RESEARCH ARTICLE

WILEY

On the implementation of a material point-based arc-length method

Nathan D. Gavin[1] | Giuliano Pretti[1] | William M. Coombs[1] |
John C. Brigham[2] | Charles E. Augarde[1]

[1]Department of Engineering, Durham University, Durham, UK

[2]Swanson School of Engineering, University of Pittsburgh, Pittsburgh, Pennsylvania, USA

Correspondence
William M. Coombs, Department of Engineering, Durham University, Durham, UK.
Email: w.m.coombs@durham.ac.uk

Summary

The material point method is a versatile technique which can be used to solve various types of solid mechanics problems, especially those involving large deformations. However, the capability of the material point method to track a load-displacement response can deteriorate once a limit point, such as snap-through or snap-back, in the response is encountered. One way of overcoming this is to use path following techniques, such as an arc-length method. This technique is well established in finite element analysis but not within any material point method formulation. This paper provides for the first time an arc-length controlled implicit, quasi-static material point method. The modifications to the standard arc-length scheme to allow for the stable execution of an arc-length solver within the material point method are detailed. The capability of the material point-based arc-length method is demonstrated through a number of problems, which include linear elastic, non-linear elastic, linear elastic-perfectly plastic and linear elastic-plastic softening material behaviour under large deformations. The techniques presented in this paper are essential for arc-length techniques to be applied effectively to the material point method and the combination of these techniques makes the method suitable for new problems that cannot be solved with existing implicit material point approaches.

KEYWORDS
arc-length methods, elastoplasticity, large deformation mechanics, material point method, snap-through

1 | INTRODUCTION

In the finite element method (FEM), for a quasi-static, implicit solution of a non-linear problem, the applied load (or applied displacement) is incremented at the start of each load step and a Newton-Raphson scheme is often used to determine the corresponding displacement (or load) such that the out-of-balance forces of the system are below a given tolerance. However, this procedure only works if the incremented variable is strictly increasing throughout the simulation such that the equilibrium path can be traced successfully. Figure 1A shows an increasing equilibrium path in which a single value of force corresponds to a single value of displacement. It is possible to continue increasing either variable
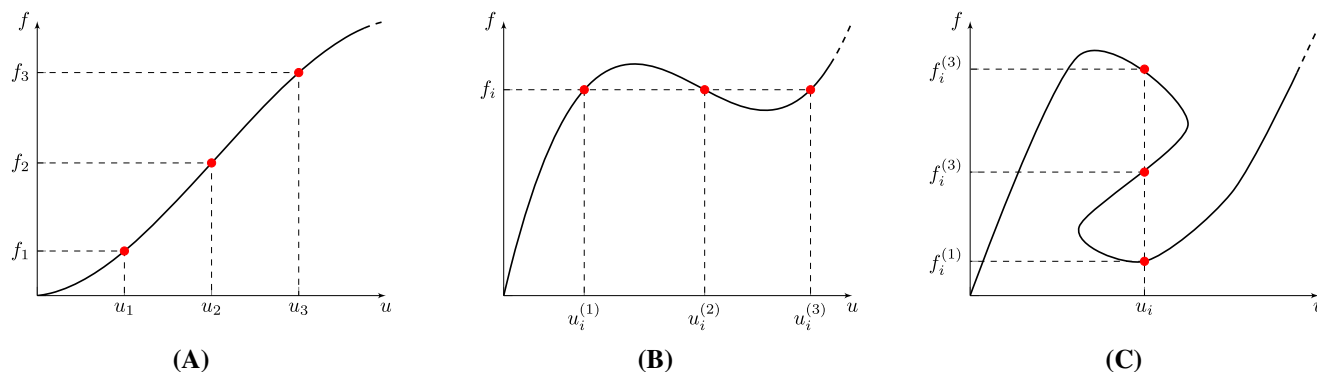
FIGURE 1 Equilibrium paths of (A) increasing response, (B) snap-through response and (C) snap-back response.

and determine a solution on the equilibrium path. On the other hand, problems involving large deformations[1] and some types of non-linear material behaviour[2] will often have critical points along the equilibrium path where, after the critical point, it is not possible to increment the variable further and obtain a solution. A snap-through response is when an increase in displacement for a decrease in load is seen, as shown in Figure 1B. In the case of a snap-through response, a single force value can be associated with multiple displacement states and it is may not be possible to determine the correct state, meaning that it is not possible to model with a load controlled scheme. It is still possible to model a snap-through response with a displacement controlled scheme as the equilibrium path is strictly increasing in terms of displacement, but this can also change the physical problem being modelled depending on the nature of the imposed boundary conditions. A snap-back response is when a decrease in displacement is seen after a snap-through response, as shown in Figure 1C. As with a snap-through response, it is not possible to use a load controlled scheme due to the multiple possible displacement states for a given force, a snap-back response also has multiple load states for a given displacement. As the equilibrium path is not strictly increasing in terms of displacement for a snap-back response, it is not possible to model this response using a displacement controlled scheme. In order to overcome this issue and be able to trace the equilibrium paths of each response, a path following technique is required, where the most common seems to be the arc-length method.[3–5]

The arc-length method was independently developed by Riks[3,4] and Wempner[5] in the 1970s and is one of the main techniques used in finite element analysis (FEA) to overcome the issues associated with the solving of snap-through and snap-back equilibrium paths.[6] Crisfield[7] adapted the method proposed by Riks such that the arc-length method is more suited for use in the FEM when used with the modified Newton-Raphson method. Since its conception, many variations of the arc-length method have been proposed to solve complex equilibrium paths including (but not limited to) Bergan et al.,[8] Ramm,[9] Powell and Simons,[10] Lam and Morley,[11] Fafard and Massicotte,[12] Krenk,[13] Nicholson,[14] May[15] and Pretti et al.[16] The essence of the arc-length method comes from the constraint equation, or arc-length equation, which is included in the non-linear equations of the problem and solved to give an incremental load factor and corresponding incremental displacements. By solving for both load and displacement throughout the simulation, the issues which may arise in traditional load/displacement controlled schemes due to snap-through or snap-back responses are mitigated. The arc-length method has been used to solve a variety of non-linear problems including buckling of cylindrical shells,[17–20] structural analysis,[21] modelling of crack propagation[22,23] and failure of brittle materials.[24,25]

While the arc-length method is well established in the FEM, when modelling problems involving large deformation mechanics the FEM can run into issues involving volumetric locking and large mesh distortion[26,27] leading to inaccurate results and may necessitate the expensive process of re-meshing. The material point method (MPM) is an alternative to standard Lagrangian schemes and was developed by Sulsky and co-workers[28,29] as a solid mechanics extension of the FLuid Implicit Particle (FLIP) method[30] which is itself an extension of the Particle-In-Cell (PIC) method.[31] As a mixture of Eulerian and Lagrangian schemes, the physical domain is discretised by Material Points (MPs) which are immersed in a background mesh of finite elements. The MPs act as integration points and deform through the background mesh which remains unchanged throughout the analysis, avoiding the large mesh distortion seen in the FEM. The MPM is a tool that has been widely used to model many engineering problems including manufacturing processes,[32] large deformation problems in geomechanics,[33–35] fracture mechanics[36] and contact.[37] This paper will focus on an implicit

quasi-static implementation of the MPM, adapting the AMPLE (A Material Point Learning Environment) framework[38] in order to incorporate the arc-length solver.

To the authors' knowledge, there is no current literature detailing the implementation of the arc-length method in a MPM framework and this paper describes the first Material Point-Based Arc-Length Method (MP-BALM). The combination of an arc-length method with the MPM expands the applicability of the MPM to load-controlled problems involving snap-through, which are currently not possible with techniques in the published literature. Section 2 covers the basic principles of the arc-length method as it is implemented in the FEM, followed by an explanation of the MPM and the adaptations needed for the arc-length method to be used successfully with the MPM in Section 3. Section 4 then presents numerical examples using the MP-BALM, including problems involving linear elastic, non-linear elastic, linear elastic-perfectly plastic and linear elastic-plastic softening material behaviour under large deformations. While the arc-length method and the concepts presented in this paper are able to analyse problems involving both snap-through and snap-back responses, the MPM is a continuum method for which, to the authors' knowledge, there are no known problems involving global snap-back responses. Therefore, only problems involving snap-through responses are analysed in Section 4.

2 | THE ARC-LENGTH METHOD

In the standard FEM, an arc-length solver (or Riks solver) solves for both load and displacement based on the converged state of the previous load step and a specified arc length. The equilibrium path is formed of load factor-displacement pairs, $\{\lambda, \mathbf{u}\}$, where λ is the load factor and $\mathbf{u}$ are the nodal displacements. The load factor acts as a scalar multiplier of the external force vector applied to the material domain such that

$$\mathbf{f}^{ext} = \lambda \, \mathbf{f}_0^{ext} \tag{1}$$

where $\mathbf{f}_0^{ext}$ is the initial applied force vector and $\mathbf{f}^{ext}$ is the force vector corresponding to the current load factor. A key feature of the arc-length method is that the distance from the current converged state to the next state along the equilibrium path is limited by a specified arc length, Δl. The most general form of the arc-length constraint is the spherical arc-length constraint[2] which is given by

$$\Delta \mathbf{u}^T \Delta \mathbf{u} + \Delta \lambda^2 \psi^2 \mathbf{f}_0^{ext^T} \mathbf{f}_0^{ext} = \Delta l^2 \tag{2}$$

where $\Delta \mathbf{u}$ and $\Delta \lambda$ are the converged values of the change in nodal displacements and the change in load factor respectively for the current load step, and ψ is a prescribed scaling parameter. This paper utilises the more widely used version of the arc-length constraint, the cylindrical arc-length method, which sets the scaling parameter ψ to be 0, thus only ensuring that the Euclidean norm of the change in nodal displacements is equal to the prescribed arc length.[2] The cylindrical arc-length constraint is given by

$$\Delta \mathbf{u}^T \Delta \mathbf{u} = \Delta l^2. \tag{3}$$

Figure 2 shows the use of the cylindrical arc-length method to track through the equilibrium path for a simple two degree of freedom system. Given a converged state, $(\lambda_i, \mathbf{u}_i)$, when visualised in terms of two displacement components, u_1 and u_2, the constraint equation in (3) creates a cylindrical zone around the converged point. The equilibrium path will intersect the cylinder at two points, A and B. These are the two points on the equilibrium path which satisfy the arc-length constraint based on the change in nodal displacements from the converged state. Based on the direction of travel along the equilibrium path taken in the previous load step, either point A or point B is chosen to be the next converged state and the process is repeated for the next load step. The arc-length scheme is often paired with a Newton-Raphson solution scheme such that equilibrium is achieved as well as satisfying the specified arc-length constraint. In each iteration of a load step, the load factor increment, $\delta\lambda$, is found by solving the quadratic equation[7]

$$\alpha_1 \delta\lambda^2 + \alpha_2 \delta\lambda + \alpha_3 = 0 \tag{4}$$
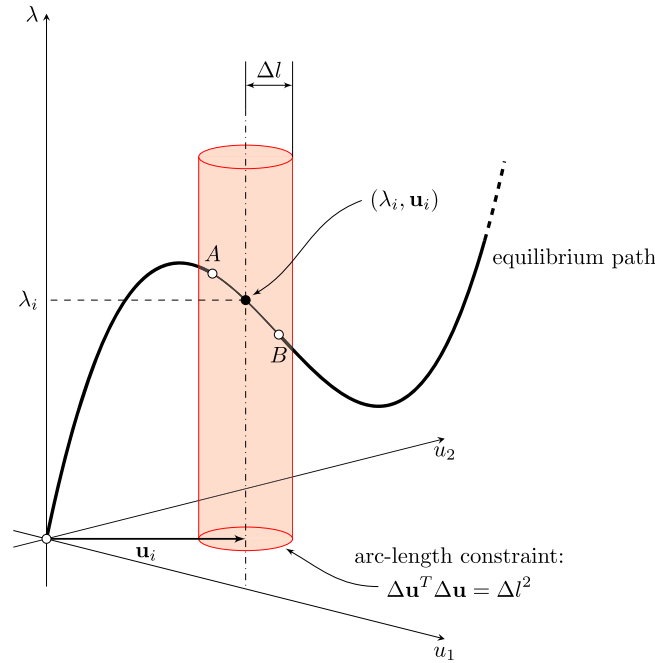
FIGURE 2 The cylindrical arc-length method (reproduced from de Souza Neto et al.[2]).

with the coefficients of the quadratic given by

$$\alpha_1 = \delta\overline{\mathbf{u}}^T \delta\overline{\mathbf{u}} \tag{5a}$$

$$\alpha_2 = 2\,\delta\overline{\mathbf{u}}^T \left\{ \Delta\mathbf{u}^{(k-1)} + \delta\mathbf{u}^* \right\} \tag{5b}$$

$$\alpha_3 = \left\{ \Delta\mathbf{u}^{(k-1)} + \delta\mathbf{u}^* \right\}^T \left\{ \Delta\mathbf{u}^{(k-1)} + \delta\mathbf{u}^* \right\} - \Delta l^2 \tag{5c}$$

where $\Delta\mathbf{u}^{(k-1)}$ is the change in nodal displacements calculated so far in the load step. The tangential displacement, $\delta\overline{\mathbf{u}}$, and the Newton-Raphson displacement[*], $\delta\mathbf{u}^*$, are given by

$$\delta\overline{\mathbf{u}} = \mathbf{K}^{-1}\,\mathbf{f}_0^{ext} \tag{6a}$$

$$\delta\mathbf{u}^* = \mathbf{K}^{-1}\,\mathbf{f}^{oobf} \tag{6b}$$

where $\mathbf{K}$ is the global stiffness matrix and $\mathbf{f}^{oobf}$ is the out-of-balance force vector. As the form of (4) is a quadratic equation, two possible solutions for $\delta\lambda$ will be found; one root will progress along the equilibrium path while the other will track back along the path. Therefore, the selection of the appropriate root is required to ensure that the the simulation progresses in the desired direction. One option to find the most appropriate load factor increment is the "angle criteria" which aims to minimise the angle between the change in nodal displacements so far in the load step and the newly calculated change in nodal displacements. This is done by maximising the dot product of the two vectors in question such that the value of $\delta\lambda$ is chosen such that

$$\delta\lambda^{(k)} = \arg\left[\max_{\delta\tilde{\lambda}}\left(\left\{ \Delta\mathbf{u}^{(k-1)} + \delta\mathbf{u}^* + \delta\tilde{\lambda}\,\delta\overline{\mathbf{u}} \right\}^T \Delta\mathbf{u}^{(k-1)} \right) \right]. \tag{7}$$

A second option for root selection is presented in Hellweg and Crisfield[39] which was proposed to counter the fact that the angle criterion encounters issues for problems involving sharp snap-back responses. This approach involves calculating the two roots of $\delta\lambda^{(k)}$ as normal, and the total displacements, stresses and internal forces are computed for each root and the root which gives the smallest residual is taken. As stated by Hellweg and Crisfield, this method need

only be used in the presence of sharp snap-backs, however, this paper only focuses on snap-through responses thus no further consideration of this method is taken.

