

iGIMP: An implicit Generalised Interpolation Material Point Method for large deformations

T.J.Charlton, W.M.Coombs and C.E.Augarde

*School of Engineering and Computing Sciences, Durham University, South Road, Durham,
DH1 3LE, United Kingdom.*

Abstract

The Material Point Method (MPM) uses a combined Eulerian-Lagrangian approach to solve problems involving large deformations. A problem domain is discretised as material points which are advected on a background grid. Problems are encountered with the original MPM when material points cross between grid cells, and this has been tackled by the development of the Generalised Interpolation MPM, where material points' domains of influence extend beyond the currently occupied grid cell. In this paper, the Generalised Interpolation Material Point (GIMP) Method has been implemented implicitly in a manner that allows a global stiffness matrix to be constructed similar to that in the Finite Element Method (FEM) by combining contributions from individual elements on the background grid. An updated Lagrangian finite deformation framework has been used to ensure non-linear behaviour within each of the loadsteps. The weighting functions used for this which make the GIMP method different to standard MPM are presented and the implementation is explained. Specific details on computing the deformation gradient to be consistent with the updated Lagrangian framework and the updating of the material point influence domains are outlined, both of which are currently unclear in the published literature. It is then shown through numerical examples that for both small and large deformation elastic and elasto-plastic problems, the implicit GIMP method agrees well with analytical solutions and exhibits convergence properties between that

Email address: t.j.charlton@durham.ac.uk (T.J.Charlton)

of linear and quadratic FEM.

Keywords: material point method; generalised interpolation material point method; finite deformation; geometric non-linearity; elasto-plasticity

1. Introduction

When modelling continuum mechanics problems, specifically those involving large deformation, difficulties often occur when using mesh-based methods; such as displacement without re-meshing being limited by mesh distortion. Due to this, there has been increased interest recently in particle-based methods, in particular the Material Point Method (MPM). The MPM models a problem domain as a collection of Material Points (or particles) which move through a fixed background mesh on which calculations are carried out. This offers an advantage over many other mesh-free methods, which are also good for modelling large deformations and non-linear behaviour for example [1–3], because the existence of a background mesh removes the computational expense of undertaking nearest neighbour searches. In the MPM, properties are mapped between nodes on this background mesh and material points, during each load (or time) step. The majority of previous MPM research has looked at explicit formulations [4–46], with a few exceptions [47–55]. The advantages of adopting an implicit approach include allowance of much larger loadsteps, improved stability and error control, in comparison to explicit methods. An implicit formulation has also been shown in [49] to achieve superior accuracy. For static stress analysis problems, which are commonly tackled using an implicit Finite Element Method (FEM), there are benefits in an implicit MPM approach as there are many commonalities.

One of the main issues of the MPM is a well documented grid crossing instability. This occurs when material points move between elements in the background grid. Errors are introduced at the material points due to the non-continuous nature of the shape function derivatives between elements. These errors have been investigated in [56] and there are a selection of methods that have been proposed to address this issue and improve the MPM including CPDI

[57, 58], DDMP [59] and the Generalised Interpolation Material Point (GIMP) method [60]. Research into the GIMP method has also almost exclusively used an explicit approach, for example [60–72] with one notable exception [73] where the GIMP method is implemented implicitly using a matrix-free approach for dynamic problems using hyperelasticity. In an alternative approach [74] implicit MPM is used, however, rather than tackling the instabilities using GIMP, an additional non-physical stiffness term is introduced. Although this increases the numerical stability of the method, it destroys its ability to converge to analytical solutions.

In this paper, an implicit implementation of the basic GIMP method using an updated Lagrangian formulation is described using an approach that allows the local calculation of element stiffness matrices. The large deformation elasto-plastic implicit GIMP method described in this paper is implemented using an updated Lagrangian, geometrically non-linear formulation to accurately capture the behaviour of large strain problems. The formulation presented in this paper allows technology (such as constitutive models) to be easily shared between the implicit GIMP and implicit FEM and this is demonstrated by introducing von-Mises plasticity in some of the numerical examples. This is the first time that a fully implicit formulation has been proposed for an elasto-plastic GIMP method. The formulation includes the implementation of the spatial algorithmic consistent tangent to ensure optimum convergence of the global Newton process. This has required the calculation of the deformation gradient for implicit MPM methods to be clarified and a suitable domain updating procedure to be established. Additionally, we are able demonstrate the convergence properties of the GIMP method. The GIMP method is presented first before introducing the weighting functions (Section 2), outlining the finite deformation theory (Section 3) and describing the implementation (Section 4) before demonstrating the method using numerical examples (Section 5) and presenting conclusions (Section 6).

2. Generalised Interpolation Material Point Methods

2.1. Overview of MPM methods

The MPM was first developed by Sulsky *et al.* [75, 76] as an extension to solid mechanics of the Fluid Implicit Particle (FLIP) method [77, 78], which itself was an extension to the Particle in Cell (PIC) method [79] used in fluid dynamics. The problem domain is divided into a number of background cells forming an Eulerian mesh through which the material points travel. Information from the material points is interpolated to this background mesh where the calculations are carried out. The new values are then mapped back to the material points and the material point positions are updated. This results in a method that combines advantages of both Lagrangian and Eulerian approaches, allowing boundaries and history dependent variables to be tracked while not encountering the problems associated with mesh distortion in large deformation problems in other methods such as the FEM. Since its inception, the MPM has been developed and improved as well as being used in a number of applications including granular materials [41, 50, 80, 81], fracture [16, 25, 43, 69, 70, 82] and geotechnical applications [20, 21, 36, 37, 42, 46, 54, 55, 63, 68, 72, 83–86].

In the standard MPM a problem can arise when a material point crosses the boundary between one background grid cell and another. This is due to the fact that shape function derivatives are not continuous between elements and this results in incorrect stresses being calculated. A significant advance in the MPM was made in [60] where the cell crossing instability was reduced with the GIMP method. To attempt to alleviate the problem, the GIMP method introduces a material point characteristic function describing the influence domain of each particular material point. This modification from the MPM means that it is possible that a material point can influence nodes other than those associated with the element it is inside. This occurs when the material point is close enough to the edge of an element that the domain overlaps adjacent elements. The introduction of the GIMP method is shown in [60] to give an improved stress response to the MPM. The errors introduced when material points cross

element boundaries are reduced (although not completely removed [87]) because of the increased smoothness of the shape functions. Despite not being a complete remedy to grid crossing error the improvement shown by GIMP is significant.

2.2. Weighting functions

In the MPM, shape functions identical to those used in the FEM are used to relate nodal values to values at material points. For example, forces can be mapped to grid nodes from material points through

$$f_v = \sum_{i=1}^{n_p} f_{p_i} N_i, \quad (1)$$

where subscripts v and p refer to grid nodes (or vertices) and material points (or particles) respectively, n_p is the number of material points in elements surrounding the node and N_i are the standard shape functions as used in the FEM. A straightforward choice would be linear Lagrange shape functions given in 1D as

$$N_i = \frac{1 \pm \xi}{2} \quad (2)$$

where ξ is the local coordinate (in a domain $-1,+1$). These shape functions are also used to map properties from grid nodes to material points at the end of each step and their derivative are used to compute the stiffness matrix.

In the GIMP method [60], the standard FEM shape functions used in the MPM are replaced by weighting functions S_{vp} which are constructed based not only on the FEM shape functions but also a material point characteristic function χ_p specifying the influence domain of the material point. This is the key difference between the GIMP method and MPM, it can be thought of as giving each particle an associated domain rather than being a single point in a specific location. The weighting function (S_{vp}) can be calculated in a local coordinate system in one dimension (ξ) as

$$S_{vp} = \frac{1}{V_p} \int_{\Omega_p \cap \Omega} \chi_p(\xi) N_v(\xi) d\xi, \quad (3)$$

where V_p is the material point volume (or length in 1D), N_v are the shape functions as shown in (2) with subscript v indicating values are associated with grid nodes (or vertices), Ω is problem domain and Ω_p is the influence domain of the material point. The gradient of the weighting functions (∇S_{vp}), can also be calculated using

$$\nabla S_{vp} = \frac{1}{V_p} \int_{\Omega_p \cap \Omega} \chi_p(\xi) \nabla N_v(\xi) d\xi. \quad (4)$$

The original MPM can be recovered by setting the material point characteristic function equal to the Dirac delta function, that is

$$\chi_p(\xi) = \delta(\xi) V_p. \quad (5)$$

In the GIMP method, the use of different functions for $\chi_p(\xi)$ means that smoother weighting functions can be obtained. The simplest extension is to use a hat function with a value of unity within the material point's influence domain and zero elsewhere. This characteristic function, which is used in the development below can be expressed as

$$\chi_p(\xi) = \begin{cases} 1, & \text{if } \xi \in \Omega_p \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

Figure 1 demonstrates graphically how the GIMP weighting functions (3) can be constructed in one dimension from a convolution of the standard finite element shape functions and the material point characteristic function. The weighting function at a particular point can be thought of as the area of overlap between a material point's influence domain and the standard shape functions of the associated background grid node.

Figure 2 shows weighting functions (dashed lines) overlain on standard shape functions (solid lines) for a set of adjacent nodes. It can be seen that a material point positioned at a node would not solely contribute to that node and would instead also have a small amount of influence on the surrounding nodes. Despite this the GIMP weight functions still possess partition of unity. To construct weighting functions in multiple dimensions, the tensor product of one

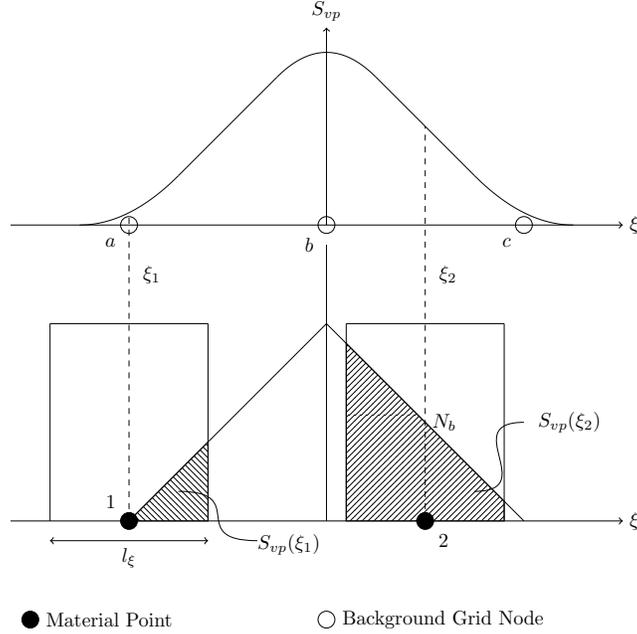
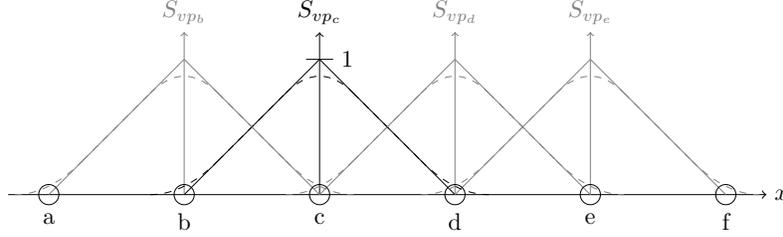


Figure 1: Weighting function (S_{vp}) associated with b can be thought of as a convolution of the standard shape function at (N_b) and a material point's characteristic function χ_p , with width l_ξ .

dimensional functions is taken. This use of separate functions relating to material points has similarities to that of other meshless methods [88].

Using weighting functions that extend outside of an element presents a problem if one wishes to calculate element stiffness matrices in a manner similar to the FEM. In this case it must be ensured that although material points outside an element can have influence, the influence is only for the part of that material point's influence domain overlapping with the element. To address this problem, new weighting functions are constructed, where the integrations in (3) and (4) are only calculated over the area of each element.

Figure 3 shows (in a similar manner to the GIMP weighting functions) how the overlapping area between the material point characteristic function and the standard FEM shape functions within an element can produce new functions S_{vp_a} and S_{vp_b} that allow element stiffnesses to be calculated in a manner not



○ Background Grid Node

Figure 2: Weighting functions for grid nodes b, c, d and e in one dimension. Solid lines show the case in the MPM, here it can be seen that influence only extends to adjacent elements. Dashed lines show an example where the characteristic function, χ_p is the top hat function.

previously possible. The weighting functions associated with the element a-b in Figure 3 are

$$S_{vp_a} = \frac{1}{4l_p} 2\xi_2 - \xi_2^2 - 2\xi_1 + \xi_1^2 \quad \text{and} \quad S_{vp_b} = \frac{1}{4l_p} 2\xi_2 + \xi_2^2 - 2\xi_1 - \xi_1^2 \quad (7)$$

where ξ_1 and ξ_2 are the integration limits for (3) in the local coordinates of the current element which can be given as

$$\xi_1 = \begin{cases} -1, & \text{if } \xi_p - \frac{l_p}{2} < -1 \\ \xi_p - \frac{l_p}{2}, & \text{if } \xi_p - \frac{l_p}{2} > -1 \end{cases} \quad (8)$$

$$\xi_2 = \begin{cases} 1, & \text{if } \xi_p + \frac{l_p}{2} > 1 \\ \xi_p + \frac{l_p}{2}, & \text{if } \xi_p + \frac{l_p}{2} < 1 \end{cases} \quad (9)$$

where ξ_p is the material point location and l_p is the material point domain size.

