

RESEARCH ARTICLE

An implicit locking-free B-spline Material Point Method for large strain geotechnical modelling

Mian Xie¹  | Pedro Navas²  | Susana López-Querol¹ 

¹Department of Civil, Environmental and Geomatic Engineering, University College London, London, UK

²ETSI Caminos, Canales y Puertos, Universidad Politécnica de Madrid, Madrid, Spain

Correspondence

Susana López-Querol, Department of Civil, Environmental and Geomatic Engineering, University College London, London, UK.
Email: s.lopez-querol@ucl.ac.uk

Funding information

China Scholarship Council, Grant/Award Number: 202008520044

Abstract

The Material Point Method (MPM) has drawn great attention in the numerical modelling of large deformation, geotechnical problems. The popularity of MPM is mainly because its formulation shares significant similarities with the Finite Element Method. In MPM, the iteration points can move independently from the mesh, allowing for the resolution of large deformation problems. However, because of this, the original MPM formulation suffers from the well-known cell-crossing noise and volumetric-locking instabilities, resulting in a strongly oscillated stress field. A novel implicit locking-free B-spline MPM that controls stress oscillations to a negligible level is proposed in this paper. A novel, but very straightforward B-spline shape function implementation procedure, avoids the need for a complex material point searching algorithm, providing seamless transformation from the original MPM to this robust B-spline MPM, aiming at modelling large-strain geotechnical problems. The newly proposed volumetric locking mitigation strategy is also very easy to implement, which facilitates the reproducibility of this research. The proposed method is validated against three numerical studies: granular column collapse experiment, slope failure and footing with large penetration. The proposed numerical method agrees well with experiments reported in the literature and previous numerical studies. Also, these numerical examples show that the proposed method provides a more prominent stress field than other available methodologies.

KEYWORDS

B-spline MPM, F-bar, foundations, landslides, volumetric locking stabilisation

1 | INTRODUCTION AND LITERATURE REVIEW

From the design point of view, civil engineers are often interested in small deformation problems because it is feasible to undertake the analysis of structures that are not likely to experience high strains. The Finite Element Method (FEM) and the Finite Difference Method (FDM) are the most commonly used numerical approaches in the civil engineering industry due to their accuracy for small deformation problems and the ability to describe the history-dependent material behaviour.

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial-NoDerivs](https://creativecommons.org/licenses/by-nc-nd/4.0/) License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

© 2023 The Authors. *International Journal for Numerical and Analytical Methods in Geomechanics* published by John Wiley & Sons Ltd.

However, geotechnical engineering involves numerous large deformation problems, such as landslides and penetration of piles during their installation, among others.

Soga et al.¹ comprehensively reviewed the available numerical methods for large deformation in geotechnical modelling. The Material Point Method (MPM) is recommended due to the following reasons: (a) it is suitable for large-scale and large deformation problems; (b) its computational efficiency; (c) it is based on a similar mathematical formulation as for the FEM; (d) its ability to deal with the history-dependent soil constitutive models and (e) the simplicity for applying essential boundary conditions.¹ Some other comprehensive reviews of numerical modelling for large deformation in geotechnics can be found in Refs.^{2–5}

In MPM, the material domain is discretised in material points that can be treated as integration points in FEM.⁶ To solve the distorted mesh issue in FEM, the mesh is set to its original position in MPM at each time step, and the material points carry all the permanent information (e.g. stress, displacement and velocity). As a result, the material points (integration points) can move independently from the mesh.⁶ In fact, the MPM will yield to FEM if the mesh is not set to its original position after every time step.⁷

Aimed for computational efficiency, Sulsky et al.⁸ use linear FEM shape functions (hat functions) in the original MPM. The use of these shape functions in MPM causes well-known cell crossing instability when a material point crosses the cell boundary. The Generalised Interpolation MPM (GIMPM) developed by Bardenhagen and Kober⁹ is one of the most commonly used MPM methods in the literature due to its efficiency. However, it cannot fully eliminate the cell crossing instability¹⁰ since for that it is necessary to utilise a higher-order shape function that covers multiple cells.

The available higher-order MPMs are the B-spline MPM (BSMPM)¹¹ and the Local Maximum-Entropy MPM.¹² Both methods can entirely eliminate the cell crossing noise. BSMPM is chosen for this research due to its better efficiency. The latter approach requires a Newton–Raphson loop to construct the Local Maximum-Entropy shape function.^{13–15} However, the construction of the B-spline shape function still requires much more computational effort than the original MPM and the GIMPM shape function. Additionally, the conventional implementation of a higher-order shape function requires a particle-searching algorithm to know where the material points and their neighbours are located.¹⁵ Therefore, it is more appropriate to use an implicit iteration algorithm rather than an explicit one for the MPM with higher-order shape functions, since an implicit solver allows relatively larger time steps and the shape function only needs to be constructed once for each step. Also, due to this feature, the implicit solver is more preferable for relatively slow simulations, especially for the quasi-static analysis. Apart from utilising the higher-order shape function, the Moving Least Squares (MLS) shape function is also usually used to mitigate the cell crossing noise.^{16–18} In the current literature, the MLS shape function has only been used in the explicit MPM solver, and the implicit MLS–MPM still needs to be studied.

Volumetric locking is the other source of instability in MPM besides the cell crossing noise. The volumetric locking will result in over-stiff behaviour and non-physical stress oscillations. In FEM, the volumetric locking instability can be easily overcome by a reduced quadrature scheme.¹⁹ However, utilising the reduced quadrature scheme is not feasible in MPM for two reasons: (a) a sufficient number of material points per cell is required to ensure the desired quadrature accuracy; (b) it is difficult to limit the number of material points per cell since they can move freely between them.⁶ The available volumetric locking mitigation strategies^{6,20–27} are mainly suitable for the MPM with linear shape functions (e.g. original MPM and GIMPM). The BSMPM shows less volumetric locking than the MPMs with linear shape functions. However, fully overcoming the volumetric locking for BSMPM is challenging since its shape functions are associated with adjacent cells.²⁸ Navas et al.²⁹ mitigated the volumetric locking in the Optimal Transportation Meshfree (OTM) Method, a numerical method based on Local Maximum-Entropy shape functions. This approach relies on the triangular mesh, however, the rectangular mesh is a more popular choice in the MPM literature. To the best of our knowledge, there are currently only three available pieces of research^{28,30,31} related to the mitigation of volumetric locking for a higher-order MPM. Two of them^{30,31} are based on the F-bar projection method proposed by Elguedj et al.,³² which was originally developed for B-spline FEM. However, the application of this method is not straightforward due to the introduction of an additional background grid with low-order shape functions.²⁸ Also, it is hard to apply to other types of MPM (i.e. GIMPM) because of this feature. The method proposed by Zhao et al.²⁸ is simple enough, but its performance has not been well tested by a wide range of large deformation geotechnical problems.

This paper proposes a total locking-free, implicit BSMPM implementation to overcome the previously mentioned research gaps. This newly proposed method is simple, as does not require the implementation of any sophisticated particle search algorithm. The application of the newly proposed volumetric locking migration strategy is also very straightforward and can be easily implemented. Additionally, the proposed method has been validated with various large deformation geotechnical problems.

This paper is structured as follows: Section 2 explains the proposed numerical model in detail, including the construction of B-spline shape functions, large strain formulation of the constitutive model, volumetric locking migration strategy

and the proposed implicit BSMPM algorithm. Then, we will present some verification numerical examples for large deformation geotechnical problems in Section 3: the granular column collapse, slope failures and the large penetration of strip footing problem. Finally, we present the conclusions of this research in Section 4.

2 | DESCRIPTION OF THE NUMERICAL MODEL

2.1 | BSMPM shape function: A new efficient searching algorithm

In MPM, the kinematic quantities (e.g. displacement, velocity and acceleration) at any material point can be approximated by interpolating the nodal quantities of the related grid using the shape functions. For example, the incremental displacement of a material point ($\Delta \mathbf{u}_p$) can be approximated from the incremental nodal displacement of related grid nodes (\mathbf{u}_I) by

$$\Delta \mathbf{u}_p = \sum_{I=1}^{n_n} N_{Ip} \Delta \mathbf{u}_I, \quad (1)$$

where n_n is the number of grid nodes that influences the material point. In this research, the subscript p means that the variables are related to the material point. Similarly, the subscript I denotes that the variables are related to the grid nodes. N_{Ip} is the shape function associated with node I evaluated at the position of material point p . In the original MPM, the employed shape function is the traditional hat function; however, different shape functions are used for improved MPMs, such as the B-spline function. Also, the total nodal displacements are equivalent to the incremental nodal displacements in MPM once a brand-new grid has been used for each time step.

The communication between nodes and material points is not one-way only. The information can also be mapped from the material points to the related node. For example, the diagonal components of lumped nodal mass (m_I) can be calculated by taking into account the contribution of all the material points related to this node:

$$m_I = \sum_{p=1}^{n_p} N_{Ip} m_p, \quad (2)$$

where n_p is the number of material points associated with node I , and m_p is the mass of the material point p .

BSMPM follows the exactly same computational procedures as the original MPM except for utilising a different shape function N_{Ip} .³³ Like the other MPM shape functions, we can first construct the B-spline shape functions in 1-D and then convolute them in multi-dimensions.^{34–37} A 1-D BSMPM shape function is usually constructed based on a knot vector, Ξ , which contains a series of knots with non-decreasing order for their spatial coordinates:

$$\Xi = \{x_1, x_2, \dots, x_{n+q}, x_{n+q+1}\}, \quad (3)$$

where n is the total number of B-spline shape functions; q denotes the polynomial order and $x_i (i = 1, 2, \dots, n + q + 1)$ is the i th knot.³⁶ These knots can be treated as grid nodes in the FEM or MPM. However, in the case of a boundary grid node, the corresponding boundary knot needs to be repeated q times. As we can see from the definition of the knot vector, at least $q + 2$ knots are required to construct a B-spline shape function. In this research, we use a quadratic ($q = 2$) B-spline shape function; therefore, at least four knots are needed to construct a B-spline shape function. In the case of a boundary grid, only two grid nodes are available. To have four knots, its boundary knot needs to be repeated twice.

Figure 1 graphically illustrates the 2-D discretisation of quadratic BSMPM. Figure 1A shows a discretisation of a rectangular continuum body using 16 material points. This continuum body moves in a domain (parametric grid) uniformly discretised into 12 patches (four patches in the x -direction and three patches in the y -direction) with a constant unit spacing. The entire parametric grid can be represented by two open global knot vectors: $\Xi_x = \{0, 0, 0, 1, 2, 3, 4, 4, 4\}$ and $\Xi_y = \{0, 0, 0, 1, 2, 3, 3, 3\}$ in x - and y -directions, respectively. The knot vectors are open because their first and last knot appear $q + 1$ times. As shown in Figure 1A, these knots appear three times for a quadratic B-spline, and these repeated knots are the boundary knots. Figure 1B shows a 2-D tensor product grid formed by the tensor product of the knot vectors in the x - and y -directions. In BSMPM, the information is mapped from the material points to the tensor product grid nodes.³⁷ Then, the partial differential equations are solved in the tensor product grid. Finally, all the permanent