The incremental change in nodal displacements, $\delta\mathbf{u}$, is then calculated as a combination of the Newton-Raphson displacement and the tangential displacement (which is scaled by the load factor increment determined by (4) and (7)) such that

$$\delta\mathbf{u}^{(k)} = \delta\mathbf{u}^* + \delta\lambda^{(k)} \, \delta\overline{\mathbf{u}}. \tag{8}$$

At the start of the simulation, an initial arc length is required. It is possible to simply prescribe this value, however, this is not easy for the user to determine a reasonable initial arc length as it is dependent on the mesh size. Instead, the initial arc length can be calculated as the Euclidean norm of the iterative increment of the tangential displacement at the start of the problem, where

$$\Delta l_0 = \sqrt{\delta\overline{\mathbf{u}}^T \, \delta\overline{\mathbf{u}}} \tag{9}$$

where the $\delta\overline{\mathbf{u}}$ used is the initial tangential displacement based on the reference load where an initial iterative load factor increment is prescribed as 1 to produce an initial arc length value. A choice can be made to set this initial arc length value from (9) to be the maximum allowed arc length throughout a simulation, however, this choice is not a condition to be satisfied in the arc-length method. In principle, the arc length can take any value but there may be cases where the arc length becomes too large and causes issues in the analysis, so imposing an upper limit for this value is often the best action. If the number of Newton-Raphson iterations needed in a load step exceeds a prescribed desired number, the arc length for the following load step is adjusted accordingly, allowing for an increase or decrease in the distance away from the current converged state to the next converged state. The arc length for the following load step is calculated at the end of each load step such that

$$\Delta l_i = \max\left[\min\left[\left(\frac{k_{des}}{k_n}\right)^\alpha \Delta l_0, \Delta l_0\right], \frac{\Delta l_0}{\gamma}\right] \tag{10}$$

where k_{des} and k_n are the desired and actual number of Newton-Raphson iterations needed in the load step respectively and α and γ are positive scalars. In the literature, α is taken to be 1[7] or 0.5,[11] however, Ramm[9] explains that when $\alpha = 1$, oscillations in the number of Newton-Raphson iterations needed in each load step are seen and therefore taking $\alpha = 0.5$ is a more stable approach. The value of γ controls the minimum allowed arc length through the entire simulation such that Δl_i is a value in the interval $\left[\frac{\Delta l_0}{\gamma}, \Delta l_0\right]$, this ensures that if the required Newton-Raphson process continues to exceed the desired number of iterations, the arc length will not continue to decrease so much that the tracing of the equilibrium path begins to halt as the allowed distance becomes tiny. There is no theory on the choice of value of γ, therefore $\gamma = 4$ is chosen here based on Pretti et al.[16]

In the first iteration of each load step, there is clearly no information as to the change in nodal displacements, $\Delta\mathbf{u}^{(0)} = \mathbf{0}$. In this case, solving (4) produces the solutions for the initial load factor increment of

$$\delta\lambda^{(1)} = \pm\frac{\Delta l}{\sqrt{\delta\overline{\mathbf{u}}^T \, \delta\overline{\mathbf{u}}}} \tag{11}$$

giving a magnitude of the initial increment. However, it is not possible to use (7) to determine which solution to choose and a predictor is required. It is important to choose the correct value of the predictor solution as one will progress the simulation along the equilibrium path and the other will track back along the path.[1] Multiple options of predictor solution criteria have been presented in the literature to choose the correct sign for $\delta\lambda^{(1)}$ to progress along the path, including:

1. Stiffness determinant[40]–sign determined by the determinant of the current stiffness matrix:

$$sign(\delta\lambda^{(1)}) = sign\left(\left|\mathbf{K}(\mathbf{u}^{(0)})\right|\right) \tag{12}$$

2. Incremental work[10]–sign determined by predictor work increment:

$$sign\left(\delta\lambda^{(1)}\right) = sign\left(\delta\overline{\mathbf{u}}^T \mathbf{f}^{ext}\right) \tag{13}$$

3. Secant path[41,42]–sign determined by deformation of the previous load step $(\Delta\mathbf{u}_n)$:

$$sign\left(\delta\lambda^{(1)}\right) = sign\left(\Delta\mathbf{u}_n^T \delta\overline{\mathbf{u}}\right) \tag{14}$$

Option 1 is widely used and is easily implemented as the stiffness matrix will be known, however, issues may arise if the stiffness matrix is close to singular[7] or in the presence of bifurcations where the solution will oscillate around the bifurcation point.[2] Option 2 is able to trace an equilibrium path through a bifurcation point but is not able to effectively trace snap-back responses.[2] Both options 1 and 2 use information based on the current equilibrium state and nothing about the history of the equilibrium path. Option 3 involves the use of $\Delta\mathbf{u}_n$ which takes the deformation of the mesh in the previous load step into account when determining the deformation of the current load step. The secant path predictor is able to overcome the issues seen with the stiffness determinant and incremental work predictor schemes. An alternative predictor solution was proposed by Kadapa et al.[20] where the nodal displacements and load factor for the first iteration of a load step are calculated as a linear combination of the respective nodal displacements and load factor of the previous two load steps such that change in nodal displacements and load factor increment in the predictor step are given by

$$\left. \begin{aligned} \delta\mathbf{u}^{(1)} &= \alpha\Delta\mathbf{u}_n \\ \delta\lambda^{(1)} &= \alpha\Delta\lambda_n \end{aligned} \right\} \quad \text{where } \alpha = \frac{\Delta l_0}{\Delta l_n} > 0. \tag{15}$$

This is a simpler and less expensive predictor solution which does not require any additional computation to determine the direction of the progression along the equilibrium path in the predictor step.

Figure 3 shows the process of the arc-length solver through the iterations of a single load step. From the previously converged point, $(\lambda_i, \mathbf{u}_i)$, the predictor iteration determines the initial load factor increment (with direction determined by the secant predictor) which is used to calculate an initial change in nodal displacement in order to satisfy the arc-length constraint. In subsequent iterations, the load factor increment is calculated using (4) and then the change in nodal displacements are calculated using (8), these steps are repeated until the Newton-Raphson error is below an allowed tolerance to give the next converged point, $(\lambda_{i+1}, \mathbf{u}_{i+1})$.
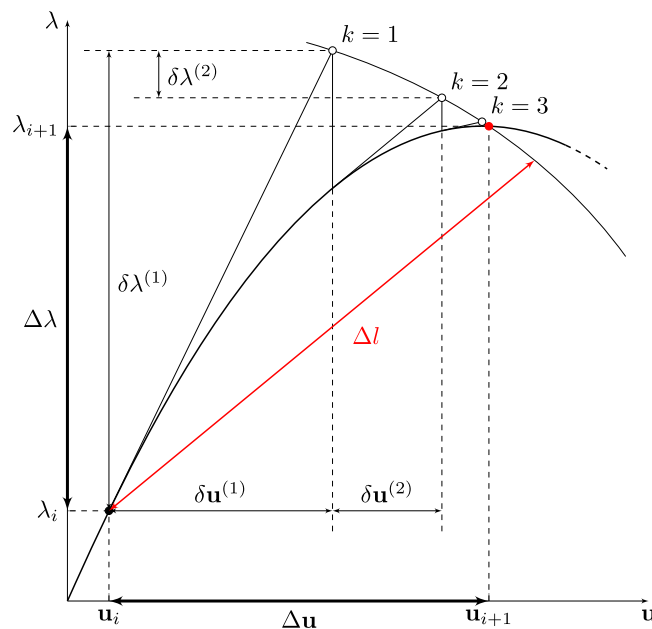

FIGURE 3 Arc-length solution scheme using Newton-Raphson iterations.

The arc-length method detailed in this section is presented in the form of a FEM-based method to solve problems with the path following technique. However, there are some key differences that must be considered when implementing the arc-length method into the MPM formulation, and these will be explained in the next section.

3 | THE MATERIAL POINT-BASED ARC-LENGTH METHOD

The key steps of the standard MPM algorithm for solid mechanics are depicted in Figure 4, starting from the upper left image and moving left to right, top to bottom, the procedure for a load step is:[38]

(a) the physical domain is discretised by MPs which carry data such as mass, stress, deformation, and so forth, of the material and the positions of these points within the background mesh are determined;

(b) the MP data for stress, volume and stiffness is mapped onto the element nodes associated with that MP using the basis functions to generate the internal force and stiffness quantities;

(c) the governing equations are assembled on the background mesh nodes;

(d) the governing equations are solved for the unknown nodal displacements;

(e) the mesh information is then mapped back to the MPs to update the deformation, stress, volume, and so forth, and the MP displacements are calculated;

(f) the MP positions are updated and the background mesh is reset.

For more information on the formulation of the MPM, the reader is directed towards the original work of Sulsky and co-workers[28,29] as well as the many papers[34,36,43,44] and textbooks[45-47] that cover the MPM.
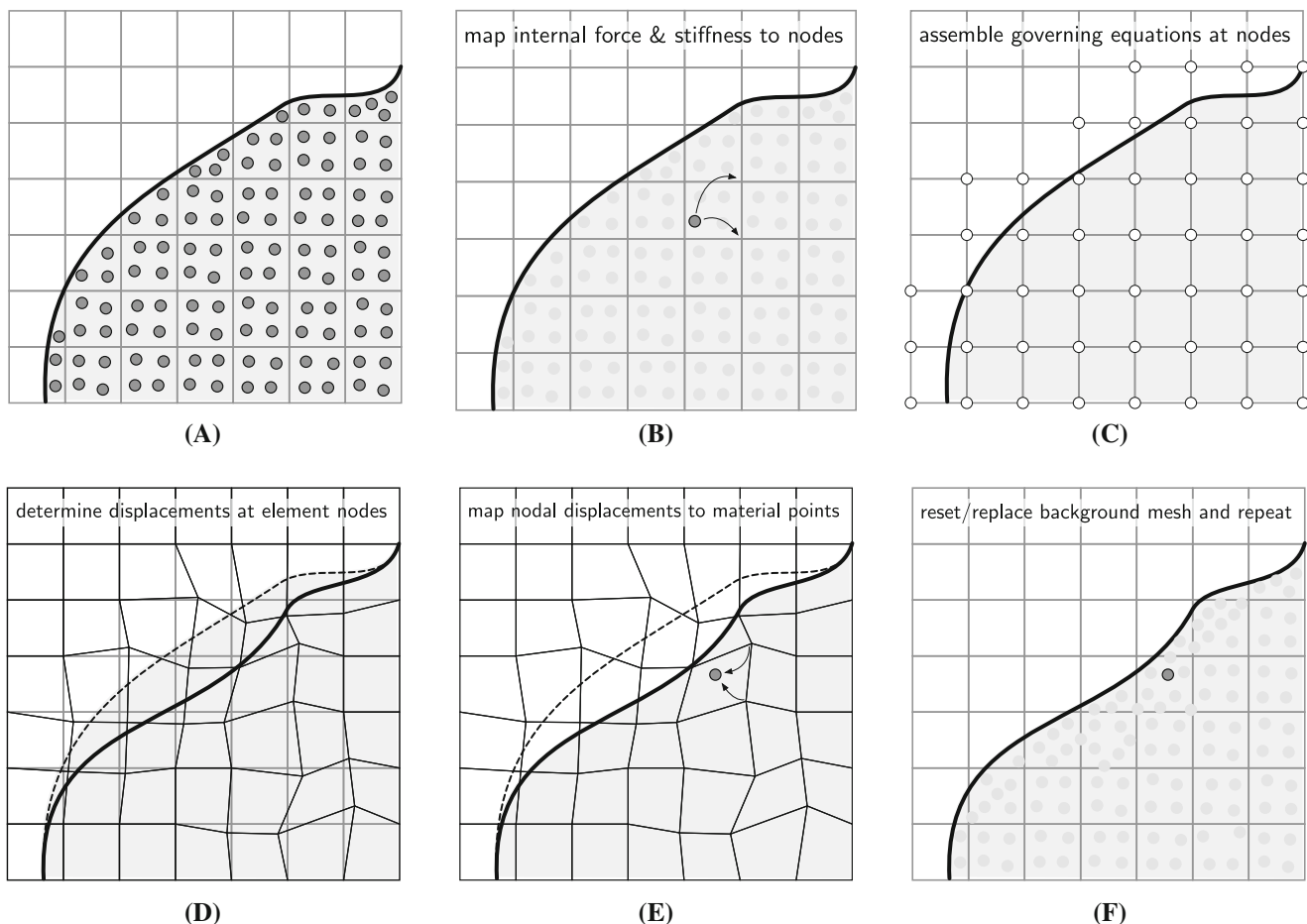

FIGURE 4　MPM algorithm steps (adapted from Coombs et al.[38]).