By summing these weighting functions at nodes from the contributions from different elements it is possible to recover the GIMP weighting functions as introduced earlier. This is illustrated in Figure 4, where S_{vp_b} is reconstructed from contributions from elements $a - b$ and $b - c$. It can be seen that these weighting functions mean that each node is not over- or under- accounted for.

The gradients of the weighting functions are calculated similarly using (4) and can be visualised as the overlap between the material point characteristic

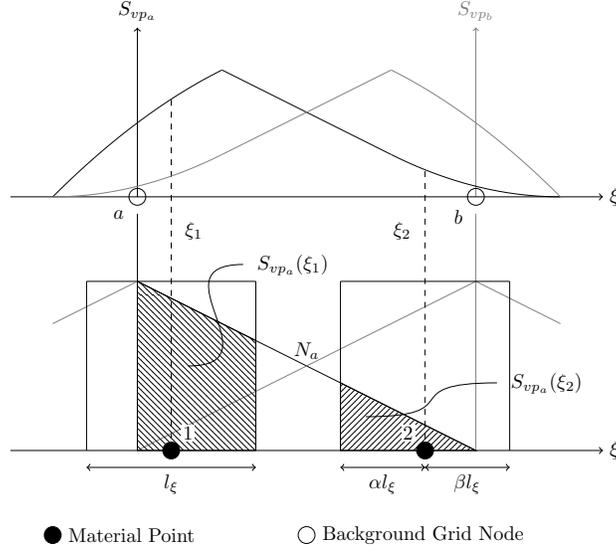


Figure 3: Element based weighting functions can be constructed from the convolution of the material point characteristic function with shape functions of nodes of the element.

function and the gradients of the standard shape functions. Figure 5 shows the gradient of the weighting functions within each element and the sum of these at node b . It can again be seen that these functions extend beyond the element but the gradients of the GIMP shape functions are recovered when contributions from both elements are considered. The area with a constant gradient is the section where the material point's influence domain is fully inside the element; at this point it is equal to the standard FEM shape functions. The gradient weighting functions for element $a - b$ in Figure 5 can be expressed as

$$\nabla S_{vp_a} = \frac{\xi_1 - \xi_2}{2l_p} \quad \text{and} \quad \nabla S_{vp_b} = \frac{\xi_2 - \xi_1}{2l_p} \quad (10)$$

where ξ_1 and ξ_2 are given by (8) and (9).

3. Geometrically non-linear GIMP

In this paper an updated Lagrangian finite deformation formulation is combined with logarithmic strain and Kirchhoff stress measures that control the

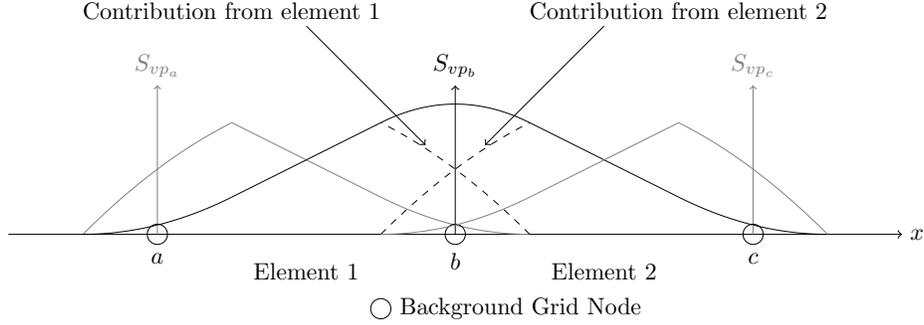


Figure 4: The GIMP weighting function shown for node b by the solid dark line. This function is constructed from the sum of the weighting functions within elements 1 and 2, which are shown by the dashed lines.

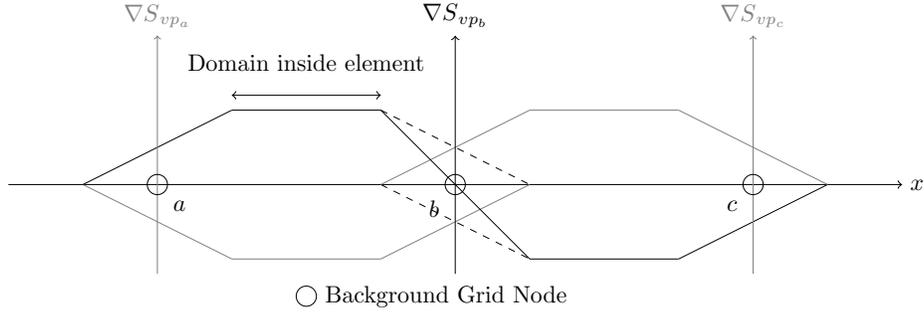


Figure 5: The GIMP gradient weighting function shown for node b by the solid dark line. This function is constructed from the sum of the weighting functions within elements 1 and 2, which are shown by the dashed lines.

constitutive behaviour at each material point. This formulation is one of the most straightforward ways to implement large strain elasto-plasticity within a finite element framework [89]. In this framework all static and kinematic variables are referred to the previously converged state, rather than the original state in a total Lagrangian formulation. The majority of this section uses index notation to detail the geometric non-linear formulation; only the discrete equations are expressed in matrix-vector form for convenience. The finite deformation framework adopted in this paper is based on implementations given in [90] and [91]. The framework has more generally been accepted and used by

a number of authors including [92–94]. It is possible to extend this to allow plastic anisotropy using the formulation of Eterovic and Bathe [95] or to allow elastic and plastic anisotropy following the work of Caminero et al [96] without modifying the overall framework. In this paper examples are restricted to isotropic elasto-plasticity for simplicity to ensure clarity of the GIMP method.

The weak form of equilibrium for an updated Lagrangian formulation can be expressed as

$$\int_{\varphi_t(\mathcal{B})} (\sigma_{ij}(\nabla\eta)_{ij} - b_i\eta_i) dv - \int_{\varphi_t(\partial\mathcal{B})} (t_i\eta_j) ds = 0 \quad (11)$$

where φ_t is the motion of a material body, \mathcal{B} , subject to body forces, b_i , and tractions, t_i , on the boundary of the material domain, $\partial\mathcal{B}$. The weak form is derived using a field of admissible virtual displacements η_i . Within this statement of equilibrium, the deformation gradient is the fundamental variable that characterises the deformation at a material point

$$F_{ij} = \frac{\partial x_i}{\partial X_j}, \quad (12)$$

where X_j are the original reference coordinates and $x_i = \varphi(X_i, t)$ are the updated coordinates of the material point. It is assumed that the deformation gradient can be multiplicatively split into elastic and plastic components [97, 98]

$$F_{ij} = F_{ik}^e F_{kj}^p, \quad (13)$$

where the superscripts e and p denote elastic and plastic terms, respectively.

When implementing large strain elasto-plasticity there is a choice of which stress and strain measures to adopt. Here, we use the combination of logarithmic strains with Kirchhoff stresses as it allows the use of conventional small strain constitutive equations within a finite deformation framework. In particular, the stress integration algorithm of plasticity models does not change provided that these stress and strain measures are combined with an exponential map of the plastic flow equation (see [99] or [100] for full details). All of the constitutive models used in this paper adopt a fully implicit stress integration algorithm based on backward Euler. This allows the updated stress state to be determined

given a trial stress state (or trial elastic strain state) and the relevant constitutive parameters, again see [100] for full details.

Here we restrict this framework to the case where a linear relationship exists between the elastic logarithmic strains ε_{ij}^e and the Kirchhoff stresses τ_{ij} that is

$$\tau_{ij} = D_{ijkl}^e \varepsilon_{kl}^e, \quad (14)$$

where D_{ijkl}^e is the linear elastic isotropic material stiffness tensor. The elastic logarithmic strain is defined as

$$\varepsilon_{ij}^e = \frac{1}{2} \ln(b_{ij}^e), \quad (15)$$

where $b_{ij}^e = F_{ik}^e F_{jk}^e$ are the components of the elastic left Cauchy-Green strain tensor. The Cauchy stress can be obtained from the Kirchhoff stress using the relationship

$$\sigma_{ij} = J^{-1} \tau_{ij}, \quad (16)$$

where $J = \det(F_{ij})$ is the volume ratio. In order to obtain the current Kirchhoff stress state, τ_{ij} , the constitutive model requires a trial stress (or elastic strain state) to act as the initial estimate in the backward Euler stress integration algorithm. The trial elastic left Cauchy-Green strain tensor $(b_t^e)_{ij}$ is obtained from

$$(b_t^e)_{ij} = \Delta F_{ik} (b_n^e)_{kl} \Delta F_{jl}, \quad (17)$$

where the subscripts t and n denote trial and previously converged states, respectively, rather than a physical index. ΔF_{ij} is the increment in the deformation gradient for the current loadstep (see Section 3.2). The previous elastic left Cauchy-Green strain tensor $(b_n^e)_{ij}$ can be obtained from the elastic strain state from the previously converged loadstep

$$(b_n^e)_{ij} = \exp(2(\varepsilon_n^e)_{ij}) \quad (18)$$

and the trial elastic strain is obtained using

$$(\varepsilon_t^e)_{ij} = \frac{1}{2} \ln((b_t^e)_{ij}). \quad (19)$$

The updated Kirchhoff stress and the updated elastic strain states can then both be obtained from the constitutive algorithm.

3.1. Discrete implementation

Introducing the element approximation for the displacements at a material point

$$\varphi_i = \sum_{a=1}^{n_{en}} (S_{vp})_a (d_i)_a \quad \text{and} \quad \eta_i = \sum_{a=1}^{n_{en}} (S_{vp})_a (d_i^\eta)_a, \quad (20)$$

where d_i and d_i^η are the physical and virtual nodal displacements, respectively, a denotes the node number and n_{en} is the number of nodes associated with an element. The Galerkin form of the weak statement of equilibrium over an element, E , can be obtained from (11) and (20) as

$$\{f^{RE}\} = \int_{\varphi_t(E)} [G]^T \{\sigma\} dv - \int_{\varphi_t(E)} [S_{vp}]^T \{b\} dv - \int_{\varphi_t(\partial E)} [S_{vp}]^T \{t\} ds = \{0\}, \quad (21)$$

where $[S_{vp}]$ the GIMP shape function matrix and $[G]$ is the tensorial form of the strain-displacement matrix containing the derivatives of the GIMP shape functions with respect to the updated nodal coordinates. The first term in (21) is the internal force within an element and the combination of the second and third terms is the external force vector. Equation (21) is non-linear in terms of the unknown nodal displacements and can be efficiently solved using the standard Newton-Raphson (NR) procedure. The nodal displacements within a load step, $\{\Delta d\}$, are obtained by iteratively updating the displacements until (21) is satisfied within a given tolerance, that is

$$\{\delta d_{k+1}\} = [K]^{-1} \{f_k^R\}, \quad (22)$$

where $k+1$ denotes the current iteration number, $\{\delta d_{k+1}\}$ are the iterative nodal displacements, $\{f_k^R\}$ is the global residual out-of-balance force vector (21) from the k th iteration and $[K]$ is the global tangent stiffness matrix. The current displacement increment within a load step can be obtained through

$$\{\Delta d_{k+1}\} = \sum_{n=1}^{k+1} \{\delta d_n\}. \quad (23)$$

Linearising (21) with respect to the unknown nodal coordinates, and assuming that the applied body forces and surface tractions are independent of the nodal

displacements, gives the element contribution to the global stiffness matrix as

$$[k^E] = \int_{\varphi(E)} [G]^T [a][G] dv. \quad (24)$$

The non-symmetric spatial material tangent modulus of a material point is given by

$$a_{ijkl} = \frac{1}{2J} D_{ijmn}^{\text{alg}} L_{mnpq} B_{pqkl} - S_{ijkl}, \quad (25)$$

where

$$L_{mnpq} = \frac{\partial \ln(b_{mn}^e)}{\partial b_{pq}^e}, \quad B_{pqkl} = \delta_{pk} b_{ql}^e + \delta_{qk} b_{pl}^e \quad \text{and} \quad S_{ijkl} = \sigma_{il} \delta_{jk}. \quad (26)$$

D_{ijmn}^{alg} is the small strain algorithmically consistent tangent, that is, the tangent that is consistent with the adopted stress integration algorithm [101]. The use of this tangent allows for asymptotic quadratic convergence of the global residual (21). L_{ijkl} can be determined as a particular case of the derivative of a general symmetric second order tensor function with respect to its argument; see Miehe [102] for details.

