Computers and Structures 212 (2019) 257–274

Contents lists available at ScienceDirect

Computers and Structures

journal homepage: www.elsevier.com/locate/compstruc

B-spline based boundary conditions in the material point method

Y. Bing, M. Cortis, T.J. Charlton, W.M. Coombs *, C.E. Augarde

Department of Engineering, Durham University, Durham, UK

ARTICLE INFO

Article history:
Received 15 August 2018
Accepted 10 November 2018

Keywords:
Material point method
B-splines
Boundary conditions
Non-matching meshes
Implicit boundary method

ABSTRACT

The material point method is an increasingly popular method for tackling solid mechanics problems involving large deformations. However, there are issues associated with applying boundary conditions in the method and, to date, no general approach for imposing both Neumann and Dirichlet boundary conditions has been proposed. In this paper, a new B-spline based boundary method is developed as a complete methodology for boundary representation, boundary tracking and boundary condition imposition in the standard material point method. The B-spline interpolation technique is employed to form continuous boundaries which are independent of the background mesh. Dirichlet boundary conditions are enforced by combining the B-spline boundaries with an implicit boundary method. Neumann boundary conditions are included by direct integration of surface tractions along the B-spline boundary. This general boundary method not only widens the problems that can be analysed by all variants of the material point method, when implemented using an implicit solver, but is also applicable to other embedded and non-matching mesh approaches. Although the Dirichlet boundary conditions are restricted to implicit methods, boundary representation, tracking and Neumann boundary condition enforcement can be applied to explicit and implicit methods.

1. Introduction

For decades, the finite element method (FEM) has dominated the computational analysis of structures and solid mechanics. However, as a method that is based on a pure Lagrangian description, the standard FEM exhibits some weaknesses for certain applications, one of which is its inability to model large deformation and fracture problems, such as landslides, without going through the expensive processes of re-meshing and mapping of the state variables. Meshless Methods (MMs) provide an alternative to the FEM. Early developments include the Smoothed Particle Hydrodynamics method [1], the Element-Free Galerkin method [2] and the Reproducing Kernel Particle method [3]. Although the elimination of a Lagrangian mesh allows MMs to be able to model problems that are challenging for the FEM, these methods are not without disadvantages. As the governing equations are solved on the particles, searching of the neighbour particles is essential and high order integration is required to accurately integrate the rational shape functions [4]. An alternative to all the choices mentioned above is the material point method (MPM).

The MPM, developed originally by Sulsky and co-workers [5,6], discretises a problem domain by a finite number of particles, or material points (MPs), on which the material properties and history dependent variables are prescribed and carried throughout a simulation. Each material point is assigned a weight representing the volume of the domain under its influence. In addition to these Lagrangian particles, an Eulerian background mesh is employed to solve the governing equations, which fully eliminates the need for neighbourhood searching. Mapping between the material points and mesh nodes is usually achieved through interpolation functions, and the positions of the material points are effectively integration points within each element. Although there are no restrictions placed on the form of the background mesh, a structured grid with regular elements is often chosen due to its high computational efficiency [7]. Major improvements of the MPM include the development of the generalised interpolation material point (GIMP) method [8], the convected particle domain interpolation (CPDI) method [9] and the second-order convected particle domain interpolation (CPDI2) method [10]. The GIMP method overcomes stress oscillations when material points move from one element to another by assigning a physical domain to each material point, and uses interpolation functions that have a higher continuity. These modifications allow the material points to partially maintain influence on elements other than the one in which they are located due to the overlap of the material point's domain with adjacent elements. However, the issue of material separation is not eliminated by the GIMP method as it does not consider the

* Corresponding author.
E-mail address: w.m.coombs@durham.ac.uk (W.M. Coombs).

https://doi.org/10.1016/j.compstruc.2018.11.003
0045-7949/

change in shapes of the volumes linked to the material points during deformation. This problem is improved by both the CPDI and the CPDI2 methods by allowing the domains of the material points to deform. However, the CPDI techniques are effectively dual mesh methods with the mesh representing the material point domains being convected through a background mesh. This increases the memory requirement of the method, and also introduces additional approximations through the calculation of the basis functions between the overlapping meshes. In this paper we adopt the standard MPM as the focus of the paper is on the imposition of boundary conditions rather than the reduction of cell crossing instabilities; however the techniques developed in this paper are applicable to all material point variants.[1]

The MPM and its variants have been widely applied to analyse fluid-structure interaction problems [11–13], impact, penetration and explosion problems [14–18], materials [19–21], landslides [22–25] and crack propagation [26,27]. Despite this, the lack of effective boundary representation when the boundaries of the physical problem do not coincide with the background mesh in the standard MPM has limited the nature of boundary conditions (BCs) and hence the scope of problems to which the MPM can be applied. Unlike the standard FEM, the boundaries of a physical domain in the MPM do not necessarily align with the background mesh. This makes the application of boundary conditions troublesome especially for tractions (Neumann) and non-zero prescribed displacement (Dirichlet) boundary conditions. Although one can always adjust the position of the problem domain to achieve boundary-mesh alignment, this is not convenient or even possible if a regular background mesh is used, especially when non-zero essential boundary conditions are applied through multiple load steps.

However, this drawback of the MPM is rarely discussed. The most common approach to date to introduce tractions into the system is via boundary particles which normally are the outer layer of the material points [28]. Prescribed tractions are carried by the boundary particles, and then mapped onto the nodes of elements that contain these particles through shape functions. The downside of this approach is that tractions are applied not on a surface but across a boundary band, which increases the thickness of the actual boundary and leads to errors [28,29]. Some papers have made the effort to improve the accuracy of this standard approach with different weighting parameters [30–32], but the accuracy of these methods is limited by the element size, and mesh refinement around the boundaries is required to maintain sufficient accuracy. Alternatively, the moving mesh concept, introduced by Kafaji in 2013 [29], allows Dirichlet boundary conditions to be imposed directly in the same way as standard finite elements, but only if the essential boundary does not change shape [33] and moves in one direction [32]. Another potential solution to the problem of boundary condition imposition is the dual-grid approach [34], which uses a string of elements to locate essential boundaries with respect to the background mesh. Unfortunately, simple 1D tests have shown that this method is sensitive to the location and element size of the boundaries [35] and an improvement to the method has yet to be reported. A similar problem occurs in the FEM when using non-conforming meshes and different weighted interpolation methods have been developed to solve this issue. For example, Höllig et al. [36–38] introduced a new set of interpolation functions called weighted extended B-splines (Web-splines). By adopting these Web-splines as the shape functions in the formation of the Galerkin weak form, homogeneous Dirichlet boundary conditions can be imposed without generating an additional mesh for the boundary conditions. Recently Remmerswaal [39] presented a boundary detection approach for the MPM based on concepts from the fluid mechanics community. However, the work was restricted to moving Neumann boundary conditions and did not include a general method for the imposition of Dirichlet conditions on arbitrary boundaries.

The implicit boundary method (IBM) [33,40–44] enforces Dirichlet boundary conditions by introducing extra penalty stiffness in the system through "Dirichlet functions" which approximate step functions. Implementation of fixed Dirichlet boundary conditions with problem boundaries parallel to one of the coordinates has been demonstrated in [40–42]. In Zhang and Kumar [43], formulations to enforce displacements that do not align with the global coordinate system has been suggested. This method has recently been applied to the MPM framework and also extended to include the imposition of "roller" boundary conditions on inclined boundaries [33,44]. Another method that uses implicit equations to apply Dirichlet boundary conditions is the weighted finite cell method [45,46]. Because this method uses implicit equations directly instead of step functions, it is capable of enforcing inhomogeneous Dirichlet boundary conditions exactly without mesh generation. However, the accuracy of the overall solution largely depends on the level of the quadtree refinement that is performed around the problem boundaries and it has not been applied to MPMs.

Despite the development of methods to impose boundary conditions in the standard MPM, a general method for boundary representation and boundary condition imposition is still lacking and this deficiency is resolved by the method proposed in this paper. The most obvious advantage gained from having defined boundaries is that it allows the analyser to visualise the deformed shape of the problem domain in post-processing. More importantly, general boundary conditions, tractions (Neumann boundary conditions) and prescribed displacements (inhomogeneous Dirichlet boundary conditions) in particular, can be enforced. This paper provides, for the first time, a general way to track boundaries and to impose Neumann conditions in all variants of the MPM and also Dirichlet boundary conditions for implicit implementations of the MPM. Inspired by Kim and Young [47–49], the method presented in this paper uses a B-spline interpolation technique to represent boundaries. B-splines have been adopted as they provide a general method for the description of complex boundaries whilst being independent of the background mesh. Tractions are imposed through direct integration of the traction over the B-spline boundary segment on which it is applied and displacements are prescribed via the IBM.

This paper is organised as follows. The MPM formulation is presented in Section 2 which is followed by a brief description of the B-spline boundary representation technique in Section 3. Section 3 also details the imposition of Neumann and Dirichlet boundary conditions over the B-spline represented boundary. The proposed method is validated by a number of numerical examples in Section 4 and conclusions are drawn in Section 5. Matrix/vector notation is used to present the formulation, and linear elastic materials are considered as the focus of the paper is on the boundary conditions. The techniques presented in this paper can also be applied to non-linear material behaviour and finite deformation mechanics.

2. Material point method formulation

The MPM combines the advantages of both Lagrangian and Eulerian approaches to solid mechanics. An implicit FEM implementation is adopted for the standard MPM formulation in this paper. Although the majority of MPM implementations use an

[1] The only exception to this is the Dirichlet boundary conditions that can only be applied to implicit versions of the material point method. However, the boundary representation and tracking techniques described in this paper can be applied to all material point methods as can the method for inhomogeneous Neumann boundary condition (non-zero tractions) enforcement.

explicit approach to solve the governing equations, the implicit approach has the benefit that it allows larger load steps to be used in analyses, which reduces the number of load steps required. Additionally, Guilkey and Weiss [50] have shown that implicit MPM formulations give more accurate results than explicit methods for certain classes of problems.

A background mesh, as used in the FEM, is generated at the start of the simulation. The problem domain is then discretised using material points which are placed in the relevant elements. The background mesh should cover the physical domain, however no requirement is placed on the alignment of the background grid with the domain; the background mesh can be changed through the simulation and should be adapted, extended or replaced if the physical problem exceeds the mesh extent. At the beginning of each load step, the location of each material point is identified. Information carried by the material points is mapped to the nodes where the governing equations are solved.

The focus of this paper is on the imposition of boundary conditions within the MPM. Therefore, we restrict the MPM formulation to quasi-static analysis with the assumptions of small strains and linear elastic material behaviour. With these assumptions, the weak form of equilibrium within an element is given by

$$\underbrace{\int_{\Omega} [B]^T [D^e][B]\, \mathrm{d}V}_{[k^e]} \{d^e\} = \underbrace{\int_{\partial\Omega} [M]^T \{t\}\, \mathrm{d}S + \int_{\Omega} [M]^T \left\{f^b\right\} \mathrm{d}V}_{\{f^{ext}\}}, \tag{1}$$

where Ω represents the domain of the element, $[B]$ is the strain-displacement matrix, $[D^e]$ is the linear elastic stiffness matrix, $[k^e]$ denotes the element stiffness matrix and $\{d^e\}$ is the element nodal displacement; $\partial\Omega$ represents the domain boundary of the element, $[M]$ contains the standard finite element shape functions, $\{t\}$ is the traction applied to boundary $\partial\Omega$, $\left\{f^b\right\}$ is the body forces and $\left\{f^{ext}\right\}$ denotes the external force vector.

The calculation of the stiffness matrix in the MPM is similar to that used in the standard FEM, except that material points, regardless of their positions inside the element of interest, along with their global volumes are used directly to perform the numerical integration. The element stiffness matrix is then calculated as

$$[k^e] \cong \sum_{i=1}^{n_{mp}} [B]_i^T [D^e][B]_i V_i, \tag{2}$$

where n_{mp} is the number of material points inside the element of interest and V_i is the volume associated with material point i.