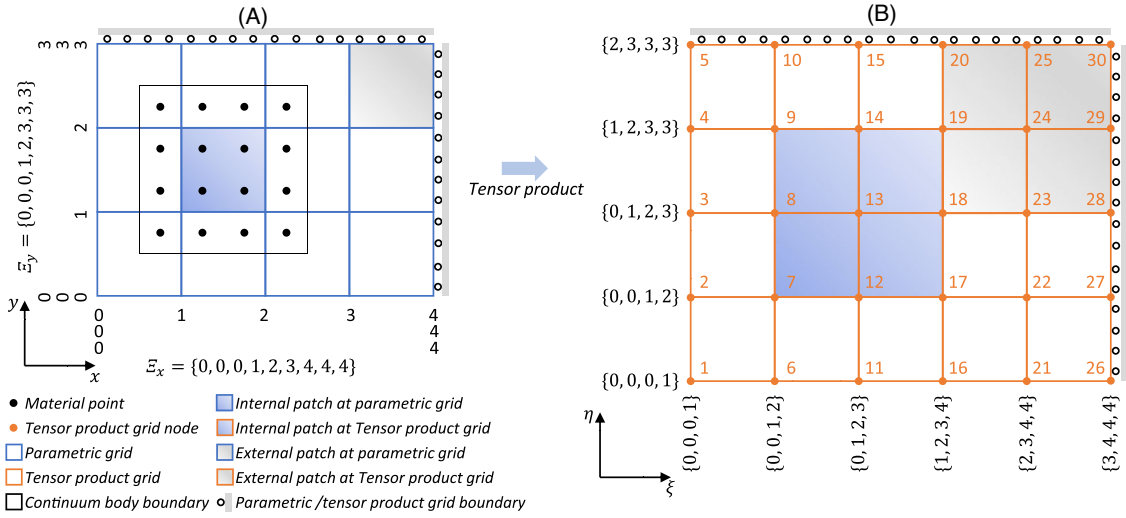


FIGURE 1 Graphical illustration of 2-D quadratic B-spline MPM discretisation: (A) parametric grid, (B) tensor product grid.

information is mapped back to the material points, and the tensor product grid is set to its original position. As shown in Figure 1, the parametric grid and the tensor product grid only coincide at the boundary grid nodes where the Dirichlet boundary conditions apply.³⁶

We use an internal patch (blue hatched in Figure 1) which contains four material points, as an example to illustrate the construction of the BSMPM shape function. To construct shape functions for each material point inside this patch, we need to obtain sub-knot vectors from the global knot vectors Ξ_x and Ξ_y . As shown in Figure 1A, this internal patch sits on the second span of the parametric grid in both directions. Therefore, we need to obtain sub-knot vectors from the second knot in the global knot vectors Ξ_x and Ξ_y . For each patch, there are $q + 1$ shape functions in each direction, and $q + 2$ knots are required to construct a shape function. As a result, the sub-knot vectors $\{0, 0, 1, 2\}$ and $\{0, 1, 2, 3\}$ in the x -direction and $\{0, 0, 1, 2\}$ and $\{1, 2, 3, 4\}$ in the y -direction are obtained by reading four knots three times one by one from the second knot in the global knot vectors.

Given the coordinate of a material point (x) and a knot vector ($\Xi = \{x_1, x_2, \dots, x_{n+q}, x_{n+q+1}\}$), we can use the Cox-de Boor formula³⁸ to determine the shape functions recursively starting from polynomial order zero ($q = 0$):

$$N_{i,0}(x) = \begin{cases} 1 & \text{if } x_i \leq x < x_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

In the case of polynomial order larger than zeros ($q > 0$):

$$N_{i,q}(x) = \frac{x - x_i}{x_{i+q} - x_i} N_{i,q-1}(x) + \frac{x_{i+q+1} - x}{x_{i+q+1} - x_{i+1}} N_{i+1,q-1}(x). \quad (5)$$

The shape function derivatives can be obtained using the following:

$$\frac{dN_{i,q}(x)}{dx} = \frac{q}{x_{i+q} - x_i} N_{i,q-1}(x) - \frac{q}{x_{i+q+1} - x_{i+1}} N_{i+1,q-1}(x). \quad (6)$$

Note that the above Cox-de Boor recursion formula assumes $0/0 = 0$.

As a result, there are three shape functions (i.e. $i = 3$) for each material point in each direction. After the tensor product, there are nine (three times three) shape functions evaluated at the tensor product grid nodes for each material point. In other words, one material point is governed by nine tensor product grid nodes. These four material points in the hatched internal patch are governed by the same tensor product grid nodes (i.e. grid nodes number 7, 8, 9, 12, 13, 14, 17, 18 and 19), as shown in Figure 1B. Therefore, their material point (p) to tensor product grid nodes (N) connectivity matrix ($p2N$ matrix) is [7, 8, 9, 12, 13, 14, 17, 18, 19]. Similarly, if a material point moves to the hatched external patch, its $p2N$ matrix will be [18, 19, 20, 23, 24, 25, 28, 29, 30], as shown in Figure 1B. This $p2N$ matrix is crucial in MPM during the communication between the material points and their related grid nodes (in BSMPM, they are tensor product grid nodes). Summations

over the grid nodes and material points require this $p2N$ matrix to know the relationship between the material points and grid nodes, as shown in Equations (1) and (2). The conventional method constructs the BSMPM shape functions patch by patch. Utilising a particle search algorithm, we can determine the material point (p) to patch/element (e) connectivity matrix ($p2e$). Since the patch (e) to tensor product grid nodes (N) connectivity matrix ($e2N$) is an inherent property of a tensor product grid, we can obtain the $p2N$ matrix through $p2e$ and $e2N$ (i.e. material point to patch to tensor product grid nodes).

This research proposes a new implementation to avoid the searching algorithm. The basic idea is constructing shape functions for the entire parametric grid instead of patches. As shown in Figure 1A, for each material point, we can construct its B-spline shape functions for the entire background mesh instead of each patch using only two global knot vectors Ξ_x and Ξ_y . Using the Cox-de Boor formula, we can obtain the shape functions evaluated at every tensor product grid node for each material point. However, these shape functions are non-zero only at the patch where the interested material point locates. Because of this property, we can eliminate the zero shape functions to obtain the exact same shape functions as constructing them patch by patch. Also, we can find and record the locations (indices) of these non-zero shape functions. These indices are the components of the $p2N$ matrix for this material point. Finally, this method yields the same shape functions and $p2N$ matrix as the conventional patch-by-patch method with a particle searching algorithm. The algorithm of the proposed method for a 2-D problem is shown in Appendix A.

A quadratic B-spline is used in this research because of its efficiency and capability of fully overcoming the cell crossing noise.¹¹ Although a cubic B-spline shape function results in a more accurate result,³⁷ it is more cumbersome compared with the quadratic one because of the larger matrix size. The construction of a cubic B-spline shape function follows the same recursion procedure (Equations 4–6) and the proposed methodology can be easily extended to the employment of such a higher-order shape function.

In the computer graphics community,^{39,40} the explicitly defined B-spline shape function is usually used. This type of B-spline shape function is derived recursively assuming a uniform grid space and internal zero-centred material point.¹¹ The employment of this type of B-spline shape function is very efficient because the recursion procedure is avoided during construction. However, implementing this explicitly defined B-spline shape function requires ghost nodes (i.e. additional nodes) outside the boundary to ensure the partition of unity.⁴¹ The presence of these ghost nodes introduces massive numerical errors around the boundaries. In the case of MPM application in computer graphics,^{39,40} the material body is usually not placed against the boundaries. However, the massive numerical errors around the boundaries are not acceptable for an engineering problem because the symmetric boundary is frequently used to reduce the computational effort, for example, the strip footing problem in geotechnical engineering (Section 3.3). Therefore, it is more appropriate to use a knot vector to construct the B-spline shape functions recursively (Equations 4–6) to avoid introducing ghost nodes.

2.2 | Large strain constitutive framework

In MPM, the constitutive model is evaluated at the material point level. Therefore, for the notations in this section, we omit the subscript p for clarity purposes.

In the large strain constitutive model, the left elastic Cauchy–Green strain \mathbf{b}^e is stored as a state variable, and it is updated every time step from t to $t + dt$ ⁴²:

$$\mathbf{b}^e(t + dt) = \Delta \mathbf{F} \mathbf{b}^e(t) \Delta \mathbf{F}^T, \quad (7)$$

where $\Delta \mathbf{F}$ is the deformation gradient increment. The deformation gradient, \mathbf{F} , can be written as

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \frac{\partial (\mathbf{X} + \mathbf{u})}{\partial \mathbf{X}} = \mathbf{I} + \frac{\partial \mathbf{u}}{\partial \mathbf{X}}. \quad (8)$$

One of the most commonly used strain measurements for the large strain formulation is logarithmic (also called Hencky strain tensor), and, therefore, the elastic logarithmic strain $\boldsymbol{\epsilon}^e$ is defined by⁴²

$$\boldsymbol{\epsilon}^e = \frac{1}{2} \ln \mathbf{b}^e. \quad (9)$$

Note that the evaluation of Equation (9) requires polar decomposition since \mathbf{b}^e is a matrix. The corresponding stress measure is the Kirchhoff stress $\boldsymbol{\tau}$ defined by

$$\boldsymbol{\tau} = \mathbf{D}^e \boldsymbol{\epsilon}^e, \quad (10)$$

where \mathbf{D}^e is the linear elastic consistent tangent in Voigt notation. In the MPM formulation, the Cauchy stress $\boldsymbol{\sigma}$ is still used.⁴³ The Kirchhoff stress $\boldsymbol{\tau}$ has only been used in the constitutive model. The Cauchy stress $\boldsymbol{\sigma}$ can be recovered from the Kirchhoff stress $\boldsymbol{\tau}$ by

$$\boldsymbol{\sigma} = \frac{\boldsymbol{\tau}}{\det(\mathbf{F})} = \frac{\boldsymbol{\tau}}{J}, \quad (11)$$

where $J = \det(\mathbf{F})$ is the Jacobian of the deformation gradient.

With a continuum-based method, such as the FEM or the MPM, the stress needs to be updated along with the incremental displacement. Usually, the elastic predictor return mapping algorithm is used to update the stress in an elastoplastic constitutive model. Utilising the logarithmic strain and Kirchhoff stress allows us to use the same elastic predictor return mapping algorithm derived from the small strain theory.⁴² In this research, we use the Mohr–Coulomb constitutive model as documented by de Souza Neto et al.,⁴² who also report the return mapping algorithm in detail.

2.3 | Overcoming volumetric locking: A new F-bar approach

This research employs the F-bar Method to overcome volumetric locking. This method is originally proposed by de Souza Neto et al.⁴⁴ to avoid the volumetric locking in FEM. Coombs et al.⁶ summarised the advantages of the F-bar Method as: (a) it does not introduce any additional unknown or unphysical parameter; (b) it is possible to apply to any type of constitutive model and MPM; (c) it does not require additional material points to capture the volumetric behaviour. Same as the previous section, the subscript p is omitted for the notations in this section. The notation with no subsection means that this value is evaluated at the material point level.

The idea of the F-bar Method is based on the multiplicative split of the deformation gradient \mathbf{F}

$$\mathbf{F} = \mathbf{F}_d \mathbf{F}_v, \quad (12)$$

where

$$\mathbf{F}_d = \det(\mathbf{F})^{-1/dof} \mathbf{F} \quad \text{and} \quad \mathbf{F}_v = \det(\mathbf{F})^{1/dof} \mathbf{I} \quad (13)$$

are the deviatoric and volumetric parts of the deformation gradient, respectively.⁴⁴ \mathbf{I} is the identity matrix. dof is the spatial dimensions of the problem (e.g. $dof = 2$ for a 2-D problem).

In the F-bar Method, the volumetric locking is resolved by replacing the deformation gradient \mathbf{F} in the constitutive model with a modified deformation gradient $\bar{\mathbf{F}}$.⁴⁴ In this modified deformation gradient, the volumetric part \mathbf{F}_v of \mathbf{F} is replaced by $(\mathbf{F}_0)_v$, a volumetric part of $\bar{\mathbf{F}}$ that has less volumetric constraint⁴⁴

$$\bar{\mathbf{F}} = \mathbf{F}_d (\mathbf{F}_0)_v = \left[\frac{\det(\mathbf{F}_0)}{\det(\mathbf{F})} \right]^{1/dof} \mathbf{F}. \quad (14)$$

In this research, we use the incremental version of the F-bar Method since our constitutive model is driven by the incremental deformation gradient shown in Equation (7). Therefore, the incremental version of the F-bar Method can be obtained by rewriting Equation (14)⁶:

$$\Delta \bar{\mathbf{F}} = \Delta \mathbf{F}_d (\Delta \mathbf{F}_0)_v = \left[\frac{\det(\Delta \mathbf{F}_0)}{\det(\Delta \mathbf{F})} \right]^{1/dof} \Delta \mathbf{F} = \left(\frac{\Delta \bar{J}}{\Delta J} \right)^{1/dof} \Delta \mathbf{F}, \quad (15)$$

where $\Delta J = \det(\Delta \mathbf{F})$ is the Jacobian of the deformation gradient increment; $\Delta \bar{J}$ is the averaged Jacobian that has less volumetric constraint compared to ΔJ . By replacing $\Delta \mathbf{F}$ by $\Delta \bar{\mathbf{F}}$ in Equation (7), the volumetric locking can be overcome.⁶

From the above illustration, the key to the F-bar approach is to determine the averaged Jacobian $\Delta \bar{J}$ in Equation (15). Departing from the research conducted by Zhao et al.,²⁸ we calculate the averaged Jacobian $\Delta \bar{J}$ by mapping the original one ΔJ to the related grid nodes and then mapping back to the material points. The mathematical formulation of this mapping and remapping procedure is stated as follows:

(1) The nodal mass-weighted Jacobian can be calculated by mapping the product of mass and ΔJ from material points as

$$(m\Delta J)_I = \sum_{p=1}^{n_p} N_{Ip} m_p \Delta J, \quad (16)$$

where m_p is the mass of material point p ; $(m\Delta J)_I$ is the nodal mass-weighted Jacobian.