In material point methods, (24) is evaluated through the summation of the material point stiffness contributions where the nodal stiffness components of a single material point can be obtained through

$$[k^p] = [G]^T [a][G] V_p, \quad (27)$$

where V_p is the material point volume in the spatial frame, that is

$$V_p = \det(\Delta F_{ij}) V_p^n. \quad (28)$$

V_p^n is the material point volume at the previously converged state, obtained from the product of the global influence domain lengths in the Cartesian directions. A material point's contribution to the internal force vector is given by

$$\{f^p\} = [G]^T \{\sigma\} V_p \quad (29)$$

Note that it is essential to use the volume in the spatial frame (28) in both (27) and (29) to ensure the correct order of convergence of the NR process. It may not initially be clear that $\det(\Delta F_{ij})$ must be included to obtain the correct volume.

3.2. Deformation gradient calculation

One point of departure of implicit MP methods from conventional finite elements is the calculation of the deformation gradient. In an updated Lagrangian formulation, the deformation gradient is normally obtained through [99]

$$F_{ij} = \Delta F_{ik} F_{kj}^n \quad \text{where} \quad \Delta F_{ij} = \left(\delta_{ij} - \frac{\partial \Delta u_i}{\partial x_j} \right)^{-1}. \quad (30)$$

F_{ij}^n is the deformation gradient from the previously converged state, Δu_i is the displacement increment in the current load step, x_i are the updated coordinates (position in the spatial configuration determined from the nodal positions). However, in the MPM the concept of the current (or original) coordinates of the nodes does not exist. The reason for this is that in MP methods the shape functions, and their derivatives, are defined assuming that the global coordinates of the background mesh remain in a regular grid. It is therefore not possible to use (30)₂ to determine the deformation gradient increment. Instead, the increment in the deformation gradient must be obtained using (see [99] amongst others)

$$\Delta F_{ij} = \delta_{ij} + \frac{\partial \Delta u_i}{\partial \tilde{X}_j}, \quad (31)$$

where $\tilde{X}_i = x_i - \Delta u_i$ are the coordinates at the start of the load step. $\frac{\partial \Delta u_i}{\partial \tilde{X}_j}$ is constructed at each material point by summing contributions from each element it overlaps.

$$\frac{\partial \Delta u_i}{\partial \tilde{X}_j} = \sum_1^{n_{els}} \Delta u_i \frac{\partial (S_{vp})}{\partial \tilde{X}_i}. \quad (32)$$

The derivatives with respect to global coordinates at the start of a loadstep can then be obtained as

$$\frac{\partial (S_{vp})_a}{\partial \tilde{X}_i} = \left(\frac{\partial \tilde{X}_i}{\partial \xi_j} \right)^{-1} \frac{\partial (S_{vp})_a}{\partial \xi_j} \quad (33)$$

where

$$\frac{\partial \tilde{X}_i}{\partial \xi_j} = \sum_{a=1}^{n_{en}} \frac{\partial (N)_a}{\partial \xi_j} (\tilde{X}_i)_a. \quad (34)$$

Equation (31) allows the determination of the increment in the deformation gradient based on a regular (undeformed) background grid. However, in order

to form the stiffness matrix and internal force vector for an updated Lagrangian formulation we require the derivatives of the shape functions with respect to the current coordinates, x_i . The mapping that links the current coordinate, x_i , to the coordinate at the start of the load step, \tilde{X}_i , is

$$\frac{\partial \tilde{X}_i}{\partial x_j} = \delta_{ij} - \frac{\partial \Delta u_i}{\partial x_j} = (\Delta F_{ij})^{-1}, \quad (35)$$

that is, the inverse mapping of the increment in the deformation gradient (obtained from (31)). The derivatives of the shape functions with respect to the updated coordinates follow as

$$\frac{\partial (S_{vp})_a}{\partial x_j} = \frac{\partial (S_{vp})_a}{\partial \tilde{X}_i} \frac{\partial \tilde{X}_i}{\partial x_j} = \frac{\partial (S_{vp})_a}{\partial \tilde{X}_i} (\Delta F_{ij})^{-1}, \quad (36)$$

where a is the node number. These derivatives are required for the construction of $[G]$, first seen in (21), as this matrix contains the derivatives of the basis functions with respect to the current nodal coordinates.

3.3. Domain updating

MP methods usually model a problem over a number of loadsteps and this presents an opportunity to update the influence domains of the material points at the end of each load step. Two ways of doing this labelled uGIMP and cpGIMP were presented by Wallstedt and Guilkey in [62]. uGIMP keeps the material point influence domains unchanged between loadsteps. This is the simplest approach to take, however it often results in domains overlapping each other or separating. cpGIMP addresses this problem by updating the size of the influence domain using the diagonal components of the deformation gradient. This rectifies the problem when deformation is aligned with the grid, however the method fails to improve matters when any rotation of the material occurs. Sadeghirad *et al.* [57] developed another approach known as the Convected Particle Domain Interpolation (CPDI) method which improves the MPM by updating the influence domains associated with material points. In the CPDI method, the initially rectangular material point domains are allowed to deform into parallelograms. Thus, the CPDI method is an extension to GIMP which

removes a problem that exists when rotations occur for updating the material points. A further extension can be achieved by tracking the domain corners as shown in [58, 73] which can ensure material points are contiguous. However, the CPDI method induces an additional approximation in the way that the basis functions of a material point are determined. Unlike in the GIMP method where the basis functions are determined analytically, in the CPDI method, to obtain the integration of the grid shape functions over the particle domain, a linear approximation between domain corner points is introduced. If all of the corners of a particle domain do not lie in the same element then errors are introduced into the basis function determination as the discontinuous nature of the grid shape functions is not captured by the linear approximation.

A simpler way of addressing the rotation problem is to use only the stretch part of the deformation gradient for updating domains in a cpGIMP fashion. In the cpGIMP method, it is necessary to update the material point domain lengths instead of updating the material point volumes as an independent variable. In 1D this is the same, however in 2D or 3D it is important to take note of the changes in each direction. One option is to update the domain lengths, l_i^p , using components of the deformation gradient according to

$$l_i^p = l_i^{p_0} F_{ii} \quad (\text{no implied sum on } i), \quad (37)$$

where $l_i^{p_0}$ are the original domain lengths. However, problems arise when the rotational component of the deformation gradient is non-zero [57]. Instead, here we propose a new approach where the domain lengths are updated according to the symmetric material stretch tensor

$$U_{ij} = \sqrt{F_{ki} F_{kj}}, \quad (38)$$

where $F_{ij} = R_{ik} U_{kj}$ and R_{ij} is the rotational component of the deformation gradient. It should be clear from the above equation that the material stretch tensor is equivalent to the deformation gradient rotated back into the original reference frame. The material point domains can then be updated according to

$$l_i^p = l_i^{p_0} U_{ii} \quad (\text{no implied sum on } i). \quad (39)$$

This updating is performed at the end of a load step once the NR process has converged. The important consequence of this rather minor modification to the theory is demonstrated numerically in Section 5.3.

4. Numerical implementation

When implementing the iGIMP algorithm, the first step is to discretise the problem into a set of material points spread over the domain, within a regular background grid which extends beyond the physical domain. A notable difference to the standard FEM is this requirement for the grid to extend to where material is expected to move into during a simulation. It is possible to keep track of how much of the grid is covered by material points and extend it if necessary. As in the standard MPM, the background grid is not restricted to any particular shape, however for convenience a regular mesh is usually selected. In this work, two-noded line elements are used in 1D and four-noded elements are used in 2D aligned with the coordinate axis. It is possible to use the same techniques in 3D. Elements initially containing material are populated with material points and a weight is assigned based on the volume of material represented by each material point. The influence domains are defined to initially cover the whole of the material with no gaps or overlaps. In this work, material points are arranged inside the elements in a uniform manner, however other initial positions can be chosen.

At the start of each load step the location of each material point with respect to the background grid must be determined. The local coordinates of the material point (ξ, η) are calculated and, from these coordinates, the weighting functions (3) can be computed. Grid elements void of material points are also determined and not included in the calculation during the current load step. At this stage, any external forces on the material points should be incremented

and then mapped to the grid nodes using

$$\{f_v\} = \sum_{i=1}^{n_p} \{f_{p_i}\} [S_{vp_i}]. \quad (40)$$

The out-of-balance force is calculated as in (11) and the NR process is started. Displacements are calculated from the out-of-balance force and the stiffness calculated on the previous iteration (22). Afterwards, the strain displacement matrix and the derivatives of the displacement required for calculation of the deformation gradient can be calculated. It should be noted here that these quantities at the material points must be calculated as the sum of the different contributions from elements the material point overlaps, and particular attention should be made when calculating $[G]$ to take into account the mapping outlined in (36). Due to material points potentially having influence over different numbers of nodes, the size of $[G]$ can change between material points. The structure of $[G]$ in 1D for a material point overlapping two elements is as follows

$$[G] = \left[\left[\frac{\partial S_{vp1}^a}{\partial X} \right] \quad \left[\frac{\partial S_{vp2}^a}{\partial X} \right] + \left[\frac{\partial S_{vp2}^b}{\partial X} \right] \quad \left[\frac{\partial S_{vp3}^b}{\partial X} \right] \right] [\Delta F]^{-1}, \quad (41)$$

where superscripts a and b refer to the derivatives of the weighting functions in elements a and b . The stress, internal force and stiffness can be calculated as shown in Section 3.1, and the out-of-balance force can be evaluated to determine whether the NR process has converged.

At the end of each load step, once the NR process has converged to within the designated tolerance, the material point positions and domains are updated and the background grid is reset. The algorithm is outlined in more detail in Figures 6 and 7 showing the implicit Generalised Interpolation Material Point (iGIMP) algorithm in a similar way to [100]. It is possible to replace the material model in 7 with other models such as those presented in [96] to allow both elastic and plastic anisotropy, however for clarity the elasto-plastic model outlined in this paper has been included.

$lstp$	1	2	...lstps	FOR EACH Loadstep
$\{f_{n+1}^{ext}\}$				Calculate which elements each particle overlaps, and therefore free degrees of freedom increment and map the external forces $\{f_v^{ext}\} = \{f_p^{ext}\}[S_{vp}]$ (40)
$NRit$	1	2	...	WHILE $ f^R < tol$
$\{u_v\}$				solve for displacements $\{\delta d_{k+1}\} = [K]^{-1}\{f_k^R\}$, (22)
$\{u_p\}$				particle displacement $\{u_p\} = [S_{vp}]\{u_v\}$
i	1	2	... n_{mp}	FOR EACH Material Point
$[\frac{\partial S_{vp}^a}{\partial X}]\{\Delta u_{n+1}\}$				calculate derivatives of displacements by looping through overlapping elements
i	1	2	... n_{mp}	FOR EACH Material Point
				see figure 7
$\{f^R\}$				out of balance force $\{f^R\} = \{f^{int}\} - \{f^{ext}\}$
$\{l_p\}$				update particle domain lengths $l_i^p = l_i^{p0} U_{ii}$ No implied sum (39)
$\{x_p^{n+1}\}$				update particle positions $\{x_p^{n+1}\} = \{x_p^n\} + \{u_p\}$

Figure 6: Implicit GIMP algorithm.

i	1	2	$\dots n_{mp}$	FOR EACH Material Point
$[\Delta F]$		↓		deformation gradient increment $[\Delta F_{n+1}] = [1] + \left[\frac{\partial \Delta u_{n+1}}{\partial \bar{X}} \right]$ (31)
$[F]$				updated deformation gradient $[F_{n+1}] = [\Delta F_{n+1}][F_n]$ (30)
$[b]$				trial elastic left Cauchy Green strain $[b_t^e] = [\Delta F_{n+1}][b_n^e][\Delta F_{n+1}]^T$ (17)
$[\varepsilon_t^e]$				logarithmic elastic trial strain $[\varepsilon_t^e] = \frac{1}{2} \log([b_t^e])$ (15)
$[\tau], [\varepsilon^e]$				Kirchhoff stress and elastic logarithmic strain use small strain constitutive model
$[\sigma]$				Cauchy stress $[\sigma] = J^{-1}[\tau]$ (16)
$[b^e]$				left elastic cauchy green strain $[b_{n+1}^e] = \exp(2[\varepsilon_{n+1}^e])$ (18)
$[a]$				spatial tangent $[a] = \frac{1}{2J} [D^{\text{alg}}][L][B] - [S]$, (25)
n_{els}	1	2	$\dots n_{els}$	FOR EACH overlapping element
$[G]$		↓		strain displacement matrix $[G] = \left[\frac{\partial S_{vp1}^a}{\partial \bar{X}} \right] [\Delta F]^{-1} \dots$ (41)
$\{f_p^{int}\}$	●			particle internal force $\{f_p^{int}\} = [G]^T \{\sigma\} V_p$ (29)
$[K_p]$	●			particle stiffness $[K_p] = [G]^T [a][G] V_p$ (27)

Material model

Figure 7: Implicit GIMP algorithm.