Although $\left\{f^{ext}\right\}$ in (1) includes both tractions and body forces, only the calculation of body forces is presented at this stage as traction boundary conditions are detailed in Section 4 of this paper. The nodal body forces for an element containing material points are obtained through

$$\left\{f^{ext}\right\} \cong \sum_{i=1}^{n_{mp}} [M_i]^T \left\{f^b_{mp_i}\right\} V_i, \tag{3}$$

where $\left\{f^b_{mp_i}\right\}$ is the body force assigned to material point i. The element stiffness matrix and the element nodal force vector are assembled into the structure stiffness matrix $[K]$ and the structure force vector $\{f\}$ respectively. These form the linear system

$$[K]\{d\} = \{f\}, \tag{4}$$

by which the global vector of nodal displacements in the current loadstep, $\{d\}$, can be obtained.

After evaluating the nodal displacements, the strain and stress increments at material points resulting from the current load step can be computed by

$$\{\Delta\varepsilon\} = [B]\{d^e\} \quad \text{and} \quad \{\Delta\sigma\} = [D^e]\{\Delta\varepsilon\}. \tag{5}$$

Then the positions of the material points are updated. For material point i within element e, its displacement increment over the loadstep, $\{\Delta u_{mp}\}$, is found by mapping the nodal displacement $\{d^e\}$ to the particle through

$$\{\Delta u_{mp}\} = [M_i]\{d^e\}. \tag{6}$$

Global coordinates of the material points are then updated as

$$\{x_{mp}\}_{n+1} = \{x_{mp}\}_n + \{\Delta u_{mp}\}, \tag{7}$$

where n represents the number of load steps. The total displacement of a material point, $\{u_{mp}\}$, can be obtained by summing the incremental displacement, $\{\Delta u_{mp}\}$, over the n loadsteps. The strains and the stresses are updated in the same fashion at the end of the load step by using the strain and stress increments respectively, that is

$$\{\varepsilon_{n+1}\} = \{\varepsilon_n\} + \{\Delta\varepsilon\} \quad \text{and} \quad \{\sigma_{n+1}\} = \{\sigma_n\} + \{\Delta\sigma\}. \tag{8}$$

In order to have a stationary background mesh throughout the simulation, the positions of the mesh nodes are reset at the end of each load step. This leaves the deformed problem domain being represented by the new positions of the material points within the original background mesh.

The mapping of spatial variables from material points to grid nodes is a key aspect of the MPM. In this paper we adopt the standard mapping procedure using the basis functions of the background grid. As discussed by Sulsky and Gong [51] this is a special case of Shepard function interpolation and, in general, is only exact for constant functions and this mapping can limit the convergence rate of the method. However, the focus of this paper is on the imposition of boundary conditions in the MPM and not on the accuracy of the material point to grid mapping.

3. B-spline boundary conditions

As mentioned in the previous sections, a key feature that is missing from the standard MPM is the capacity to apply traction boundary conditions and this requires a method to track the evolving boundary of the physical domain. A local cubic B-spline interpolation method is chosen here to achieve the boundary representation and tracking. Full details of the construction of B-spline curves and the local cubic method can be found in [52,53]; only the essential equations are reviewed here.

3.1. B-spline basics

A pth-degree B-spline curve is defined using a vector with a sequence of non-decreasing real numbers (the knots)

$$\{\Xi\} = \left\{ \underbrace{0, \ldots, 0}_{p+1}, \xi_{p+1}, \ldots, \xi_{r-p-1}, \underbrace{1, \ldots, 1}_{p+1} \right\}, \tag{9}$$

where $r+1$ is number of knots and ξ is the local coordinate within the knot vector. Here we adopt an open knot vector, with $p+1$ repeated knots at the beginning and the end (normally taking the value of 0 and 1 respectively) of the knot vector, which ensures interpolation with the initial and final control points. Combining this knot vector with a set of n control points, $\{P_i\}$, a B-spline curve can be expressed as

$$\{C(\xi)\} = \sum_{i=0}^{n} N_{i,p}(\xi)\{P_i\}, \tag{10}$$

where $N_{i,p}(\xi)$ are the pth-degree B-spline basis functions defined recursively; for $p = 0$

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{if} \quad \xi_i \leqslant \xi < \xi_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

and for $p > 0$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi). \tag{12}$$

3.2. Local cubic interpolation for boundary representation

In order to impose boundary conditions in the MPM we first need to establish a B-spline boundary representation. The first step in developing a boundary representation technique for the standard MPM is to identify a set of boundary material points. These can be additional boundary tracking points[2] inserted along the desired boundaries or simply the outer layer of material points for the physical domain. Spline curve segments are then fitted between adjacent sampling points to calculate the control points. Finally, a suitable knot vector is determined and the overall cubic B-spline curve is formed by (10). It should be highlighted that the calculated control points are not convected with the physical deformation, instead they are recalculated based on the deformed material point coordinates at the start of each load or time step.

The reasons for choosing the local cubic interpolation method relate to the nature of the fitting and the order of the spline. In contrast to an approximation method which only gives an estimated fitting to the sampling points, the interpolation approach passes through the sampling points precisely. This means there is greater control on the B-spline represented boundaries and ensures that the spline passes through key boundary points. Although a global fitting [52] results in a curve with higher continuity, it is incapable of reproducing sharp corners which are important features in engineering problems. Local fitting, on the other hand, constructs curves in a piecewise fashion. Only local data are used at each step, so a fluctuation in data only affects the curve locally. More importantly, sharp corners can be handled properly by the local method. In terms of the spline order, the local cubic interpolation has a more straightforward formulation than the local quadratic interpolation [52]; no special cases or angle calculations are involved in the former method. Although the local quadratic interpolation is more robust when a large number of sampling points are used [53], the inability of this method to handle more complicated shapes that include sharp corners means that the local cubic interpolation method is adopted here. It should also be noted that changing the local B-spline fitting technique does not impact on the rest of the algorithm presented in this paper and a different B-Spline order could be adopted if appropriate.

3.3. Intersection of boundaries and the background mesh

Having described a method of boundary representation in the previous subsection, the issue of enforcing boundary conditions can now be discussed. Tractions are applied by integrating over the B-spline represented boundary directly, and prescribed displacements are imposed with the IBM [33]. In order to minimise numerical quadrature errors, integration is performed over the boundaries element by element. Therefore, a search algorithm is required to determine the intersection points of the boundary and the mesh. To find the values of ξ at the points where the interpolated boundary intersects with the background mesh, a search algorithm based on the Newton-Raphson method is adopted. As the focus of this paper is the imposition of boundary conditions, the search algorithm is not detailed here for the sake of brevity (see Bing [53] for details of the search algorithm).

[2] Boundary tracking points are essential material points with zero volume (or mass) so that they do not contribute to the stiffness or internal force calculations.

3.4. Neumann boundary conditions

With a robust boundary definition using B-splines, the application of traction to a boundary is straightforward. Recall that

$$\left\{f^t\right\} = \int_{\partial\Omega} [M]^T \{t\}\, \mathrm{d}S, \tag{13}$$

where $\left\{f^t\right\}$ is the resultant nodal force due to traction, $[M]$ contains the standard finite element shape functions and $\{t\}$ is the prescribed traction.

A pth-degree B-spline curve can be integrated numerically by using $(p-1)$th order Gauss quadrature [54]. The local coordinate of 1D Gauss quadrature has a range of $[-1, 1]$, whereas, the local coordinate of a B-spline curve has positive values only, therefore mapping between these two coordinate systems is required (in addition to the standard map between the local and global coordinates, or physical space). To allow for this, a separate space, called the parent domain [55], is introduced over which the numerical quadrature is conducted. Fig. 1 shows an illustration of the three spaces: the physical space, $\{x\}$, the parametric space, ξ, and the parent domain, $\tilde{\xi}$. In the physical space, the boundary geometry is defined in global coordinates. The parametric space contains the knots (local coordinates) which run along the curve from one end to the other. The parent domain is simply a local system where $\tilde{\xi} \in [-1, 1]$ on which numerical integration is performed.

To perform the integration, the desired B-spline curve segment is pulled back from the physical space to the parametric space. In other words, the local coordinates (ξ_j and ξ_{j+1}) of the start and the end point of the segment are identified by using their global coordinates. A linear transformation between the parent domain, $\tilde{\xi} \in [-1, 1]$, and the parametric space, $[\xi_j, \xi_{j+1}]$, is adopted to map the locations of Gauss points between these two spaces. By inspection, we can obtain the following link between the parent and parametric spaces

$$\xi = \xi_j + \frac{\left(\tilde{\xi} + 1\right)\left(\xi_{j+1} - \xi_j\right)}{2}. \tag{14}$$

As we are applying Gauss quadrature to integrate (13), a Jacobian mapping is required between the parent and physical spaces to map the Gauss quadrature lengths, which are defined over the local coordinates $\tilde{\xi} \in [-1, 1]$, to the physical space. Due to the use of a two local coordinate systems, the Jacobian contains two components

$$\{J_B\} = \left\{\frac{dC}{d\tilde{\xi}}\right\} = \left\{\frac{dC}{d\xi}\right\}\frac{d\xi}{d\tilde{\xi}}, \tag{15}$$

where

$$\frac{d\xi}{d\tilde{\xi}} = \frac{\xi_{j+1} - \xi_j}{2}. \tag{16}$$

Applying Gauss quadrature to (13), we obtain

$$\left\{f^t\right\} \cong \sum_{i=1}^{n_{gp}} [M_i]^T \{t\} \left\|\{J_B\}_i\right\| w_i, \tag{17}$$

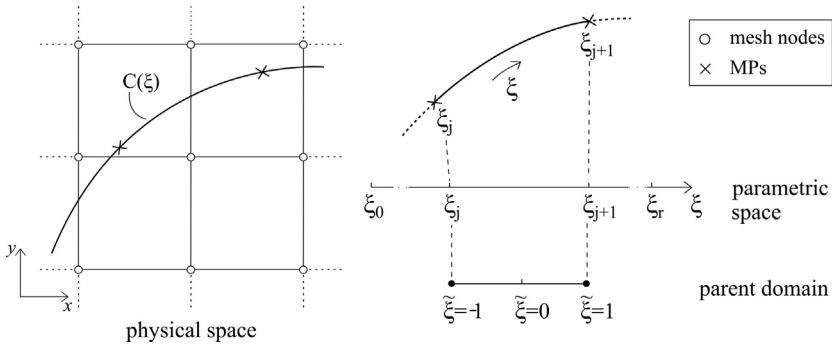


Fig. 1. Physical and parametric spaces for numerical integration of boundary conditions.

where n_{gp} is the number of Gauss points used to integrate over the segment within the background grid element, w_i is the weight associated with Gauss point i, $\|(\cdot)\|$ denotes the L2 norm of $(\cdot)$ and in this case the L2 norm of the Jacobian, $\{J_B\}$, which maps the length of the boundary between the parent and physical spaces.

3.5. Dirichlet boundary conditions

The methodology behind the IBM [40–42] is that essential boundary conditions are enforced by introducing extra stiffness to the system via the use of Dirichlet functions (D-functions) which constrain displacement on essential boundaries.