(2) Then, the nodal Jacobian can be obtained by

$$\Delta J_I = \frac{(m\Delta J)_I}{m_I}, \quad (17)$$

where $m_I = \sum_{p=1}^{n_p} N_{Ip} m_p$ is the nodal mass.

(3) Finally, the nodal Jacobian is mapped back to the material points via

$$\Delta \bar{J} = \sum_{I=1}^{n_n} N_{Ip} \Delta J_I. \quad (18)$$

Although the above F-bar Method can overcome the volumetric locking, the spurious stress oscillation cannot be entirely cured in MPM. In this research, in addition to modifying $\Delta \mathbf{F}$, we apply the same mapping re-mapping technique to the last converged left elastic Cauchy–Green strain $\mathbf{b}^e(t)$. As a result, the specious stress oscillation can be cured by averaging both $\Delta \mathbf{F}$ and $\mathbf{b}^e(t)$. We name this method *the modified F-bar Method*. In this method, Equation (7) is replaced by

$$\mathbf{b}^e(t + dt) = \Delta \bar{\mathbf{F}} \bar{\mathbf{b}}^e(t) (\Delta \bar{\mathbf{F}})^T, \quad (19)$$

where $\bar{\mathbf{b}}^e$ is the averaged left elastic Cauchy–Green strain, which is obtained by

$$\bar{\mathbf{b}}^e = \sum_{I=1}^{n_n} N_{Ip} \left(\frac{1}{m_I} \sum_{p=1}^{n_p} N_{Ip} m_p \mathbf{b}^e \right). \quad (20)$$

Note that Equation (20) represents the same mass-weighted mapping and re-mapping procedure as Equations (16)–(18) but in a compact format. Applying Equation (20) only (i.e. without utilising the F-bar) can also mitigate some form of volumetric locking. However, similarly to solely applying the F-bar Method, the stress oscillation cannot be controlled at a satisfactory level.

This mass-weighted mapping and re-mapping technique has been applied in the post-processing stage to smooth the contour map of stresses in MPM.^{45,46} The rationale behind using this mapping and re-mapping technique is that the velocity field of the original MPM is stable despite the highly oscillated stress field because the velocity of a material point has been mass projected to the grids and mapped back to the material point in an MPM computational circle. However, applying this technique only in the post-processing stage cannot prevent the un-physical oscillation of the stress field.

The idea of our modified F-bar Method is similar to the method proposed by Zhao et al.,²⁸ but these are different methods because (a) we use a different incremental version of the F-bar Method; (b) we apply the mapping and re-mapping technique to the term ΔJ instead of J and (c) we also smooth the left elastic Cauchy–Green strain. These modifications improve the algorithm's performance under extremely large deformation situations, which will be shown in the later numerical examples. A reader may refer to Zhao et al.²⁸ for a detailed explanation of the original method.

2.4 | Proposed implicit B-spline MPM algorithm

This research follows the implicit MPM algorithm proposed by Guilkey and Weiss,⁷ based on the unconditionally stable Newmark integration. This implicit algorithm was designed for the original MPM. We tailored this implicit MPM algorithm to fit our proposed methods provided in the previous sections. In the original implicit MPM algorithm,⁷ the Newton–Raphson Method is adopted to solve the non-linear equations. However, in this research, the quasi-Newton

Method proposed by Broyden⁴⁷ is used. It allows neglecting the linearisation of the stiffness contribution due to the proposed F-bar Method. Therefore, quadratic convergence cannot be achieved. Instead, the employment of the aforementioned quasi-Newton Method results in a larger number of iterations but less computational time for each iteration. The following algorithm describes a loading step from time t to $t + dt$ for the proposed implicit locking-free BSMPM.

(1) *Shape functions and topology*

Given the coordinates of material points $\mathbf{x}_p(t)$ and the global knot vectors, construct the B-spline shape functions N_{Ip} and derivatives with respect to global coordinates at the start of a loadstep ∇N_{Ip} using the algorithm mentioned in Appendix A.

(2) *Map information from material point to tensor product grid nodes*

(a) The nodal mass m_I can be obtained by

$$m_I = \sum_{p=1}^{n_p} N_{Ip} m_p.$$

(b) The nodal velocity $\mathbf{v}_I(t)$ can be obtained by

$$\mathbf{v}_I(t) = \frac{1}{m_I} \sum_{p=1}^{n_p} N_{Ip} \mathbf{v}_p(t) m_p.$$

(c) The nodal acceleration $\mathbf{a}_I(t)$ can be obtained by

$$\mathbf{a}_I(t) = \frac{1}{m_I} \sum_{p=1}^{n_p} N_{Ip} \mathbf{a}_p(t) m_p,$$

where $\mathbf{a}_p(t)$ is the material point acceleration.

(d) The nodal external force $\mathbf{Fext}_I(t + dt)$ can be obtained by

$$\mathbf{Fext}_I = \sum_{p=1}^{n_p} N_{Ip} \mathbf{Fext}_p(t + dt),$$

where $\mathbf{Fext}_p(t + dt)$ is the material point external force.

(3) *Initilisation of the quasi-Newton iteration*

(a) Set the number of iterations $k = 0$.

(b) Use the last converged nodal displacement as the initial guess: $\mathbf{u}_I^k(t + dt) = \mathbf{u}_I^k(t)$.

(4) *Update the permanent information at each material point*

(a) Given $\mathbf{u}_I^k(t + dt)$ calculate the incremental deformation gradient by

$$\Delta \mathbf{F}_p^k(t + dt) = \mathbf{I} + \sum_{I=1}^{n_n} \mathbf{u}_I^k(t + dt) \otimes \nabla N_{Ip},$$

(b) Given $\Delta \mathbf{F}_p^k(t + dt)$ and last converged deformation gradient from the last time step $\mathbf{F}_p(t)$, update the deformation gradient using

$$\mathbf{F}_p^k(t + dt) = \Delta \mathbf{F}_p^k(t + dt) \mathbf{F}_p(t).$$

(c) Compute the determinants of the deformation gradient and incremental deformation gradient by

$$J_p^k(t + dt) = \det[\mathbf{F}_p^k(t + dt)] \text{ \& } \Delta J_p^k(t + dt) = \det[\Delta \mathbf{F}_p^k(t + dt)].$$

(d) Update the volume of material point:

$$V_p(t + dt) = \Delta J_p^k(t + dt) V_p(t)$$

(e) Compute the averaged Jacobian increment:

$$\Delta J_p^k(t + dt) = \sum_{I=1}^{n_n} N_{Ip} \left[\frac{1}{m_I} \sum_{p=1}^{n_p} N_{Ip} m_p \Delta J_p^k(t + dt) \right].$$

(f) Compute the incremental F-bar by

$$\Delta \bar{\mathbf{F}}_p^k(t + dt) = \left[\frac{\Delta \mathbf{F}_p^k(t + dt)}{\Delta J_p^k(t + dt)} \right]^{1/dof} \Delta \mathbf{F}_p^k(t + dt).$$

(g) Given the incremental F-bar $\Delta \bar{\mathbf{F}}_p^k(t + dt)$ and the last converged averaged left elastic Cauchy–Green strain $\bar{\mathbf{b}}_p^e(t)$, update the left elastic Cauchy–Green strain $\mathbf{b}_p^{e,k}(t + dt)$ by

$$\mathbf{b}_p^{e,k}(t + dt) = \Delta \bar{\mathbf{F}}_p^k(t + dt) \bar{\mathbf{b}}_p^e(t) \Delta \bar{\mathbf{F}}_p^k(t + dt)^T$$

(h) Evaluate the elastic logarithmic strain $\boldsymbol{\epsilon}_p^{e,k}(t + dt)$ by

$$\boldsymbol{\epsilon}_p^{e,k}(t + dt) = \frac{1}{2} \ln \mathbf{b}_p^{e,k}(t + dt)$$

(i) Utilising the return-mapping algorithm, update the Kirchhoff stress $\boldsymbol{\tau}_p^k(t + dt)$ and the left elastic Cauchy–Green strain $\mathbf{b}_p^{e,k}(t + dt)$ based on $\boldsymbol{\epsilon}_p^{e,k}(t + dt)$. Then, restore the Cauchy stress $\boldsymbol{\sigma}_p^k(t + dt)$ by

$$\boldsymbol{\sigma}_p^k(t + dt) = \frac{\boldsymbol{\tau}_p^k(t + dt)}{J_p^k(t + dt)}$$

(5) *Obtain global out-of-balance force matrix*

(a) The global nodal internal force $\mathbf{Fint}_I^k(t + dt)$ is obtained by

$$\mathbf{Fint}_I^k(t + dt) = \sum_{p=1}^{n_p} \mathbf{G}^T \boldsymbol{\sigma}_p^k(t + dt) V_p(t + dt),$$

where $\mathbf{G} = \nabla_x N_{Ip} = \nabla N_{Ip} [\Delta \mathbf{F}_p^k(t + dt)]^{-1}$ is the strain–displacement matrix.

(b) The out-of-balance force $\mathbf{Foob}_I^{k+1}(t + dt)$ is calculated by

$$\mathbf{Foob}_I^{k+1}(t + dt) = \mathbf{Fext}_I(t + dt) - \mathbf{Fint}_I^k(t + dt) - \mathbf{M}_I \left[\frac{4}{dt^2} \mathbf{u}_I^k(t + dt) - \frac{4}{dt} \mathbf{v}_I(t) - \mathbf{a}_I(t) \right]$$

where \mathbf{M}_I is the nodal mass matrix. The lumped nodal mass $\mathbf{M}_I = \text{diag}(m_I)$ is typically used, which is a diagonal matrix.

(6) Obtain global nodal stiffness matrix

if $k = 0$, get the initial guess of the global stiffness matrix by

$$\mathbf{K}_I^k(t + dt) = \frac{4}{dt^2} \mathbf{M}_I + \sum_{p=1}^{n_p} \mathbf{G}^T \mathbf{D}_p^{e,k}(t + dt) \mathbf{G} V_p(t + dt)$$

where, $\mathbf{D}_p^{e,k}(t + dt)$ is the elastic tangent modulus of a material point.

if $k > 0$, estimate the global stiffness matrix by

$$\mathbf{K}_I^k(t + dt) = \mathbf{K}_I^k(t) - \frac{\mathbf{Foob}_I^{k+1}(t + dt) \delta \mathbf{u}_I^k}{\delta \mathbf{u}_I^k \delta \mathbf{u}_I^k}$$

where, $\delta \mathbf{u}_I^k$ is the nodal displacement increment from the last quasi-Newton iteration.

(7) Solve the linear equation

At quasi-Newton iteration number k , the nodal displacement increment $\delta \mathbf{u}_I^{k+1}$ can be obtained by solving the following linear equation with Dirichlet boundary conditions

$$\mathbf{K}_I^k(t + dt) \cdot \delta \mathbf{u}_I^{k+1} = \mathbf{Foob}_I^{k+1}(t + dt)$$

(8) Update grid nodal displacement

$$\mathbf{u}_I^{k+1}(t + dt) = \mathbf{u}_I^k(t + dt) + \delta \mathbf{u}_I^{k+1}$$

(9) Check convergency

$$\text{If } f^{k+1} = \frac{\|\delta \mathbf{u}_I^{k+1} \mathbf{Foob}_I^{k+1}\|}{\|\delta \mathbf{u}_I^{k+1} \mathbf{Foob}_I^{k+1}\|} < \text{tol}$$

go to step (10). Otherwise, set $k = k + 1$ and go to step (4).