5. Numerical Examples

In this section, three numerical examples are presented to demonstrate the iGIMP method. For each example, the geometrically non-linear updated Lagrangian GIMP method described in Section 3 is used.

5.1. One dimensional compression under self weight

The response of a column to the application of a body force due to increasing gravity, as shown using an explicit GIMP method in [60]. The column has an initial length (L_0) of 50 and is restrained at the bottom with $u(z = 0) = 0$. A total force of $w = 40,000$ is applied by assigning a density of $\rho = 80$ and incrementing gravity (up to $g = 10$). The Young's modulus is $E = 1 \times 10^6$, in compatible units. The analytical solution for the vertical Cauchy stress σ in this 1D problem is now derived. The initial vertical position within the column is Z , therefore

$$\sigma = \rho_0 b(l_0 - Z), \quad (42)$$

where ρ_0 is the initial density of the material and b is the body force. The Cauchy and Kirchhoff stresses are linked through (16). In one dimension, the logarithmic strain is defined as

$$\varepsilon^{(0)} = \frac{1}{2} \ln(F^2) = \ln(F) \quad (43)$$

and we assume that the Kirchhoff stress is linked to the logarithmic strain through

$$\tau = E\varepsilon^{(0)}. \quad (44)$$

By combining the above equations, the Cauchy stress can be expressed as

$$\sigma = \frac{1}{F} E \ln(F). \quad (45)$$

From (42) we can obtain σ for any point in the problem domain and the deformation gradient can be found using a Newton process to solve for F in (45). This analytical solution differs from the incrementally linear solutions given in

[60] and [59] to be consistent with the fact that the method described in this paper is geometrically non-linear within a load step.

For the case shown in Figure 8, the domain is discretised into 50 elements with each element initially containing two material points positioned so that the influence domain of each material point consists of half the element, or $V_p = 0.5$. The stresses at the end of the simulation using the iGIMP method are compared against stresses calculated using the standard MPM using the same discretisation, and the analytical solution given below. The MPM and iGIMP simulations were both run using 20 loadsteps. It can be seen that the MPM simulation experiences an oscillation in its response, deviating significantly from the analytical solution, due to material points crossing element boundaries. In the iGIMP method this problem is alleviated as movements of material points between elements happen more gradually giving a smoother change in stiffness as opposed to a sudden jump. The effect of changing the element size in the

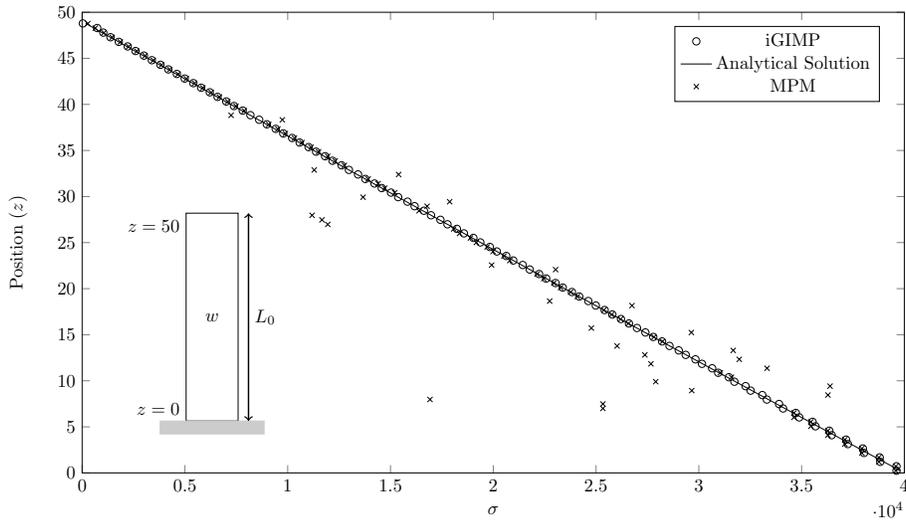


Figure 8: Numerical solutions using iGIMP and MPM of a column under self weight plotted against the analytical solution.

background grid is now demonstrated. Both the number of background grid elements and the number of material points per element are changed and the

error relative to the exact solution plotted both for the problem outlined above with a total load of $w = 40,000$ shown in Figure 9(b), and with a load of $w = 10,000$ as shown in Figure 9(a).

To aid comparison and to study convergence with refinement, a dimensionless error measure is used

$$\text{error} = \sum_p \frac{|\sigma(Z_p) - \sigma_p| v_{p0}}{WL_0}. \quad (46)$$

It can be seen in Figure 9 that varying the number of material points does not

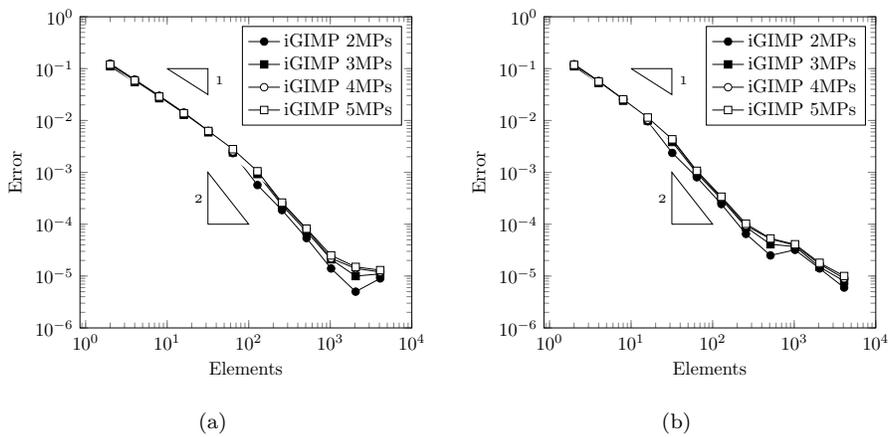


Figure 9: Convergence study at (a) $w = 10,000$ and (b) $w = 40,000$.

have a large influence on the solution to this problem. The convergence rate varies between 1 and 2 for most numbers of elements with a degradation towards higher numbers of elements. It is possible that this can be attributed to the fact that there will be more material points crossing boundaries contributing additional error which cancels out the benefit of additional elements.

The same problem is also modelled with a weight of $w = 400,000$, ten times larger than the initial problem to show the large deformation capabilities of the method. Figure 10 shows the stress against position and the corresponding analytical solution, and Figure 11 shows the convergence of the error with an increasing number of elements. Here, the convergence is also compared against linear and quadratic finite element solutions. It can be seen that for a given

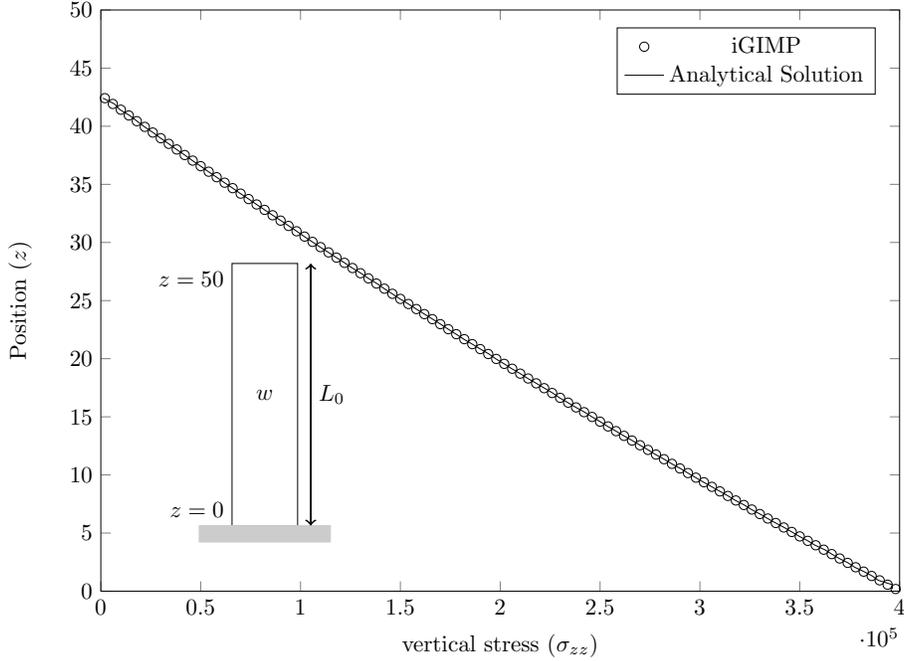


Figure 10: Stress against position for $w = 400,000$.

number of elements, the error for the iGIMP code is less than the linear FEM simulation with 2 Gauss points per element. The convergence rate for the linear FEM simulation is constant at 1 whereas the convergence rate for the iGIMP simulations varies between 1.8 and 0.6. The convergence rate for the quadratic FEM code with 3 Gauss points per element is 2, again as expected. It should be noted that if 2 Gauss points per element are used then the FEM code achieves machine precision for any number of elements. This is because the two sampling points are correctly positioned to approximate the solution exactly. For a linear finite element the same applies to a single Gauss point in the centre of the element. In iGIMP the material points are not located at Gauss quadrature positions so the same does not apply.

From the analytical solution given in Equation (45), the deformation gradient at the base of the column is calculated to be 0.74292 and the displacement at the top of the column to be -7.3347 . Using two material points per element,

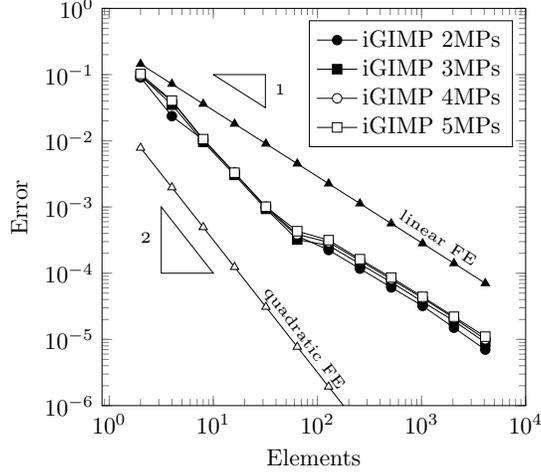


Figure 11: Convergence study for iGIMP shown against linear and quadratic FEM.

and taking the top displacement from the top material point and bottom deformation gradient from the bottom most material point, it can be seen in Table 1 that the displacement is accurate to 5 significant figures for all numbers of grid elements shown, and the error in deformation gradient decreases with increasing elements with a linear rate of convergence, where the deformation gradient error is given as

$$F_{\text{error}} = |F_p - F_{(0)}|/F_{(0)}. \quad (47)$$

The error and rate of convergence are comparable to that of linear finite elements.

The same problem of a column under self weight is now investigated but this time using a Von-Mises constitutive model with a deviatoric yield stress of $\rho_y = 3 \times 10^4$. The yield surface is defined as

$$f = \rho - \rho_y = 0 \quad (48)$$

where

$$\rho = \sqrt{3J_2}, \quad J_2 = \frac{1}{2}(s_i s_i) \quad \text{and} \quad s_i = \tau_i - \frac{1}{3}\sum_{j=1}^3 \tau_j \quad (49)$$

The material will yield when $\tau_{zz} = \rho_y$ which should occur at a position of $Z = l_0 - \frac{\rho_y}{\rho_\sigma b_0}$. Below this point, the material will experience elasto-plastic behaviour

Table 1: Error in Deformation Gradient and Displacement

number of elements	256	512	1024	2048
iGIMP				
top displacement (m)	-7.3347	-7.3347	-7.3347	-7.3347
base deformation gradient	0.74322	0.74307	0.74300	0.74296
displacement error	-	-	-	-
deformation gradient error	4.091×10^{-4}	2.041×10^{-4}	1.017×10^{-4}	5.040×10^{-5}
Linear FEM				
top displacement (m)	-7.3347	-7.3347	-7.3347	-7.3347
base deformation gradient	0.74325	0.74308	0.74300	0.74296
displacement error	-	-	-	-
deformation gradient error	4.477×10^{-4}	2.237×10^{-4}	1.118×10^{-4}	5.593×10^{-5}

and despite zero deformation being enforced in the out of plane directions, stresses τ_{xx} and τ_{yy} will be introduced. Due to fact that the boundary conditions are the same these two stresses will be equal, because of this from here on variables in the y direction will not be discussed. Using this it is possible to write the deviatoric stress in this situation as

$$\rho = |\tau_{xx} - \tau_{zz}|. \quad (50)$$

Here the stress in the vertical direction should remain the same, following the analytical solution (42) however in the section near the base of the column where this stress is reached, stresses will appear in the out of plane direction. This is shown in Figure 12 where it can be seen that in the elasto-plastic region there are stresses in both the vertical and horizontal directions. The analytical solution for these out of plane stresses is not immediately obvious, using knowledge of the deformation due to the boundary conditions a relationship can be found between elastic parts of the deformation gradient

$$F_{xx}^e = F_{zz}^e e^{\frac{\rho y}{E}}. \quad (51)$$

It can further be shown that in the vertical direction a relationship between plastic and elastic components of the deformation gradient exists as

$$F_{zz}^p = F_{zz}^e 2 e^{\frac{2\rho_y}{E}}. \quad (52)$$

Using this the Cauchy stress in the vertical direction can be written as

$$\sigma_{zz} = \frac{1}{F_{zz}^e 3 e^{\frac{2\rho_y}{E}}} E \ln(F_{zz}^e). \quad (53)$$

This result allows us to calculate F_{zz}^e using a Newton process which yields F_{zz} using (52). F_{xx}^e can also be calculated using (51) which allows the calculation of

$$\sigma_{xx} = \frac{1}{F_{zz}} E \ln(F_{xx}^e). \quad (54)$$

The derivation for this can be found in [Appendix A](#).