The approximation for displacements used in the IBM is

$$\{u\} = [D][M]\{d^e\} + \{u^a\}, \tag{18}$$

where $\{u^a\}$ contains the essential boundary conditions and $[D]$ is a diagonal matrix that contains the D-functions, i.e. $[D] = diag(D_i)$ $(i = 1, \ldots, n_d)$ and n_d is the dimensionality. Deriving the weak form of equilibrium by using (18) alters the element stiffness matrix $[k^e]$, more specifically, the entries within the conventional strain-displacement matrix, $[B]$. For 2D plane stress and strain problems using the IBM, the strain displacement matrix is a $3 \times 2n_{en}$ matrix that takes the following form, with $k = 1, \ldots, n_{en}$ and n_{en} is the number of nodes per element,

$$[\overline{B}_k] = \begin{bmatrix} D_1 \frac{\partial M_k}{\partial x} + M_k \frac{\partial D_1}{\partial x} & 0 \\ 0 & D_2 \frac{\partial M_k}{\partial y} + M_k \frac{\partial D_2}{\partial y} \\ D_1 \frac{\partial M_k}{\partial y} + M_k \frac{\partial D_1}{\partial y} & D_2 \frac{\partial M_k}{\partial x} + M_k \frac{\partial D_2}{\partial x} \end{bmatrix}, \tag{19}$$

where the bar has been included to distinguish the matrix from the conventional strain-displacement matrix, $[B]$. Examples of suitable D-functions can be found in [33,40,41]. Here we follow the approach of Cortis et al. [33] and adopt a D-function of the form

$$D(\phi) = \begin{cases} 0 & \phi < 0 \\ 1 - \left(1 - \frac{\phi}{\delta}\right)^2 & 0 \leqslant \phi \leqslant \delta \\ 1 & \phi > \delta. \end{cases} \tag{20}$$

An illustration of the variation of $D(\phi)$ over the bandwidth is shown in Fig. 2. This D-function provides an approximation of a step function using a signed distance function ϕ which has a value of zero on the boundary of interest, and $\phi < 0$ indicates the exterior whereas $\phi > 0$ implies the interior of the problem domain (see Fig. 3). When $\phi < 0, D(\phi) = 0$ and no stiffness integration is performed over the mesh that is outside of the domain as no material points should be outside of the domain. When $\phi > \delta, D(\phi) = 1$ which allows $[B]$ to return to its regular form.

The stiffness matrix for the IBM can be obtained by decomposing $[\overline{B}]$ into two matrices ($[B_1]$ which contains the gradient of the shape functions and $[B_2]$ which contains the gradient of the D-functions), which allows the stiffness matrix be expanded as

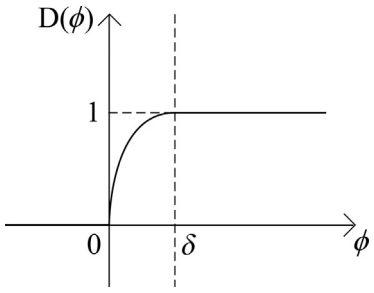


Fig. 2. D-function variation as a function of distance from the boundary, ϕ, where the physical domain resides in the region $\phi > 0$.

$$[k^e] \cong \underbrace{\int_\Omega [B_1]^T [D^e][B_1]\, dV}_{[k_1^e]} + \underbrace{\int_\Omega [B_1]^T [D^e][B_2]\, dV}_{[k_2^e]} + \underbrace{\int_\Omega [B_2]^T [D^e][B_1]\, dV}_{[k_2^e]^T} + \underbrace{\int_\Omega [B_2]^T [D^e][B_2]\, dV}_{[k_3^e]}, \tag{21}$$

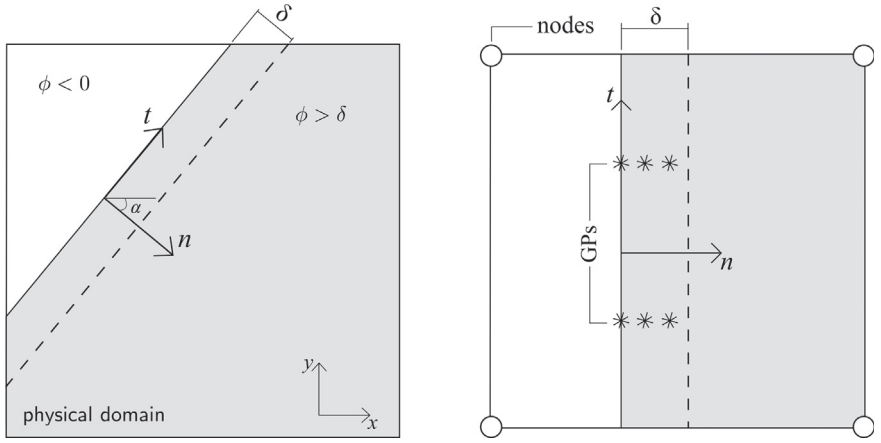
where for a single node, k, of an element containing an implicit boundary the $[B_1]$ and $[B_2]$ matrices are given by

$$[B_{1k}] = \begin{bmatrix} D_1 \frac{\partial M_k}{\partial x} & 0 \\ 0 & D_2 \frac{\partial M_k}{\partial y} \\ D_1 \frac{\partial M_k}{\partial y} & D_2 \frac{\partial M_k}{\partial x} \end{bmatrix} = \begin{bmatrix} D_1 & 0 & 0 & 0 \\ 0 & D_2 & 0 & 0 \\ 0 & 0 & D_1 & D_2 \end{bmatrix} \begin{bmatrix} \frac{\partial M_k}{\partial x} & 0 \\ 0 & \frac{\partial M_k}{\partial y} \\ \frac{\partial M_k}{\partial y} & 0 \\ 0 & \frac{\partial M_k}{\partial x} \end{bmatrix} = [\overline{D}_1][\overline{B}_{1k}], \tag{22}$$

and

$$[B_{2k}] = \begin{bmatrix} M_k \frac{\partial D_1}{\partial x} & 0 \\ 0 & M_k \frac{\partial D_2}{\partial y} \\ M_k \frac{\partial D_1}{\partial y} & M_k \frac{\partial D_2}{\partial x} \end{bmatrix} = \begin{bmatrix} \frac{\partial D_1}{\partial x} & 0 \\ 0 & \frac{\partial D_1}{\partial y} \\ \frac{\partial D_1}{\partial y} & \frac{\partial D_1}{\partial x} \end{bmatrix} \begin{bmatrix} M_k & 0 \\ 0 & M_k \end{bmatrix} = [\overline{D}_2][\overline{B}_{2k}]. \tag{23}$$

$[k_1^e]$ can be viewed as the regular stiffness matrix when $\phi > \delta$ and $[k_2^e]$ and $[k_3^e]$ contribute additional stiffness terms that only exist if the element contains an essential boundary. As proposed in [33], $[k_2^e]$ and $[k_3^e]$ can be determined by using a local coordinate system based on the tangent and the normal of the boundary, i.e. (t, n), as



(a) Inclined boundary. (b) Integration point positions.

Fig. 3. Implicit boundary coordinate system and integration scheme.

$$[k_2^e] = \int_S [B_1]^T [D^e][B_2]\,\mathrm{d}S = \int_t [\overline{B}_1]^T \left(\int_0^\delta [\overline{D}_1][D^e][\overline{D}_2][J_T]\,\mathrm{d}n \right) [\overline{B}_2]\,\mathrm{d}t, \tag{24}$$

$$[k_3^e] = \int_S [B_2]^T [D^e][B_2]\,\mathrm{d}S = \int_t [\overline{B}_2]^T \left(\int_0^\delta [J_T]^T [\overline{D}_2][D^e][\overline{D}_2][J_T]\,\mathrm{d}n \right) [\overline{B}_2]\,\mathrm{d}t, \tag{25}$$

where $[J_T]$ is a transformation matrix that provides the mapping between (t,n) and (x,y). Because $[\overline{D}_1]$ only contains the D-functions which are invariant with respect to the coordinate system while $[\overline{D}_2]$ is populated by the derivatives of the D-functions with respect to (x,y), and the latter has a significantly larger value than the former, $[J_T]$ is only applied to transform $[\overline{D}_2]$. The transformation matrix has been shown in [33] to be

$$[J_T] = \begin{bmatrix} \frac{\partial n}{\partial x} & \frac{\partial n}{\partial y} \\ \frac{\partial t}{\partial x} & \frac{\partial t}{\partial y} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}, \tag{26}$$

where θ, as indicated in Fig. 3a, is the angle required to align direction n with x and t with y.

Roller boundary conditions, permitting the material to have free movement in the tangential direction, requires D_2 to have a constant value of 1 throughout the band, while D_1 depends on (20).

Incorporating the IBM with the B-spline represented boundaries only requires a separate calculation of $[k_2^e]$ and $[k_3^e]$ in addition to $[k_1^e]$. In most of the examples shown later, the additional stiffness matrices are evaluated by using the Gauss quadrature scheme shown in Fig. 3b for each element. The Gauss points that are placed on the boundary are responsible for the outer integral of (24) and (25) and the Gauss points that are placed across the band are used to perform the inner integral. In this paper we adopt two-point Gauss quadrature in the tangential direction and three-point quadrature in the normal direction; giving a total of six Gauss points for each boundary segment within each element (see Fig. 3b). For boundaries with high curvature it may be necessary to increase the quadrature order in the tangential direction.

Applying Gauss quadrature to (24) and (25), $[k_2^e]$ and $[k_3^e]$ can be approximated as

$$[k_2^e] \cong \sum_{i=1}^{n_{gp1}} [\overline{B}_1]^T \left(\sum_{j=1}^{n_{gp2}} [\overline{D}_1]^T [D^e][\overline{D}_2][J_T] \det\left([J_D]_j\right) w_j \right) [\overline{B}_2] \det([J_B]_i) w_i, \tag{27}$$

and

$$[k_3^e] \cong \sum_{i=1}^{n_{gp1}} [\overline{B}_2]^T \left(\sum_{j=1}^{n_{gp2}} [J_T]^T [\overline{D}_2]^T [D^e][\overline{D}_2][J_T] \det\left([J_D]_j\right) w_j \right) [\overline{B}_2] \det([J_B]_i) w_i, \tag{28}$$

respectively. $[J_T]$ is the transformation matrix formulated by (26); $[J_D]$ provides the mapping between the implicit boundary coordinate n and the Gauss quadrature coordinate; $[J_B]$ is the link between the global coordinates and the boundary integration domain (15); w indicates the weights associated with the Gauss points, and n_{gp1} and n_{gp2} are the number of Gauss points along the boundary and across the band respectively.

One thing to note here is that in order to compute the angle θ within $[J_T]$, the inward normal, $\{n\}$, of the B-spline boundary at the Gauss point positions is required. $\{n\}$ of a point with local coordinate ξ and global coordinates $\{C(\xi)\} = [C_x, C_y]$ on a B-spline curve takes the following form [56],

$$\{n\} = \begin{Bmatrix} n_x \\ n_y \end{Bmatrix} \quad \text{with} \quad n_x = \frac{1}{J(\xi)} \frac{\mathrm{d}C_y}{\mathrm{d}\xi}, \quad n_y = \frac{-1}{J(\xi)} \frac{\mathrm{d}C_x}{\mathrm{d}\xi}. \tag{29}$$

Computation of the derivatives of the B-spline curve with respect to the knot coordinate, ξ, in $(29)_2$ and $(29)_3$ can be found in [52], and the Jacobian transformation $J(\xi)$ is defined as

$$J(\xi) = \sqrt{\left(\frac{\mathrm{d}C_x}{\mathrm{d}\xi}\right)^2 + \left(\frac{\mathrm{d}C_y}{\mathrm{d}\xi}\right)^2}. \tag{30}$$

3.5.1. Inhomogeneous Dirichlet boundary conditions

Inhomogeneous (or non-zero) Dirichlet boundary conditions introduce an additional term into the external force vector [41,42]

$$\left\{f^{ext}\right\} = \int_{\partial\Omega} [M]^T \{t\}\,\mathrm{d}S + \int_\Omega [M]^T \left\{f^b\right\} \mathrm{d}V - \int_\Omega [\overline{B}]^T \{\sigma^a\} dV, \tag{31}$$

where $\{\sigma^a\} = [D^e]\{\varepsilon^a\}$ is the stress associated with the inhomogeneous Dirichlet boundary condition, $\{u^a\}$, which is imposed through nodal displacements, $\{d^a\}$, $\{\varepsilon^a\} = [B]\{d^a\}$ is the strain associated with Dirichlet boundary condition and $[B]$ is the conventional strain-displacement matrix. This additional inhomogeneous Dirichlet force can be expressed as

$$\left\{f^d\right\} = \int_\Omega [\overline{B}]^T \{\sigma^a\} dV = \int_\Omega [\overline{B}]^T [D^e][B]\{d^a\} dV \tag{32}$$

and expanding (32) gives

$$\left\{f^d\right\} = \int_\Omega [[B_1] + [B_2]]^T [D^e][B]\{d^a\} dV, \tag{33}$$

which can be evaluated in a similar way to the stiffness integrals (27) and (28). An important aspect of imposing inhomogeneous Dirichlet boundary conditions is the determination of the nodal displacements, $\{d^a\}$. It is straightforward to determine the required displacements if the entire boundary segment is subjected to the same non-zero boundary condition.[3] However, in the case where the required $\{u^a\}$ varies along the boundary, an error minimisation technique must be adopted to globally minimise the difference between the required displacement solution along the boundary, $\{u^a\}$, and the finite element approximation, $\{u^h\} = [M]\{d^a\}$. In this paper we minimise the following residual integrated over the B-spline boundary

$$\int_{\partial\Omega} \|\{R\}\| dS = \int_{\partial\Omega} \|\{u^h\} - \{u^a\}\| dS, \tag{34}$$

using a least squares weighted residual technique

$$\int_{\partial\Omega} \left[\frac{\partial R}{\partial d^a}\right] \{R\} dS = \int_{\partial\Omega} [M]^T \{[M]\{d^a\} - \{u^a\}\} dS = \{0\}. \tag{35}$$

Rearranging in terms of the unknown nodal displacements, $\{d^a\}$, gives

$$\{d^a\} = \left[\int_{\partial\Omega} [M]^T [M] dS\right]^{-1} \left\{\int_{\partial\Omega} [M]^T \{u^a\} dS\right\}, \tag{36}$$

where the integrals can be approximated using Gauss quadrature. Note that $\{d^a\}$ must be evaluated globally, not on an element by element basis, in order to minimise the residual across the entire boundary. This requires the integrals in (36) to be assembled into a global matrix and vector, respectively.