The MPM approach used in this paper is based on the open-source AMPLE code,[38] a quasi-static implicit finite deformation MPM based on the updated Lagrangian formulation with the weak form of equilibrium given by

$$\int_{\Omega} \left(\sigma_{ij}(\nabla_x \eta)_{ij} - b_i \eta_i \right) \, \mathrm{d}v - \int_{\partial\Omega} (t_i \eta_i) \, \mathrm{d}s = 0 \tag{16}$$

assuming a field of admissible virtual displacements, η_i, where Ω is the domain of the body with boundary $\partial\Omega$. The domain is subjected to a traction, t_i, on the boundary of the domain (with surface, s) and body forces, b_i, applied to the volume, v, of the domain. These tractions and body forces result in a Cauchy stress field, σ_{ij}, through the body. Discretising (16) with background mesh elements, E, and imposing Galerkin's weak form produces

$$\mathcal{A}_{\forall E} \left(\int_E \nabla_x \mathbf{S}_{vp}^T \, \sigma_p \, \mathrm{d}v - \int_E \mathbf{S}_{vp}^T \, \mathbf{b} \, \mathrm{d}v \right) - \int_{\partial\Omega} \mathbf{S}_v^T \, \mathbf{t} \, \mathrm{d}s = \mathbf{0} \tag{17}$$

where $\mathcal{A}$ is the standard assembly operator (in this case acting over all elements in the mesh), $\mathbf{S}_{vp}$ is a matrix containing the basis functions to provide the mapping between the MPs and the background mesh, $\nabla_x \mathbf{S}_{vp}$ is the strain-displacement matrix containing the derivatives of the basis functions with respect to the global coordinates. In this paper, the boundary of the MP domain is generated using a B-spline representation[48] which allows Neumann boundary conditions (such as the tractions in (17)) to be applied directly to the MP domain. The shape functions used to integrate the traction forces over the domain boundary, $\mathbf{S}_v$, are the standard Finite Element shape functions. The weak form in (17) can be discretised by the MPs such that

$$\mathcal{A}_{\forall p} \left(\nabla_x \mathbf{S}_{vp}^T \, \sigma_p \, V_p - \mathbf{S}_{vp}^T \, \mathbf{b} \, V_p \right) - \int_{\partial\Omega} \mathbf{S}_v^T \, \mathbf{t} \, \mathrm{d}s = \{0\} \tag{18}$$

where the standard assembly operator is applied over each MP, p, and V_p is the MP volume.

There are multiple options for the choice of basis functions used to map data between the MPs and the mesh nodes. The original approach was to use standard FEM shape functions[28] in which the MPs were assumed to be concentrated point volumes and the shape functions of the finite element mesh are used. An alternative to standard FEM shape functions are Generalised Interpolation MPM (GIMPM) basis functions[43] which are generated as the convolution of the finite element shape functions with a point characteristic function (taken to be a hat unity function). This generates a set of C^1 continuous basis functions using a domain centred on the MP position over which the volume of the MP is evenly distributed. Another option is to use higher order B-splines[49] instead of the piecewise-linear basic functions, which is seen to significantly reduce spatial quadrature errors.[50] The higher order B-spline basis functions are generated based on a knot vector, with values associated with positions of the mesh nodes, and the location of the point in question relative to the knots within the knot vector. The value of the basis function is calculated recursively based on the order of the B-spline. Using a cubic B-spline produces C^2 continuous basis functions centred at the MP location and spanning four mesh elements in each dimension. This paper however, utilises the GIMPM basis functions throughout.

It is widely documented that the MPM can be unstable, mostly due to cell crossing errors, a problem mitigated by the use of GIMPM basis functions, however there are other issues linked to boundary representation. A key feature of the MPM is that the physical domain of the problem is made up of MPs, therefore, there is no physical boundary representation. For illustrative purposes, the "boundary" of the physical domain is depicted with a thick black line in Figure 4, however, this boundary is not explicitly represented in the method and the edge of the physical domain is implied by the extent of the MPs and their associated domains. If desired, the boundary of the MP domain can be represented using B-splines[48] where points with zero volume are placed along the boundary of the domain which are used as sample points for the creation of a B-spline curve in 2D, or surface in 3D. These boundary points positions are updated at the end of each load step in the same fashion as the MPs, allowing for the tracking of the boundary throughout the analysis. This B-spline boundary also allows Dirichlet and Neumann boundary conditions to be applied directly to the MP domain by integrating over the B-spline curve/surface. The reader should refer to Bing et al.[48] for more information on B-spline boundary representation in the MPM. However, irrespective if the boundary is explicitly represented or not, the fact that the extent of the physical domain will not necessarily coincide with the background mesh means that there is the potential for background elements to only overlap with a small portion of the physical domain. This "small cut" problem can lead to poor conditioning of the global system of equations and impact on the stability of the MPM. In more detail, the ill-conditioning

of the system of equations due to the presence of partially filled elements can be quantified by the system's condition number (the ratio of the largest and smallest eigenvalues of the global system matrix). The smallest eigenvalue of the system is related to the smallest intersection between the physical domain and the background mesh, as this intersection can be much smaller than the size of the elements in the background mesh, the condition number is unbounded, which is detrimental to the solution stage of the analysis. The ghost stabilisation technique proposed by Burman[51] penalises jumps in the gradient of the solution field across elements cut by the physical domain.[52] This is especially useful for the MPM due to the fact that it is often the case that MPs around the edge of the physical domain are located in partially filled elements. Coombs[53] implemented the ghost stabilisation method within the MPM where boundary and faces are determined over which the ghost penalty was integrated, resulting in a well-conditioned system and improved stability in analysis.

When modelling large deformations of nearly incompressible solids, the issue of volumetric locking can occur due to low order elements being used and the low order interpolation polynomials inadequacy to represent the volume preserving displacement fields.[54] One common method of avoiding this volumetric locking is the F-bar approach developed by de Souza Neto and co-workers[54,55] where the deformation gradient is decomposed into deviatoric and volumetric components such that

$$\mathbf{F} = \mathbf{F}_d \, \mathbf{F}_v, \tag{19}$$

where $\mathbf{F}, \mathbf{F}_d$ and $\mathbf{F}_v$ are the deformation gradient, its deviatoric and volumetric components respectively with the two components being defined as

$$\mathbf{F}_d = \det{(\mathbf{F})}^{-1/3} \, \mathbf{F} \text{ and } \mathbf{F}_v = \det{(\mathbf{F})}^{1/3} \, \mathbf{I} \tag{20}$$

A deformation gradient, $\mathbf{F}_0$, is computed which results from the displacement at centroid of the element. The modified deformation gradient, $\overline{\mathbf{F}}$ is calculated as the composition of the deviatoric component of $\mathbf{F}$ and the volumetric component of $\mathbf{F}_0$ such that

$$\overline{\mathbf{F}} = \mathbf{F}_d(\mathbf{F}_0)_v = \left(\frac{\det(\mathbf{F}_0)}{\det(\mathbf{F})} \right)^{-1/3} \mathbf{F}. \tag{21}$$

The modified deformation gradient is then used to compute the stiffness matrix and internal force vector. Coombs et al.[26] implemented the F-bar approach in the MPM, which was chosen as it: does not introduce any additional unknowns; is straightforward to implement; does not restrict the choice of constitutive model; does not require any tuning parameters; and does not require additional MPs to track volumetric behaviour.

Numerical examples presented in this paper utilise the B-spline boundary representation, ghost stabilisation and the F-bar formulation.

3.1 | Reformulation of the arc-length method into the MPM

As the MPM allows the MPs to move through the background mesh, it is possible that the MPs will populate small portions of an element around the boundary of the physical domain. This can be seen in Figure 4 where the domain represented by the MPs has a boundary which cuts through elements of the background mesh. This causes issues with elements that participate in the analysis having very small stiffnesses, resulting in extremely large nodal displacements relative to the applied external load. This means that if an arc-length solver is implemented into the MPM in its FEM form, the large nodal displacements will account for the majority of the allowed nodal displacements of the arc-length constraint, manifesting in very small increments of the load factor and an inefficient tracing of the equilibrium path. We reformulate the arc-length constraint such that it is based on the change in MP displacements to produce the Material Point-Based Arc-Length Method (MP-BALM). The change in MP displacement, $\Delta\mathbf{u}_p$, is calculated by mapping the change in nodal displacements to the location of the MP such that

$$\Delta\mathbf{u}_p = \sum_{v=1}^{n} S_{vp}(x_p)\Delta\mathbf{u}_v = \mathbf{N}_p\Delta\mathbf{u} \tag{22}$$

where n is the number of nodes in the background mesh, $S_{vp}(x_p)$ is the basis function of node (or vertex) v based on the position of the MP, x_p, $\Delta\mathbf{u}_v$ is the change in displacement of node v and $\mathbf{N}_p$ is a $3 \times 3n$ matrix containing the basis functions for all nodes in the mesh. A diagonal matrix, $\overline{\mathbf{N}}$, of size $3n \times 3n$ can be formed which contains the sum of all MP basis functions for every node in the mesh such that

$$\overline{\mathbf{N}} = \begin{bmatrix} \sum_{\forall p} S_{1p}(x_p) & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & \sum_{\forall p} S_{1p}(x_p) & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & \sum_{\forall p} S_{1p}(x_p) & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \sum_{\forall p} S_{np}(x_p) & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & \sum_{\forall p} S_{np}(x_p) & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & \sum_{\forall p} S_{np}(x_p) \end{bmatrix}. \tag{23}$$

The values within $\overline{\mathbf{N}}$ are scaled by the maximum diagonal value such that the matrix contains values between 0 and 1. The values in $\overline{\mathbf{N}}$ are representative of the influence of the change in nodal displacements on the physical body, where nodes surrounded by fully populated elements have a value equal or close to 1 and the nodes associated with less physical material have a value of less than 1. Due to the nature of the MPM, the displacements of the MPs have more of a physical meaning compared to the displacements of the nodes of the mesh as it is the MPs that represent the deformation of the physical domain in a MPM analysis. Therefore, in a similar fashion to the MP displacement calculation in (22), a vector of scaled nodal displacements, $\Delta\mathbf{u}_s$, is calculated as

$$\Delta\mathbf{u}_s = \overline{\mathbf{N}}\Delta\mathbf{u}, \tag{24}$$

therefore, the arc-length constraint in (3) can then be modified using scaled nodal displacements giving

$$\begin{aligned} \Delta\mathbf{u}_s^T \Delta\mathbf{u}_s &= \left(\overline{\mathbf{N}}\Delta\mathbf{u}\right)^T \left(\overline{\mathbf{N}}\Delta\mathbf{u}\right) \\ &= \Delta\mathbf{u}^T \left(\overline{\mathbf{N}}^T \overline{\mathbf{N}}\right) \Delta\mathbf{u} \\ &= \Delta\mathbf{u}^T \tilde{\mathbf{N}} \Delta\mathbf{u} = \Delta l^2 \end{aligned} \tag{25}$$

where $\tilde{\mathbf{N}} = \overline{\mathbf{N}}^T \overline{\mathbf{N}}$ is a diagonal matrix with values between 0 and 1. The result of the modification in (25) is that the nodes associated with partially filled elements will not take up as much of the allowed change in nodal displacements, improving the performance of the arc-length solver in tracing the equilibrium path with more uniform increments. This will be demonstrated with numerical examples in Section 4. The values in $\overline{\mathbf{N}}$ are scaled in order to ensure that changing the number of MPs used in the analysis does not have a significant effect on the progression of tracing along the equilibrium path. For example, if no scaling is present, increasing the number of MPs will increase the values on the diagonal of $\overline{\mathbf{N}}$ thus increasing the left hand side of (25) and resulting in smaller displacements per load step and a slower progression through the analysis. Changing the number of MPs should not result in significant changes in how an analysis progresses. The $\overline{\mathbf{N}}$ matrix scales the nodal displacements in the arc-length condition based on a node's influence on the physical domain, however, as the values in the matrix are rescaled such that they are between 0 and 1, $\overline{\mathbf{N}}$ has no physical meaning in terms of the problem. The reason for the use of $\overline{\mathbf{N}}$ is to dampen the large nodal displacements of the partially filled elements around the boundary of the domain (as the corresponding values in $\overline{\mathbf{N}}$ will be close to zero) while maintaining the displacements of the nodes which have more influence on the physical domain.

The addition of the $\tilde{\mathbf{N}}$ matrix in (25) is then incorporated throughout the MP-BALM formulation to calculate the load factor increment where the new coefficients of (4) become

$$\alpha_1 = \delta\overline{\mathbf{u}}^T \tilde{\mathbf{N}} \delta\overline{\mathbf{u}} \tag{26a}$$

$$\alpha_2 = \delta\overline{\mathbf{u}}^T \tilde{\mathbf{N}} \{\Delta\mathbf{u}^{(k-1)} + \delta\mathbf{u}^*\} \tag{26b}$$

$$\alpha_3 = \{\Delta\mathbf{u}^{(k-1)} + \delta\mathbf{u}^*\}^T \tilde{\mathbf{N}} \{\Delta\mathbf{u}^{(k-1)} + \delta\mathbf{u}^*\} - \Delta l^2. \tag{26c}$$

The initial arc-length calculation in (9) also includes $\tilde{\mathbf{N}}$ such that

$$\Delta l_0 = \sqrt{\delta\overline{\mathbf{u}}^T \tilde{\mathbf{N}} \delta\overline{\mathbf{u}}} \tag{27}$$

as well as for the predictor solution where the initial load factor increment is calculated as

$$\delta\lambda^{(1)} = sign\left(\Delta\mathbf{u}_n^T \tilde{\mathbf{N}} \delta\overline{\mathbf{u}}\right) \frac{\Delta l}{\sqrt{\delta\overline{\mathbf{u}}^T \tilde{\mathbf{N}} \delta\overline{\mathbf{u}}}}. \tag{28}$$

It is still important to include the scaling matrix in (28), as $\delta\overline{\mathbf{u}}$ is calculated from the assembled stiffness matrix for the MPs in the current state meaning that the partially filled elements at the boundary of the MP domain may still see large nodal displacements. The large nodal displacements may also be seen within $\Delta\mathbf{u}_n$ as these are simply the calculated nodal displacements from the previous load step where the large nodal displacements may have also occurred.

