Reactive Flow Code Package for Detonation. *Propellants, Explosives, Pyrotechnics*. https://doi.org/10.1002/(SICI)1521-4087(200004)25:2<54::AID-PREP54>3.0.CO;2-3

Steffen, M., Wallstedt, P. C., Guilkey, J. E., Kirby, R. M., & Berzins, M. (2008). Examination and analysis of implementation choices within the Material Point Method (MPM). *CMES - Computer Modeling in Engineering and Sciences*. https://doi.org/10.3970/cmes.2008.031.107

Steffen, Michael, Kirby, R. M., & Berzins, M. (2008). Analysis and reduction of quadrature errors in the material point method (MPM). *International Journal for Numerical Methods in Engineering*, *76*(6), 922–948. https://doi.org/10.1002/nme.2360

Steffen, Michael, Kirby, R. M., & Berzins, M. (2010). Decoupling and balancing of space and time errors in the material point method (MPM). *International Journal for Numerical Methods in Engineering*. https://doi.org/10.1002/nme.2787

Stomakhin, A., Schroeder, C., Chai, L., Teran, J., & Selle, A. (2013). *A material point method for snow simulation*.

Stomakhin, A., Schroeder, C., Jiang, C., Chai, L., Teran, J., & Selle, A. (2014). Augmented MPM for phase-change and varied materials. *ACM Transactions on Graphics*, *33*(4). https://doi.org/10.1145/2601097.2601176

Sulsky, D., Chen, Z., & Schreyer, H. L. (1994). A particle method for history-dependent materials. *Computer Methods in Applied Mechanics and Engineering*, *118*(1–2), 179–

196. https://doi.org/10.1016/0045-7825(94)90112-0

Sulsky, D., & Kaul, A. (2004). Implicit dynamics in the material-point method. *Computer Methods in Applied Mechanics and Engineering*, *193*(12–14), 1137–1170. https://doi.org/10.1016/j.cma.2003.12.011

Sulsky, Deborah. (2017). *MPM 2017 Keynote Lecture 4 , 11 January 2017 ACCURACY AND STABILITY OF THE MATERIAL-POINT METHOD Department of Mechanical Engineering University of New Mexico*. *January*.

Sulsky, Deborah, & Gong, M. (2016). Improving the material-point method. *Lecture Notes in Applied and Computational Mechanics*. https://doi.org/10.1007/978-3-319-39022-2_10

Sulsky, Deborah, Zhou, S.-J., & Schreyer, H. L. (1995). Application of a particle-in-cell method to solid mechanics. *Computer Physics Communications*, *87*(1–2), 236–252. https://doi.org/10.1016/0010-4655(94)00170-7

Tampubolon, A. P., Gast, T., Klár, G., Fu, C., Teran, J., Jiang, C., & Museth, K. (2017). Multi-species simulation of porous sand and water mixtures. *ACM Transactions on Graphics*, *36*(4), 1–11. https://doi.org/10.1145/3072959.3073651

Tan, H., & Nairn, J. A. (2002). Hierarchical, adaptive, material point method for dynamic energy release rate calculations. *Computer Methods in Applied Mechanics and Engineering*, *191*(19–20), 2123–2137. https://doi.org/10.1016/S0045-7825(01)00377-2

Tielen, R., Wobbes, E., Möller, M., & Beuth, L. (2017). A High Order Material Point Method. *Procedia Engineering*, *175*(2016), 265–272. https://doi.org/10.1016/j.proeng.2017.01.022

Tran, L. T., Kim, J., & Berzins, M. (2010). Solving time-dependent PDEs using the material point method, a case study from gas dynamics. *International Journal for Numerical*

*Methods in Fluids*. https://doi.org/10.1002/fld.2031

Tran, Q.-A., & Sołowski, W. (2019). Generalized Interpolation Material Point Method modelling of large deformation problems including strain-rate effects – Application to penetration and progressive failure problems. *Computers and Geotechnics*, *106*, 249–265. https://doi.org/10.1016/j.compgeo.2018.10.020

Tran, Q.-A., Solowski, W., Karstunen, M., & Korkiala-Tanttu, L. (2017). Modelling of Fall-cone Tests with Strain-rate Effects. *Procedia Engineering*, *175*(2016), 293–301. https://doi.org/10.1016/j.proeng.2017.01.029

Tran, Q.-A., & Sołowski, W. T. (2017). Large strain consolidation modelling using Dual Domain Material Point Method. *15th IACMAG, September*.