4. Numerical examples

This section will present a series of numerical examples to validate the method of imposing boundary conditions described in the previous section. Body forces are not included in the analyses as the focus of this paper is on the imposition of Neumann and Dirichlet boundary conditions.

The performance of the methods is measured by errors between the numerical results and the analytical solutions. Relative stress error R_{σ_i} at material point i is computed as

$$R_{\sigma_i} = \frac{|(\sigma_i^h - \sigma_i^a)|}{|\sigma_i^a|}, \tag{37}$$

where σ^h and σ^a are one component of the numerical and analytical stresses, and $|(\cdot)|$ is the absolute value of $(\cdot)$. Similarly, error of displacements at a material point can be calculated as

$$R_{d_i} = \frac{|\left(d_i^h - d_i^a\right)|}{|d_i^a|}, \tag{38}$$

where d^h and d^a are one component of the numerical and analytical displacements, respectively. Note that u and v are used later to indicate horizontal and vertical displacements respectively. Absolute errors are used when the analytical results contain zero values in the domain, which are

$$A_{\sigma_i} = |(\sigma_i^h - \sigma_i^a)| \quad \text{and} \quad A_{d_i} = |\left(d_i^h - d_i^a\right)|. \tag{39}$$

[3] In that case the essential boundary displacements of the nodes associated with the elements containing the boundary are simply set to the required value.

In order to have an overview of the performance, global errors over the problem domains are used. Measures at material point i are weighted by the global volume of the material point, V_{g_i}, before summing together and then the error summation is divided by the total volume of the problem domain, $V_G = \sum_{i=1}^{n_{mp}} V_{g_i}$. For example, the global error in a particular stress component is calculated as

$$\overline{R}_\sigma = \frac{1}{V_G} \sum_{i=1}^{n_{mp}} R_{\sigma_i} V_{g_i}. \tag{40}$$

Four numerical examples are presented in the following subsections:

1. compression of a 2D square: to validate the imposition of Dirichlet boundary condition for boundaries both aligned with and inclined to the background mesh. The example covers both roller displacement boundary conditions and non-zero tractions and investigates convergence with: background element size, number of material points and the boundary width, δ;
2. cantilever beam under an applied traction: to demonstrate the imposition of Neumann (traction) boundary conditions over an evolving boundary;
3. pressurised thick-walled cylinder: to validate the imposition of Neumann boundary conditions over a curved boundary and to demonstrate convergence with: background element size and number of material points; and
4. circular hole in an infinite plate: to validate the imposition of complex Neumann and inhomogeneous Dirichlet boundary conditions, involving spatially-varying non-normal tractions, and displacements, and demonstrate convergence with background element size.

4.1. Compression of a 2D square domain

The first numerical example is the one-dimensional compression of a 2D plane strain square domain, as shown in Fig. 4a. The material had a Young's modulus of $E = 1$ MPa and Poisson's ratio of $\nu = 0.25$. Three sides of the square were subjected to roller boundary conditions and a constant traction of 8 kPa imposed perpendicular to the fourth edge over a single load step. This example has been included to investigate the errors and convergence of the proposal B-spline boundary method under different combinations of boundary conditions, namely:

4.1.1. physical domain aligned with the background grid (0° - model, Fig. 4a): conventional Dirichlet roller boundary conditions on three sides and B-spline applied traction on the fourth; and
4.1.2. implicit homogeneous Dirichlet boundary conditions to impose the roller boundary conditions on three sides of the problem domain and B-spline applied normal traction on the fourth boundary.

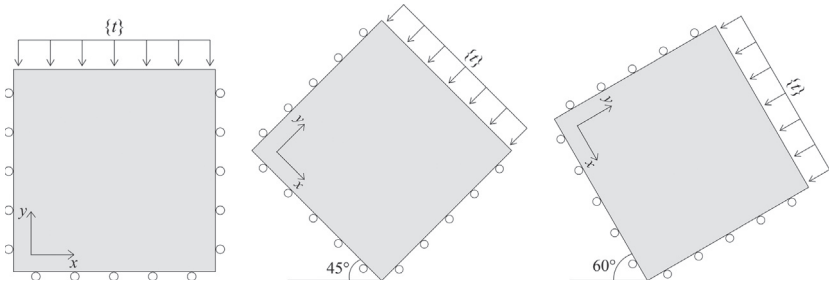
This is a constant stress problem and the stress over the whole domain should be equal to

$$\left\{\sigma_{xx} \quad \sigma_{yy} \quad \sigma_{xy}\right\}^T = 8 \times \left\{3^{-1} \quad 1 \quad 0\right\}^T \text{kPa} \tag{41}$$

with the strain in the y-direction of $\varepsilon_{yy} = 6.67 \times 10^{-3}$ and zero strain in the other directions.

4.1.1. Conventional Dirichlet and B-spline Neumann boundary conditions

Fig. 5a shows the initial discretisation of the problem with a domain size of 1 m by 1 m. A 2 by 2 background mesh of quadrilateral elements with $h = 0.5$ m and 4 material points per element



(a) 0° model. (b) 45° model. (c) 60° model.

Fig. 4. Uniaxial compression of a square.

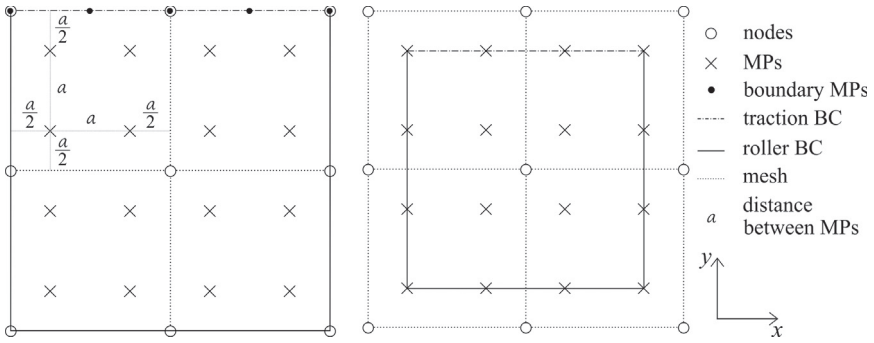
was used to model the domain. These material points were positioned such that they had the same distance, $a = h/2$, between the adjacent material points and $a/2$ between the mesh and material points that were located closest to the mesh. All boundaries were coincident with the mesh. The roller boundary conditions were imposed directly on the background mesh and 5 zero-volume boundary material points were inserted to form a B-spline description of the boundary, on which the traction was applied (marked by the dot-dash line in Fig. 5a).

The average errors in $\bar{R}_{\sigma_{xx}}$, $\bar{R}_{\sigma_{yy}}$ and $\bar{A}_v$ all achieved machine precision (order of 10^{-16} for the stress errors and 10^{-18} for the displacements, despite only using 16 material points), indicating that the traction boundary conditions have been implemented correctly.

4.1.2. Implicit homogeneous Dirichlet and B-spline Neumann boundary conditions

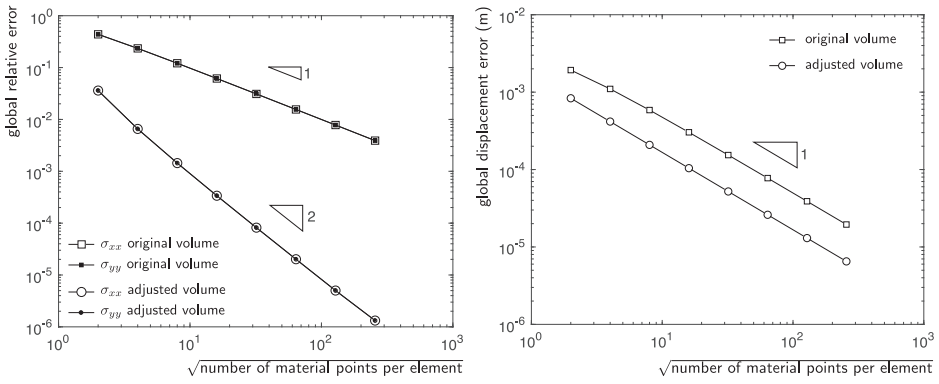
As this is a constant stress problem, the size of the problem domain does not have an impact on the comparison of results. To test the B-spline based IBM, the problem boundaries were modified to coincide with the outer layer of the material points, as shown in Fig. 5b. Due to the intersection of the boundary with the material points, volumes of the material points on the boundaries were adjusted accordingly to maintain the correct overall volume. That is, material points at the domain corners had their volumes set to a quarter of their original volumes, and the volumes of the other boundary material points were halved.

Convergence plots of the average stress and displacement errors with respect to the number of material points in each direction per element with $h = 0.5$ m are shown in Fig. 6. Results are shown for



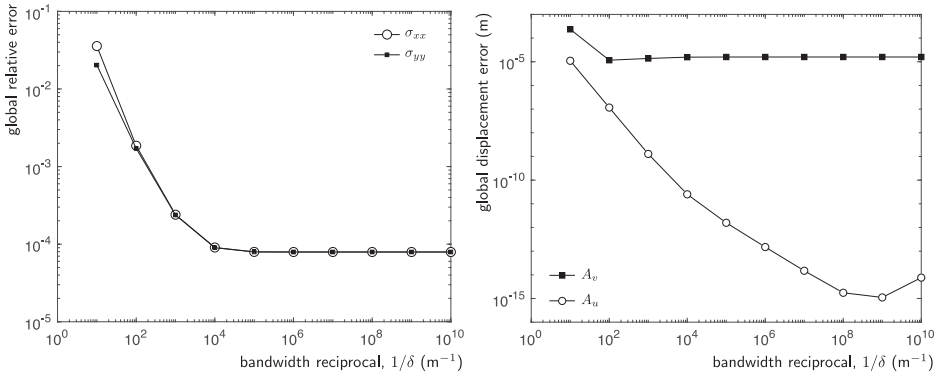
(a) Mesh-coincident boundaries. (b) MP-coincident boundaries.

Fig. 5. Discretisation of the 2D square domain.



(a) $\bar{R}_{\sigma_{xx}}$ and $\bar{R}_{\sigma_{yy}}$. (b) $\bar{A}_v$.

Fig. 6. Implicit homogeneous Dirichlet with B-spline Neumann boundary conditions for 2D compression (0° model): convergence with material points/element.



(a) $\bar{R}_{\sigma_{xx}}$ and $\bar{R}_{\sigma_{yy}}$. (b) $\bar{A}_u$ and $\bar{A}_v$.

Fig. 7. Implicit homogeneous Dirichlet with B-spline Neumann boundary conditions for 2D compression (0° model): $1/\delta$ convergence with 32^2 material points/element.

Table 1
Implicit homogeneous Dirichlet with B-spline Neumann boundary conditions for 2D compression: global relative errors.