3.2 | Nodal displacement reconstruction

This paper utilises the secant path predictor in (28) (scaled form of (14)) for the determination of the direction of the equilibrium path in the first iteration of a load step where the converged nodal displacement of the previous load step is required. However, since a key feature of the MPM is that the background mesh is reset at the end of every load step, obtaining these nodal displacements is not trivial and instead using the nodal displacement vector at the end of the previous load step is not feasible as the body has deformed and the MPs have changed position, meaning that the domain is not influenced by the mesh nodes in the same way as in the previous load step. A method of reconstructing the nodal displacement field based on the MP displacements of the previous load step and the updated MP positions is therefore needed. This is done using a least squares weighted residual technique[48]

$$\int_\Omega \mathbf{S}_{vp}^T \{\mathbf{S}_{vp} \, \Delta\mathbf{u}_n - \Delta\mathbf{u}_p^a\} \, dv = \{0\} \tag{29}$$

where $\Delta\mathbf{u}_p^a$ are the known MP displacements from the previous load step and $\Delta\mathbf{u}_n$ are the unknown nodal displacements which are to be solved for. Rearranging for the unknown nodal displacements we obtain

$$\Delta\mathbf{u}_n = \left[\int_\Omega \mathbf{S}_{vp}^T \, \mathbf{S}_{vp} \, dv\right]^{-1} \left\{\int_\Omega \mathbf{S}_{vp}^T \, \Delta\mathbf{u}_p^a \, dv\right\} \tag{30}$$

It is worth noting that $\Delta\mathbf{u}_n$ is evaluated globally in order to minimise the residual error across the whole domain meaning that the integrals in Equation (30) must be assembled into a global matrix and vector respectively. This means that $\Delta\mathbf{u}_n$ is calculated as

$$\Delta\mathbf{u}_n = \left[\underset{\forall p}{A}\left(\mathbf{S}_{vp}^T \, \mathbf{S}_{vp} \, V_p\right)\right]^{-1} \left\{\underset{\forall p}{A}\left(\mathbf{S}_{vp}^T \, \Delta\mathbf{u}_p^a \, V_p\right)\right\} \tag{31}$$

As the nodal displacements of the previous load step are reconstructed based on the MPs' position in the current state, there may still be cases where large nodal displacements are calculated for partially filled elements around the boundary of the MP domain, therefore, the scaling matrix is needed in the predictor meaning (28) should be used over (14) in order to mitigate issues involving large nodal displacements.

Algorithm 1 shows the standard solution procedure for the MP-BALM. It is worth pointing out that a zeroth load step is analysed to generate the initial state of the domain and the arc-length solver is not used in this load step. Compared

---

Algorithm 1. MP-BALM procedure

---

for $i = 0 : i_{max}$ **do**
    $\Delta\mathbf{u}^{(0)} = 0$
    determine MP data in current state
    calculate scaling matrix, $\bar{\mathbf{N}}$ - (23)
    determine free degrees of freedom
    calculate external force vector, $\mathbf{f}_0^{ext}$
    nodal displacement reconstruction, $\Delta\mathbf{u}_n$ - (31)
    initialise Newton-Raphson error, $fErr = 1$
    start iteration counter, $k = 0$
    while $fErr > tolerance$ **do**
        if $i > 0$ **then**
            if $k > 0$ **then**
                if $i = 1$ and $k = 1$ **then**
                    calculate initial arc length - (27)
                end if
                calculate tangential displacement, $\delta\bar{\mathbf{u}}$ - (6a)
                if $k = 1$ **then**
                    determine $\delta\lambda^{(1)}$ using predictor solution - (28)
                else
                    calculate Newton-Raphson displacement, $\delta\mathbf{u}^*$ - (6b)
                    determine $\delta\lambda^{(k)}$ by solving quadratic equation - (4) with (26)
                end if
                calculate incremental change in nodal displacement, $\delta\mathbf{u}^{(k)}$ - (8)
            else
                zero incremental displacement in zeroth iteration, $\delta\mathbf{u}^{(0)} = 0$
            end if
        end if
        increment change in nodal displacements, $\Delta\mathbf{u}^{(k)} = \Delta\mathbf{u}^{(k-1)} + \delta\mathbf{u}^{(k)}$
        assemble stiffness matrix, $\mathbf{K}$, and internal force vector, $\mathbf{f}^{int}$
        calculate out-of-balance forces, $\mathbf{f}^{oobf}$
        calculate Newton-Raphson error
    end while
    update MP positions
    update Δl - (10)
end for

---

to the AMPLE framework, the MP-BALM algorithm is reasonably similar. The nodal displacement reconstruction takes place at the start of each load step before entering the Newton-Raphson iteration loop. The arc-length solver itself requires multiple if statements in order to determine how to find the load factor increment and the arc length needs to be updated at the end of each load step based on the iterative performance of said load step. This algorithm is used to solve the numerical examples presented in the next section.

4 | NUMERICAL EXAMPLES

A series of numerical examples are presented to demonstrate the capability of the proposed arc-length solver with a MPM formulation. The first example demonstrates that the solver can be implemented into a MPM code and produce correct results for simple problems with analytical solutions. Further examples are given to demonstrate the capabilities of an arc-length solver within the MPM with problems which encounter snap-through responses.

4.1 | Compression of an elastic column under self weight

To demonstrate that the implemented solver is functional within a MPM code, a simple benchmark test is utilised, namely the 1D compression of an elastic column with an initial height of $l_0 = 50$ m under self weight. Although this problem is in essence 1D, a 2D plane strain implementation is used. The material is modelled as a Henky material with a Young's modulus of 10 kPa and a Poisson's ratio of zero. The background mesh is generated using square elements with roller boundary conditions applied over the base and sides of the mesh. The column is discretised using a distribution of 2×2 evenly distributed generalised interpolation material points in each populated element of the background mesh. A target body load of 800 N m^{-2} ($g = 10$ m s^{-2} and a density of $\rho_0 = 80$ kg m^{-3}) is applied. As the load at any point in the analysis is determined by the initial applied load and the load factor (as per (1)), an initial gravitational acceleration is prescribed and the problem is run until the load at the end of a load step exceeds the target body load. Selecting a smaller initial load means that more load steps are required to reach the target load.

To study convergence, the number of elements along the height of the column (y-direction) is varied between 2^2 and 2^{13} (between 4 and 8192) while a single element is used through the width of the column (x-direction) throughout. The initial MP to mesh element ratio is also maintained throughout. The problem is run with the arc-length solver for three different initial body loads, 8, 4 and 2 N m^{-2} which results in the simulations running over 17, 33 and 67 load steps respectively.

The analytical solution for the normal stress in the y-direction at the end of the analysis is

$$\sigma_{yy}^a = \rho_0 g_n (l_0 - Y) \tag{32}$$

where Y is the original position of the point in the body, l_0 is the original height of the column and g_n is the gravitational acceleration due to the load factor at the end of the simulation. Figure 5 shows the numerical results for the column under self weight using a mesh of 2^6 elements and an initial body force of 2 N m^{-2}, the calculated vertical stress is plotted against the original vertical position of each MP. The solid line in Figure 5 is the the analytical solution in (32).

The error between the calculated stress in the column and the analytical solution is calculated as

$$\text{error} = \sum_{p=1}^{n_p} \frac{\|\sigma_{yy}^p - \sigma_{yy}^a(Y_p)\| V_p^0}{(g_n \rho_0 l_0) V_0} \tag{33}$$

where σ_{yy}^p is the normal stress in the y-direction at each MP position and $V_0 = \sum V_p^0$ is the initial volume of the column. As the arc-length method will not perfectly achieve the target body load and the simulation is simply run until the body load first exceeds the target load, each run will have a different value for g_n. Running over 17, 33 and 67 load steps gives final g_n values of 12.9, 11.7 and 10.2 m s^{-2} respectively. Figure 6 shows the convergence of the solution as the mesh is refined. It is clear that the rate of convergence matches the rates of convergence when running the same problem with a linear solver. It is also clear to see that the error value of the solver compared to the linear solver depends on total number of load steps
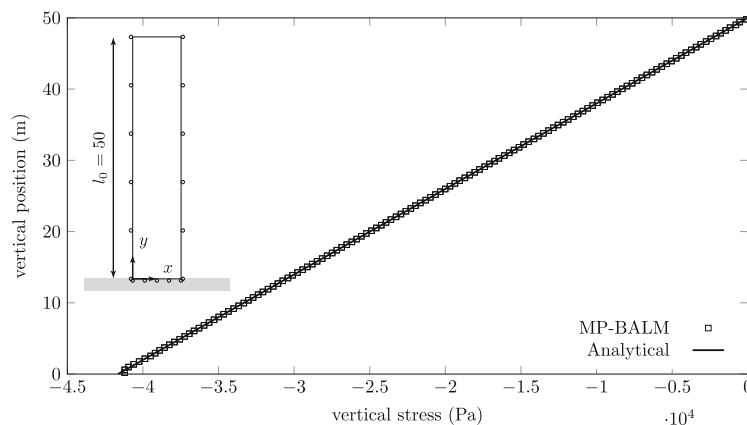

FIGURE 5 Numerical solution of a column under self weight plotted against the analytical solution.
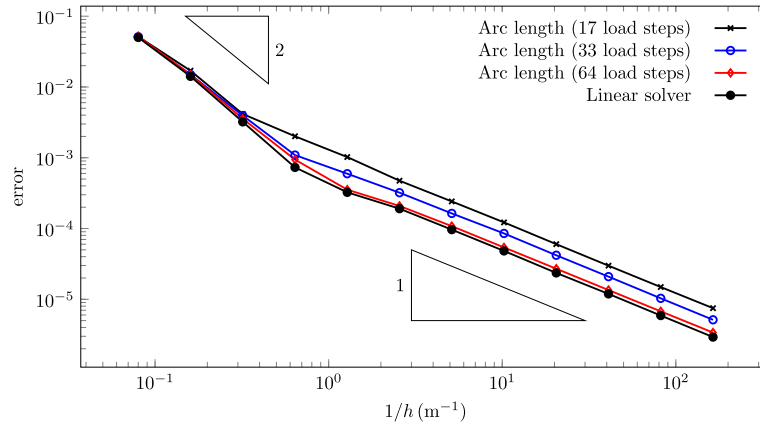
FIGURE 6 Compression of column under self weight convergence.

required to reach the target body force, this is due to the fact that the linear solver is run over 40 evenly spaced load steps whereas in the arc-length solver, the load increment is not evenly distributed throughout the simulation. As the number of load steps used to run the simulation is increased, the convergence behaviour of the MP-BALM tends towards that of the linear solver.

4.2 | Hyperelastic cylinder under internal pressure

The next example is that of a 2D incompressible, hyperelastic cylinder under internal pressure. The cylinder has an inner radius $R_i = 1$ m with the outer radius, R_o, set such that the ratio $\frac{R_i}{R_o} = 0.85$. The cylinder is modelled as a hyperelastic material using the regularised, isotropic Ogden model[2,56] with the strain-energy function, W, given in terms of the principal stretches, $\overline{\lambda}_i$ $(i = 1, 2, 3)^\dagger$, such that

$$W(\overline{\lambda}_1, \overline{\lambda}_2, \overline{\lambda}_3, J) = \sum_{r=1}^{N} \frac{\mu_r}{\alpha_r} \left[(\overline{\lambda}_1)^{\alpha_r} + (\overline{\lambda}_2)^{\alpha_r} + (\overline{\lambda}_3)^{\alpha_r} - 3 \right] + \frac{1}{2} K (\ln J)^2 \tag{34}$$

where μ_r and α_r are pairs of material parameters, K is the material bulk modulus, J is the volume ratio. It is worth noting that due to the fact the cylinder is incompressible, it is assumed that $J \approx 1$, however, this is not strictly enforced, therefore, incompressibility is ensured through a suitably large value for the bulk modulus. The material parameters are defined as[57]

$$\alpha_1 = 1.3 \qquad \alpha_2 = 5.0 \qquad \alpha_3 = -2.0$$
$$\mu_1^* = 1.491 \qquad \mu_2^* = 0.003 \qquad \mu_3^* = -0.023$$

where $\mu_i^* = \frac{\mu_i}{\mu}$ and μ is the ground-state shear modulus. The ground-state shear modulus is chosen arbitrarily as 0.375 MPa, which is the equivalent shear modulus based on a Young's modulus of 1 MPa and a Poisson's ratio of 0.3. The bulk modulus is set such that it is sufficiently large to enforce near incompressibility and takes the value of 99.8 MPa.

Due to symmetry, one quarter of the cylinder cross section is modelled with a plane strain assumption. A diagram of the problem is shown in Figure 7. The background mesh is generated with an overall width and height of 3.5 m in order to provide enough space for the domain to deform up to a final inner stretch of 3, in other words, until the deformed inner radius is three times the initial inner radius. Each element in the background mesh is a square with size $h = 0.05$m. In order to generate the initial domain of the cylinder, the entire mesh is filled with MPs where each element contains 8×8 evenly spaced generalised interpolation material points. It is important to ensure that there is a large number of MPs initially populating the mesh for optimal integration over the domain. The MPs which do not lie within the boundaries of the cylinder are then removed, leaving the desired physical domain made up of a total of 7722 MPs. This check is performed at the centre of the MP domain, determining whether the centre of the domain is within the boundaries of the cylinder meaning that some portions of the GIMP domains may exist outside of the cylinder boundary. Roller
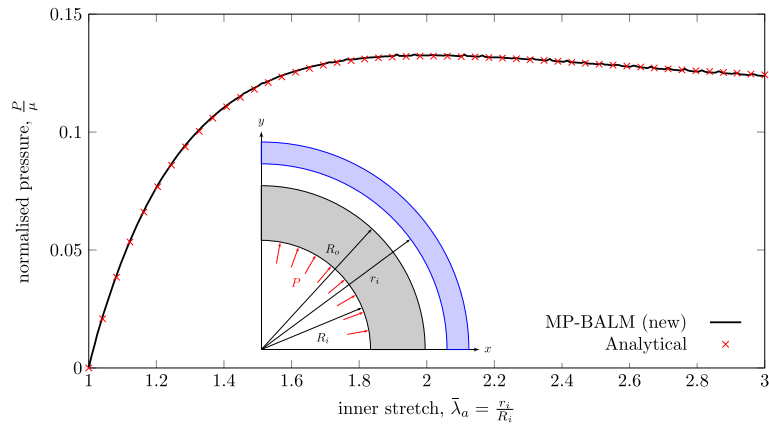
FIGURE 7 Two-dimensional hyperelastic cylinder under internal pressure. Plot: inner stretch against normalised applied internal pressure comparing calculated results (solid black line) and analytical solution (red crosses). Inset: diagram of cylinder problem setup showing initial inner radius and outer radius and deformed inner radius.

boundary conditions are applied over each edge of the background mesh. The problem is run using the ghost stabilisation technique[53] in order to reduce errors and improve the stability of the simulation due to partially filled background elements. The ghost stabilisation parameter is set to $\gamma = 10^6$. An initial normalised pressure, $P^* = \frac{P}{\mu}$, of 5×10^{-3} is applied over the inner boundary of the cylinder using a B-spline boundary representation.[48] No pressure is applied over the outer radius of the cylinder. An analytical solution for this problem is given as[57]

$$P = \int_{\bar{\lambda}_a}^{\bar{\lambda}_b} \frac{\hat{W}_{\bar{\lambda}} \, \mathrm{d}\bar{\lambda}}{(\bar{\lambda}^2 \bar{\lambda}_z - 1)} \tag{35}$$

where $\hat{W}(\bar{\lambda}, \bar{\lambda}_z) = W(\bar{\lambda}, \bar{\lambda}_z, (\bar{\lambda}\bar{\lambda}_z)^{-1}, 1)$ and $\hat{W}_{\bar{\lambda}}$ is the derivative of $\hat{W}$ with respect to $\bar{\lambda}$ and the integration is performed between the radial stretch at the inner radius, $\bar{\lambda}_a$, and the radial stretch at the outer radius, $\bar{\lambda}_b$. As the problem in question is run using the plane strain assumption, there is zero stretch in the axial direction meaning $\bar{\lambda}_z = 1$ throughout the analysis. Figure 7 shows the calculated inner stretch against applied internal pressure (normalised with the ground-state shear modulus) for the analysis using the new (scaled) approach. There is a clear snap-through behaviour displayed with the maximum in pressure seen at an inner stretch of 1.9. Comparing the calculated load-displacement curve with the analytical solution calculated using (35), the MP-BALM matches the analytical solution very closely throughout the simulation.