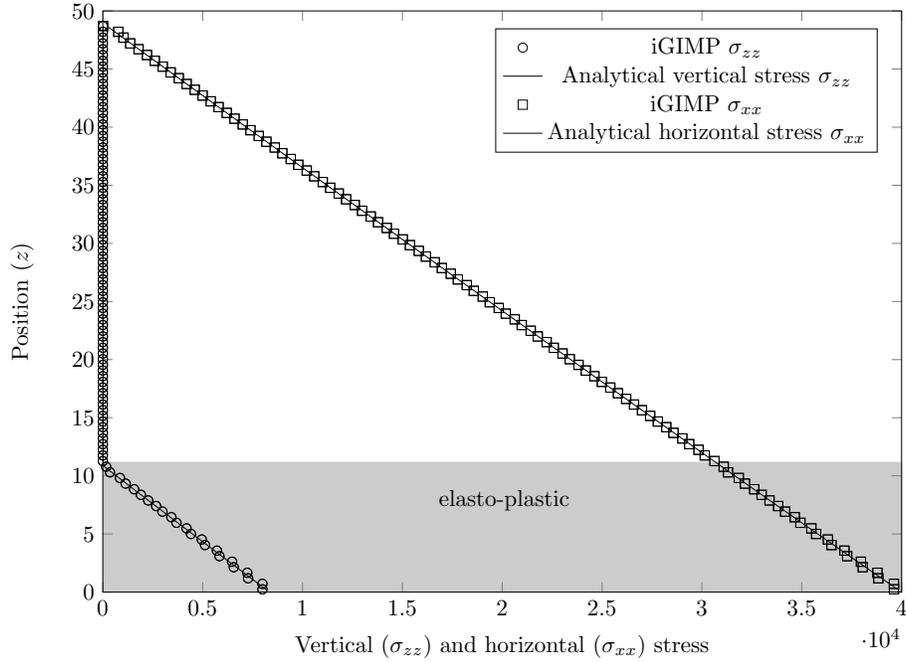


Figure 12: Vertical (σ_{zz}) and horizontal ($\sigma_{xx} = \sigma_{yy}$) stress using iGIMP plotted against analytical solution for a load of 40000 and yield stress of $\rho_y = 3 \times 10^4$.

The problem was repeated with a body force increased by a factor of 10. The solutions are shown in [Figure 13](#) to also show close agreement with the

analytical solution. Figure 14 shows the convergence for these simulations when

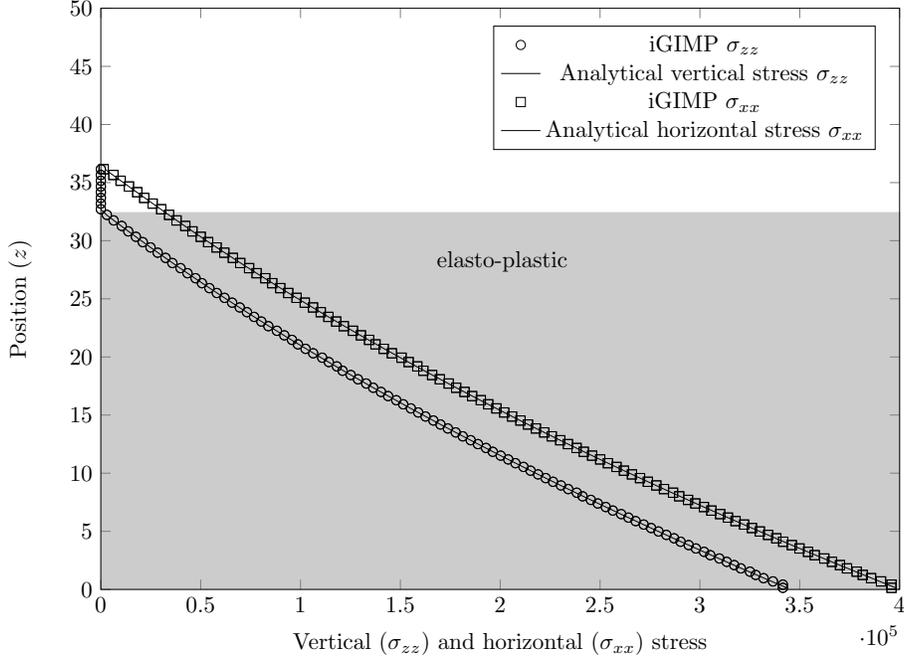


Figure 13: Vertical (σ_{zz}) and horizontal ($\sigma_{xx} = \sigma_{yy}$) stress using iGIMP plotted against analytical solution for a load of 400000 and yield stress of $\rho_y = 3 \times 10^4$.

refining the mesh which exhibits the same behaviour as for the elastic case. The convergence plot for quadratic finite elements is not shown in this case as it experienced locking with a 3x3 quadrature scheme and reached machine precision in one step using reduced integration.

5.2. 2D compaction under self weight

The second problem presented is the behaviour of a material compacting under self weight. The response of the material with increasing gravity is modelled. The problem domain at the beginning of the simulation has a height of 8 units and a width of 8 units, Young's modulus of $E = 1 \times 10^5$ and Poisson's ratio of $\nu = 0.3$, as shown in Figure 15. Vertical movement along the bottom edge is prevented and due to symmetry, only half the problem is modelled. Self weight

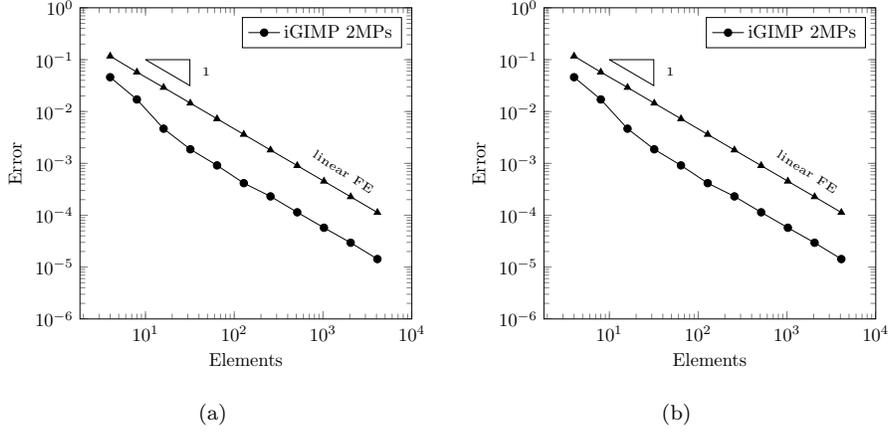


Figure 14: Convergence study at (a) $w = 40,000$ and (b) $w = 400,000$ for the plastic case.

Table 2: Newton Raphson residuals showing near asymptotically quadratic convergence.

Step	16	17	18	19	20
Iteration 1	1.357×10^{-3}	1.246×10^{-3}	1.147×10^{-3}	1.094×10^{-3}	9.897×10^{-4}
Iteration 2	4.805×10^{-7}	4.202×10^{-7}	3.658×10^{-7}	3.478×10^{-7}	2.872×10^{-7}
Iteration 3	6.903×10^{-14}	5.399×10^{-14}	4.283×10^{-14}	1.839×10^{-12}	2.809×10^{-14}

is applied incrementally over 20 loadsteps with a total weight of $w = 4 \times 10^5$. A 10 by 10 background grid is used to allow for material movement during the simulation. The initial position of the material is modelled using 4 material points per element (shown in a single shade of grey) along with the final (non-exaggerated) displaced shape of the material. The shading of the material point domains corresponds to the vertical stress at each material point at the end of the simulation.

Figure 16 shows that the convergence within a load step is near asymptotically quadratic. This convergence rate of the NR process indicates correct implementation of the method. The values shown in Table 2 correspond to the norm of the out-of-balance force ($\{f^{ob}\}$) at the end of each NR iteration for the final 5 loadsteps. This is calculated as outlined in Section 4.

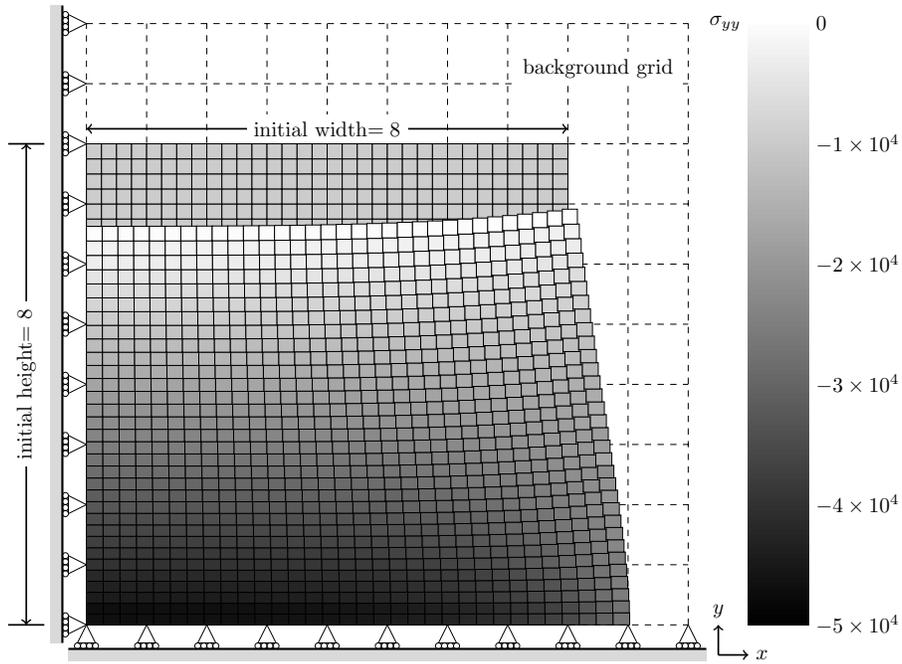


Figure 15: Compaction under self weight problem showing initial (grey) and final (coloured) material point positions and influence domains with vertical stress shown.

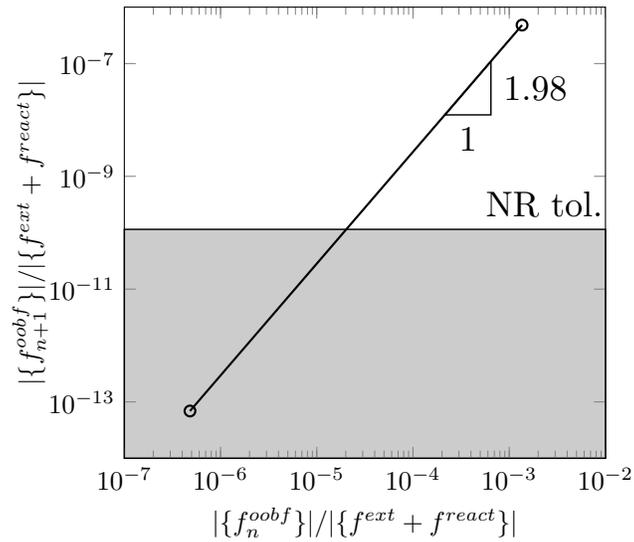


Figure 16: Out-of-balance force at the end of each loadsteps showing near asymptotically quadratic convergence. Displayed graphically for load step 16.

In Figure 17, the maximum horizontal displacement of the material is compared against results obtained from an updated lagrangian Finite Element analysis of the same problem. The material properties are as described above. The FEM analysis used linear finite elements with 2 x 2 Gauss quadrature with 1,000,000 elements. By changing the number of grid elements, it can be seen that the displacement converges towards a constant value. Displacements have been normalised to the displacement given by the finest FEM mesh. When calculating the displacement using the iGIMP method, the value used is the displacement of the bottom right material point plus half of any extension to its influence domain. The convergence is investigated using different numbers of material points per element. With an element size of 1 and smaller, the solutions using different numbers of material points per element all come within 1% of the converged solution. This suggests that the number of elements in the background mesh has more influence on the accuracy of the solution than the number of material points per element; this can be seen clearly in Figure 17 by the fact that for finer meshes the displacements for varying numbers of material points are all very similar.