Model	45°	60°
$\overline{R}_v$	4.84×10^{-7}	4.90×10^{-7}
$\overline{R}_{\sigma_{yy}}$	9.68×10^{-9}	7.25×10^{-9}

both the case where the original material point volumes are used and for where the volumes are adjusted as described above. For stress errors, adjusting the material point volumes increases the convergence rate as the discretised domain maintains the correct volume. The convergence rates of the displacement errors are the same but the simulation with the correct volume starts at, and maintains, a lower overall error.

Fig. 7a and b illustrates the convergence rates of stress and displacement errors with respect to δ for the 0° model with 32^2 material points per element. Errors of σ_{xx} and σ_{yy} converge to a value of 8.18×10^{-5}, and $\delta < 10^{-6}$ m does not reduce the stress errors. $\overline{A}_u$ and $\overline{A}_v$ reach their minimum with a value of 1.20×10^{-5} when $\delta = 10^{-2}$ m and 1.14×10^{-15} when $\delta = 10^{-9}$ m respectively. Vertical displacement error settles to 1.64×10^{-5} for $\delta < 10^{-5}$ m; whereas, horizontal displacement error starts increasing after the minimum because the impact on the minor changes to the conditioning of the stiffness matrix on results that are close to machine precision. Conducting the same analysis with various numbers of material points per element found that to achieve the minimum stresses errors, the optimum bandwidth needs to be reduced when the number of material points increases. As for displacement errors, there was little variation in the optimum bandwidth when the number of material points was changed. Therefore, consistent with the work of Cortis et al. [33], a δ/h value of the order of 10^{-6} is recommended.

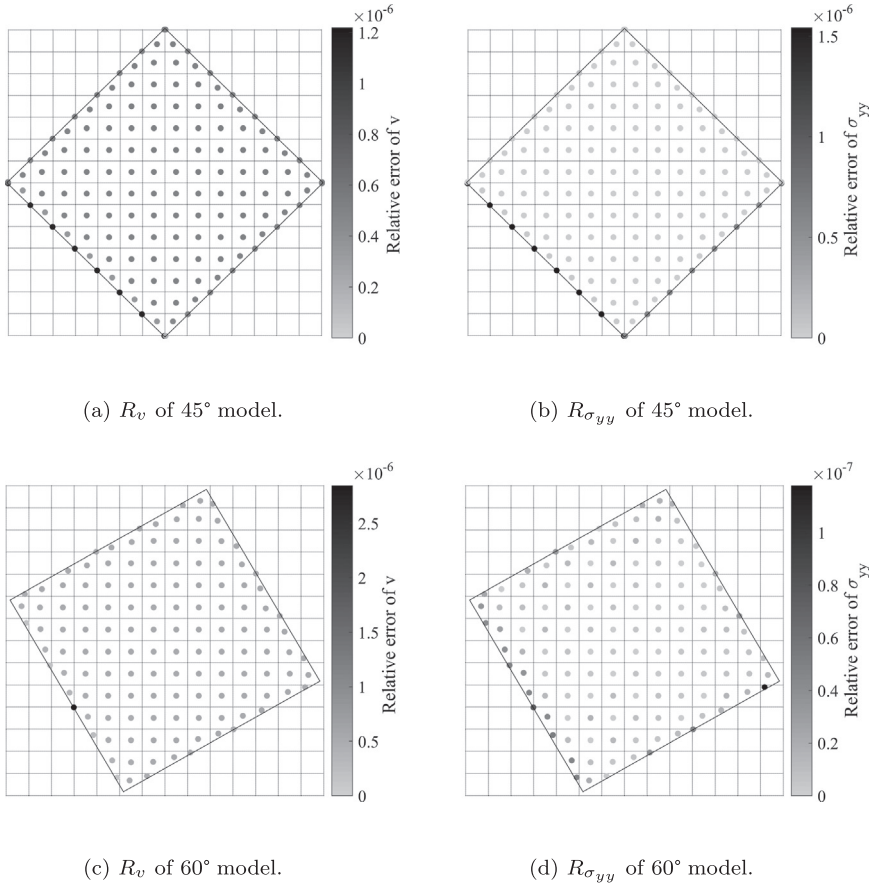
To test the ability of the B-spline based IBM on modelling inclined boundaries, the problems shown in Fig. 4b and c were analysed using the discretisation of Cortis et al. [33]. In this case the problem domain was 100 mm by 100 mm and the background mesh consisted of $14^2, h = 10$ mm elements with one material point per element in the physical domain. Additional zero-volume boundary points were introduced along the boundaries to form B-spline descriptions and the roller boundary conditions were imposed using the IBM. Global relative errors of displacement v and stress σ_{yy} for both models using the new discretisation are shown in Table 1. Error distributions illustrated in Fig. 8 indicate that both roller and traction boundary conditions have been applied successfully on the inclined boundaries. Consistent with the findings of Cortis et al. [33], the larger errors are located in partially-filled elements where errors are introduced through non-optimum placement of the material points in terms of the stiffness integration.

Convergence analyses of the displacement and stress errors with respect to the bandwidth, δ, for both rotated models were performed and the results are shown in Fig. 9. Average stress error, $\overline{R}_{\sigma_{yy}}$, of the 45° model reaches its minimum when $\delta = 10^{-6}$ mm; whereas, $\overline{R}_v$ reaches its minimum with $\delta = 10^{-5}$ mm. The same pattern also occurs for the 60° model: the optimum bandwidth for σ_{yy} is 10^{-6} mm and that for v is 10^{-5} mm. All errors converge at relatively the same rate towards their minimum and then diverge after the optimum bandwidth. The divergence shown is due to the increasing condition number of the stiffness matrix as the bandwidth reduces. Therefore, in accordance with the values suggested in Cortis et al. [33] and with the results from the previous section, a relative bandwidth of approximately $\delta/h = 10^{-6}$ is recommended.

4.2. Cantilever beam with applied traction

The second example is of a plane strain cantilever beam subjected to a uniform traction that remains perpendicular to the top surface of the beam throughout the analysis (as shown in Fig. 10). There is no analytical solution for this problem but it is included to demonstrate the boundary representation abilities of the proposed method. In particular, this example includes a perpendicular traction on an evolving boundary. The cantilever beam had a length of 10 m and a depth of 2 m and the material had a Young's modulus of $E = 1$ MPa and Poisson's ratio of $v = 0.25$. The left hand side boundary was fixed at the middle and rollers above and below. A constant pressure of 1500 Pa was applied along the top boundary.

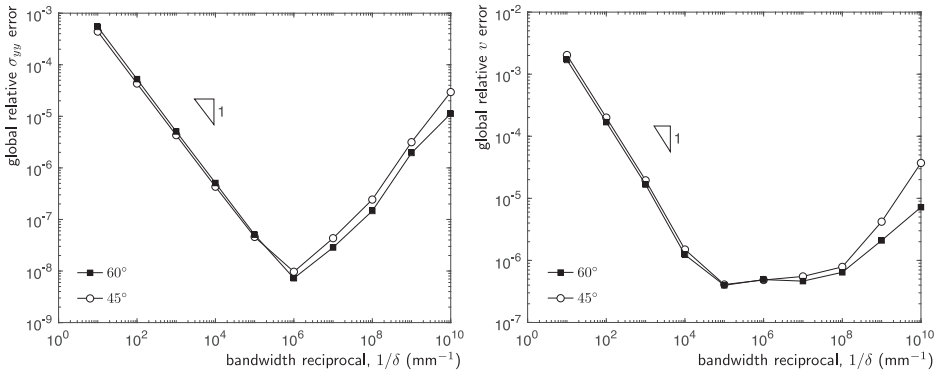
A background mesh with 1.5 m by 1.5 m elements was used, and the problem domain was discretised using 896 uniformly distributed material points (arranged in a regular 56 by 16 grid as shown in Fig. 11a). The material points were positioned such that none of the boundaries coincided with the mesh; the x and y coordinates of the material points were perturbed by 0.25 and 0.50 from the background mesh position. The outer layer of the material points were identified as the problem boundaries which were approximated using B-splines. boundary conditions on the left boundary ware applied by the B-spline based IBM and the pressure applied through 5 equal load steps. The initial discretisation and the final deformed cantilever beam are shown in Fig. 11a. The



(a) R_v of 45° model. (b) $R_{\sigma_{yy}}$ of 45° model.

(c) R_v of 60° model. (d) $R_{\sigma_{yy}}$ of 60° model.

Fig. 8. Implicit homogeneous Dirichlet with B-spline Neumann boundary conditions for 2D compression: error distribution.



(a) $\bar{R}_{\sigma_{yy}}$. (b) $\bar{R}_v$.

Fig. 9. Implicit homogeneous Dirichlet with B-spline Neumann boundary conditions for 2D compression: error convergence plots vs. bandwidth $1/\delta$.

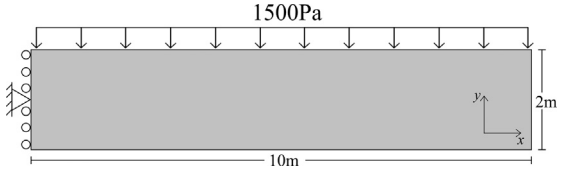
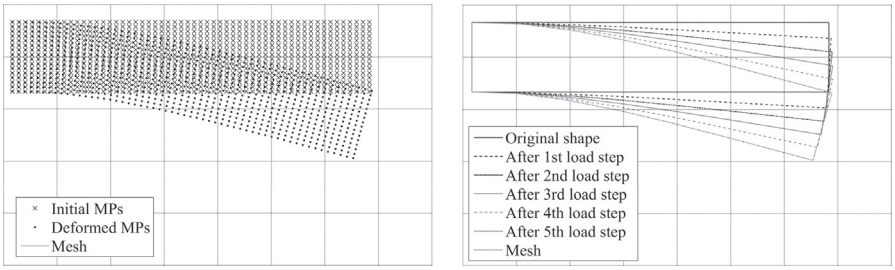


Fig. 10. Cantilever beam geometry.

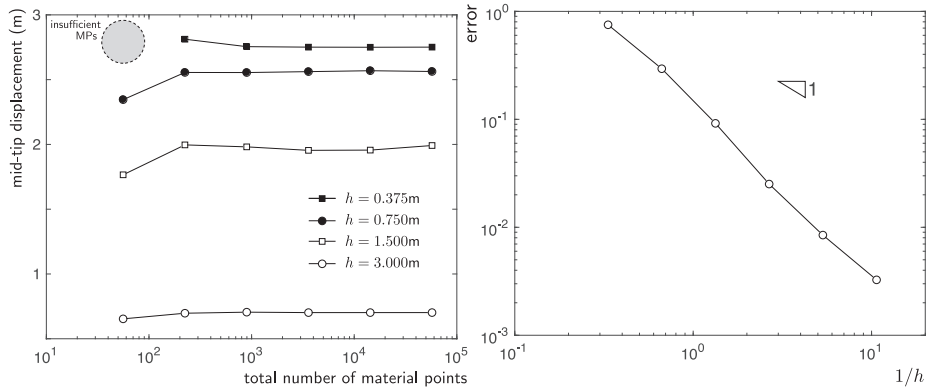
advantage of this approach is that boundaries can be tracked after each load step without plotting out all the material points (see Fig. 11b) and that tractions can be applied to the deformed B-spline boundary over a number of loadsteps. Fig. 11b also shows that the deformed shape has been successfully captured by the B-spline approximation.

The material point method presented in this paper makes the assumption of infinitesimal strains, resulting in an incrementally linear method but global response that is non-linear due to updating the position of the material points at the end of each linear load step. Therefore it is not possible to compare the results of this analysis with a finite deformation solution of this problem. Instead, a convergence test on the displacement at the centre of the beam tip was carried out by performing both *h*-refinement of the background mesh and increasing the number of material points



(a) final deformation of the material points. (b) boundary visualisation after each load step.

Fig. 11. Cantilever beam deformation.