At large displacements, small fluctuations can be seen in the calculated load-displacement curve, this may be due to the fact that the physical domain does not have a defined boundary in the MPM with the use of the B-spline boundary being used as an approximation for the boundary. This means that the pressure will not be applied perfectly to the true boundary of the domain. Also, at large deformations, the incompressible cylinder is stretched by a large distance and the thickness of the cylinder is reduced, potentially meaning that a small number background elements are active through the thickness of the cylinder thus the integration over the physical domain is sub-optimal, especially over the boundaries. Figure 8 shows the undeformed and deformed configurations of the cylinder where it can be seen that at least two background elements span the thickness of the cylinder in the undeformed configuration whereas only a single element contains the full thickness of the cylinder in the deformed configuration for this problem setup.

The problem is run using the traditional (unscaled) and new arc-length approaches for the same problem setup. The traditional approach takes 299 load steps to reach the desired final inner stretch, whilst the new approach takes only 233 load steps in order to complete the analysis. As mentioned above, the results from the new approach fit the analytical solution very well, a similar set of results are also seen with the traditional approach, however, more erratic fluctuations in the curve are seen for the highly deformed state ($\bar{\lambda}_a > 2.6$). In order to compare the two approaches, a study is performed over the first 100 load steps of the analysis for mesh sizes of 0.1, 0.05, 0.025 and 0.0125 m, each for initial MP distributions of 6^2 and 8^2 MPs per element. Figure 9 shows the inner stretch of the cylinder after 100 load steps
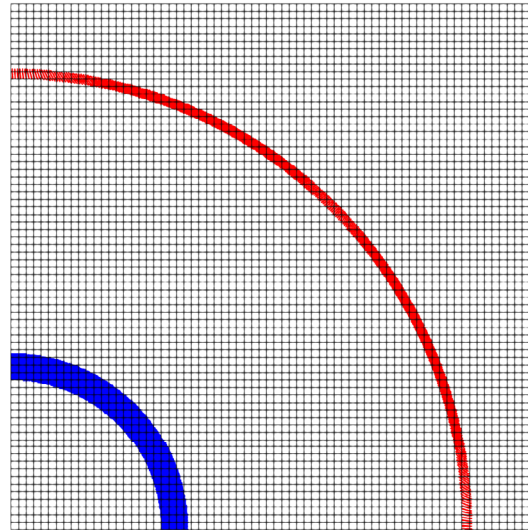
FIGURE 8 Two-dimensional hyperelastic cylinder under internal pressure in the undeformed (blue) and deformed (red) configurations.
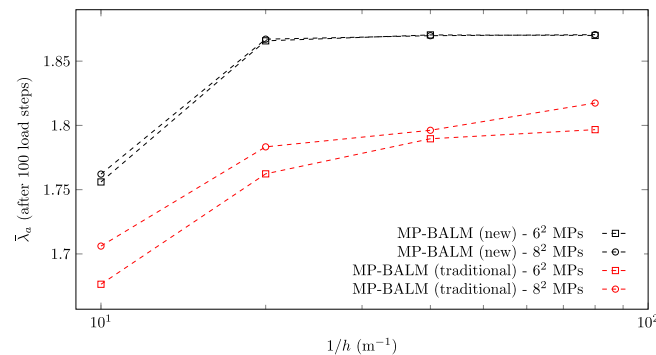
FIGURE 9 Two-dimensional hyperelastic cylinder under internal pressure comparison inner stretch after 100 load steps for both traditional and new arc-length approaches for varying mesh sizes and initial MP distributions.

for varying mesh size and initial distribution. The results shown in Figure 9 does not represent a plot of accuracy and is not representative of the convergence of the analyses to the exact solution. By showing the inner stretch of the cylinder after 100 load steps, it is possible to see the sensitivity of the arc-length approaches (both the traditional and new approaches) to the discretisation of the problem with the scaled (new) approach displaying much less sensitivity to mesh refinement and changes in MP distribution compared to the unscaled (traditional) approach. It is clear to see that the traditional approach is consistently slower in terms of progression along the equilibrium path due to the fact that the displacement of the cylinder domain is limited by the large nodal displacements of partially filled elements accounting for the majority of the allowed nodal displacements in each load step. Using the new arc-length approach also produces more consistent results when varying the initial MP displacements as seen in Figure 9 where for the case of $1/h = 10^1$, there is a significant difference between the calculated inner stretches with the traditional approach for the two MP distributions whereas the inner stretches for the new approach are much closer. As the mesh is refined, the difference in inner stretches for the two MP distributions reduce, producing virtually the same results while the traditional approach still shows a disparity in results for the two MP distributions. While both the traditional and new approaches produce the same equilibrium curve, it is clear that scaling nodal displacements results in a more efficient analysis which requires less load steps overall and allows for the use of coarser meshes and a reduction in the number of MPs. The benefits of scaling the nodal displacement in the arc-length constraint will also be shown in Section 4.3, including a load-displacement curve for the traditional and new arc-length approaches to demonstrate their respective progression along the equilibrium path.

4.3 | Double notched plate under tension

The third numerical example is the plane strain, elasto-plastic analysis of a deep double-edge-notched plate under tensile load to demonstrate the ability of the MP-BALM implementation to track a snap-through response when modelling an elasto-plastic material. The specimen has an overall width, $2w$, of 10 m and length, $2l$, of 30 m. Two deep notches are made along the horizontal midline of the plate creating a ligament with a width, $2b$, of 2 m. The geometry of the plate is shown in Figure 10A, which also shows a pressure, P, applied along the top and bottom edges of the plate. Due to the symmetry of the problem, one quarter of the geometry is modelled as shown in Figure 10A. The computational setup of the problem is shown in Figure 10B with roller boundary conditions applied along the side of the mesh where $x = 0$ m and only part of the section along the bottom of the mesh which corresponds to the ligament of the plate, $y = 0$ m and $x \leq 1$ m. The physical domain is discretised using 2×2 evenly spaced generalised interpolation material points within each background mesh element. The problem is analysed using an F-bar implementation[26] in order to reduce the effect of volumetric locking. The plate material is modelled using a Henky model with a von Mises plastic model using with Young's modulus, E, of 206.9 GPa, a Poisson's ratio, v, of 0.29, and a uniaxial yield stress, σ_y, of 0.45 GPa. An initial pressure of 50 MPa is applied by integrating the traction over a B-spline boundary generated along the top edge of the domain and the simulation is run until a top edge displacement of 0.2 m is achieved.

The simulation is run four times with multiple stages of mesh refinement, starting at a mesh size of 2^{-1} m and down to a mesh size of 2^{-4} m, with the mesh size reducing by a factor of 2 each time. Throughout each simulation, the reaction force along the nodes which are situated along the ligament (i.e., where the roller boundary conditions are applied along the bottom of the mesh) are calculated as the top edge of the plate is displaced due to the applied force. Letting R denote the total reaction force of the restrained edge of the ligament, the net vertical stress through the ligament is given by

$$\bar{\sigma} = \frac{R}{b}. \tag{36}$$

The normalised load-displacement response for this problem is shown in Figure 11. The snap-through behaviour is clearly shown for all mesh sizes with varying intensities, with only a slight snap-through behaviour seen for the mesh
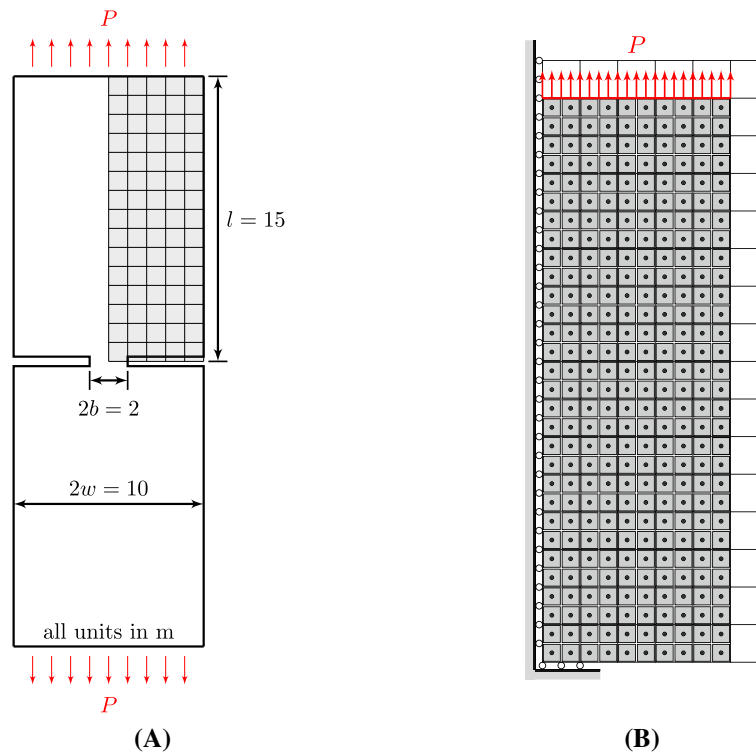

FIGURE 10   Double notched plate: (A) Problem geometry and (B) computational mesh and MP distribution.

size of 2^{-1} m. Also shown in Figure 11 is the Prandtl limit load[58] which is the analytical small strain limit load for this problem which is controlled by the stress at the notch in which $\sigma_{lim} \approx 2.97\sigma_y$. It is not until the mesh is refined to a size of 2^{-3} m that the load does not exceed the theoretical small strain load limit.

Due to the geometry of the problem, a singularity is formed at the end of the notch which is difficult to model accurately. As the mesh is refined, this singularity is amplified and causes the issue of MPs separating from the base. These points of separation can be seen in Figure 11 by the sudden changes in load-displacement response for mesh sizes 2^{-3} m (point A) and 2^{-4} m (points B and C). This is especially prominent for the mesh size of 2^{-4} m as this singularity causes the simulation to fail at a displacement of 0.15 m. Figure 12 shows the MP positions for the final load step of the simulation for each mesh size, the stress distribution is also shown. It is clear from Figure 12A,B that the singularity is not prominent as there is a smooth change in the MPs' vertical position moving away from the ligament. In Figure 12C the separation of the MPs away from the base can be seen as well as the increased separation between MPs around the singularity. Figure 12D shows the MP distribution for the end displacement of 0.15 m as this is the point at which the simulation fails, it is clear to see that large stress concentrations of the MPs at the end of the ligament as well as the disordering of the MPs around the singularity.

In order to demonstrate the need for special adaptation of the arc-length method for the MPM, the double notched plate problem is run with a mesh size of $2^{-1}m$ and an initial pressure of 10 MPa over the top edge of the domain. This problem is run twice, using the traditional arc-length constraint in (3) where there is no scaling of the nodal displacements and with the arc-length constraint in (25) in which the nodal displacements are scaled based on the basis function contributions of associated MPs. Figure 13 shows the normalised load-displacement curve for the simulation run over 100 load steps. It is clear that the case with the traditional arc-length constraint begins to stall at a vertical displacement of approximately 0.02 m. After this point, the progression along the equilibrium path slows such that the physical material of the plate shows only slight deformation over each load step. This is due to the fact that there are partially filled elements and extremely large nodal displacements are calculated for nodes that are weakly influenced by the MPs. Referring to (3), these large nodal displacements mean that the load factor increment must be very small in order to satisfy the arc-length constraint. On the other hand, by weighting the calculated nodal displacements by the strength of the node's association with the MP domain, using (25), the effect of these large nodal displacements are mitigated, allowing the simulation to progress along the equilibrium path much further than the unscaled case. In this case, no ghost stabilisation is utilised in order to solely demonstrate the effect of the scaling matrix with the new arc-length approach.