The same problem is analysed with a Von-Mises constitutive model with a deviatoric yield stress of $\rho_y = 1.2 \times 10^4$; the yield function being defined in the same way as in (48). This leads to significantly larger displacements as can be seen in Figure 18. The convergence for the final 5 steps is given in Table 3 where it can be seen that more steps are needed for the NR algorithm to find the correct path now that the material behaviour is elasto-plastic but then reaches near asymptotic quadratic convergence as before, until running into the precision of the machine for lower errors.

5.3. 2D Cantilever Beam

The final example is an elastic cantilever beam with a point load of 100kN applied at the vertical mid point on its free end. To achieve this loading using iGIMP, this load is split between the two end most material points above and below the neutral axis, as highlighted in Figure 19. The beam has a length

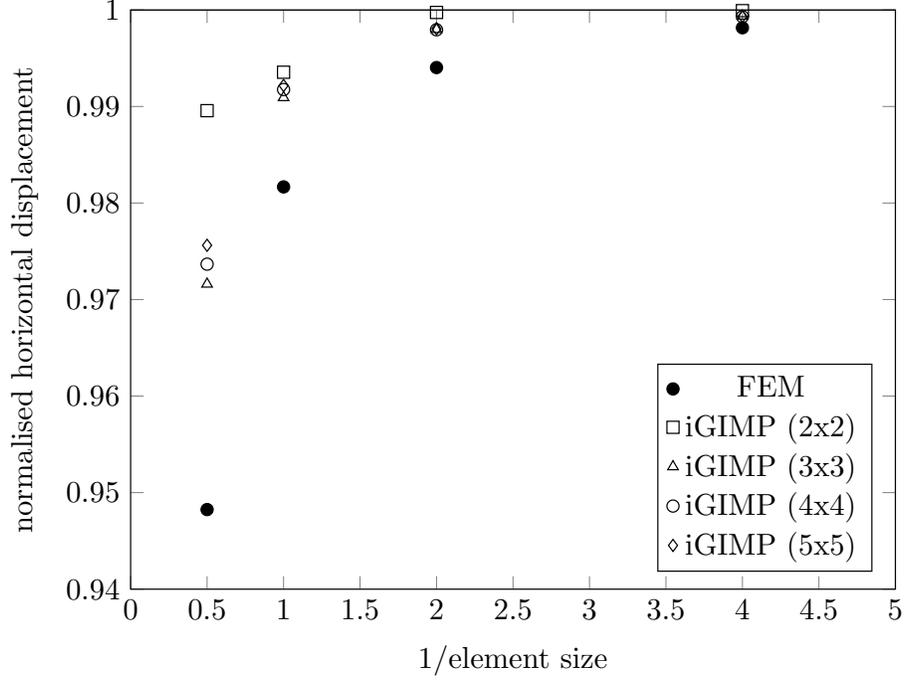


Figure 17: Physical convergence of displacement with changing mesh density.

Table 3: Newton Raphson Residuals reaching near asymptotically quadratic convergence for elasto-plastic case.

Step	16	17	18	19	20
Iteration 1	2.987×10^{-2}	2.787×10^{-2}	2.600×10^{-2}	2.467×10^{-2}	2.372×10^{-2}
Iteration 2	4.631×10^{-2}	4.703×10^{-2}	3.072×10^{-2}	2.504×10^{-2}	2.013×10^{-2}
Iteration 3	9.236×10^{-3}	2.483×10^{-3}	8.893×10^{-4}	6.861×10^{-4}	3.863×10^{-4}
Iteration 4	2.379×10^{-4}	1.530×10^{-5}	3.494×10^{-6}	1.633×10^{-6}	1.237×10^{-6}
Iteration 5	8.852×10^{-6}	1.562×10^{-10}	4.098×10^{-11}	2.642×10^{-12}	3.505×10^{-12}
Iteration 6	2.028×10^{-10}	8.386×10^{-16}			
Iteration 7	7.290×10^{-16}				

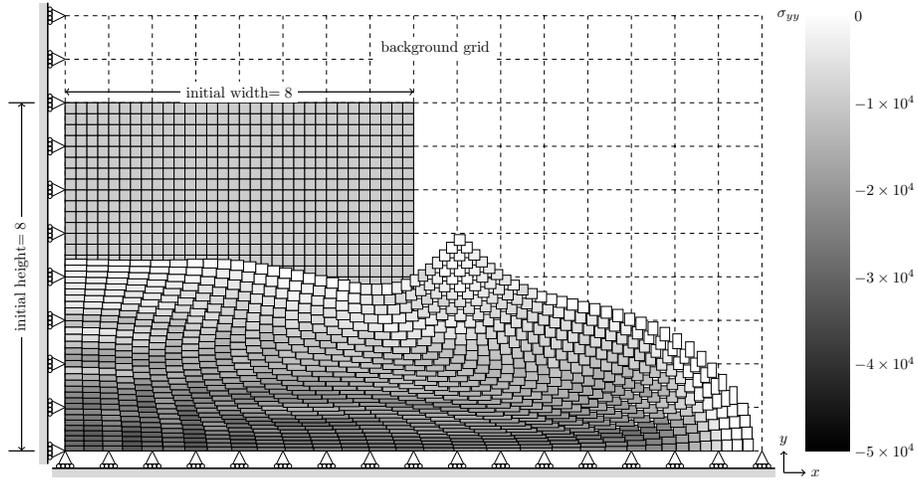


Figure 18: Compaction under self weight with a yield stress introducing elasto-plastic deformation.

of 10m and depth of 1m, and the material has a Young's modulus of 12MPa and Poisson's ratio of 0.2. The load is applied over 50 loadsteps with the problem initially split into 40 elements each containing 3×3 material points. The beam is fixed at the left hand end in both directions at the neutral axis with roller boundary conditions applied to other nodes along the boundary. Figure 19 shows the original and final (unexaggerated) configurations. Here, it is important to note that the material point influence domains are updated using the stretch rather than the full deformation gradient. The reasons for this are as discussed in Section 3 and it can be seen in Figure 20(a) how the analysis collapses when using the deformation gradient for these updates (highlighted by the circled region on the right hand figure). Due to the material point rotations (a rotation of 90 degrees would cause the leading diagonal of the deformation gradient to go to zero) the size of the material points gets very small leading to a non-physically small stiffness in those elements.

In Figure 21, the normalised horizontal and vertical displacement at the loading point are plotted against the analytical solution and results from a finite element analysis. For the iGIMP solution, this is the average between the

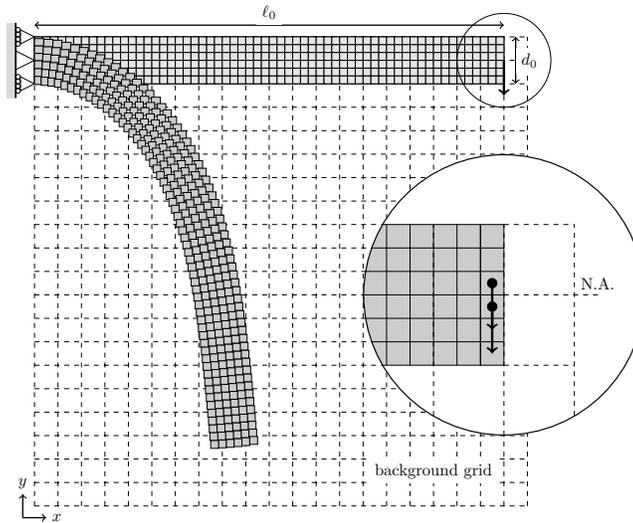
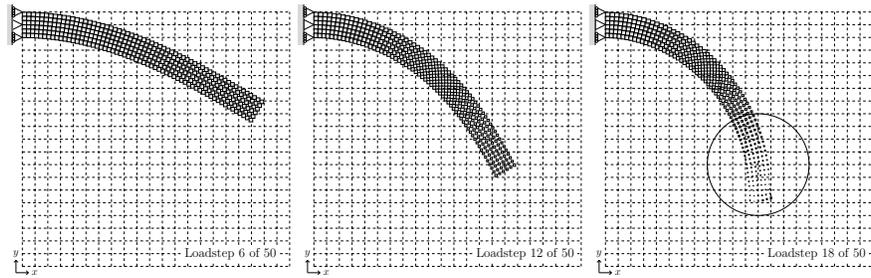


Figure 19: Figure showing initial and final material point positions and domains as well as boundary conditions. There are 3x3 material points in each element. It is also shown how the load is approximated by splitting between the two endmost material points above and below the neutral axis (N.A).

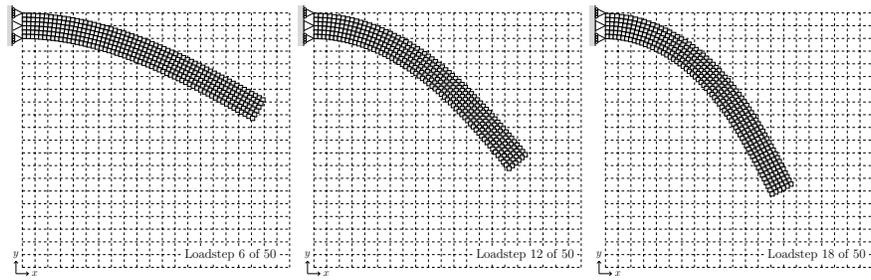
two loading points above and below the neutral axis. The analytical solution is provided in [103]. The FEM analysis uses 8 noded quadratic elements with 3 x 3 Gauss quadrature with the same 20 by 2 element discretisation as initially used in the iGIMP analysis. The load is applied at the neutral axis and the boundary conditions applied at the left hand end of the beam the same as the iGIMP analysis. Good agreement can be seen between the iGIMP displacements and both the FEM and the analytical solutions.

6. Conclusions

In this paper a fully implicit formulation of an elasto-plastic GIMP method is presented for the first time. The construction of the weighting functions and implementation of the method have been explained in detail. The element stiffnesses are shown to be calculated based on contributions from overlapping parts of material point influence domains allowing the global stiffness matrix to



(a)



(b)

Figure 20: Material point domains during deformation updated using the deformation gradient $[F]$ (a) and using the right symmetric stretch matrix $[U]$ (b). It is highlighted in (a), load step 18 where the material point domains become problematic, the simulation failed on the following step.

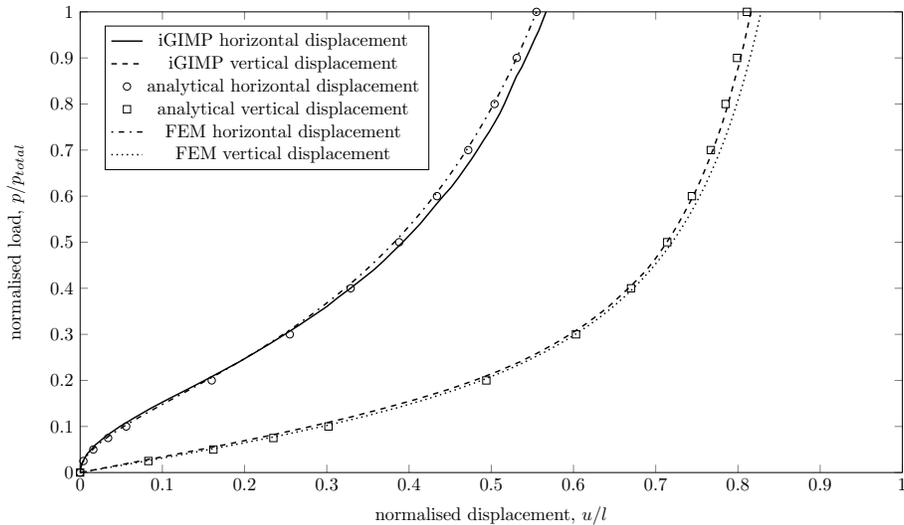


Figure 21: Graph showing normalised horizontal and vertical displacements at the midpoint of the free end of the cantilever beam as load is incrementally applied. The iGIMP results are shown alongside FEM results and the analytical solution.

be assembled from the element stiffness matrices as in the FEM. In this paper, a new way of computing the deformation gradient and updated derivatives, which are consistent with the updated Lagrangian approach has been developed. Using the implicit GIMP method it has been shown that for both small and large deformation elasto-plastic problems, accurate results can be achieved. It has been shown that by using these consistent values of deformation gradient and derivatives of shape functions when forming the consistent global stiffness matrix it is possible to maintain the correct convergence rate of the global equilibrium equations, and that within a load step the Newton Raphson process converges asymptotically quadratically as expected indicating a correct implementation. By increasing the number of background elements, the iGIMP method shows convergence properties between that of linear and quadratic FEM. Additionally, the novel use of updating material point influence domain lengths using the stretch tensor has been shown in Figure 20 to allow the iGIMP method to model problems including rotation of material which previously has been problematic

for the standard GIMP method when updating material point influence domains using the deformation gradient.

Acknowledgements

The first author is supported by a UK EPSRC PhD Studentship.