(a) material point refinement (b) convergence with h refinement

Fig. 12. Cantilever beam: convergence of mid-tip displacement with (a) increasing numbers of material points for $h = 3.0, 1.5, 0.75$ and 0.375 m and (b) convergence with h refinement..

representing the problem domain. As shown in Fig. 12a, for all four mesh configurations, the mid-tip displacement converges with increasing numbers of material points.[4] However, it is important to note that increasing the number of material points reduces the errors associated with integrating the internal force and stiffness of the material and, unlike h refinement with optimally integrated finite elements (via Gauss-Legendre quadrature, for example), monotonic convergence is not guaranteed, as shown by the $h = 1.5$ m analyses.

As there is no analytical solution to this problem it is only possible to demonstrate the convergence of the method by comparing the material point tip displacement with a reference solution. In this case the reference solution was determined through analysing the problem with a bi-linear four noded quadrilateral finite element mesh with $h = 2^{-7}$ m (327,680 elements arranged in a regular 1280×256 grid). The elements were integrated using 2 by 2 Gauss quadrature and the traction was imposed directly on the upper boundary of the elements. The mid-tip vertical downwards displacement from this analysis was 2.8220 m, which was taken as the *reference* value in order to determine the convergence of the material point analysis which is shown in Fig. 12b. The material point analyses are clearly converging towards the reference finite element solution. Note that, due to the incrementally linear nature of the formulation, the final value of displacement will depend on the number of load steps used to analyse the problem.

To further demonstrate the ability of the method to applied Neumann boundary conditions to problems with significant deformation, Fig. 13 shows the deformed material point positions coloured according to σ_{xx} for $h = 0.75$ m and 28×8 material points. In this case the surface traction was increased to 3000 Pa and applied over 10 loadsteps and the mid-tip displacement at the end of the analysis was 5.09 m.

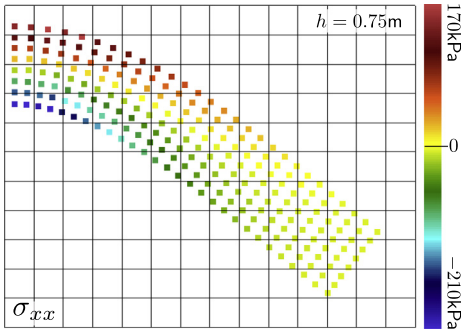
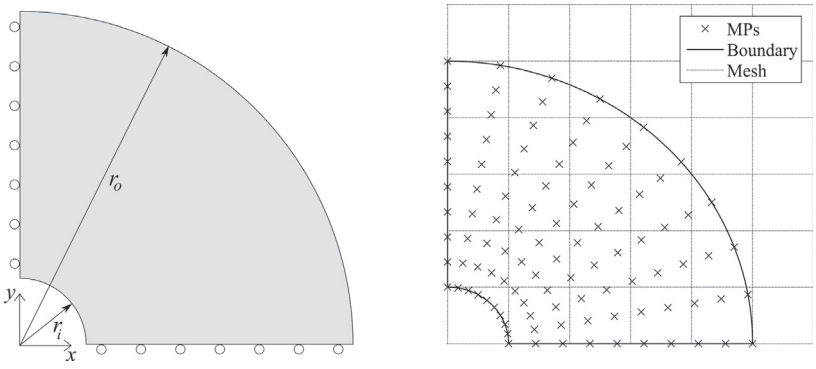


Fig. 13. Cantilever beam: deformed material point positions coloured according to σ_{xx} for $h = 0.75$ m, 28×8 material points and a surface traction of 300 Pa.

4.3. Internally pressurised thick-walled cylinder

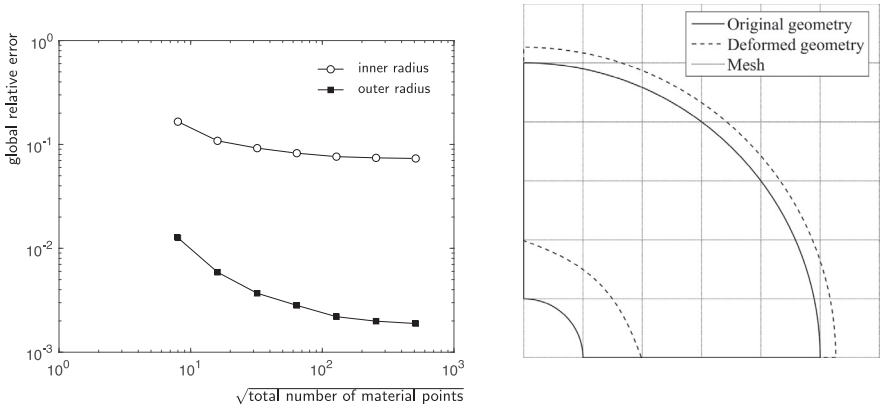
The next example is an internally pressurised plane strain thick-walled cylinder. Only a quarter of the cross-section was analysed due to symmetry; the problem domain and boundary conditions are shown in Fig. 14a where $r_i = 1$ m and $r_o = 5$ m. The material had a Young's modulus of $E = 1$ MPa and Poisson's ratio of $\nu = 0.25$. Using the analytical solution [57], it was calculated that an internal pressure of 0.756 MPa would double the inner radius,

[4] Note that for the $h = 0.375$ m mesh, 56 material points (arranged in a 14 by 4 grid) is insufficient to analyse the problem, as shown by the grey-dashed circle in Fig. 12a.

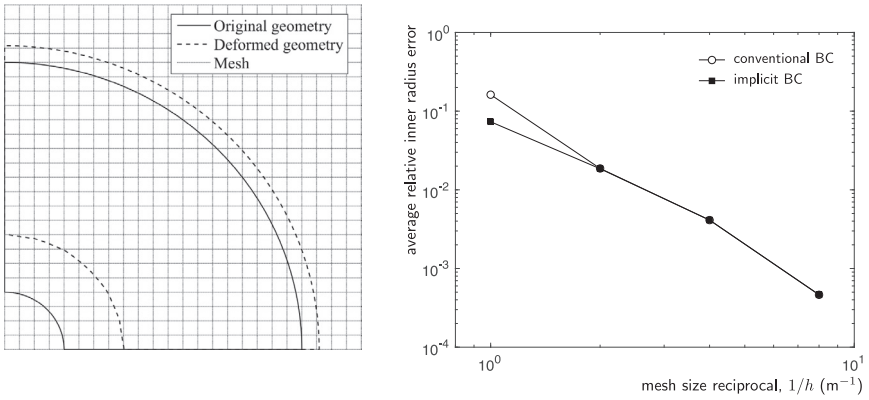


(a) Problem definition. (b) Domain discretisation.

Fig. 14. Thick-walled cylinder: geometry and discretisation.



(a) MP convergence with $h = 1$m. (b) $h = 1$m original/deformed geometry.



(c) $h = 0.25$m original/deformed geometry. (d) h-refinement convergence.

Fig. 15. Thick-walled cylinder: convergence and deformation.

and this pressure was then applied in the simulation in a single load step.

To discretise the cylinder, the same number of material points were placed uniformly along the circumferences and across the radii on a 6 m by 6 m background mesh with 1 m by 1 m elements. Fig. 14b illustrates the discretisation with 10^2 material points. Boundaries were approximated by interpolating the outer layer of material points using cubic B-splines.

Average relative errors of the deformed inner and outer radii are used to demonstrate convergence. Numerical values of the deformed inner and the outer radii were found by sampling at 1000 points along the deformed B-spline curve and calculating the distance between each point and the origin. Starting with 8^2 material points, the average relative error of both the inner and outer radii converge when the number of material points was increased (see Fig. 15a). However, the inner radius shows an error almost two orders of magnitude higher than that seen at the outer radius. This is because of the higher curvature within the inner elements which is not captured by the linear basis of the background mesh, whereas, the outer boundary has lower curvature within

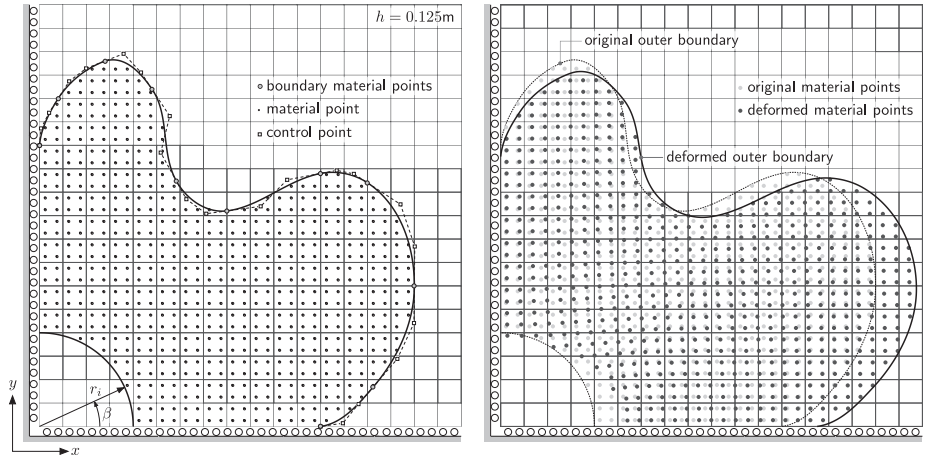


Fig. 16. Infinite plate with a circular hole and B-spline outer boundary: initial problem definition and deformed material point positions and outer boundary for $h = 0.125$ m and 2^2 material points per element.

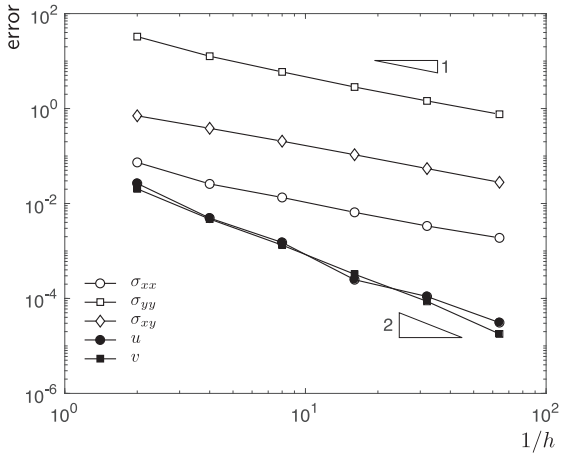


Fig. 17. Infinite plate with a circular hole (Neumann): convergence under uniform h-refinement with 8^2 material points per element.

each element. Support for this explanation is provided in Fig. 15b and c which show the original and deformed inner and outer boundaries with $h = 1$ m and $h = 0.25$ m, respectively (128^2 material points were used for both analyses). It is visually clear that a more accurate simulation of the deformation at the inner radius is achieved with a finer mesh.

To quantify the errors in the analyses, a convergence study of the inner radius error with decreasing h was performed and the results are shown in Fig. 15d. 512^2 material points were used for all of the analyses. This number of material points was chosen as further increasing the number of material points only reduced the error by less than 1% with the same mesh. Enforcing the roller boundary conditions using the B-spline based IBM results in a convergence rate that is very similar to that using the conventional boundary conditions imposed directly on the background mesh. The reason behind this high convergence rate could be that the high curvature of the inner radius was represented more accurately as the mesh was refined.

4.4. Infinite plate with a circular hole under far field stress

The final example is that of a plane stress infinite plate with a circular hole under a far field stress in the x-direction of $S = 10$ MPa applied in a single load step. Due to symmetry only one quarter of the plate was analysed and the extent of the plate was truncated by a B-spline boundary defined using 11 zero-volume material points with the following global coordinates

$$\begin{bmatrix} \{x_p\}^T \\ \{y_p\}^T \end{bmatrix} = \begin{bmatrix} 0.00 & 0.10 & 0.35 & 0.60 & 0.73 & 1.00 & 1.50 & 1.75 & 2.00 & 1.78 & 1.50 \\ 1.50 & 1.75 & 1.95 & 1.80 & 1.31 & 1.15 & 1.35 & 1.30 & 0.75 & 0.21 & 0.00 \end{bmatrix} \text{ m.}$$

The circular hole at the centre of the plate had a radius of $r_i = 0.5$ m, and the material used to model this problem had a Young's modulus of $E = 10$ MPa and Poisson's ratio of $\nu = 0.2$. The problem geometry along with the boundary conditions are shown on a regular finite element grid with $h = 0.125$ m in the left hand subfigure of Fig. 16. In all cases the roller boundary conditions at $x = 0$ and $y = 0$ were applied through the B-spline based IBM and the implicit boundary bandwidth, δ, was set to $10^{-6}h$. The boundary material points are shown by the grey-filled circles and the standard material points, arranged in a 2-by-2 grid, are shown by the black dots. The figure also shows the control points (white-filled squares) used to construct the B-spline outer boundary. The material points were uniformly distributed within each element any material point with a distance from the origin less than the circle's radius, i.e. 0.5 m, were deleted. Material points outside the outer B-spline boundary were also removed from the analysis. The right hand subfigure of Fig. 16 shows the original and deformed outer boundary along with the original and deformed material point positions, shown by light and dark grey points, respectively. A 2-by-2 grid of material points is only shown for illustrative purposes; all of the analyses reported in this section used 8^2 material points per element.