Figure 14 shows the normalised top edge displacement of the double notched plate after 100 load steps for varying mesh sizes and MP distributions with the same problem setup as above to compare the traditional and new arc-length approaches. In order to improve the stability of the analyses, a ghost stabilisation parameter of $\gamma = 2.069 \times 10^6$ is used (five orders of magnitude less than the Young's modulus of the material). As ghost stabilisation essentially incorporates extra stiffness around the boundary of the domain, the solution of the unknown nodal displacements is improved and a reduction in large nodal displacements is seen, therefore, the stalling seen in Figure 13 is less frequent. However, as seen in Figure 14, the new arc-length approach still results in more efficient analysis with further
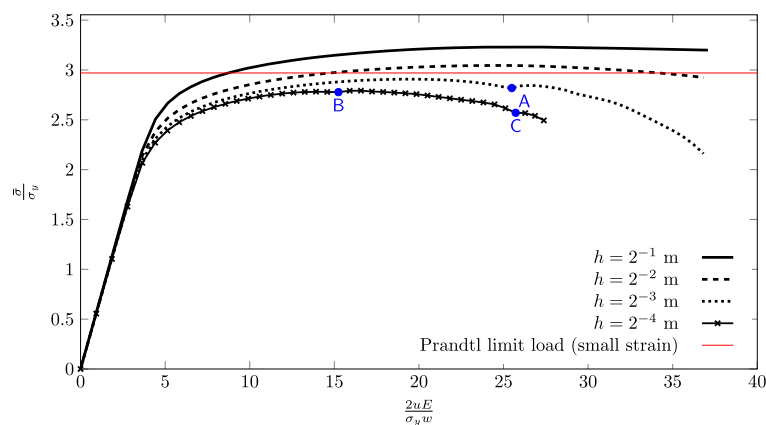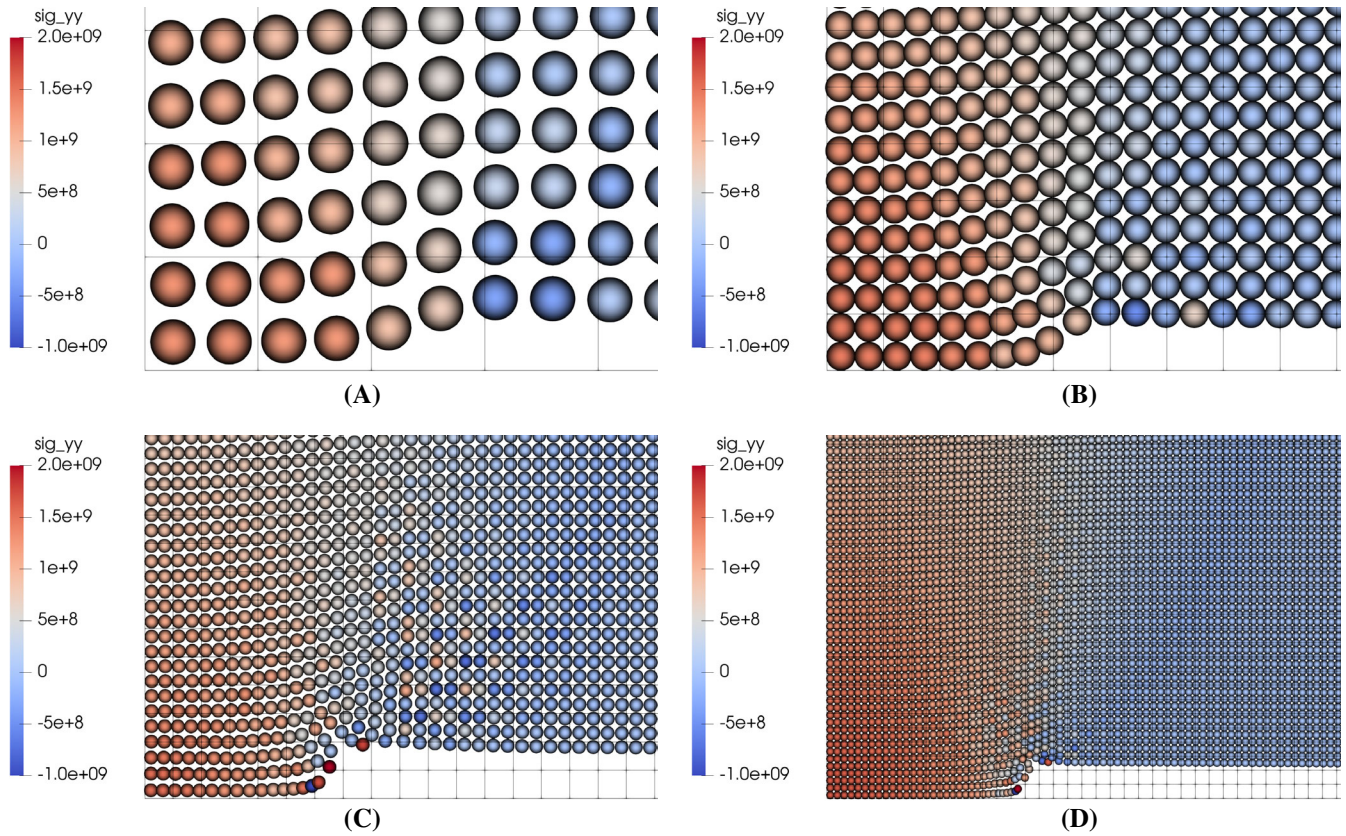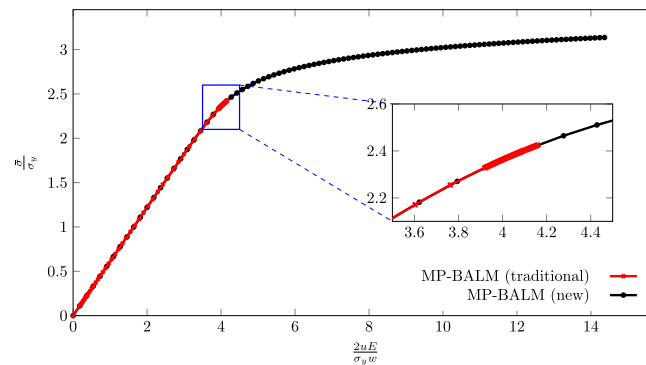

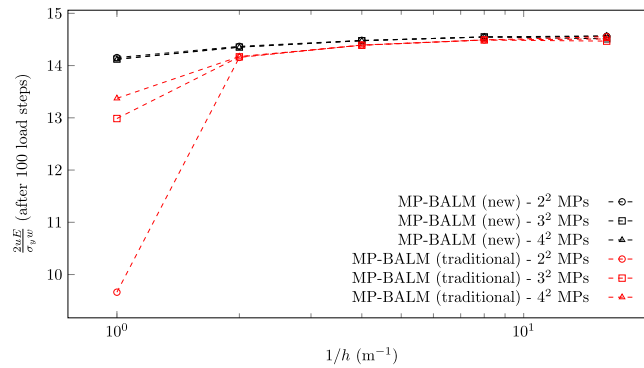**FIGURE 11** Double notched plate load-displacement curves for various mesh sizes.

FIGURE 12 MP positions and stress distribution around the notch singularity at the end of the simulation for mesh sizes of (A) 2^{-1} m, (B) 2^{-2} m, (C) 2^{-3} m and (D) 2^{-4} m.


FIGURE 13 Normalised load-displacement response over 100 load steps for double notched plate problem using the traditional arc-length approach and the new arc-length approach for a mesh size of 2^{-1} m.

progression along the equilibrium path for all mesh sizes and MP distributions compared to the traditional arc-length approach. It is not until a mesh size of 2^{-4} m that the traditional and new arc-length approaches produce the same displacement after 100 load steps. When using the new arc-length approach, the top edge displacement after 100 load steps is consistent for all initial MP distributions meaning that it is reasonable to reduce the overall number of MPs needed for the analysis compared to the traditional approach as well as having the option to increase the mesh size to reduce the overall runtime if desired. As with the hyperelastic cylinder example, the results shown in Figure 13 are not representative of the convergence to the exact solution with mesh refinement but a means to demonstrate the fact that the traditional arc-length approach is more sensitive to the problem discretisation compared to the new approach.

FIGURE 14 Double notched plate comparison of top edge displacement after 100 load steps for both traditional and new arc-length approaches for varying mesh sizes and MP distribution.
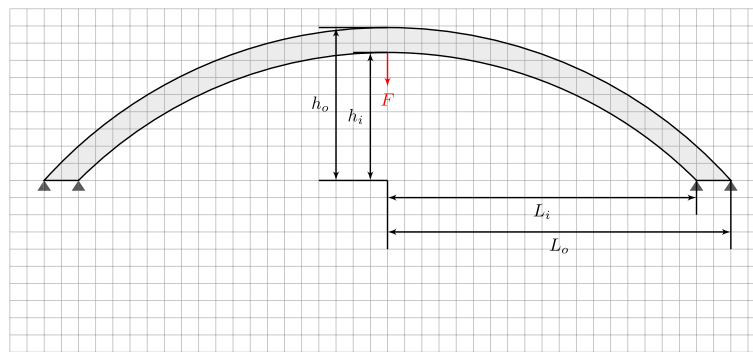

FIGURE 15 Problem geometry and basic mesh schematic.

4.4 | Arch with applied point load

The next numerical example is a 2D arch with a plane strain assumption subjected to a point load to demonstrate the ability to track the equilibrium path of snap-through behaviour using a force controlled scheme. This problem is akin to the Crisfield arch[1] also presented by Hrinda.[59] Here we adapt the arch to be modelled with continuum elements. A schematic of the problem geometry and mesh setup is given in Figure 15 in which the arch is formed with an inner length, L_i, and outer length, L_o, from the origin of 9.5 and 10 m respectively. The height of the inner arch wall, h_i, and the outer wall, h_o, are set as 4.09 and 4.45 m respectively. The inner and outer radius of the curves formed by the arch are then 13.09 and 13.45 m respectively. The mesh is generated such that the nodes of the mesh align with the base of the arch in order to apply fixed boundary conditions to the nodes coincident with the base of the arch. The mesh is also extended below the line of the base of the arch to allow for further displacement past the initial height of the arch. The arch is modelled as a Henky material with a Young's modulus of 100 kPa and a Poisson's ratio of 0.25. The mesh is fully populated with each element containing 4×4 evenly spaced generalised interpolation material points. The points which are not contained within the domain of the arch are then discarded in a similar fashion to the cylinder problem in Section 4.2. A mesh of square elements each with size 2^{-3} m is used for the simulation.

An initial point load of 100 N is applied at the centre of the inner edge of the arch (indicated by the red arrow in Figure 15), this load is applied evenly between the bottom-most MPs in the arch, one either side of the centre line. The simulation is run until the MPs which have the load applied to them have displaced by 8.2 m, taking a total of 139 load steps to complete. Figure 16 shows the load-displacement response at the MPs at which the load is applied. Four stages are highlighted along the equilibrium path, the unscaled MP deformations at each of these four points are shown in Figure 17.

From Figure 16 it is clear to see that there is a snap-through response as the arch is pulled down and through the baseline. A peak load is encountered at point (a) (at load step 19), where the centre of the arch has been pulled down and
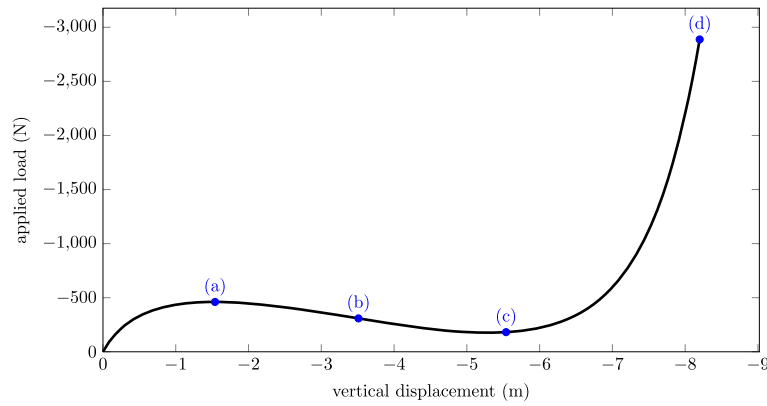
FIGURE 16   Load-displacement response of elastic arch with applied point load.
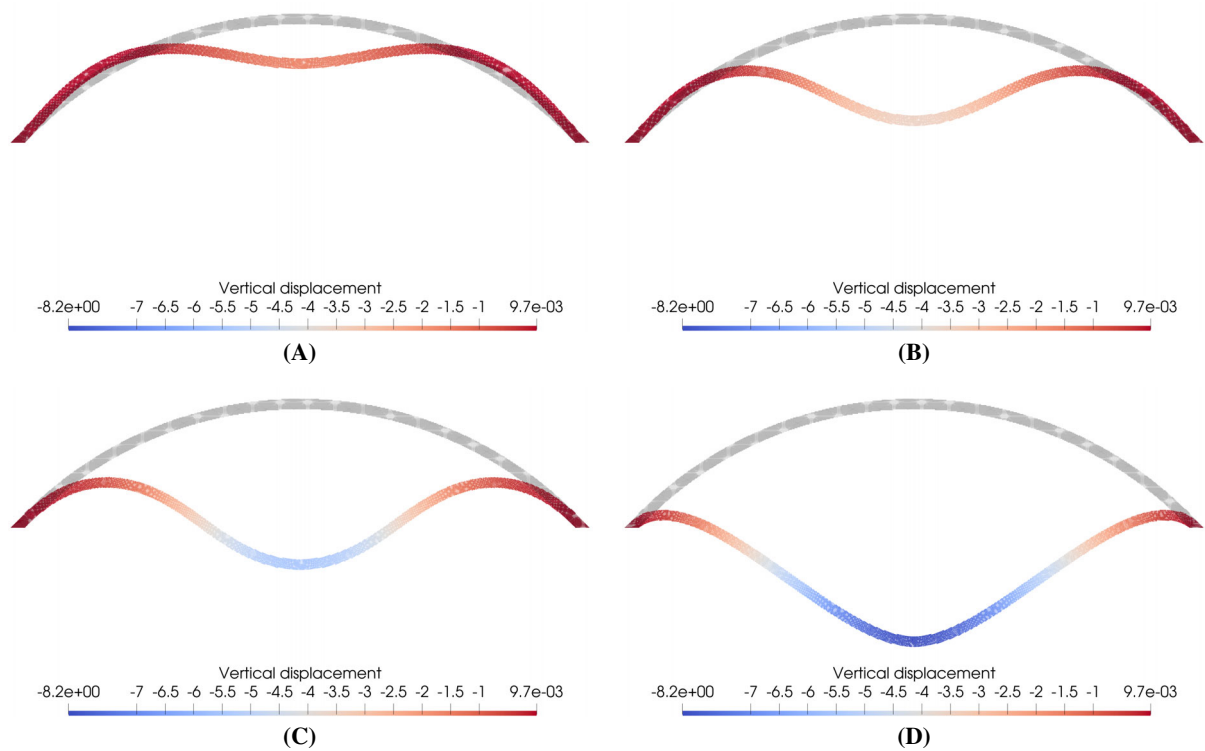

FIGURE 17   Unscaled MP deformation for shallow arch problem at stages (A) load step 19, (B) load step 48, (C) load step 84 and (D) load step 139 (see Figure 16 for position along equilibrium path).

bulging of the arch can be seen on the left and right sides of the arch. After this point, the snap-through response can be observed as the material softens and deforms through point (b) (at load step 48), is pulled through the base of the arch until the next stationary point at point (c) (at load step 84). After point (c), the material begins to stiffen again as the load required to further deform the arch begins to increase until the centre of the arch has displaced by 8.2 m at point (d) (at load step 139).