(10) Update the grid kinematics

(a) The nodal acceleration $\mathbf{a}_I(t + dt)$ can be updated by

$$\mathbf{a}_I(t + dt) = \frac{4}{dt^2} \mathbf{u}_I^{k+1}(t + dt) - \frac{4}{dt} \mathbf{v}_I(t) - \mathbf{a}_I(t)$$

(b) The nodal velocity $\mathbf{v}_I(t + dt)$ can be updated by

$$\mathbf{v}_I(t + dt) = \frac{2}{dt} \mathbf{u}_I^{k+1}(t + dt) - \mathbf{v}_I(t)$$

(11) Save the converged information into the material points

(a) The spatial coordinates of material points $\mathbf{x}_p(t + dt)$ can be updated by

$$\mathbf{x}_p(t + dt) = \mathbf{x}_p(t) + \sum_{I=1}^{n_n} N_{Ip} \mathbf{u}_I^{k+1}(t + dt)$$

(b) The displacement of material points $\mathbf{u}_p(t + dt)$ can be updated by

$$\mathbf{u}_p(t + dt) = \mathbf{u}_p(t) + \sum_{I=1}^{n_n} N_{Ip} \mathbf{u}_I^{k+1}(t + dt)$$

(c) The acceleration of material points $\mathbf{a}_p(t + dt)$ can be updated by

$$\mathbf{a}_p(t + dt) = \sum_{I=1}^{n_n} N_{Ip} \mathbf{a}_I(t + dt)$$

(d) The velocity of material points $\mathbf{v}_p(t + dt)$ can be updated by

$$\mathbf{v}_p(t + dt) = \mathbf{v}_p(t) + \frac{\mathbf{a}_p(t) + \mathbf{a}_p(t + dt)}{2} dt$$

(e) Given the updated left elastic Cauchy–Green strain $\mathbf{b}_p^{e,k}(t + dt)$, compute the averaged left elastic Cauchy–Green strain $\bar{\mathbf{b}}_p^e(t + dt)$ by

$$\bar{\mathbf{b}}_p^e(t + dt) = \sum_{I=1}^{n_n} N_{Ip} \left[\frac{1}{m_I} \sum_{p=1}^{n_p} N_{Ip} m_p \mathbf{b}_p^{e,k}(t + dt) \right]$$

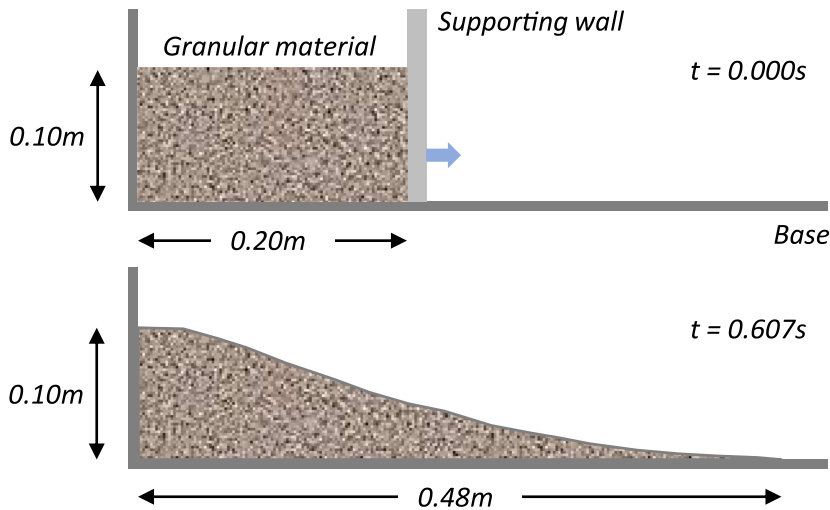
(f) Finally, save all the converged permanent variables like $\mathbf{F}_p(t + dt)$ and $V_p(t + dt)$, then go to the next time step.

3 | NUMERICAL EXAMPLES

Both dynamic and quasi-static large deformation numerical examples are included in this research to illustrate the performance of the new method proposed in this paper. We set the mass matrix to zero for the quasi-static analysis, and hence the dynamic implicit algorithm automatically becomes quasi-static.⁷ As mentioned before, the Mohr–Coulomb constitutive model has been applied in this research. The third numerical example is based on the Tresca yield criterion, which is a special case of Mohr–Coulomb with zero friction and dilation angles. Also, the soil is assumed as weightless for the third

TABLE 1 Material properties for three numerical examples.

Numerical example Section	Unit weight γ [kN/m ³]	Young's modulus E [kPa]	Poisson's ratio ν	Friction angle ϕ [°]	Cohesion c [kPa]
3.1	20.4	5840	0.3	21.9	0
3.2	20	100,000	0.3	20	10
3.3	0	100	0.495	0	1

**FIGURE 2** Graphical illustration of granular column collapse experiment.

example. The material input parameters for these numerical models are summarised in Table 1. The dilation angle for these three examples is zero. The simulations are conducted using an in-house developed MATLAB code. The VTK files are generated during the simulations, and the post-processing of these VTK files is performed using the software ParaView.

3.1 | Granular column collapse

The granular column collapse experiment was initially designed to study natural catastrophe granular flow propagation.⁴⁸ This research uses a granular column collapse experiment documented by Nguyen et al.⁴⁹ in order to validate the proposed numerical model. Nguyen et al.⁴⁹ use a Smoothed Particle Hydrodynamics (SPH) Method with Mohr–Coulomb constitutive model to validate this experiment, where aluminium rods are used to represent the granular soil. The material properties of these aluminium rods were calibrated using a series of experimental tests,⁴⁹ summarised in Table 1.

Figure 2 graphically illustrates this granular column collapse experiment. Initially, the granular materials were set to be rectangular, with 0.1 m in height and 0.2 m in width.⁴⁹ Then, the granular flow was initialised by suddenly removing the supporting wall. The evolution of the granular flow was recorded using a high-speed camera. Finally, the granular flow became stationary at 0.607 s after removing the supporting wall.

The aluminium rods have neglectable movement in the z -direction (in page direction), which allows us to model it under a 2-D plane strain condition. In the 2-D numerical model, the supporting wall that blocks the aluminium bars shown in Figure 2 has not been modelled. The gravity acceleration is applied at the initial time step since the wall was removed quickly during the experiment.⁴⁹ The width and height of the background mesh (parametric grid) are set to be 550 and 102.5 mm, respectively. Both the width and height for each parametric grid cell (patch) are 2.5 mm. Thirty-six (6^2) material points are set in each grid cell to minimise the quadrature error. As a result, the aluminium rods have been discretised into 115,200 material points. The tensor product grid is formed from the parametric grid. The fixed and roller boundary conditions are applied on the bottom and sides of the grid, respectively.

Viscous damping was applied by Nguyen et al.⁴⁹ to stabilise their SPH-based model. In this research, no numerical damping is applied because the energy loss during the rotation of the aluminium rods is negligible.⁵⁰ Finally, the time increment is set as 0.00025 s, resulting in 2428 steps in total. The tolerance for the Newton–Raphson iteration is set to be 10^{-8} .

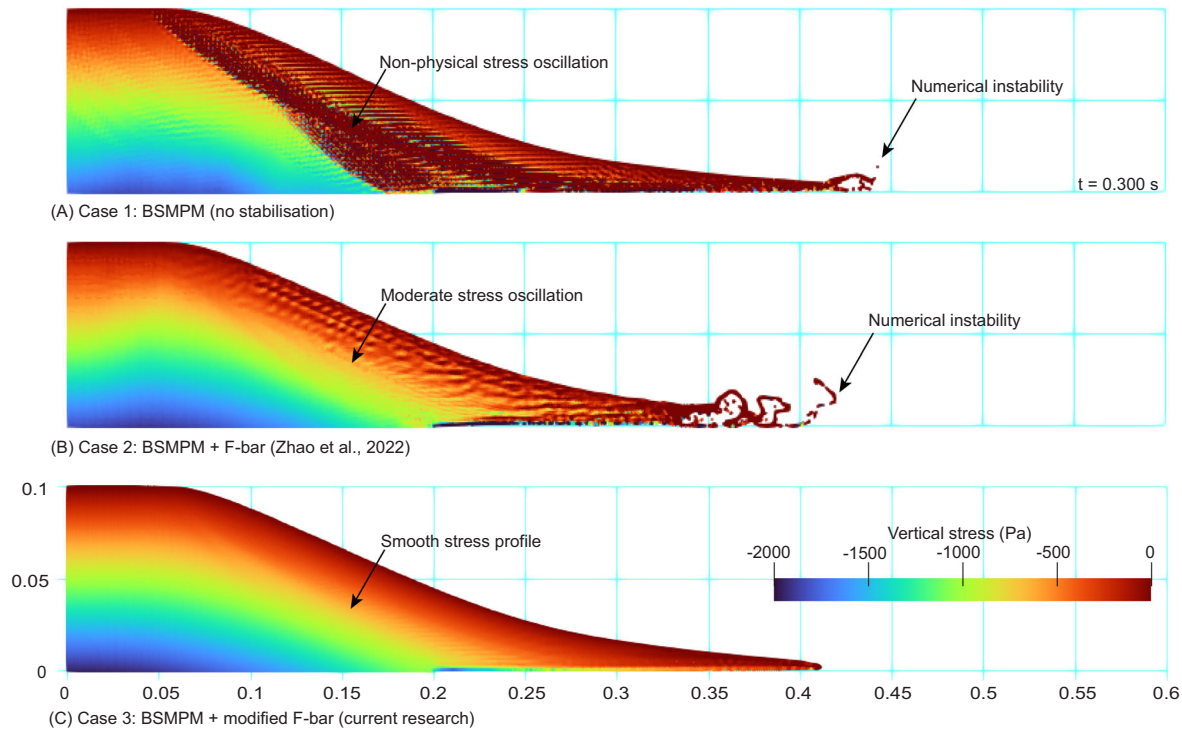


FIGURE 3 Simulations of granular column collapse by different methods: (A) BSMPM with no stabilisation; (B) BSMPM with F-bar Method; (C) BSMPM with modified F-bar Method. BSMPM, B-spline Material Point Method.

In this numerical example, three scenarios have been examined: the BSMPM with no stabilisation method, the BSMPM with F-bar Method²⁸ and BSMPM with the newly proposed modified F-bar Method. The first two methods experience numerical instability after the granular flow reaches a relatively long runout distance (around 0.37 m). The second one, the BSMPM with the F-bar Method,²⁸ cannot have a converged result after 0.351 s due to the significant numerical instability. The first simulation, the BSMPM with no stabilisation, has a converged result, but the unstable granular flow clashes with the right boundary (i.e. 0.55 m) at about 0.42 s. These findings are illustrated in Figure 3, which shows the vertical stress contour maps at 0.300 s given by these three methods. Figure 3A shows that the BSMPM with no stabilisation experiences severe stress oscillation due to the kinematic locking. The over-stiff behaviour is not observed in the BSMPM for this granular column collapse problem. However, the non-physical stress oscillation causes kinematic instability and over-prediction of the runout distance. Figure 3B shows that the F-bar Method²⁸ significantly reduces stress oscillation. However, we can still observe moderate stress oscillation and amplified kinematic instability, as shown in Figure 3B. Figure 3C shows that only minimal stress oscillation is observed in our modified F-bar Method, and the stress profile is smooth. The remaining stress oscillation can be prevented by reducing the time step, resulting in losing computational efficiency. This research uses the relatively large time step by trading off a limited amount of accuracy.

Figure 4 shows the comparison of quasi-Newton and Newton–Raphson Methods. A linear convergence can be observed in this quasi-Newton Method, while the convergence rate for the Newton–Raphson Method is quadratic. However, the computational effort for each iteration is very small for this quasi-Newton Method. The overall efficiency of the quasi-Newton Method can be higher than the Newton–Raphson Method, especially for a highly non-linear large-scale problem.