The analytical displacement solution [57] for this problem is

$$u = \frac{Sr_i}{8\mu}\left(\frac{r}{r_i}(k+1)\cos(\beta) + \frac{2r_i}{r}((1+k)\cos(\beta)+\cos(3\beta)) - \frac{2r_i^3}{r^3}\cos(3\beta)\right) \tag{42}$$

and

$$v = \frac{Sr_i}{8\mu}\left(\frac{r}{r_i}(k-3)\sin(\beta) + \frac{2r_i}{r}((1-k)\sin(\beta)+\sin(3\beta)) - \frac{2r_i^3}{r^3}\sin(3\beta)\right) \tag{43}$$

where $\mu = E/(2(1+\nu))$ and $k = (3-\nu)/(1+\nu)$ for plane stress analysis.[5] r is the distance from the centre of the circle to a point on the plate, β is the angle between r and the positive x-direction as shown in Fig. 16, r_i is the radius of the circular hole and S is the

[5] For plane strain analysis $k = 3 - 4\nu$.

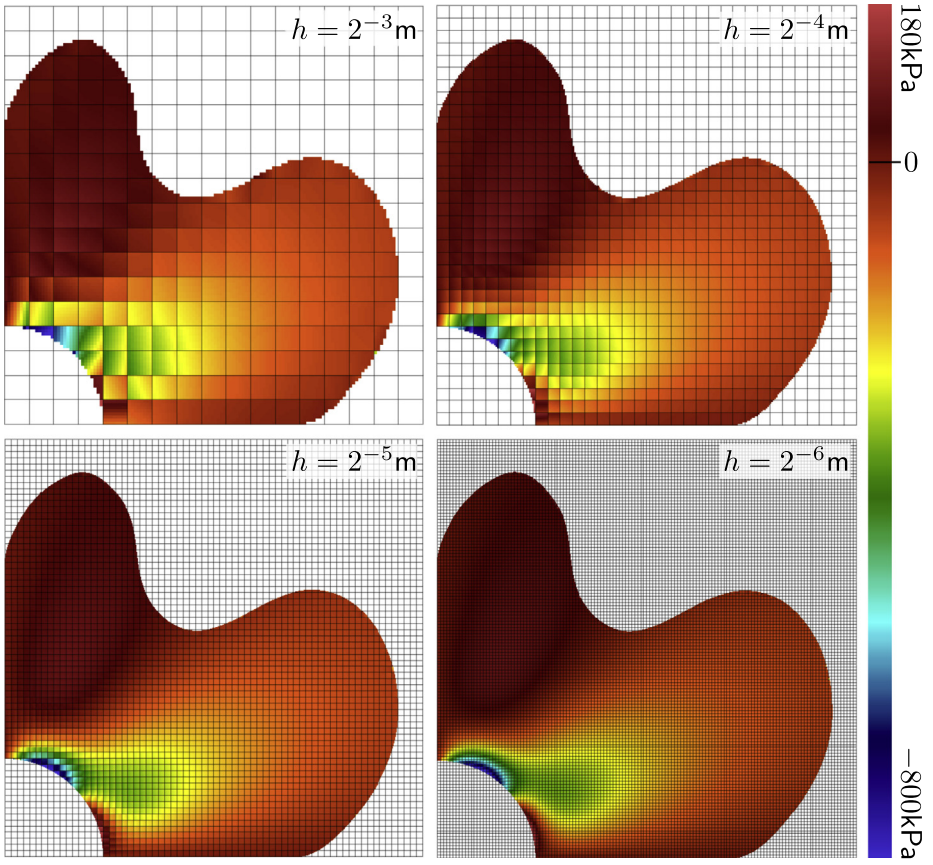


Fig. 18. Infinite plate with a circular hole (Neumann): original material point positions coloured according to σ_{xy} for $h = 2^{-3}, 2^{-4}, 2^{-5}$ and 2^{-6} m with 8^2 material points per element.

far field normal stress acting in the x-direction. The analytical stress solution [57] described in the Cartesian coordinate system is given by

$$\sigma_{xx} = S - S\left(\frac{r_i^2}{r^2}\right)\left(\frac{3}{2}\cos(2\beta) + \cos(4\beta)\right) + S\left(\frac{3r_i^4}{2r^4}\right)\cos(4\beta), \quad (44)$$

$$\sigma_{yy} = -S\left(\frac{r_i^2}{r^2}\right)\left(\frac{1}{2}\cos(2\beta) + \cos(4\beta)\right) - S\left(\frac{3r_i^4}{2r^4}\right)\cos(4\beta) \quad \text{and} \quad (45)$$

$$\sigma_{xy} = -S\left(\frac{r_i^2}{r^2}\right)\left(\frac{1}{2}\sin(2\beta) + \sin(4\beta)\right) + S\left(\frac{3r_i^4}{2r^4}\right)\sin(4\beta), \quad (46)$$

The problem was analysed using two different boundary conditions on the outer B-spline boundary:

1. inhomogeneous Neumann boundary condition, obtained by determining the appropriate traction at each Gauss point using the analytical stress solution (44)–(46); and
2. inhomogeneous Dirichlet boundary condition using the IBC based on the displacement solution (42) and (43).

The analysis of the problem using a Neumann condition will be presented first.

4.4.1. Inhomogeneous Neumann

The problem was analysed using 6 different background grid sizes, with $h = 2^{-1}$ to $h = 2^{-6}$ m, in each case 8^2 material point points per background grid cell were used.[6] The traction was integrated using 4 Gauss points per element segment. Fig. 17 shows the absolute displacement error and relative stress error convergence under this uniform h refinement. The correct convergence rates for both stress and displacement are obtained demonstrating the correct implementation of the non-uniform Neumann boundary condition.

Fig. 18 shows the original material point positions coloured according to σ_{xy} for $h = 2^{-3}, 2^{-4}, 2^{-5}$ and 2^{-6} m. The smoothness of the stress solution increases with h refinement with the finest two discretisations obtaining good agreement with the analytical stress solution.

Fig. 19 shows the displacement error distribution for $h = 2^{-5}$ m and 8^2 material points per element, where the error scales are $A_u \in \left[0, 1.5 \times 10^{-4}\right]$ m and $A_v \in \left[0, 1.0 \times 10^{-4}\right]$ m. Although the maximum errors in Fig. 19 are 5.61×10^{-4} m and 3.59×10^{-4} m, for A_u and A_v, respectively, the error scales have been truncated to make the error distribution more apparent. Oscillations in the displacement errors can be observed on the outer B-spline boundary. These are primarily due to the poor integration of the global stiffness matrix caused by partially filled background elements. Other areas of high error are associated with the inner boundary, again partially due to non-full elements but also due to the higher stress gradients in this area.

4.4.2. Inhomogeneous Dirichlet

The same problem was analysed by imposing the analytical displacement solution on the outer B-spline boundary using the implicit boundary method (as described in Section 3.5.1). The nodal displacements for the elements containing the outer boundary, $\{d^a\}$, were determined using the least squares weighted residual technique described in Section 3.5.1. The convergence of this

[6] 8^2 material point points per background grid cell were adopted in these analyses to ensure that the convergence rate of the analyses were not limited by integration inaccuracies of the material stiffness and internal force. The impact of reducing the number of material points is explored in Section 4.4.2.

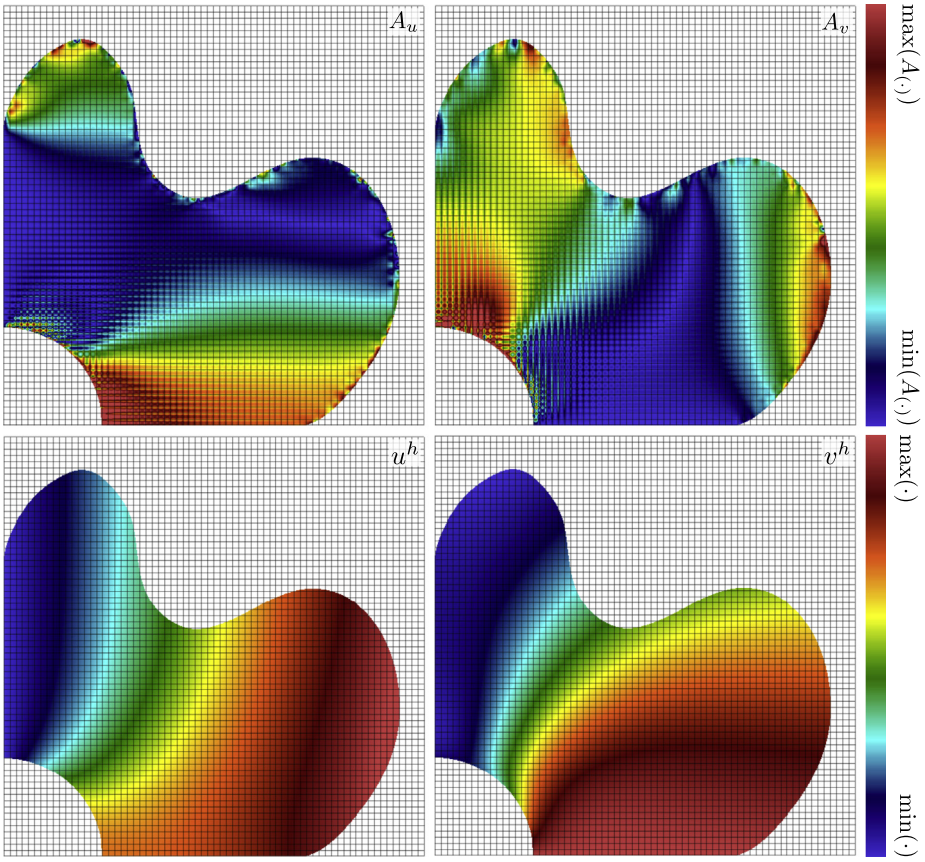


Fig. 19. Infinite plate with a circular hole (Neumann): displacement error distribution (top) and displacement solution (bottom) for $h = 2^{-5}$ m and 8^2 material points per element. The error scales are $A_u \in \left[0, 1.5 \times 10^{-4}\right]$ m and $A_v \in \left[0, 1.0 \times 10^{-4}\right]$ m and the displacement scales $u^h \in [0, 0.22]$ m and $v^h \in [-0.055, 0]$ m.