4.5 | Slope failure

The final numerical example examines the deformation of a 45° elasto-plastic, plane strain slope collapse due to gravitational loading. The geometry of the slope is shown in Figure 18 where roller boundary conditions are applied over the
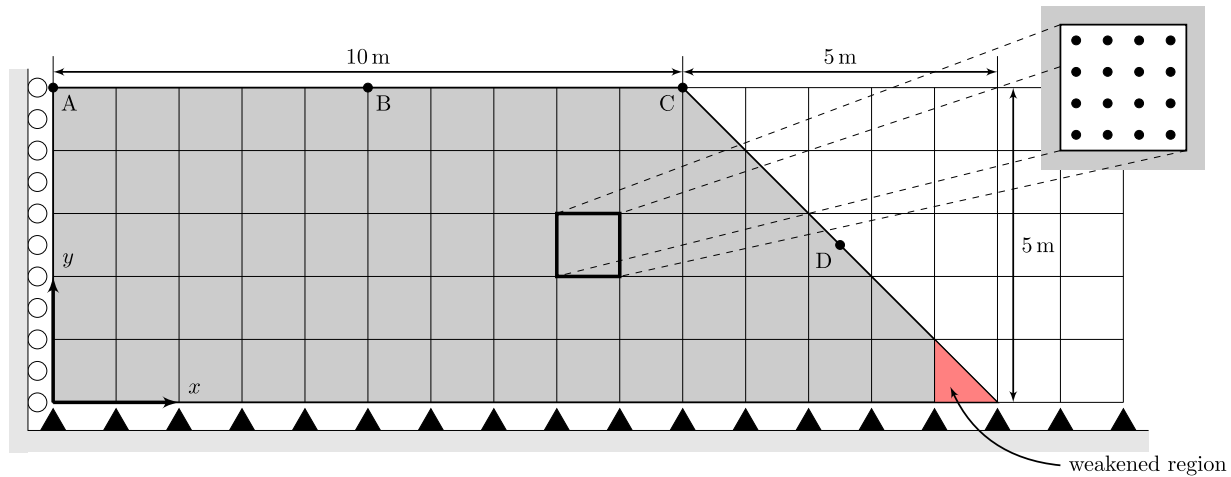
FIGURE 18 Slope geometry and boundary conditions (note that background mesh is for illustrative purposes only and does not represent the actual size of the mesh used in the analysis).

left edge of the domain and the bottom of the domain is fully fixed. The material is represented using a von Mises yield criterion with softening behaviour with a Young's modulus of 1 MPa, a Poisson's ratio of 0.33 and an initial von Mises yield stress of $\sigma_y^0 = 20$ kPa. The yield function is defined as

$$f = \sqrt{3J_2} - \sigma_y(\overline{\varepsilon}^{\mathrm{p}}) = 0, \tag{37}$$

where $\sigma_y(\overline{\varepsilon}^{\mathrm{p}}) = \sigma_y^0 + H\overline{\varepsilon}^{\mathrm{p}}$ is the current yield stress of the material, $\overline{\varepsilon}^{\mathrm{p}} = \sqrt{\frac{2}{3}\mathrm{tr}([\varepsilon^{\mathrm{p}}][\varepsilon^{\mathrm{p}}])}$ is the magnitude of the accumulated plastic strains, σ_y^0 is the initial yield stress, H is the plastic hardening (or softening) modulus, J_2 is the second invariant of the deviatoric stress

$$J_2 = \frac{1}{2}\mathrm{tr}([s][s]), \qquad \text{where} \qquad [s] = [\tau] - \frac{1}{3}\mathrm{tr}([\tau]) \tag{38}$$

and $[\tau]$ is the Kirchhoff stress. A strain softening modulus of $H = -50$ kPa‡ is used and a minimum von Mises yield stress of 13.3 kPa is set (the initial yield stress reduced by a factor of 1.5). A weaker region ($x \in [14.5, 15]$) at the toe of the slope is introduced, as shown by the red area in Figure 18, where the yield stress is reduced to $\sigma_y = 18$ kPa. The material has a uniform initial density of 2400 kg m^{-3}. It is worth noting that this problem is provided for the purpose of demonstrating the MP-BALM and is not representative of any real-life slope due to only using the J_2 yield criterion as the basis of the material failure behaviour.

The slope domain is generated by fully populating the background mesh (with a mesh size of $h = 2^{-2}$ m) with 4×4 evenly spaced generalised interpolation material points per element and removing MPs which are not contained within the desired area, giving a total of 16,040 MPs. The problem is run with both load control and the MP-BALM schemes with gravitational loading of $g = 9.81$ m s^{-2}. For both cases, the deformation of four positions on the slope are tracked through the analysis, the points A = (0, 5) m, B = (5, 5) m, C = (10, 5) m and D = (12.5, 2.5) m are shown in Figure 18. Ghost stabilisation is utilised for both cases with a ghost stabilisation parameter set to $\gamma = 1$ (six orders of magnitude less than the Young's modulus of the material). The problem is analysed using an F-bar implementation in order to reduce the effect of volumetric locking.

For the load controlled scheme, the gravitational load is applied over 160 equal load steps and the problem is run until convergence over a load step is not achieved over 10 Newton-Raphson iterations. This occurs after 79 load steps with no convergence seen in load step 80. Figure 19 shows the displacement magnitude of positions A, B, C and D as gravity is increasingly applied up to load step 79 and are shown as red crosses in each plot.

The problem is then run with the MP-BALM with an initial gravitational load of $g_0 = \frac{9.81}{160}$ ms^{-2}, which is equivalent to the initial gravitational load for the load controlled case. The analysis is run until position C has displaced by 0.5 m,
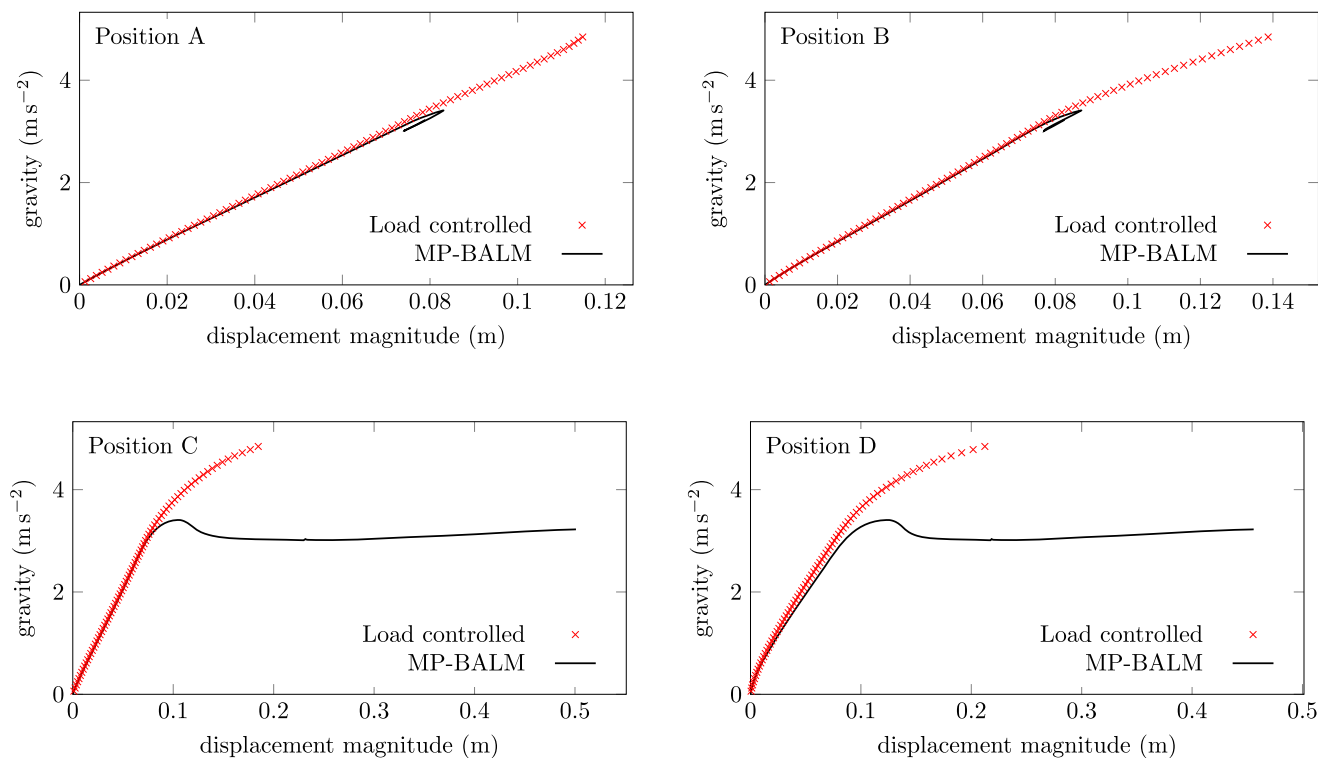
FIGURE 19 Load-displacement responses at specified positions around edge of slope domain throughout elasto-plastic collapse.

a significant increase from the maximum displacement seen by the load controlled case, which fails at a position C displacement magnitude of approximately 0.19 m. The MP-BALM analysis took a total of 477 load steps to achieve a position C displacement magnitude of 0.5 m. The load-displacement curves of the MP-BALM analysis are shown in Figure 19 and are depicted as solid black lines in each plot. It is worth reinforcing the point that the reported displacements in Figure 19 are the magnitudes of the displacements of the positions in question. In terms of local deformation, it is clear that positions A and B display local snap-back responses and positions C and D undergo snap-through responses. The difference in the load-displacement response between the arc-length and load controlled cases are stark and can be explained that at the point at which the paths begin to diverge that the material points will continue to displace with smaller increases in gravity than imposed by the load-controlled case. Implicit MPMs satisfy equilibrium using the state of the material points at the end of the current load step, where the stress carried by the material at their deformed positions are in equilibrium with the externally applied load. Once this equilibrium point has been found the points are then advected (displaced) through the mesh based on the incremental nodal displacements associated with the current step. However, once the material point positions are updated there is no guarantee that the stresses at the material points are still in equilibrium with the externally applied loads; this is the case for all MPMs, not just implicit MPMs or the implementation considered in this paper. This can be clearly seen by considering Equation (18) where the body forces are transferred to the nodes using the basis functions whereas the stress in the material is transferred using the derivatives of the basis functions and once the material points have changed position within the element it is likely, apart from for trivial cases, that there will be a non-zero residual at the start of the next step without increasing the imposed load. This non-zero residual will lead to additional material displacement in order to satisfy (18) without any increase in load. Increasing the load will lead to a different equilibrium path, as seen in Figure 19. The use of the MP-BALM ensures that the unbounded displacements seen by the load controlled schemes are contained due to the fact that the displacements of the domain are limited within each load step, this means that the simulation is stabilised around the critical point where the unbounded displacements begin.

Figure 20 shows the deformed configurations of the slope collapse analysis using a load controlled scheme (left column) and the MP-BALM (right column). For each analysis the final shear stress (top) and yield stress (bottom) states are shown. It is clear from Figure 20 that the MP-BALM results in far greater overall displacement with a larger failure
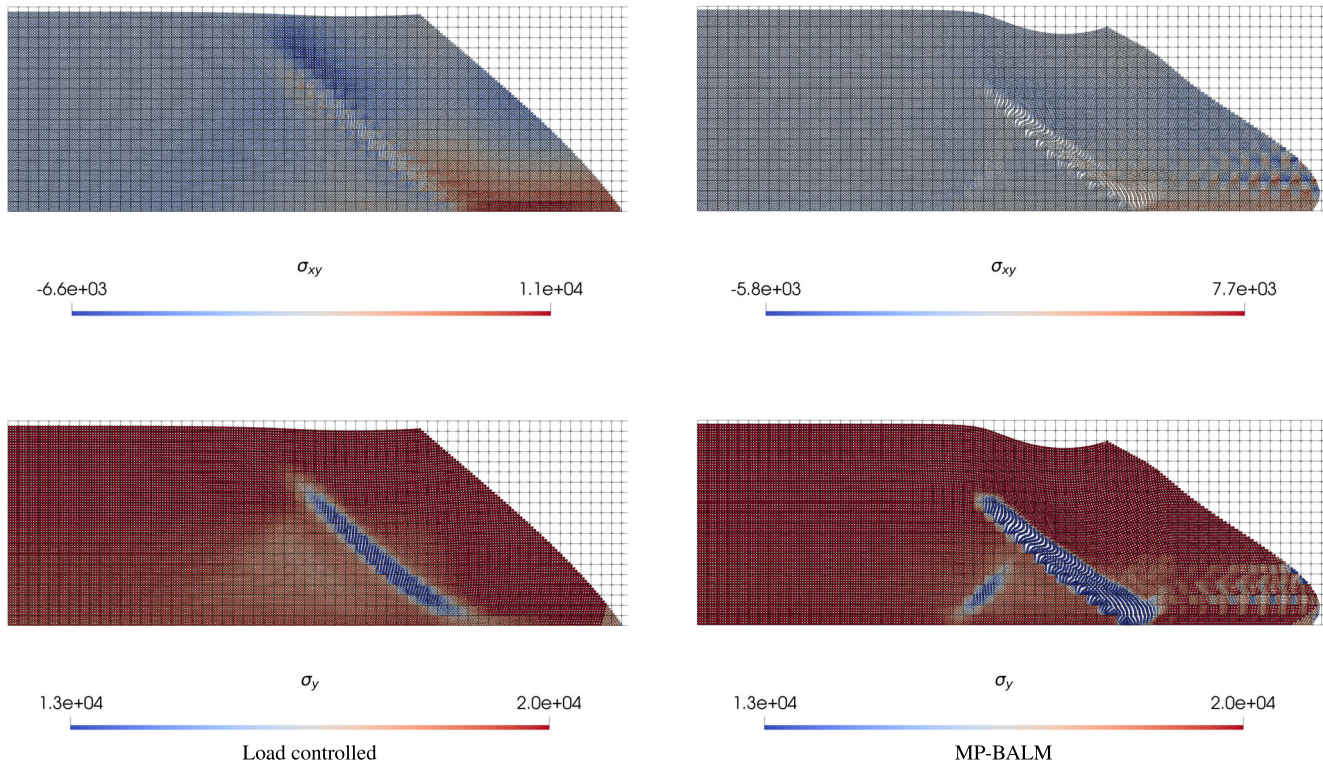
FIGURE 20 Final deformed configurations of the slope collapse problem for the load controlled analysis (left column) and MP-BALM analysis (right column). For each analysis, the shear stress (top row) and yield stress (bottom row) is shown.

band being seen in the material and importantly at a lower load compared to the load-controlled case, identifying a more critical failure mode for the slope.