References

- [1] T. Rabczuk, P. Areias, T. Belytschko, A simplified mesh-free method for shear bands with cohesive surfaces, *International Journal for Numerical Methods in Engineering* 69 (5) (2007) 993–1021.
- [2] P. Areias, M. Msekh, T. Rabczuk, Damage and fracture algorithm using the screened poisson equation and local remeshing, *Engineering Fracture Mechanics* 158 (2016) 116–143.
- [3] T. Rabczuk, E. Samaniego, Discontinuous modelling of shear bands using adaptive meshfree methods, *Computer Methods in Applied Mechanics and Engineering* 197 (6) (2008) 641–658.
- [4] Y. Higo, F. Oka, S. Kimoto, Y. Morinaka, Y. Goto, Z. Chen, A coupled MPM-FDM analysis method for multi-phase elasto-plastic soils, *Soils and Foundations* 50 (4) (2010) 515–532.
- [5] A. Gilmanov, S. Acharya, A hybrid immersed boundary and material point method for simulating 3d fluid–structure interaction problems, *International Journal for Numerical Methods in Fluids* 56 (12) (2008) 2151–2177.
- [6] W. Hu, Z. Chen, A multi-mesh MPM for simulating the meshing process of spur gears, *Computers & structures* 81 (20) (2003) 1991–2002.
- [7] J. Burghardt, R. Brannon, J. Guilkey, A nonlocal plasticity formulation for the material point method, *Computer Methods in Applied Mechanics and Engineering* 225 (2012) 55 – 64.

- [8] Z. Chen, S. Jiang, Y. Gan, H. Liu, T. D. Sewell, A particle-based multi-scale simulation procedure within the material point method framework, *Computational Particle Mechanics* 1 (2) (2014) 147–158.
- [9] L. Shen, Z. Chen, A silent boundary scheme with the material point method for dynamic analyses, *Computer Modeling in Engineering and Sciences* 7 (2005) 305–320.
- [10] Y. Lian, X. Zhang, Y. Liu, An adaptive finite element material point method and its application in extreme deformation problems, *Computer Methods in Applied Mechanics and Engineering* 241 (2012) 275–285.
- [11] Z. Chen, R. Brannon, An evaluation of the material point method, Tech. rep., Sandia National Laboratories (2002).
- [12] X. Zhang, K. Sze, S. Ma, An explicit material point finite element method for hyper-velocity impact, *International Journal for Numerical Methods in Engineering* 66 (4) (2006) 689–706.
- [13] J. E. Guilkey, S. Bardenhagen, K. Roessig, J. Brackbill, W. Witzel, J. Foster, An improved contact algorithm for the material point method and application to stress propagation in granular material, *Computer Modeling in Engineering & Sciences* 2 (4) (2001) 509–522.
- [14] D. Sulsky, H. L. Schreyer, Axisymmetric form of the material point method with applications to upsetting and Taylor impact problems, *Computer Methods in Applied Mechanics and Engineering* 139 (1996) 409 – 429.
- [15] L. Xu, H. Schreyer, D. Sulsky, Blast-induced rock fracture near a tunnel, *International Journal for Numerical and Analytical Methods in Geomechanics* 39 (1) (2015) 23–50.
- [16] Y. Guo, J. Nairn, Calculation of j -integral and stress intensity factors using the material point method, *Computer Modeling in Engineering & Sciences* 6 (3) (2004) 295–308.

- [17] S. Ma, X. Zhang, X. Qiu, Comparison study of MPM and SPH in modeling hypervelocity impact problems, *International Journal of Impact Engineering* 36 (2) (2009) 272 – 282.
- [18] P. Huang, X. Zhang, S. Ma, X. Huang, Contact algorithms for the material point method in impact and penetration simulation, *International journal for numerical methods in engineering* 85 (4) (2011) 498–517.
- [19] Y. Lian, X. Zhang, Y. Liu, Coupling of finite element method with material point method by local multi-mesh contact method, *Computer Methods in Applied Mechanics and Engineering* 200 (47) (2011) 3482–3494.
- [20] C. Coetzee, A. Basson, P. Vermeer, Discrete and continuum modelling of excavator bucket filling, *Journal of Terramechanics* 44 (2) (2007) 177 – 186.
- [21] C. Coetzee, Discrete and continuum modelling of soil cutting, *Computational Particle Mechanics* 1 (4) (2014) 409–423.
- [22] X. Ma, P. T. Giguere, B. Jayaraman, D. Z. Zhang, Distribution coefficient algorithm for small mass nodes in material point method, *Journal of Computational Physics* 229 (20) (2010) 7819 – 7833.
- [23] W. Shin, G. R. Miller, P. Arduino, P. Mackenzie-Helnwein, Dynamic meshing for material point method computations, *World Academy of Science, Engineering and Technology* 4 (2010) 64–72.
- [24] S. Bardenhagen, Energy conservation error in the material point method for solid mechanics, *Journal of Computational Physics* 180 (1) (2002) 383 – 403.
- [25] J. A. Nairn, Material point method calculations with explicit cracks, *Computer Modeling In Engineering & Sciences* 4 (6) (2003) 649–663.
- [26] H. Zhang, K. Wang, Z. Chen, 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 (2009) 1456 – 1472.

- [27] L. Xue, O. Borodin, G. D. Smith, J. Nairn, Micromechanics simulations of the viscoelastic properties of highly filled composites by the material point method (mpm), *Modelling and Simulation in Materials Science and Engineering* 14 (4) (2006) 703.
- [28] W. Hu, Z. Chen, Model-based simulation of the synergistic effects of blast and fragmentation on a concrete wall using the MPM, *International Journal of Impact Engineering* 32 (12) (2006) 2066–2096.
- [29] H. Schreyer, D. Sulsky, S.-J. Zhou, Modeling delamination as a strong discontinuity with the material point method, *Computer Methods in Applied Mechanics and Engineering* 191 (2324) (2002) 2483 – 2507.
- [30] P. Mackenzie-Helnwein, P. Arduino, W. Shin, J. A. Moore, G. R. Miller, Modeling strategies for multiphase drag interactions using the material point method, *International Journal for Numerical Methods in Engineering* 83 (3) (2010) 295–322.
- [31] F. Li, J. Pan, C. Sinka, Modelling adhesive contact between fine particles using material point method, *Mechanics of Materials* 43 (3) (2011) 157–167.
- [32] F. Li, J. Pan, C. Sinka, Modelling brittle impact failure of disc particles using material point method, *International Journal of Impact Engineering* 38 (7) (2011) 653–660.
- [33] F. Gilibert, V. Cantavella, E. Sánchez, G. Mallol, Modelling fracture process in ceramic materials using the material point method, *EPL (Europhysics Letters)* 96 (2) (2011) 24002.
- [34] Y. Lian, X. Zhang, X. Zhou, S. Ma, Y. Zhao, Numerical simulation of explosively driven metal by material point method, *International Journal of Impact Engineering* 38 (4) (2011) 238–246.

- [35] Y. Wang, H. Beom, M. Sun, S. Lin, Numerical simulation of explosive welding using the material point method, *International Journal of Impact Engineering* 38 (1) (2011) 51 – 60.
- [36] F. Zabala, E. Alonso, Progressive failure of Aznalcóllar dam using the material point method, *Géotechnique* 61 (9) (2011) 795–808.
- [37] S. Zhou, J. Stormont, Z. Chen, Simulation of geomembrane response to settlement in landfills by using the material point method, *International Journal for Numerical and Analytical Methods in Geomechanics* 23 (15) (1999) 1977–1994.
- [38] J. Li, Y. Hamamoto, Y. Liu, X. Zhang, Sloshing impact simulation with material point method and its experimental validations, *Computers & Fluids* 103 (1) (2014) 86 – 99.
- [39] A. R. York, D. Sulsky, H. L. Schreyer, The material point method for simulation of thin membranes, *International Journal for Numerical Methods in Engineering* 44 (10) (1999) 1429–1456.
- [40] Z. Wieckowski, The material point method in large strain engineering problems, *Computer Methods in Applied Mechanics and Engineering* 193 (2004) 4417 – 4438.
- [41] S. Bardenhagen, J. Brackbill, D. Sulsky, The material-point method for granular materials, *Computer Methods in Applied Mechanics and Engineering* 187 (3) (2000) 529–541.
- [42] C. Coetzee, P. Vermeer, A. Basson, The modelling of anchors using the material point method, *International Journal for Numerical and Analytical Methods in Geomechanics* 29 (9) (2005) 879–895.
- [43] Y. Guo, J. Nairn, Three-dimensional dynamic fracture analysis using the material point method, *Computer Modeling in Engineering and Sciences* 16 (3) (2006) 141–155.

- [44] I. Jassim, D. Stolle, P. Vermeer, Two-phase dynamic analysis by material point method, *International Journal for Numerical and Analytical Methods in Geomechanics* 37 (15) (2013) 2502–2522.
- [45] D. Sulsky, H. Schreyer, K. Peterson, R. Kwok, M. Coon, Using the material-point method to model sea ice dynamics, *Journal of Geophysical Research: Oceans* 112 (2007) 1–18.
- [46] S. Bandara, K. Soga, Coupling of soil deformation and pore fluid flow using material point method, *Computers and Geotechnics* 63 (1) (2015) 199–214.
- [47] J. E. Guilkey, J. B. Hoying, J. A. Weiss, Computational modeling of multicellular constructs with the material point method, *Journal of Biomechanics* 39 (11) (2006) 2074–2086.
- [48] L. Moresi, F. Dufour, H.-B. Mühlhaus, A Lagrangian integration point finite element method for large deformation modeling of viscoelastic geomaterials, *Journal of Computational Physics* 184 (2) (2003) 476 – 497.
- [49] J. E. Guilkey, J. A. Weiss, 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) (2003) 1323–1338.
- [50] S. Cummins, J. Brackbill, An implicit particle-in-cell method for granular materials, *Journal of Computational Physics* 180 (2) (2002) 506 – 548.
- [51] D. Sulsky, A. Kaul, Implicit dynamics in the material-point method, *Computer Methods in Applied Mechanics and Engineering* 193 (2004) 1137 – 1170.
- [52] E. Love, D. Sulsky, An unconditionally stable, energy-momentum consistent implementation of the material-point method, *Computer Methods in Applied Mechanics and Engineering* 195 (33) (2006) 3903 – 3925.

- [53] E. Love, D. L. Sulsky, An energy-consistent material-point method for dynamic finite deformation plasticity, *International Journal for Numerical Methods in Engineering* 65 (10) (2006) 1608–1638.
- [54] L. Beuth, T. Benz, P. A. Vermeer, Z. Wieckowski, Large deformation analysis using a quasi-static material point method, *Journal of Theoretical and Applied Mechanics* 38 (1-2) (2008) 45–60.
- [55] B. Wang, P. J. Vardon, M. A. Hicks, Z. Chen, Development of an implicit material point method for geotechnical applications, *Computers and Geotechnics* 71 (2016) 159–167.
- [56] M. Steffen, R. M. Kirby, M. Berzins, Decoupling and balancing of space and time errors in the material point method, *International Journal for Numerical Methods in Engineering* 82 (10) (2010) 1207–1243.
- [57] A. Sadeghirad, R. M. Brannon, J. Burghardt, 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* 86 (12) (2011) 1435–1456.
- [58] A. Sadeghirad, R. M. Brannon, J. Guilkey, Second-order convected particle domain interpolation (cpdi2) with enrichment for weak discontinuities at material interfaces, *International Journal for Numerical Methods in Engineering* 95 (11) (2013) 928–952.
- [59] D. Z. Zhang, X. Ma, P. T. Giguere, Material point method enhanced by modified gradient of shape function, *Journal of Computational Physics* 230 (16) (2011) 6379 – 6398.
- [60] S. Bardenhagen, E. Kober, The generalized interpolation material point method, *Computer Modeling in Engineering and Sciences* 5 (6) (2004) 477–496.

- [61] O. Buzzi, D. Pedroso, A. Giacomini, Caveats on the implementation of the generalized material point method, *Computer Modeling in Engineering and Sciences* 1 (1) (2008) 1–21.
- [62] P. Wallstedt, J. Guilkey, An evaluation of explicit time integration schemes for use with the generalized interpolation material point method, *Journal of Computational Physics* 227 (22) (2008) 9628–9642.
- [63] J. Ma, D. Wang, M. Randolph, A new contact algorithm in the material point method for geotechnical simulations, *International Journal for Numerical and Analytical Methods in Geomechanics* 38 (11) (2014) 1197–1210.
- [64] R. Ambati, X. Pan, H. Yuan, X. Zhang, Application of material point methods for cutting process simulations, *Computational Materials Science* 57 (2012) 102–110.
- [65] P. Wallstedt, J. Guilkey, Improved velocity projection for the material point method, *CMES: Computer Modeling in Engineering & Sciences* 19 (3) (2007) 223–232.
- [66] K. Abe, K. Soga, S. Bandara, Material point method for coupled hydromechanical problems, *Journal of Geotechnical and Geoenvironmental Engineering* 140 (3) (2013) 04013033.
- [67] S. Andersen, L. Andersen, Material-point-method analysis of collapsing slopes, in: *Proceedings of the 1 st International Symposium on Computational Geomechanics (COMGEO I)*, 2009, pp. 817–828.
- [68] S. Andersen, L. Andersen, Modelling of landslides with the material-point method, *Computational Geosciences* 14 (1) (2010) 137–147.
- [69] N. P. Daphalapurkar, H. Lu, D. Coker, R. Komanduri, Simulation of dynamic crack growth using the generalized interpolation material point (GIMP) method, *International Journal of Fracture* 143 (1) (2007) 79–102.