Figure 5 compares the evolutions of the runout distance. From Figure 5, only our approach has excellent agreement with the experiment. Figure 6 compares the free surface of granular flow given by our approach and the experiment at 0.109, 0.235 and 0.607 s, respectively. The grey dots in Figure 6 are the digitised experimental results, representing the free surface of granular flow. Figure 6 shows that the proposed method has a remarkable agreement with the experiment during the entire propagation process of the granular flow.

There is discontinuity of the stress field on the base of the simulation between 0.2 and 0.25 in Figures 3 and 6. This discontinuity is caused by the material point moving to the fixed boundary condition on the bottom surface. Modelling a frictional contact can solve this issue and result in a more realistic result. However, in this research, the fixed boundary condition is used to represent the rough surface for simplicity.

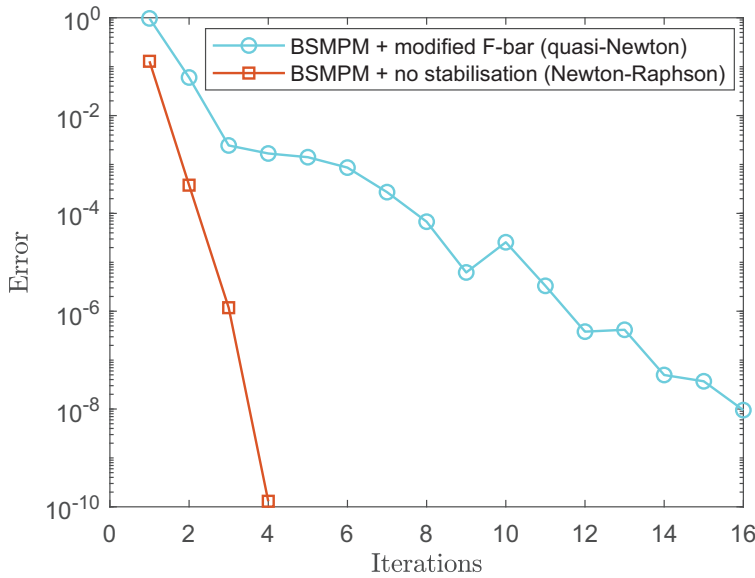


FIGURE 4 Granular column collapse, load step 10. Comparison of quasi-Newton and Newton-Raphson Methods.

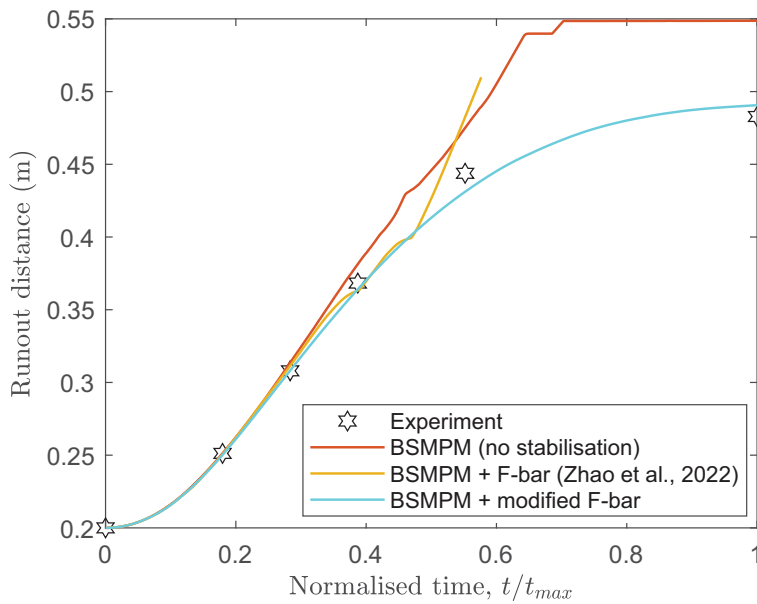


FIGURE 5 Evolutions of the runout distance.

3.2 | Slope stability

The second numerical example is a static slope stability problem introduced by Griffiths and Lane,⁵¹ who used the shear strength reduction technique to determine the factor of safety (FOS) of the slope. In the strength reduction method, the FOS of a slope can be represented by the ratio between the actual soil strength and the reduced one. Therefore, the factored shear strength parameter (i.e. cohesion and friction angle) c_f and ϕ_f can be determined by⁵¹ $c_f = c/\text{SRF}$ and $\phi_f = \arctan(\frac{\tan\phi}{\text{SRF}})$, where SRF is the shear strength reduction factor (SRF). The critical value of SRF represents the FOS of the slope.

The geometry of this slope is shown in Figure 7, and its material properties are summarised in Table 1. As shown in Figure 7, the height of the slope H is 10 m, and the inclination angle is 26.57°. We keep the dimensionless soil property $c/(\gamma H) = 0.05$ same as the one used by Griffiths and Lane.⁵¹ The fixed and roller boundary conditions are applied on the bottom and sides of the slope, respectively.

Griffiths and Lane⁵¹ analysed this slope using small deformation FEM up to $\text{SRF} = 1.4$. They plotted the SRF versus $E\delta_{max}/(\gamma H^2)$ (a dimensionless displacement), as shown in Figure 8. δ_{max} is the maximum displacement in this slope. However, due to the highly distorted mesh, the FEM yields an unconverged solution at $\text{SRF} = 1.4$, and the displacement at

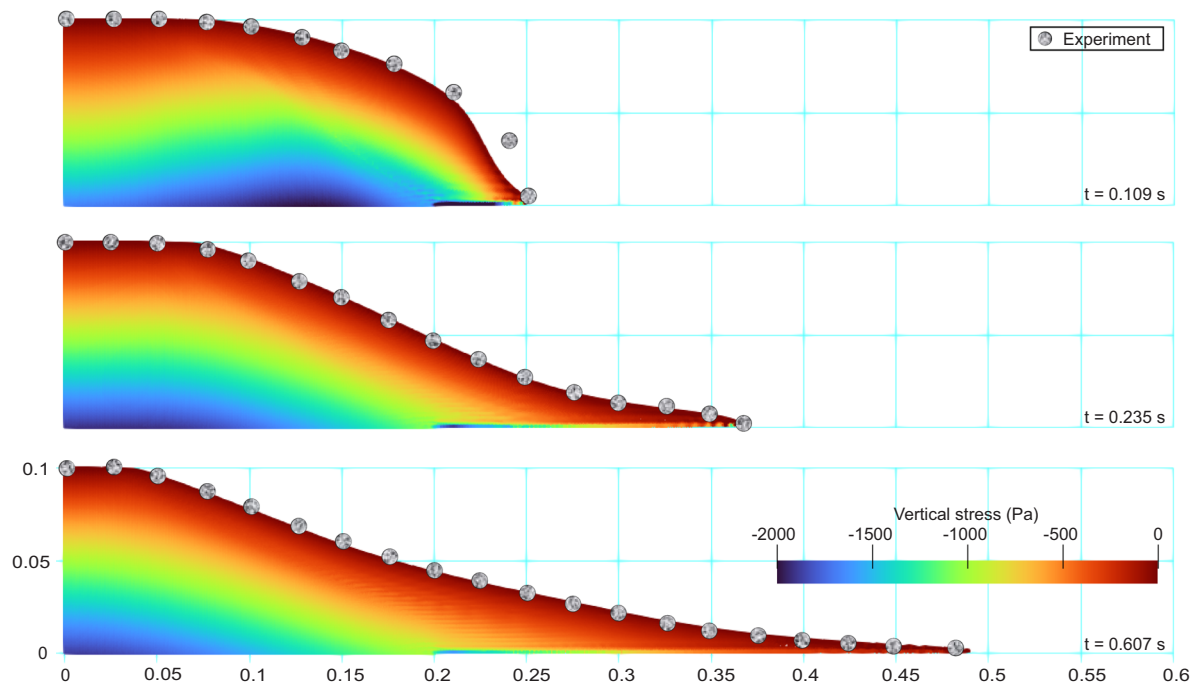
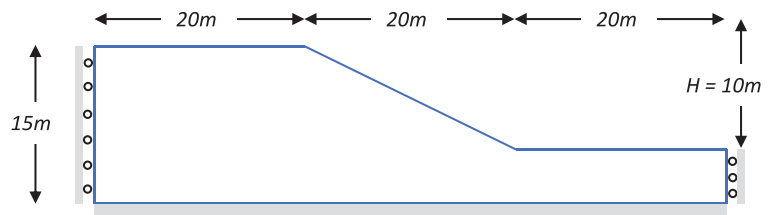


FIGURE 6 Comparison between the BSMPM with modified F-bar Method with the experiment. BSMPM, B-spline Material Point Method.

FIGURE 7 Geometry and boundary conditions of the slope example.



SRF = 1.4 in Figure 8 results from the unconverged 1000 iterations.⁵¹ In FEM, the failure of the slope is defined as the stage that the algorithm cannot find a converged solution within a user-defined number of iterations, which indicates the FOS of this slope is between 1.35 and 1.4.⁵¹ This FOS is consistent with the FOS = 1.38 determined by Bishop and Morgenstern⁵² using the Limit Equilibrium Method (LEM).

In this research, we follow the procedure that Griffiths and Lane⁵¹ proposed to determine the dimensionless displacement $E\delta_{max}/(\gamma H^2)$ of this slope up to SRF = 3.0. The size of the parametric grid cell (patch) is 1 m^2 , which is similar to the mesh size used in Griffiths and Lane.⁵¹ Same as the previous example, thirty-six (6^2) material points are used in each grid cell. As a result, the slope is discretised by 21,540 material points. Each SRF represents an independent analysis. For the BSMPM with F-bar Methods, the gravity load gradually increases from zeros to the maximum value in 10 steps for SRF < 1.38 and 100 steps for SRF > 1.38, respectively. For the BSMPM without stabilisation, 500 steps of loading are required after the failure (SRF > 1.38) to ensure numerical stability. In this numerical example, no kinematic instability has been observed. However, the BSMPM with no stabilisation cannot have a converged solution after SRF = 1.6, and the BSMPM with the F-bar Method (Zhao et al.²⁸ cannot have a converged result after SRF = 2.5.

Before the slope failure (SRF < 1.38), the results of three BSMPMs closely match the FEM research of Griffiths and Lane,⁵¹ as shown in Figure 8. After the failure, the BSMPM without stabilisation methods generates slightly over-stiff results due to the volumetric locking. We can observe significant stress oscillation in the contour map generated by BSMPM, as shown in Figure 9A. Similar to the previous numerical example, the F-bar Method Zhao et al.²⁸ proposed significantly reduces the volumetric locking and stress oscillation. In contrast, the modified F-bar Method completely overcomes the volumetric locking and stress oscillation. Figure 8 shows that the BSMPM (no stabilisation) has a similar result as the FEM when SRF = 1.4. However, both results are inaccurate because of either the volumetric locking instability in BSMPM (Figure 9A) or the highly distorted FEM mesh.⁵¹ Because of the volumetric locking instability, the BSMPM

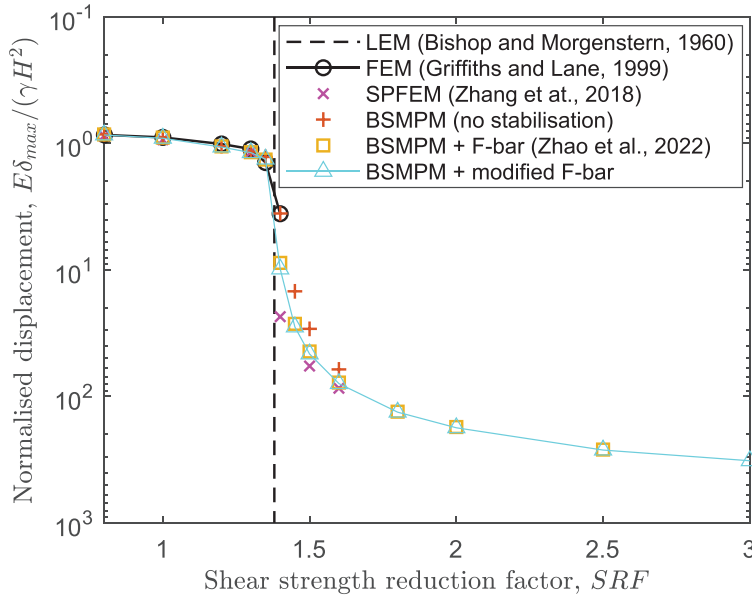


FIGURE 8 Normalised maximum displacement with different SRFs by different methods. SRFs, strength reduction factors.