5 | CONCLUSION

This paper presents an arc-length method for the MPM framework to expand the physical problems and loading scenarios that the MPM can solve, notably those involving snap-through and snap-back responses. Simply implementing the arc-length method into the MPM as it is given in the FEM approach is not suitable due to the fact that partially filled elements are a common feature in a MPM simulation. These partially filled elements mean that elements around the boundary of the domain will be poorly integrated, understiff and have the potential to experience extremely large nodal displacements. While the effect of the large nodal displacements are reduced when mapping nodal displacements to the MP positions, the effects are seen within the arc-length constraint as the nodal displacements account for the majority of the allowed displacement based on the prescribed arc length. Therefore, the arc-length constraint is reformulated such that it is based on MP displacements, essentially scaling the nodal displacements based on the node's influence on the physical domain. As the MPM involves the resetting of the background mesh at the end of each load step, nodal displacement information is lost due to the fact that the MPs have changed positions and the nodal displacements cannot simply be carried over from the previous load step. This means that in order to use the secant path predictor, an estimate for the nodal displacements for the previous load step based on the current MP positions are needed. This nodal displacement reconstruction is carried out at the start of each load step and uses a least squares weighted residual technique.

A simple benchmark test of a 2D column under self weight is given to show that the MP-BALM is able to model standard problems that the MPM with a linear solver has been proven to solve. The expansion of a hyperelastic cylinder which demonstrates a snap-through response is shown and a comparison was made between the traditional arc-length scheme and the MP-BALM scheme presented in this paper. It is clear that through scaling the nodal displacement increments, a

more robust and efficient analysis is achieved. More complex problems are shown to demonstrate that the MP-BALM can model problems with more complex deformation and model snap-through responses with problems of a double notched plate under tension, the deflection of a shallow arch and collapse of a slope including material softening. The final case has demonstrated the ability of the method to control the displacement response of a gravity-driven problem with unbounded deformation, typically of geotechnical slope collapse, which is not possible with conventional load controlled quasi-static methods.

As is the case for many arc-length papers in the current literature, the arc-length method has only been presented for the problems with load control. The case of displacement controlled problems is much more involved with an alternative arc-length constraint equation needed, please see Pretti et al.[16] for more details on the formulation of a displacement controlled arc-length method.

DATA AVAILABILITY STATEMENT
All data created during this research are openly available at https://collections.durham.ac.uk/files/r2g158bh325 (DOI: http://doi.org/10.15128/r2g158bh325). For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

ENDNOTES
*Naming convention taken from the literature, see Crisfield.[7] The Newton-Raphson displacement is the vector resulting from the elimination of the out-of-balance forces, while the tangential displacement is the vector tangential to the Newton-Raphson displacement on the load-displacement curve.

†Please note that $\bar{\lambda}$ refers to stretch is not the same as λ which denotes the load factor for the arc-length method. The Greek letter λ is conventionally used in both cases and a distinction has been made by the authors in the hope to avoid confusion.

‡A negative H value represents softening behaviour.

ORCID
Nathan D. Gavin https://orcid.org/0000-0003-1572-8283
Giuliano Pretti https://orcid.org/0000-0003-2756-9718
William M. Coombs https://orcid.org/0000-0003-2099-1676

REFERENCES
1. Crisfield MA. *Non-linear Finite Element Analysis of Solids and Structures*. first ed. John Wiley & Sons; 1991.
2. de Souza Neto EA, Perić D, Owen DRJ. *Computational Methods for Plasticity: Theory and Applications*. Wiley; 2008.
3. Riks E. The application of Newton's method to the problem of elastic stability. *J Appl Mech*. 1972;39(4):1060-1065.
4. Riks E. An incremental approach to the solution of snapping and buckling problems. *Int J Solids Struct*. 1979;15(7):529-551.
5. Wempner GA. Discrete approximations related to nonlinear theories of solids. *Int J Solids Struct*. 1971;7(11):1581-1599.
6. Memon B, Su X. Arc-length technique for nonlinear finite element analysis. *J Zhejiang Univ Sci*. 2004;5(5):618-628.
7. Crisfield MA. A fast incremental/iterative solution procedure that handles "snap-through". *Comput Struct*. 1981;13:55-62.
8. Bergan PG, Horrigmoe G, Kråkeland B, Søreide TH. Solution techniques for non-linear finite element problems. *Int J Numer Methods Eng*. 1978;12(11):1677-1696.
9. Ramm E. Strategies for tracing the nonlinear response near limit points. In: Wunderlich W, Stein E, Bathe KJ, eds. *Nonlinear Finite Element Analysis in Structural Mechanics*. Springer; 1981:63-89.
10. Powell G, Simons J. Improved iteration strategy for nonlinear structures. *Int J Numer Methods Eng*. 1981;17:1455-1467.
11. Lam WF, Morley CT. Arc-length method for passing limit points in structural calculation. *J Struct Eng*. 1992;118(1):169-185.
12. Fafard M, Massicotte B. Geometrical interpretation of the arc-length method. *Comput Struct*. 1993;46(4):603-615.
13. Krenk S. An orthogonal residual procedure for non-linear finite element equations. *Int J Numer Methods Eng*. 1995;38(5):823-839.
14. Nicholson DW. Stiff arc length constraint in nonlinear FEA. *Acta Mech*. 2005;175:123-137.
15. May S, Vignollet J, de Borst R. A new arc-length control method based on the rates of the internal and the dissipated energy. *Eng Comput*. 2016;33(1):100-115.
16. Pretti G, Coombs WM, Augarde CE. A displacement-controlled arc-length solution scheme. *Comput Struct*. 2022;258:106674.

17. Kweon JH, Hong CS. An improved arc-length method for postbuckling analysis of composite cylindrical panels. *Comput Struct*. 1994;53(3):541-549.

18. Ning X, Pellegrino S. Imperfection-insensitive axially loaded thin cylindrical shells. *Int J Solids Struct*. 2015;62:39-51.

19. White SC, Weaver PM. Towards imperfection insensitive buckling response of shell structures-shells with plate-like post-buckled responses. *Aeronaut J*. 2016;120(1224):233-253.

20. Kadapa C. A simple extrapolated predictor for overcoming the starting and tracking issues in the arc-length method for nonlinear structural mechanics. *Eng Struct*. 2021;234:111755.

21. Al-Rasby SN. Solution techniques in nonlinear structural analysis. *Comput Struct*. 1991;40(4):985-993.

22. Foster S. An application of the arc length method involving concrete cracking. *Int J Numer Methods Eng*. 1992;33(2):269-285.

23. Wang Y, Waisman H. An arc-length method for controlled cohesive crack propagation using high-order XFEM and Inwin's crack closure integral. *Eng Fract Mech*. 2018;199:235-256.

24. Alfano G, Crisfield MA. Solution strategies for the delamination analysis based on a combination of local-control arc-length and line searches. *Int J Numer Methods Eng*. 2003;58(7):999-1048.

25. Carol I, López CM, Roa O. Micromechanical analysis of quasi-brittle materials using fracture-based interface elements. *Int J Numer Methods Eng*. 2001;52(1–2):193-215.

26. Coombs WM, Charlton TJ, Cortis M, Augarde CE. Overcoming volumetric locking in material point methods. *Comput Methods Appl Mech Eng*. 2018;333:1-21.

27. Kardani M, Nazem M, Sheng D, Carter JP. Large deformation analysis of geomechanics problems by a combined rh-adaptive finite element method. *Comput Geotech*. 2013;49:90-99.

28. Sulsky D, Chen Z, Schreyer HL. A particle method for history-dependent materials. *Comput Methods Appl Mech Eng*. 1994;118:179-196.

29. Sulsky D, Zhao SJ, Schreyer HL. Application of a particle-in-cell method to solid mechanics. *Comput Phys Commun*. 1995;87:236-252.

30. Brackbill JU, Ruppel HM. FLIP: a method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions. *J Comput Phys*. 1986;65(2):314-343.

31. Harlow FH. The particle-in-cell computing method for fluid dynamics. *Methods Comput Phys*. 1964;3:319-343.

32. Sulsky D, Kaul A. Implicit dynamics in the material-point method. *Comput Methods Appl Mech Eng*. 2004;193(12):1137-1170.

33. Phuong NTV, van Tol AF, Elkadi ASK, Rohe A. Numerical investigation of pile installation effects in sand using material point method. *Comput Geotech*. 2016;73:58-71.

34. Stomakhin A, Schroeder C, Chai L, Teran J, Selle A. A material point method for snow simulation. *ACM Trans Graph*. 2013;32(4):1-10.

35. Liang W, Zhao J. Multiscale modeling of large deformation in geomechanics. *Int J Numer Anal Methods Geomech*. 2019;43(5):1080-1114.

36. Nairn JA. Material point method calculations with explicit cracks. *Comput Model Eng Sci*. 2003;4(6):649-663.

37. Wang B, Vardon PJ, Hicks MA, Chen Z. Development of an implicit material point method for geotechnical applications. *Comput Geotech*. 2016;71:159-167.

38. Coombs WM, Augarde CE. AMPLE: a material point learning environment. *Adv Eng Softw*. 2020;139:102748.

39. Hellweg HB, Crisfield MA. A new arc-length method for handling sharp snap-backs. *Comput Struct*. 1998;66(5):704-709.

40. Bergan PG, Søreide TH. Solution of large displacement and instability problems using the current stiffness parameter. In: Bergan PG, ed. *Finite Elements in Nonlinear Mechanics*. Vol 2. Tapir; 1978:647-669.

41. Feng YT, Perić D, Owen DRJ. Determination of travel directions in path-following methods. *Math Comput Model*. 1995;21(7):43-59.

42. Feng YT, Perić D, Owen DRJ. A new criterion for dertermination of initial loading parameter in arc-length methods. *Comput Struct*. 1996;58(3):479-485.

43. Bardenhagen SG, Kober EM. The generalized interpolation material point method. *Comput Model Eng Sci*. 2004;5(6):477-496.

44. Charlton TJ, Coombs WM, Augarde CE. iGIMP: an implicit generalised interpolation material point method for large deformations. *Comput Struct*. 2017;190:108-125.

45. Zhang X, Chen Z, Liu Y. *The Material Point Method: A Continuum-Based Particle Method for Extreme Loading Cases*. Elsevier; 2017.

46. Fern E, Rohe A, Soga K, Alonso E. *The Material Point Method for Geotechnical Engineering: A Practical Guide*. Taylor & Francis; 2019.

47. Nguyen VP, de Vaucorbiel A, Bordas S. *The Material Point Method: Theory*. Springer; 2023.

48. Bing Y, Cortis M, Charlton TJ, Coombs WM, Augarde CE. B-spline based boundary conditions in the material point method. *Comput Struct*. 2019;212:257-274.

49. Steffen M, Kirby RM, Berzins M. Analysis and reduction of quadrature errors in the material point method (MPM). *Int J Numer Methods Eng*. 2008;76(6):922-948.

50. Steffen M, Wallstedt PC, Guilkey JE, Kirby RM, Berzins M. Examination and analysis of Implentation choices within the material point method (MPM). *Comput Model Eng Sci*. 2008;31(2):107-128.

51. Burman E. Ghost penalty (La pénalisation fantôme). *Compt Rendus Math*. 2010;348(21–22):1217-1220.

52. Sticko S, Ludvigsson G, Kreiss G. High-order cut finite elements for the elastic wave equation. *Adv Comput Math*. 2020;46(45):1-28.

53. Coombs WM. Ghost stabilisation of the material point method for stable quasi-static and dynamic analysis of large deformation problems. *Int J Numer Methods Eng*. 2023;124(21):4841-4875.

54. de Souza Neto EA, Perić D, Dutko M, Owen DRJ. Design of Simple low Order Finite Elements for large strain analysis of nearly incompressible solids. *Int J Solids Struct*. 1996;33(22-22):3277-3296.

55. de Souza Neto EA, Andrade Pires FM, Owen DRJ. F-bar based linear triangles and tetrahedra for finite strain analysis of nearly incompressible solids. Part I: formulation and benchmarking. *Int J Numer Methods Eng*. 2005;62(3):353-383.

56. Ogden RW. Large deformation isotropic elasticity-on the correlation of theory and experiment for incompressible rubberlike solids. *Proc R Soc London. A. Math Phys Sci.* 1972;326(1567):565-584.

57. Haughton DM, Ogden RW. Bifurcation of inflated circular cylinders of elastic material under axial loading-II. Exact theory for thick-walled tubes. *J Mech Phys Solids.* 1979;27(5–6):489-512.

58. Nagtegaal JC, Parks DM, Rice JR. On numerically accurate finite element solutions in the fully plastic range. *Comput Methods Appl Mech Eng.* 1974;4(2):153-177.

59. Hrinda GA. Geometrically nonlinear static analysis of 3D trusses using the arc-length method. *WIT Transactions on Modelling and Simulation.* Vol 46. WIT Press; 2007:243-252.