Tran, Q. A., Sołowski, W., Berzins, M., & Guilkey, J. (2020). A convected particle least square interpolation material point method. *International Journal for Numerical Methods in Engineering*, *121*(6), 1068–1100. https://doi.org/10.1002/nme.6257

Tran, Q., Berzins, M., & Sołowski, W. T. (2019). An improved moving least squares method for the Material Point Method. *2nd International Conference on the Material Point Method for Modelling Soil-Water-Structure Interaction Anura3D*, 27–34.

Tran, Q., & Sołowski, W. (2019). Temporal and null-space filter for the material point method. *International Journal for Numerical Methods in Engineering*, *120*(3), 328–360. https://doi.org/10.1002/nme.6138

Tran, Q., Solowski, W. T., & Cummings, E. (2017). Hydro-mechanical coupled dual domain material point method stabilized with a null-space filter. *5th International Conference on Particle-Based Methods - Fundamentals and Applications, PARTICLES 2017*.

Tran, Q., Solowski, W., Thakur, V., & Karstunen, M. (2017). Modelling of the quickness test

of sensitive clays using the generalized interpolation material point method. In *Advances in Natural and Technological Hazards Research* (Vol. 46). https://doi.org/10.1007/978-3-319-56487-6_29

Wallstedt, P. C., & Guilkey, J. E. (2008). An evaluation of explicit time integration schemes for use with the generalized interpolation material point method. *Journal of Computational Physics*, *227*(22), 9628–9642. https://doi.org/10.1016/j.jcp.2008.07.019

Wallstedt, P. C., & Guilkey, J. E. (2010). *A weighted least squares particle-in-cell method for solid mechanics*. https://doi.org/10.1002/nme

Walters, William, P., & Zukas, J. A. (1989). *Fundamentals of Shaped Charges*. John Wiley & Sons.

Wang, B., Vardon, P. J., & Hicks, M. A. (2014). Investigation of aspects of an implicit dynamic material point method implementation. *Numerical Methods in Geotechnical Engineering - Proceedings of the 8th European Conference on Numerical Methods in Geotechnical Engineering, NUMGE 2014*, *1*, 313–318.

Wang, B., Vardon, P. J., & Hicks, M. A. (2018). Rainfall-induced slope collapse with coupled material point method. *Engineering Geology*, *239*, 1–12. https://doi.org/10.1016/j.enggeo.2018.02.007

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

Wang, L., Coombs, W. M., Augarde, C. E., Cortis, M., Charlton, T. J., Brown, M. J., Knappett, J., Brennan, A., Davidson, C., Richards, D., & Blake, A. (2019). On the use of domain-based material point methods for problems involving large distortion. *Computer Methods*

*in Applied Mechanics and Engineering*. https://doi.org/10.1016/j.cma.2019.07.011

Wang, S., Ding, M., Gast, T. F., Zhu, L., Gagniere, S., Jiang, C., & Teran, J. M. (2019). Simulation and Visualization of Ductile Fracture with the Material Point Method. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*. https://doi.org/10.1145/3340259

Wobbes, E., Möller, M., Galavi, V., & Vuik, C. (2019). Conservative Taylor least squares reconstruction with application to material point methods. *International Journal for Numerical Methods in Engineering*, *117*(3), 271–290. https://doi.org/10.1002/nme.5956

Wolper, Joshuah; Fang, Yu; Li, Minchen; Lu, Jiecong; Gao, Ming; Jiang, C. (2019). CD-MPM: continuum damage material point methods for dynamic fracture animation. *ACM Transactions on Graphics (TOG)*, *38*(4), 1–15.