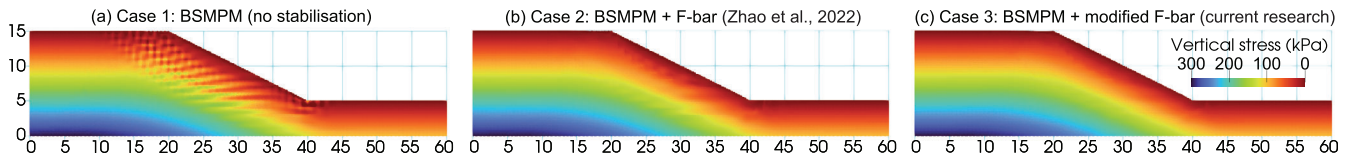


FIGURE 9 Simulations of the slope stability at SRF = 1.4 by different methods: (A) BSMPM with no stabilisation; (B) BSMPM with F-bar Method; (c) BSMPM with modified F-bar Method. BSMPM, B-spline Material Point Method; SRF, strength reduction factor.

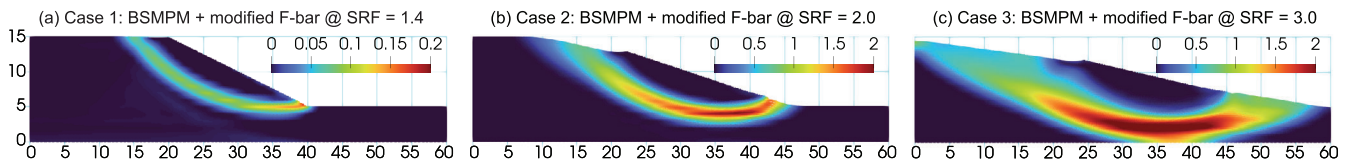


FIGURE 10 Evolution of failure surface: the accumulated plastic strain at different SRFs: (A) SRF = 1.4; (B) SRF = 2.0; (C) SRF = 3.0. SRF, strength reduction factor.

cannot have a converged solution after SRF = 1.6, even further reducing the time step. Although the maximum displacements given by the two F-bar Methods are similar, the newly proposed modified F-bar Method results in a slightly larger displacement for this numerical example, with a completely smooth stress solution, as illustrated in Figure 9C.

Zhang et al.⁵³ investigated the same slope problem using the Smooth Particle Finite Element Method (SPFEM), a large deformation Finite Element-based Method. The results of BSMPM with F-bar Methods are stiffer than Zhang et al.⁵³ because the small strain constitutive model is used in their research, overestimating the displacement in the large strain range. This phenomenon is also reported by Borja and Tamagnini⁵⁴ and Coombs and Crouch.⁵⁵

Figure 10 shows the evolution of the failure surface at the post-failure stages (SRF > 1.38). The value plotted in Figure 10 is the accumulated plastic strain generated by the BSMPM with the modified F-bar Method. The accumulated plastic strain is an internal variable that describes the development of shear plasticity and controls the Mohr–Coulomb model's hardening behaviour. However, we assume no hardening behaviour (perfectly plastic) in these numerical examples (a detailed explanation of this accumulated plastic strain and the hardening behaviour is reported by de Souza Neto et al.⁴²). As shown in Figure 10, the failure surface develops in width as the increasing SRF. Finally, at SRF = 3.0, the failure surface reaches the bottom of the slope foundation.

FIGURE 11 Geometry and boundary conditions of the strip footing example.

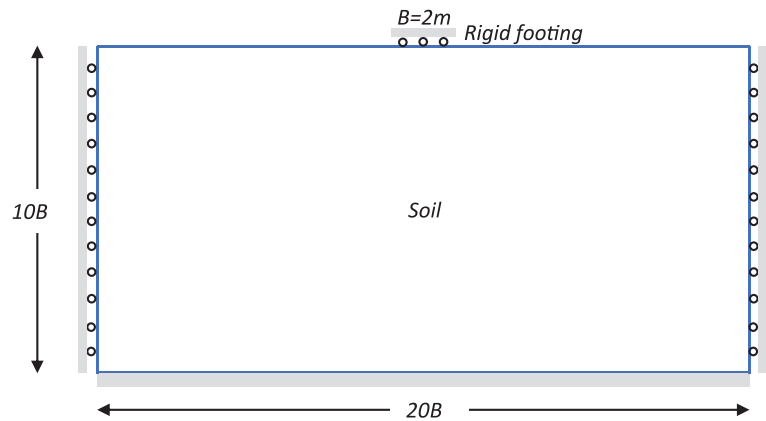
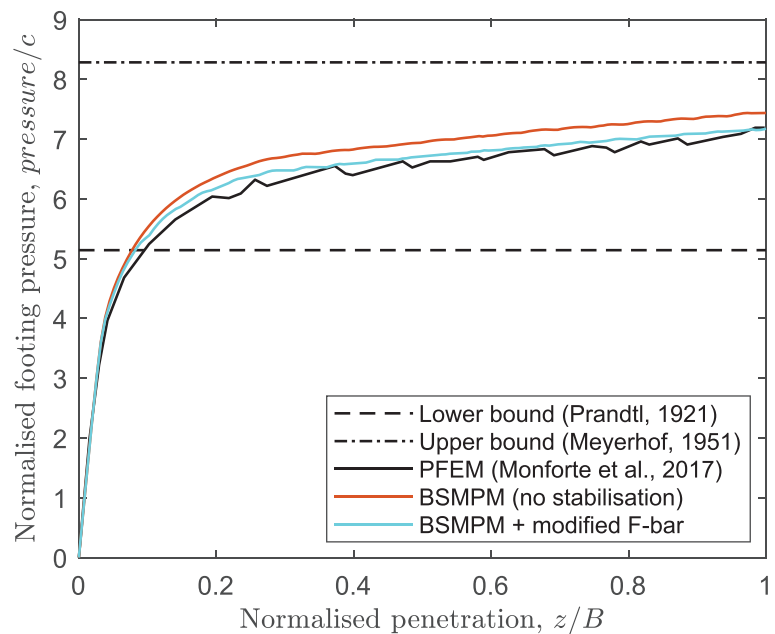


FIGURE 12 Normalised maximum displacement versus normalised pressure underneath the footing by different methods.



3.3 | Large penetration of strip footing

The third example is the large penetration of a rigid strip footing on nearly incompressible Tresca soil. We follow the same strip footing example carried out through the Particle Finite Element Method (PFEM) by Monforte et al.,⁵⁶ who report the mixed stabilised formulation to overcome the volumetric locking. Also, the same large strain constitutive model is used in their research, ensuring that the results are comparable with current research. The soil parameters are shown in Table 1. Figure 11 illustrates the geometry and boundary conditions of this example. We model the rigid strip footing as the prescribed Dirichlet boundary condition in the y -direction (roller) using the penalty method proposed by Cortis et al.⁵⁷ The width of the strip footing is 2 m ($B = 2$ m), and the final penetration reaches 2 m ($z = 2$ m) in 160 steps. The internal nodal forces where the Dirichlet boundary condition is prescribed are measured at each time step to represent the pressure underneath the footing. Due to the symmetry, only the right half of the problem has been modelled. The parametric grid cell (patch) size is 0.01 m^2 , and thirty-six (6^2) material points are used in each grid cell resulting in a total of 360,000 material points.

Figure 12 shows the normalised penetration versus normalised pressure underneath the footing, and Figure 13 shows the vertical stress contours given by the three methods. As shown in Figure 12, the result given by the proposed method is located between the lower bound, $(\pi + 2)c$, and upper bound, $(2\pi + 2)c$, analytical solutions derived by Prandtl⁵⁸ and Meyerhof,⁵⁹ respectively. The modified F-bar (current research) closely matches the PFEM⁵⁶ result, as shown in Figure 12. We can see a very small amount of fluctuations in the curves because the nodal forces are extracted as the pressure underneath the footing. On the contrary, the proposed BSMPM with the modified F-bar Method shows a less fluctuating result

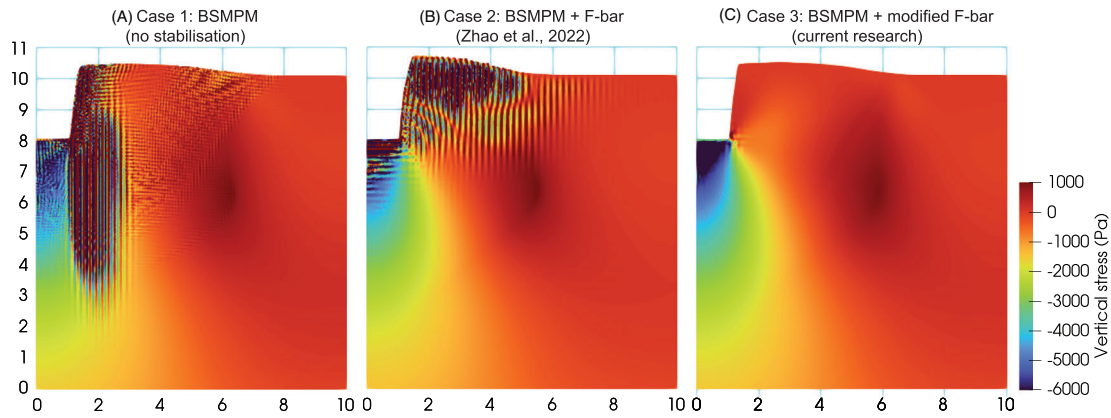


FIGURE 13 Simulations of large penetration of rigid strip footing: (A) BSMPM with no stabilisation; (B) BSMPM with F-bar Method; (C) BSMPM with modified F-bar Method. BSMPM, B-spline Material Point Method.

than the PFEM. The BSMPM with no stabilisation has an over-stiff behaviour due to the volumetric locking. The F-bar Method²⁸ has partially solved the volumetric locking issue, but its stress field highly oscillates in the area underneath the footing and the tensile region where the volume expands, as shown in Figure 13B. On the contrary, the proposed BSMPM with the modified F-bar Method has a very stable stress field despite some stress oscillation near the corner of the footing, as shown in Figure 13C. A similar corner issue can be found in the research of Yuan et al.,⁶⁰ which may be a common phenomenon for a large deformation strip footing on a nearly incompressible material. We found that slightly reducing Poisson's ratio can help with this issue.

4 | CONCLUSIONS

In this research, we propose an implicit locking-free BSMPM. The essential ingredients of this BSMPM include: (1) a special BSMPM implementation, (2) a large strain constitutive model and (3) an enhancement of the F-bar Method. Thanks to these features, the proposed BSMPM has the following advantages: (a) no cell crossing noise; (b) no volumetric locking; (c) no complex material point searching algorithm; (d) no barrier to switching from the original MPM; (e) reasonably smooth stress profile and (f) easy application of the volumetric locking strategy.

The proposed numerical model is validated with an experiment and previous numerical studies. These numerical examples disclose the accuracy of the proposed method for extremely large deformation geotechnical problems. The excellent agreement with the granular column collapse experiment indicates that the proposed method can accurately simulate granular flows. The proposed method also shows an accuracy that closely matches the previous FEM slope study in the small deformation range. Furthermore, it can simulate the post-failure stage of the slope and footing problems that involve extensive deformation without the error due to highly distorted FEM mesh. The proposed method can always maintain a smooth and stable stress field, which was a challenge for the previous MPMs.

The over-stiffening behaviour has been observed in the slope and footing problems because of the volumetric locking. However, in the granular column collapse problem, the stress oscillation due to the volumetric locking leads to an over-estimated runout distance, and no obvious over-stiffening behaviour is shown for the BSMPM with no stabilisation. The volumetric locking instability, found in previous MPMs, results in inaccurate results for all of three numerical examples. Therefore, removing the volumetric locking is essential for a geotechnical application even if the material is not nearly incompressible.