- [70] S. G. Bardenhagen, J. A. Nairn, H. Lu, Simulation of dynamic fracture with the material point method using a mixed j-integral and cohesive law approach, *International Journal of Fracture* 170 (1) (2011) 49–66.
- [71] L. Tran, J. Kim, M. Berzins, Solving time-dependent PDEs using the material point method, a case study from gas dynamics, *International journal for numerical methods in fluids* 62 (7) (2010) 709–732.
- [72] W. Sołowski, S. Sloan, Evaluation of material point method for use in geotechnics, *International Journal for Numerical and Analytical Methods in Geomechanics* 39 (7) (2015) 685–701.
- [73] A. Nair, S. Roy, Implicit time integration in the generalized interpolation material point method for finite deformation hyperelasticity, *Mechanics of Advanced Materials and Structures* 19 (6) (2012) 465–473.
- [74] B. Wang, P. J. Vardon, M. A. Hicks, Z. Chen, Development of an implicit material point method for geotechnical applications, *Computers and Geotechnics* 71 (2016) 159–167.
- [75] D. Sulsky, Z. Chen, H. L. Schreyer, A particle method for history-dependent materials, *Computer Methods in Applied Mechanics and Engineering* 118 (1) (1994) 179–196.
- [76] D. Sulsky, S.-J. Zhou, H. L. Schreyer, Application of a particle-in-cell method to solid mechanics, *Computer Physics Communications* 87 (1) (1995) 236–252.
- [77] J. Brackbill, H. Ruppel, Flip: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions, *Journal of Computational Physics* 65 (2) (1986) 314 – 343.
- [78] J. U. Brackbill, D. B. Kothe, H. M. Ruppel, Flip: A low-dissipation, particle-in-cell method for fluid flow, *Computer Physics Communications* 48 (1) (1988) 25–38.

- [79] M. W. Evans, F. H. Harlow, E. Bromberg, The particle-in-cell method for hydrodynamic calculations, Tech. rep., DTIC Document (1957).
- [80] W. Sołowski, S. Sloan, Modelling of sand column collapse with material point method, in: Proceedings of the 3rd international symposium on computational geomechanics (ComGeo III), 2013, pp. 698–705.
- [81] C. M. Mast, P. Arduino, P. Mackenzie-Helnwein, G. R. Miller, Simulating granular column collapse using the material point method, *Acta Geotechnica* 10 (1) (2015) 101–116.
- [82] H. Tan, J. A. Nairn, Hierarchical, adaptive, material point method for dynamic energy release rate calculations, *Computer Methods in Applied Mechanics and Engineering* 191 (19) (2002) 2123–2137.
- [83] P. Vermeer, L. Beuth, T. Benz, A quasi-static method for large deformation problems in geomechanics, in: Proceedings of the 12th international conference of international association for computer methods and advances in geomechanics (IACMAG), Goa, India, Citeseer, 2008, pp. 55–63.
- [84] A. Yerro, E. Alonso, N. M. Pinyol, The material point method for unsaturated soils, *Géotechnique* 65 (3) (2015) 201–217.
- [85] E. Alonso, A. Yerro, Trends in large-deformation analysis of landslide mass movements with particular emphasis on the material point method, *Géotechnique* 66 (3) (2016) 248–273.
- [86] W. Sołowski, S. Sloan, D. Wang, Material point method simulation of triaxial shear tests, in: *Computer Methods and Recent Advances in Geomechanics*, CRC Press, 2014, pp. 169–174.
- [87] M. Steffen, R. M. Kirby, M. Berzins, Analysis and reduction of quadrature errors in the material point method, *International Journal for Numerical Methods in Engineering* 76 (6) (2008) 922–948.

- [88] S. Li, W. K. Liu, Meshfree particle methods, Springer Science & Business Media, 2007.
- [89] D.-N. Kim, F. Montáns, K. Bathe, Insight into a model for large strain anisotropic elasto-plasticity, *Computational Mechanics* 44 (5) (2009) 651–668.
- [90] J. Simo, Algorithms for static and dynamic multiplicative plasticity that preserve the classical return mapping schemes of the infinitesimal theory, *Computer Methods in Applied Mechanics and Engineering* 99 (1) (1992) 61–112.
- [91] P. . D. de Souza Neto, E.A., D. Owen, A computational model for ductile damage at finite strains., in: O. E. Owen, D.R.J., E. Hinton (Eds.), *Computational Plasticity: Fundamentals and Applications Proceedings of the Third International Conference held in Barcelona, Swansea: Pineridge Press., 1992*, pp. 1425–1441.
- [92] A. Cuitino, M. Ortiz, A material-independent method for extending stress update algorithms from small-strain plasticity to finite plasticity with multiplicative kinematics, *Engineering computations* 9 (4) (1992) 437–451.
- [93] J. Simo, C. Miehe, Associative coupled thermoplasticity at finite strains: Formulation, numerical analysis and implementation, *Computer Methods in Applied Mechanics and Engineering* 98 (1) (1992) 41–104.
- [94] G. Weber, L. Anand, Finite deformation constitutive equations and a time integration procedure for isotropic, hyperelastic-viscoplastic solids, *Computer Methods in Applied Mechanics and Engineering* 79 (2) (1990) 173–202.
- [95] A. L. Eterovic, K.-J. Bathe, A hyperelastic-based large strain elasto-plastic constitutive formulation with combined isotropic-kinematic hardening using the logarithmic stress and strain measures, *International Journal for Numerical Methods in Engineering* 30 (6) (1990) 1099–1114.

- [96] M. Á. Caminero, F. J. Montáns, K.-J. Bathe, Modeling large strain anisotropic elasto-plasticity with logarithmic strain and stress measures, *Computers & Structures* 89 (11) (2011) 826–843.
- [97] E. Lee, D. Lu, Finite-strain elastic-plastic theory with application to plane-wave analysis, *Journal of Applied Physics* 38 (1967) 19–27.
- [98] E. Lee, Elastic-plastic deformation at finite strains, *Journal of Applied Mechanics* 36 (1969) 1–6.
- [99] E. de Souza Neto, D. Perić, D. Owen, *Computational methods for plasticity: Theory and applications*, John Wiley & Sons Ltd, 2008.
- [100] W. Coombs, Finite deformation of particulate geomaterials: frictional and anisotropic critical state elasto-plasticity, Ph.D. thesis, Durham University (2011).
- [101] J. Simo, R. Taylor, Consistent tangent operators for rate-independent elastoplasticity, *Computer Methods in Applied Mechanics and Engineering* 48 (1985) 101–118.
- [102] C. Miehe, Comparison of two algorithms for the computation of fourth-order isotropic tensor functions, *Computers & Structures* 66 (1998) 37–43.
- [103] T. K. Molstad, Finite deformation analysis using the finite element method, Ph.D. thesis, University of British Columbia (1977).

Appendix A. Analytical solution to column under self weight with plasticity

The response of a column to the application of a body force due to increasing gravity is modelled. The column has an initial length (L_0) and is restrained at the bottom with $u(z = 0) = 0$. Displacement is only permitted in a vertical direction. A Young’s modulus of E and a density of ρ_0 in compatible units are

assigned to give a total force, once gravity (g) is applied, of w . This time a deviatoric yield stress of ρ_y is introduced. The yield surface is defined as

$$f = \rho - \rho_y = 0 \quad (\text{A.1})$$

as outlined in section 5. The material will yield when $\tau_{zz} = \rho_y$ which should occur at a position of $Z = l_0 - \frac{\rho_y}{\rho_0 b_0}$. Below this point, the material will experience elasto-plastic behaviour and despite zero deformation being enforced in the out of plane directions, stresses τ_{xx} and τ_{yy} will be introduced. Because the boundary conditions are the same these two stresses will be equal, because of this from here on only variables in the x direction will be discussed. Using this it is possible to write the deviatoric stress as $\rho = |\tau_{xx} - \tau_{zz}|$.

The Cauchy stress in the vertical direction is the same as that for the elastic case can be determined from the initial vertical position within the column, Z , through

$$\sigma_{zz} = \rho_0 b(l_0 - Z), \quad (\text{A.2})$$

where ρ_0 is the initial density of the material and b is the body force. The Cauchy and Kirchhoff stresses (τ) are linked through $\sigma = \frac{\tau}{J}$ where $J = \det(F)$ which in this case is equal to F_{zz} .

Due to the boundary conditions it is known that $F = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & F_{zz} \end{bmatrix}$.

When there are only normal components, the logarithmic strain is defined as

$$\varepsilon^{(0)} = \frac{1}{2} \ln(F^2) = \ln(F) \quad (\text{A.3})$$

The deformation gradient can be split into elastic and plastic components using multiplicative decomposition $F = F^e F^p$. Because of this and the knowledge of F it should be noted that:

$$F_{xx}^e F_{xx}^p = 1 \quad \text{and} \quad F_{zz}^e F_{zz}^p = J. \quad (\text{A.4})$$

Using the fact that $\ln(F^e F^p) = \ln(F^e) + \ln(F^p)$ allows the strain to be split

into elastic and plastic components

$$\varepsilon^{(0)} = \varepsilon^e + \varepsilon^p = \ln(F^e) + \ln(F^p) \quad (\text{A.5})$$

and with $\nu = 0$ we can assume that the Kirchhoff stress is linked to the elastic logarithmic strain through

$$\tau = E\varepsilon^e. \quad (\text{A.6})$$

When elasto-plastic behaviour has started, using the above relationships we can write

$$E\varepsilon_{xx}^e - E\varepsilon_{zz}^e = \rho_y = E \ln(F_{xx}^e) - E \ln(F_{zz}^e) \quad (\text{A.7})$$

Rearranging this gives

$$\frac{\rho_y}{E} = \ln(F_{xx}^e) - \ln(F_{zz}^e) = \ln\left(\frac{F_{xx}^e}{F_{zz}^e}\right) \quad (\text{A.8})$$

so it can be seen that

$$F_{xx}^e = F_{zz}^e e^{\frac{\rho_y}{E}}. \quad (\text{A.9})$$

The derivative of the yield function with respect to the Kirchhoff stress ($\frac{df}{d\tau}$) can be shown through use of the chain rule to be

$$f_{,\tau} = f_{,\rho} \rho_{,J_2} \{J_2, S\}^T [S, \tau] = \frac{3}{2\rho} \{S\} = \frac{3}{2\rho} \frac{1}{3} \begin{pmatrix} \tau_{xx} - \tau_{zz} \\ \tau_{xx} - \tau_{zz} \\ 2\tau_{zz} - 2\tau_{xx} \end{pmatrix} \quad (\text{A.10})$$

Using $\rho = |\tau_{xx} - \tau_{zz}|$ and $\varepsilon^p = \dot{\gamma} \{f_{,\tau}\}$ we can arrive at the relationship

$$\frac{\varepsilon_{xx}^p}{\varepsilon_{zz}^p} = -\frac{1}{2} \quad \text{or} \quad \frac{\ln(F_{xx}^p)}{\ln(F_{zz}^p)} = -\frac{1}{2} \quad (\text{A.11})$$

Rearranging this gives

$$\ln(F_{xx}^p \sqrt{F_{zz}^p}) = 0 \quad (\text{A.12})$$

Leading to the relationship

$$F_{xx}^p \sqrt{F_{zz}^p} = 1 \quad (\text{A.13})$$

Combining [A.13](#) and [A.4](#) it can be seen that

$$\sqrt{F_{zz}^p} = F_{xx}^e. \quad (\text{A.14})$$

Substitution of this into [A.9](#) and squaring both sides gives

$$F_{zz}^p = F_{zz}^e{}^2 e^{\left(\frac{2\rho}{E}\right)} \quad (\text{A.15})$$

Using this, we can express $F_{zz} = F_{zz}^e{}^3 e^{\left(\frac{2\rho}{E}\right)}$ and, using the solution for Cauchy stress above, say that

$$\sigma_{zz} = \frac{1}{F_{zz}^e{}^3 e^{\left(\frac{2\rho}{E}\right)}} E \ln(F_{zz}^e). \quad (\text{A.16})$$

The elastic part of the deformation gradient can then be found using a Newton process to solve for F_{zz}^e in [\(A.16\)](#). Using the above relationships it is possible to calculate the remaining parts of the deformation gradient and find the out of plane stresses as

$$\sigma_{xx} = \frac{1}{F_{zz}^e} E \ln(F_{xx}^e). \quad (\text{A.17})$$