Yamaguchi, Y., Takase, S., Moriguchi, S., & Terada, K. (2020). Solid–liquid coupled material point method for simulation of ground collapse with fluidization. *Computational Particle Mechanics*. https://doi.org/10.1007/s40571-019-00249-w

Yang, W.-C., Arduino, P., Miller, G. R., & Mackenzie-Helnwein, P. (2018). Smoothing algorithm for stabilization of the material point method for fluid–solid interaction problems. *Computer Methods in Applied Mechanics and Engineering*, *342*, 177–199. https://doi.org/10.1016/j.cma.2018.04.041

Yang, Y., Sun, P., & Chen, Z. (2017). Combined MPM-DEM for Simulating the Interaction between Solid Elements and Fluid Particles. *Communications in Computational Physics*. https://doi.org/10.4208/cicp.OA-2016-0050

Yerro, A., Alonso, E. E., & Pinyol, N. M. (2015). The material point method for unsaturated soils. *Géotechnique*, *65*(3), 201–217. https://doi.org/10.1680/geot.14.P.163

Yerro, A., Alonso, E. E., & Pinyol, N. M. (2016). Run-out of landslides in brittle soils. *Computers and Geotechnics*, *80*, 427–439. https://doi.org/10.1016/j.compgeo.2016.03.001

Yin, Z. Y., Karstunen, M., Chang, C. S., Koskinen, M., & Lojander, M. (2011). Modeling time-dependent behavior of soft sensitive clay. *Journal of Geotechnical and Geoenvironmental Engineering*, *137*(11), 1103–1113. https://doi.org/10.1061/(ASCE)GT.1943-5606.0000527

Zabala, F., & Alonso, E. E. E. (2011). Progressive failure of Aznalcóllar dam using the material point method. *Géotechnique*, *61*(9), 795–808. https://doi.org/10.1680/geot.9.P.134

Zhang, D. Z., Ma, X., & Giguere, P. T. (2011). Material point method enhanced by modified gradient of shape function. *Journal of Computational Physics*, *230*(16), 6379–6398. https://doi.org/10.1016/j.jcp.2011.04.032

Zhang, D. Z., Zou, Q., VanderHeyden, W. B., & Ma, X. (2008). Material point method applied to multiphase flows. *Journal of Computational Physics*, *227*(6), 3159–3173. https://doi.org/10.1016/j.jcp.2007.11.021

Zhang, F., Zhang, X., Sze, K. Y., Lian, Y., & Liu, Y. (2017). Incompressible material point method for free surface flow. *Journal of Computational Physics*, *330*, 92–110. https://doi.org/10.1016/j.jcp.2016.10.064

Zhang, F., Zhang, X., Sze, K. Y., Liang, Y., & Liu, Y. (2018). Improved Incompressible Material Point Method Based on Particle Density Correction. *International Journal of Computational Methods*. https://doi.org/10.1142/S0219876218500615

Zhang, H. W., Wang, K. P., & Chen, Z. (2009). Material point method for dynamic analysis of saturated porous media under external contact/impact of solid bodies. *Computer Methods*

*in Applied Mechanics and Engineering*, *198*(17–20), 1456–1472. https://doi.org/10.1016/j.cma.2008.12.006

Zhang, X., Sze, K. Y., & Ma, S. (2006). An explicit material point finite element method for hyper-velocity impact. *International Journal for Numerical Methods in Engineering*. https://doi.org/10.1002/nme.1579

Zhao, J., Chen, Y., Zhang, H., Xia, H., Wang, Z., & Peng, Q. (2019). Physically based modeling and animation of landslides with MPM. *Visual Computer*. https://doi.org/10.1007/s00371-019-01709-3

Zhao, Y., & Choo, J. (2020). Stabilized material point methods for coupled large deformation and fluid flow in porous materials. *Computer Methods in Applied Mechanics and Engineering*. https://doi.org/10.1016/j.cma.2019.112742

Zheng, Y., Gao, F., Zhang, H., & Lu, M. (2013). Improved convected particle domain interpolation method for coupled dynamic analysis of fully saturated porous media involving large deformation. *Computer Methods in Applied Mechanics and Engineering*, *257*, 150–163. https://doi.org/10.1016/j.cma.2013.02.001