In future research, it is necessary to extend this method to a soil–water coupled model in order to enlarge the range of geotechnical applications to be solved. Additionally, more advanced constitutive modelling is desired to have more realistic simulations.

NOMENCLATURE

- $(\bullet)_p$: a notation associated with material point
- $(\bullet)_I$: a notation associated with grid node

- $(\bullet)^e$: the elastic part of (\bullet)
- $(\bullet)^p$: the plastic part of (\bullet)
- $(\bullet)^v$: the volumetric part of (\bullet)
- $(\bullet)^d$: the deviatoric part of (\bullet)
- $(\bullet)^k$: (\bullet) at k^{th} number of Newton-Raphson iteration
- δ : the increment of (\bullet) in the Newton-Raphson iteration
- Δ : the increment of (\bullet)
- ∇ : the gradient of (\bullet) , which is the gradient in the undeformed configuration.
- ∇_x : the spatial gradient of (\bullet) , which is the gradient in the deformed configuration.
- $(\bar{\bullet})$: the averaged (\bullet)
- \mathbf{a} : the acceleration
- \mathbf{b} : the left Cauchy-Green strain
- B : the width of the footing
- c : the cohesion
- c_f : the factored cohesion
- \mathbf{D} : the consistent tangent matrix
- dt : the time increment
- dof : the spatial dimension of the
- E : Young's modulus
- f : the error of Newton-Raphson iteration
- \mathbf{F} : the deformation gradient
- Fext** : the external force
- Fint** : the internal force
- Foob** : the out of balance force
- \mathbf{G} : the strain-displacement matrix
- H : the height of the slope
- \mathbf{I} : the identity matrix
- \mathbf{J} : the Jacobian of the deformation gradient
- k : the number of Newton-Raphson iteration
- \mathbf{K} : the global stiffness matrix
- m : the mass
- \mathbf{M} : the global mass matrix
- n : the total number of B-spline shape functions
- n_n : the number of grid nodes that influence the material point
- n_p : the number of material points associated with node I
- N_{Ip} : the shape function in the material point method
- q : the polynomial order
- t : the time
- tol : the tolerance of Newton-Raphson iteration
- \mathbf{u} : the displacement
- \mathbf{v} : the velocity
- V : the volume
- \mathbf{x} : the spatial coordinates
- x_i : the i^{th} component of knot vector
- \mathbf{X} : the material coordinates
- z : the penetration depth of the footing
- γ : the unit weight
- δ_{ij} : the Kronecker delta
- δ_{max} : the maximum displacement in the slope
- ϵ : the logarithmic strain
- ν : Poisson's ratio
- Ξ : the knot vector

- σ : the Cauchy stress
 τ : the Kirchhoff stress
 ϕ : the friction angle
 ϕ_f : the factored friction angle
 ψ : the dilation angle

ACKNOWLEDGEMENTS

The first author would like to acknowledge the University College London and China Scholarship Council (File No. 202008520044), which collectively funded his PhD study.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.


DATA AVAILABILITY STATEMENT

Data are available by the corresponding author after reasonable request.

ORCID

Mian Xie  <https://orcid.org/0000-0002-6907-3449>

Pedro Navas  <https://orcid.org/0000-0002-0482-378X>

Susana López-Querol  <https://orcid.org/0000-0002-2677-3007>

REFERENCES

1. Soga K, Alonso E, Yerro A, Kumar K, Bandara S. Trends in large-deformation analysis of landslide mass movements with particular emphasis on the material point method. *Géotechnique*. 2016;66:248-273. doi:<https://doi.org/10.1680/jgeot.15.LM.005>
2. Wang D, Bienen B, Nazem M, et al. Large deformation finite element analyses in geotechnical engineering. *Comput Geotech*. 2015;65:104-114. doi:<https://doi.org/10.1016/j.compgeo.2014.12.005>
3. Augarde CE, Lee SJ, Loukidis D. Numerical modelling of large deformation problems in geotechnical engineering: a state-of-the-art review. *Soils Found*. 2021;61:1718-1735. doi:<https://doi.org/10.1016/j.sandf.2021.08.007>
4. Qin J, Mei G, Xu N. Meshfree methods in geohazards prevention: a survey. *Arch Comput Meth Eng*. 2022;29:3151-3182. doi:<https://doi.org/10.1007/s11831-021-09686-4>
5. Shan Z, Wu H, Ni W, et al. Recent technological and methodological advances for the investigation of submarine landslides. *J Marine Sci Eng*. 2022;10:1728. doi:<https://doi.org/10.3390/jmse10111728>
6. Coombs WM, Charlton TJ, Cortis M, Augarde CE. Overcoming volumetric locking in material point methods. *Comput Methods Appl Mech Eng*. 2018;333:1-21. doi:<https://doi.org/10.1016/j.cma.2018.01.010>
7. Guilkey JE, Weiss JA. Implicit time integration for the material point method: quantitative and algorithmic comparisons with the finite element method. *Int J Numer Methods Eng*. 2003;57:1323-1338. doi:<https://doi.org/10.1002/nme.729>
8. Sulsky D, Chen Z, Schreyer H. A particle method for history-dependent materials. *Comput Methods Appl Mech Eng*. 1994;118:179-196. doi:[https://doi.org/10.1016/0045-7825\(94\)90112-0](https://doi.org/10.1016/0045-7825(94)90112-0)
9. Bardenhagen SG, Kober EM. The generalized interpolation material point method. *Comput Model Eng Sci*. 2004;5:477-496. doi:<https://doi.org/10.3970/cmesc.2004.005.477>
10. Zhang X, Chen Z, Liu Y. *The Material Point Method: A Continuum-Based Particle Method for Extreme Loading Cases*. 1st ed. Elsevier; 2016.
11. Steffen M, Kirby RM, Berzins M. Analysis and reduction of quadrature errors in the material point method (MPM). *Int J Numer Methods Eng*. 2008;76:922-948. doi:<https://doi.org/10.1002/nme.2360>
12. Molinos M, Navas P, Manzanal D, Pastor M. Local maximum entropy material point method applied to quasi-brittle fracture. *Eng Fract Mech*. 2021;241:107394. doi:<https://doi.org/10.1016/j.engfracmech.2020.107394>
13. Arroyo M, Ortiz M. Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. *Int J Numer Methods Eng*. 2006;65:2167-2202. doi:<https://doi.org/10.1002/nme.1534>
14. Molinos M, Navas P, Pastor M, Stickle MM. On the dynamic assessment of the local-maximum entropy material point method through an explicit predictor-corrector scheme. *Comput Methods Appl Mech Eng*. 2021;374:113512. doi:<https://doi.org/10.1016/j.cma.2020.113512>
15. Molinos M, Stickle MM, Navas P, Yague A, Manzanal D, Pastor M. Toward a local maximum entropy material point method at finite strain within a B-free approach. *Int J Numer Methods Eng*. 2021;122:5594-5625. doi:<https://doi.org/10.1002/nme.6765>
16. Hu Y, Fang Y, Ge Z, et al. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. *ACM Trans Graph*. 2018;37:1-14. doi:<https://doi.org/10.1145/3197517.3201293>
17. Tran QA, Sołowski W, Berzins M, Guilkey J. A convected particle least square interpolation material point method. *Int J Numer Methods Eng*. 2020;121:1068-1100. doi:<https://doi.org/10.1002/nme.6257>

18. Song JU, Kim HG. An improved material point method using moving least square shape functions. *Comput Particle Mech*. 2021;8:751-766. doi:<https://doi.org/10.1007/s40571-020-00368-9>
19. Bower AF. *Applied Mechanics of Solids*. 1st ed. CRC Press; 2009.
20. Love E, Sulsky DL. An energy-consistent material-point method for dynamic finite deformation plasticity. *Int J Numer Methods Eng*. 2006;65:1608-1638. doi:<https://doi.org/10.1002/nme.1512>
21. Mast C, Mackenzie-Helnwein P, Arduino P, Miller G, Shin W. Mitigating kinematic locking in the material point method. *J Comput Phys*. 2012;231:5351-5373. doi:<https://doi.org/10.1016/j.jcp.2012.04.032>
22. Kularathna S, Soga K. Implicit formulation of material point method for analysis of incompressible materials. *Comput Methods Appl Mech Eng*. 2017;313:673-686. doi:<https://doi.org/10.1016/j.cma.2016.10.013>
23. Zhang F, Zhang X, Sze KY, Lian Y, Liu Y. Incompressible material point method for free surface flow. *J Comput Phys*. 2017;330:92-110. doi:<https://doi.org/10.1016/j.jcp.2016.10.064>
24. Iaconeta I, Larese A, Rossi R, Oñate E. A stabilized mixed implicit Material Point Method for non-linear incompressible solid mechanics. *Comput Mech*. 2019;63:1243-1260. doi:<https://doi.org/10.1007/s00466-018-1647-9>
25. Bisht V, Salgado R, Prezzi M. Simulating penetration problems in incompressible materials using the material point method. *Comput Geotech*. 2021;133:103593. doi:<https://doi.org/10.1016/j.compgeo.2020.103593>
26. Wang L, Coombs WM, Augarde CE, et al. An efficient and locking-free material point method for three-dimensional analysis with simplex elements. *Int J Numer Methods Eng*. 2021;122:3876-3899. doi:<https://doi.org/10.1002/nme.6685>
27. Zheng X, Pisanò F, Vardon PJ, Hicks MA. Fully implicit, stabilised MPM simulation of large-deformation problems in two-phase elastoplastic geomaterials. *Comput Geotech*. 2022;147:104771. doi:<https://doi.org/10.1016/j.compgeo.2022.104771>
28. Zhao Y, Jiang C, Choo J. Circumventing volumetric locking in explicit material point methods: a simple, efficient, and general approach. *arXiv*. 2022. doi:<https://doi.org/10.48550/arXiv.2209.02466>
29. Navas P, López-Querol S, Yu RC, Pastor M. Optimal transportation meshfree method in geotechnical engineering problems under large deformation regime. *Int J Numer Methods Eng*. 2018;115:1217-1240. doi:<https://doi.org/10.1002/nme.5841>
30. Telikicherla RM, Moutsanidis G. Treatment of near-incompressibility and volumetric locking in higher order material point methods. *Comput Methods Appl Mech Eng*. 2022;395:114985. doi:<https://doi.org/10.1016/j.cma.2022.114985>
31. Sugai R, Han J, Yamaguchi Y, Moriguchi S, Terada K. Extended B-spline-based implicit material point method enhanced by F-bar projection method to suppress pressure oscillation. *Int J Numer Methods Eng*. 2023. doi:<https://doi.org/10.1002/nme.7216>
32. Elguedj T, Bazilevs Y, Calo V, Hughes T. B-bar and F-bar projection methods for nearly incompressible linear and non-linear elasticity and plasticity using higher-order NURBS elements. *Comput Methods Appl Mech Eng*. 2008;197:2732-2762. doi:<https://doi.org/10.1016/j.cma.2008.01.012>
33. Motlagh YG, Coombs WM. An implicit high-order material point method. *Procedia Eng*. 2017;175:8-13. doi:<https://doi.org/10.1016/j.proeng.2017.01.003>
34. Gan Y, Sun Z, Chen Z, Zhang X, Liu Y. Enhancement of the material point method using B-spline basis functions. *Int J Numer Methods Eng*. 2018;113:411-431. doi:<https://doi.org/10.1002/nme.5620>
35. Tran QA, Wobbes E, Solowski W, Möller M, Vuik C. Moving least squares reconstruction for B-spline Material Point Method. In: *2nd International Conference on the Material Point Method for Modelling Soil-Water-Structure Interaction*. University of Cambridge; 2019:35-41.
36. Wobbes E, Möller M, Galavi V, Vuik C. Conservative Taylor least squares reconstruction with application to material point methods. *Int J Numer Methods Eng*. 2019;117:271-290. doi:<https://doi.org/10.1002/nme.5956>
37. Sun Z, Gan Y, Tao J, Huang Z, Zhou X. An improved quadrature scheme in B-spline material point method for large-deformation problem analysis. *Eng Anal Boundary Elements*. 2022;138:301-318. doi:<https://doi.org/10.1016/j.enganabound.2022.03.004>
38. de Boor C. *A Practical Guide to Splines*. Revised ed. Springer-Verlag; 2001.
39. Stomakhin A, Schroeder C, Chai L, Teran J, Selle A. A material point method for snow simulation. *ACM Trans Graph*. 2013;32:1-10. doi:<https://doi.org/10.1145/2461912.2461948>
40. Klár G, Gast T, Pradhana A, et al. Drucker-Prager elastoplasticity for sand animation. *ACM Trans Graph*. 2016;35:1-12. doi:<https://doi.org/10.1145/2897824.2925906>
41. Steffen M, Wallstedt P, Guilkey J, Kirby R, Berzins M. Examination and analysis of implementation choices within the Material Point Method (MPM). *Comput Model Eng Sci*. 2008;2:107-128. doi:<https://doi.org/10.3970/cmesci.2008.031.107>
42. de Souza Neto E, Perić D, Owen DRJ. *Computational Methods for Plasticity*. John Wiley & Sons Ltd; 2008.
43. Coombs WM, Augarde CE. AMPLE: A material point learning environment. *Adv Eng Softw*. 2020;139:102748. doi:<https://doi.org/10.1016/j.advengsoft.2019.102748>
44. de Souza Neto E, Perić D, Dutko M, Owen D. Design of simple low order finite elements for large strain analysis of nearly incompressible solids. *Int J Solids Struct*. 1996;33:3277-3296. doi:[https://doi.org/10.1016/0020-7683\(95\)00259-6](https://doi.org/10.1016/0020-7683(95)00259-6)
45. Andersen S, Andersen L. *Post-Processing in the Material-Point Method*. Aalborg University; 2013.
46. Dunatunga S, Kamrin K. Continuum modelling and simulation of granular flows through their many phases. *J Fluid Mech*. 2015;779:483-513. doi:<https://doi.org/10.1017/jfm.2015.383>
47. Broyden CG. A class of methods for solving nonlinear simultaneous equations. *Math Comput*. 1965;19:577-593. doi:<https://doi.org/10.1090/S0025-5718-1965-0198670-6>
48. Lagrée PY, Staron L, Popinet S. The granular column collapse as a continuum: validity of a two-dimensional Navier-Stokes model with a $\mu(I)$ -rheology. *J Fluid Mech*. 2011;686:378-408. doi:<https://doi.org/10.1017/jfm.2011.335>

49. Nguyen CT, Nguyen CT, Bui HH, Nguyen GD, Fukagawa R. A new SPH-based approach to simulation of granular flows using viscous damping and stress regularisation. *Landslides*. 2017;14:69-81. doi:<https://doi.org/10.1007/s10346-016-0681-y>
50. Sołowski WT, Sloan SW. Evaluation of material point method for use in geotechnics. *Int J Numer Anal Methods Geomech*. 2015;39:685-701. doi:<https://doi.org/10.1002/nag.2321>
51. Griffiths DV, Lane PA. Slope stability analysis by finite elements. *Géotechnique*. 1999;49:387-403. doi:<https://doi.org/10.1680/geot.1999.49.3.387>
52. Bishop AW, Morgenstern N. Stability coefficients for Earth slopes. *Géotechnique*. 1960;10:129-153. doi:<https://doi.org/10.1680/geot.1960.10.4.129>
53. Zhang W, Yuan W, Dai B. Smoothed particle finite-element method for large-deformation problems in Geomechanics. *Int J Geomech*. 2018;18. doi:[https://doi.org/10.1061/\(ASCE\)GM.1943-5622.0001079](https://doi.org/10.1061/(ASCE)GM.1943-5622.0001079)
54. Borja RI, Tamagnini C. Cam-Clay plasticity part III: extension of the infinitesimal model to include finite strains. *Comput Methods Appl Mech Eng*. 1998;155:73-95. doi:[https://doi.org/10.1016/S0045-7825\(97\)00141-2](https://doi.org/10.1016/S0045-7825(97)00141-2)
55. Coombs WM, Crouch RS. Non-associated Reuleaux plasticity: analytical stress integration and consistent tangent for finite deformation mechanics. *Comput Methods Appl Mech Eng*. 2011;200:1021-1037. doi:<https://doi.org/10.1016/j.cma.2010.11.012>
56. Monforte L, Arroyo M, Carbonell JM, Gens A. Numerical simulation of undrained insertion problems in geotechnical engineering with the Particle Finite Element Method (PFEM). *Comput Geotech*. 2017;82:144-156. doi:<https://doi.org/10.1016/j.compgeo.2016.08.013>
57. Cortis M, Coombs W, Augarde C, Brown M, Brennan A, Robinson S. Imposition of essential boundary conditions in the material point method. *Int J Numer Methods Eng*. 2018;113:130-152. doi:<https://doi.org/10.1002/nme.5606>
58. Prandtl L. Hauptaufsätze: Über die Eindringungsfestigkeit (Härte) plastischer Baustoffe und die Festigkeit von Schneiden. *ZAMM - J Appl Math Mech/Zeitschrift für Angewandte Mathematik und Mechanik*. 1921;1:15-20. doi:<https://doi.org/10.1002/zamm.19210010102>
59. Meyerhof GG. The ultimate bearing capacity of foundations. *Géotechnique*. 1951;2:301-332. doi:<https://doi.org/10.1680/geot.1951.2.4.301>
60. Yuan WH, Wang HC, Zhang W, Dai BB, Liu K, Wang Y. Particle finite element method implementation for large deformation analysis using Abaqus. *Acta Geotech*. 2021;16:2449-2462. doi:<https://doi.org/10.1007/s11440-020-01124-2>

How to cite this article: Xie M, Navas P, López-Querol S. An implicit locking-free B-spline Material Point Method for large strain geotechnical modelling. *Int J Numer Anal Methods Geomech*. 2023;1-21. <https://doi.org/10.1002/nag.3599>

APPENDIX A: CONSTRUCT THE B-SPLINE SHAPE FUNCTION

The following scheme illustrates the application of the proposed method for a 2-D problem:

- (1) Given the x -coordinates of a material point x_x and the global knot vector in x -direction Ξ_x , construct the 1-D B-spline shape functions $N_{i,q}(x)$ and derivatives $\frac{dN_{i,q}(x)}{dx}$ recursively using Equations (4)–(6). Repeat the same process in y -direction to obtain $N_{i,q}(y)$ and $\frac{dN_{i,q}(y)}{dy}$.
- (2) Convolute shape functions and derivatives in multi-dimensions through the tensor product operation:

$$N_{Ip} = \text{kron}(N_{i,q}(x), N_{i,q}(y))$$

$$\partial_x N_{Ip} = \text{kron}\left(\frac{dN_{i,q}(x)}{dx}, N_{i,q}(y)\right)$$

$$\partial_y N_{Ip} = \text{kron}\left(N_{i,q}(x), \frac{dN_{i,q}(y)}{dy}\right)$$
- (3) Obtained the material point to tensor product grid nodes connectivity ($p2N$) by finding the non-zero indices of the shape functions:

$$p2N = \text{find}(N_{Ip} \neq 0)$$
- (4) Chop off the shape functions and derivatives matrices using $p2N$:

$$N_{Ip} = N_{Ip}[p2N]$$

$$\nabla N_{Ip} = [\partial_x N_{Ip}[p2N]; \partial_y N_{Ip}[p2N]]$$

In this algorithm, the functions $\text{kron}()$ and $\text{find}()$ represent the native MATLAB functions or the NumPy functions in Python. The particle searching is achieved by using this $\text{find}()$ function. The $\text{find}(A \neq 0)$ function is capable to find the indices of non-zero elements for an input vector A ; the $\text{kron}(A, B)$ function returns the Kronecker tensor product of two input vectors A and B . The proposed method also works on the non-equally spaced structured mesh by simply utilising non-uniform knot vectors. Moving from 2-D to 3-D is also straightforward by introducing an additional knot vector in the z -direction. The 3-D shape functions can be constructed following the same tensor product procedure.

The algorithm mentioned above is not efficient because it involves many non-essential zero multiplications (i.e. zero times zero) during the tensor product operations. To accelerate this algorithm, we can chop off (i.e. only keep the non-zero elements) the 1-D shape functions and derivatives before the tensor product operation. Therefore, the matrix size can be reduced significantly during the execution of $\text{kron}()$ function. Following the same concept, we can find the non-zero indices of the 1-D shape function. Then, we can use this 1-D topology information to calculate the multi-dimensions $p2N$ matrix. For 2-D quadratic B-spline MPM, a more efficient algorithm is

- (1) Given the x -coordinates of a material point x_x and the global knot vector in x -direction Ξ_x , construct the 1-D B-spline shape functions $N_{i,q}(x)$ and derivatives $\frac{dN_{i,q}(x)}{dx}$ recursively using Equations (4)–(6). Repeat the same process in y -direction to obtain $N_{i,q}(y)$ and $\frac{dN_{i,q}(y)}{dy}$.
- (2) Obtain the 1-D topology information id_x in the x -direction and id_y in the y -direction by finding the non-zero indices of the 1-D shape functions:

$$id_x = \text{find}(N_{i,q}(x) \neq 0)$$

$$id_y = \text{find}(N_{i,q}(y) \neq 0)$$
- (3) Given id_x , id_y and the number of shape functions in y -direction n_y , calculate the material point to tensor product grid nodes connectivity ($p2N$):

$$A = (id_x[1] - 1) * n_y + id_y$$

$$B = A + n_y$$

$$C = B + n_y$$

$$p2N = [A, B, C]$$
- (4) Chop off and convolute shape functions and derivatives in multi-dimensions through the tensor product operation:

$$N_{Ip} = \text{kron}(N_{i,q}(x)[id_x], N_{i,q}(y)[id_y])$$

$$\partial_x N_{Ip} = \text{kron}(\frac{dN_{i,q}(x)}{dx}[id_x], N_{i,q}(y)[id_y])$$

$$\partial_y N_{Ip} = \text{kron}(N_{i,q}(x)[id_x], \frac{dN_{i,q}(y)}{dy}[id_y])$$

$$\nabla N_{Ip} = [\partial_x N_{Ip}; \partial_y N_{Ip}]$$

In this algorithm, $id_x[1]$ means the first element of the id_x vector. For 2-D cubic B-spline MPM, a more efficient algorithm is

- (1) Given the x -coordinates of a material point x_x and the global knot vector in x -direction Ξ_x , construct the 1-D B-spline shape functions $N_{i,q}(x)$ and derivatives $\frac{dN_{i,q}(x)}{dx}$ recursively using Equations (4)–(6). Repeat the same process in y -direction to obtain $N_{i,q}(y)$ and $\frac{dN_{i,q}(y)}{dy}$.
- (2) Obtain the 1-D topology information id_x in the x -direction and id_y in the y -direction by finding the non-zero indices of the 1-D shape functions:

$$id_x = \text{find}(N_{i,q}(x) \neq 0)$$

$$id_y = \text{find}(N_{i,q}(y) \neq 0)$$
- (3) Given id_x , id_y and the number of shape functions in y -direction n_y , calculate the material point to tensor product grid nodes connectivity ($p2N$):

$$A = (id_x[1] - 1) * n_y + id_y$$

$$B = A + n_y$$

$$C = B + n_y$$

$$D = C + n_y$$

$$p2N = [A, B, C, D]$$
- (4) Chop off and convolute shape functions and derivatives in multi-dimensions through the tensor product operation:

$$N_{Ip} = \text{kron}(N_{i,q}(x)[id_x], N_{i,q}(y)[id_y])$$

$$\partial_x N_{Ip} = \text{kron}(\frac{dN_{i,q}(x)}{dx}[id_x], N_{i,q}(y)[id_y])$$

$$\partial_y N_{Ip} = \text{kron}(N_{i,q}(x)[id_x], \frac{dN_{i,q}(y)}{dy}[id_y])$$

$$\nabla N_{Ip} = [\partial_x N_{Ip}; \partial_y N_{Ip}]$$