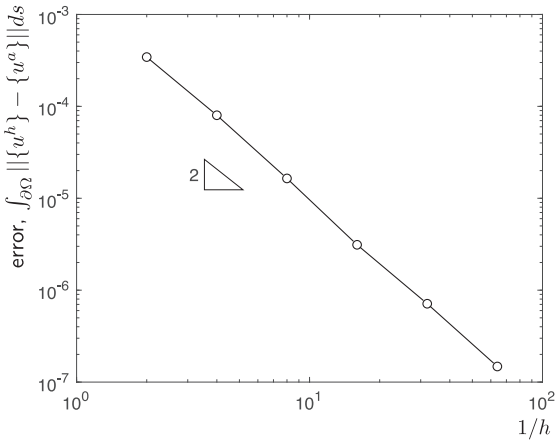


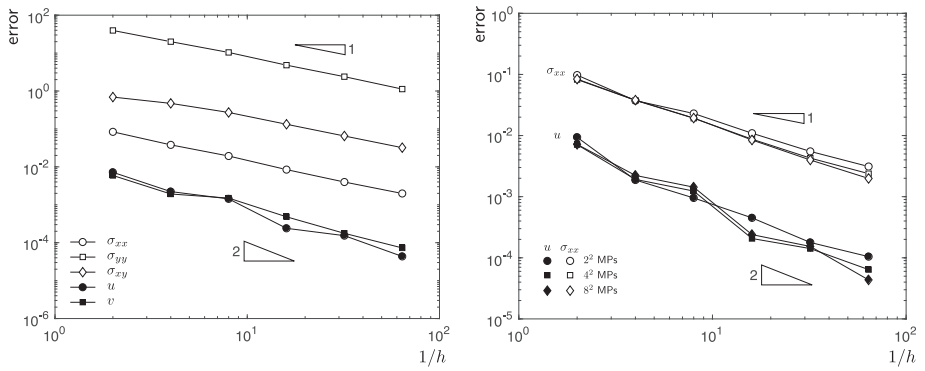
Fig. 20. Infinite plate with a circular hole (inhomogeneous Dirichlet): boundary displacement convergence under uniform h-refinement.

least squares fit with h refinement is shown in Fig. 20. The average convergence rate was super-quadratic at 2.26, providing confidence in the determination of $\{d^a\}$. These displacements where used to determine the additional external force required in (31) in order to impose the required inhomogeneous Dirichlet boundary condition.

Fig. 21 shows the convergence of the relative stress and absolute displacement errors with h refinement. As with the Neumann condition, the errors converge towards the correct solution with grid refinement. The minor oscillations in the displacement convergence are likely to be due to the poor approximation of the analytical displacement solution in the bi-linear elements containing the outer boundary. The errors are also similar in magnitude to those given in Fig. 17 for the Neumann analysis. However, the convergence rate for the displacements are lower than the equivalent inhomogeneous Neumann analysis. This is due to the fact that imposing Dirichlet conditions on the outer boundary creates discontinuities in the displacement field that do not agree with the analytical solution to the problem as they are determined via the least squares fitting approach described in Section 3.5.1. That is, the nodal values, $\{d^a\}$, ensure that the displacement along the boundary, $\{u^a\}$, converges towards the analytical solution on the boundary (as shown by Fig. 20) but there is no guarantee that the nodal displacements themselves agree with the analytical solution at their nodal locations. The discontinuities, and associated error oscillations, therefore limit the convergence rate of the method. The sub-optimal convergence rate is consistent with other papers that adopt the implicit boundary method for Dirichlet boundary conditions [40–43].

Fig. 21b also shows the convergence of u and σ_{xx} under h refinement with different numbers of material points per background grid cell, specifically $2^2, 4^2$ and 8^2 material points per element. The overall convergence rate of the analyses with different numbers of material points are similar, however degradation in the convergence rate is seen for the lower numbers of material points with progressive h refinement due to integration inaccuracies that start to limit the convergence rate with finer meshes.

Fig. 22 shows the original material point positions coloured according to σ_{xx} for $h = 2^{-3}, 2^{-4}, 2^{-5}$ and 2^{-6} m. As with Fig. 18, the smoothness of the stress solution increases with h refinement.



(a) 8^2MPs/element (b) 2^2, 4^2 & 8^2 MPs/element

Fig. 21. Infinite plate with a circular hole (inhomogeneous Dirichlet): convergence under uniform h-refinement.

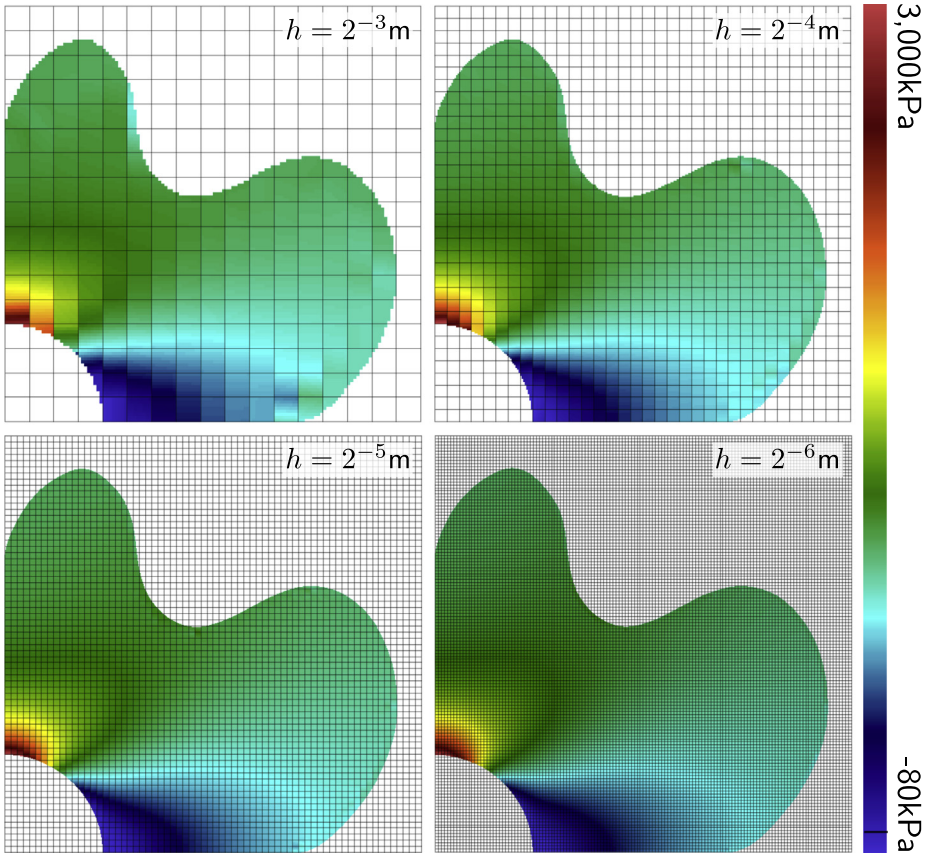


Fig. 22. Infinite plate with a circular hole (inhomogeneous Dirichlet): original material point positions coloured according to σ_{xx} for $h = 2^{-3}, 2^{-4}, 2^{-5}$ and 2^{-6} m with 8^2 material points per element.

Fig. 23 shows the displacement error distribution for $h = 2^{-6}$ m with 8^2 material points per element, where the error scales are $A_u \in \left[0, 5.0 \times 10^{-5}\right]$ m and $A_v \in \left[0, 8.0 \times 10^{-5}\right]$ m. The maximum errors in Fig. 23 are $\max(A_u) = 3.160 \times 10^{-5}$ m and $\max(A_v) = 1.93 \times 10^{-5}$ m. It should be observed that, unlike the Neumann error distribution shown in Fig. 19, the displacement error along the outer boundary is near-zero for both displacement components, this is particularly evident in the enlarged subfigures at the bottom of Fig. 23. Oscillations in the error distribution are still observed on the inner boundary, where a homogeneous Neumann (traction free) condition is automatically enforced, caused in part by partially filled elements.

This challenging problem demonstrates the capabilities of the proposed B-spline boundary formulation in analysing problems involving complex geometry and loading conditions which would not be possible using existing boundary enforcement methods available in the MPM literature.

5. Conclusions

This paper has presented, for the first time, a general method for boundary representation and boundary condition imposition in the standard MPM. A local cubic B-spline interpolation technique has been employed for boundary representation. This boundary representation technique has been combined with the IBM for homoge-

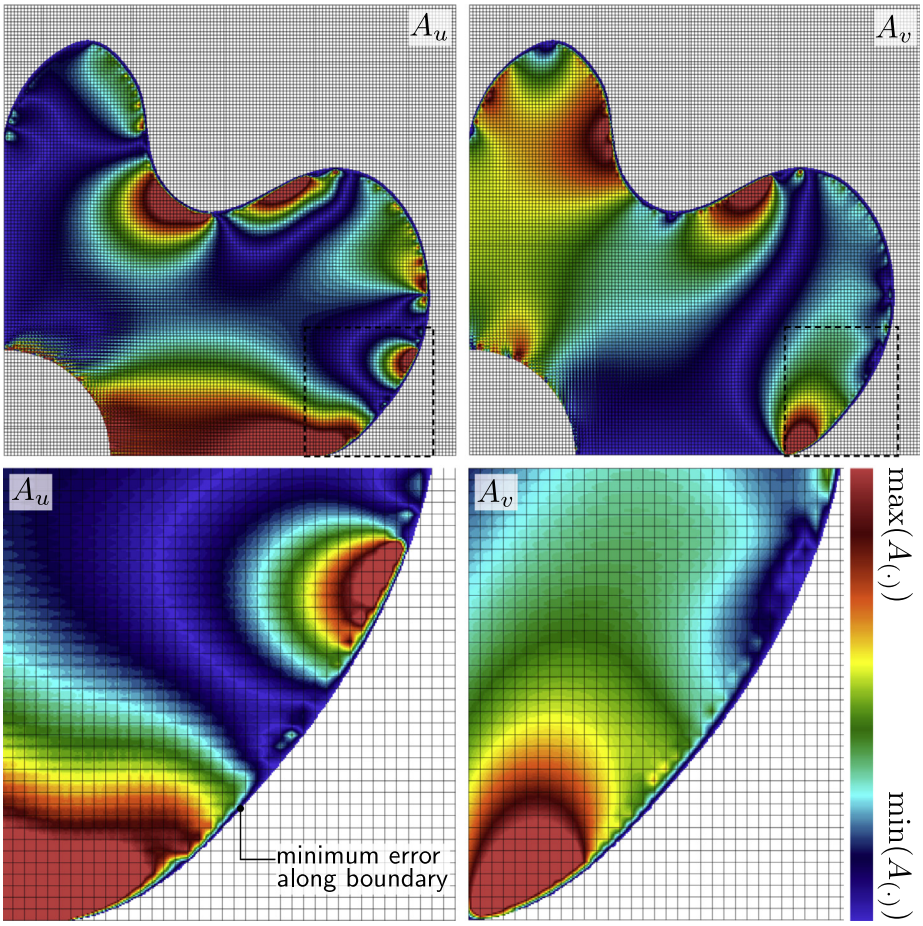


Fig. 23. Infinite plate with a circular hole (inhomogeneous Dirichlet): displacement error distribution for $h = 2^{-6}$ m and 8^2 material points per element. The error scales are $A_u \in \left[0, 5 \times 10^{-5}\right]$ m and $A_v \in \left[0, 8 \times 10^{-5}\right]$ m.

neous and inhomogeneous Dirichlet boundary conditions whereas inhomogeneous Neumann boundaries are imposed via integration of the tractions over the boundary segment. Previous approaches to boundary condition imposition within the MPM have focused on one type of boundary condition, are only applicable to piecewise linear segments [33] or are only applicable to one form of the MPM. This complete methodology of boundary representation and boundary condition imposition developed in this paper makes it possible for the standard MPM to analyse problems that were previously not feasible. Additionally, this B-spline boundary method could also be used in other non-matching mesh approaches and other types of MPM. Although the approaches for imposing Dirichlet boundary conditions presented in this paper are only applicable to implicit formulations, the boundary representation, tracking and inhomogeneous Neumann boundary condition enforcement can be applied to implicit and explicit approaches.

The focus of this paper has been on extending the boundary condition enforcement capabilities of the MPM such that it can be applied to problems that were previously not feasible, for example those containing complex, spatially varying, Neumann and Dirichlet conditions. This has been achieved through combining B-splines with the implicit boundary method and direct integration of surface tractions. However, in cases with simple boundaries, that can be represented by piecewise linear segments, the approach of Cortis et al. [33] will be more computationally efficient as it does not require B-spline reconstruction.

Throughout the paper it has been assumed that boundaries remain as boundaries through the analysis. However, when analysing large deformation problems, particularly in the area of soil mechanics, there are cases where external boundaries could come into self contact and cease to be boundaries, new boundaries could also be created. In this case special treatment, and appropriate detection routines, would be required to determine when this occurs and remove/introduce boundary tracking points as needed. This is an interesting, and challenging, area of future research but it is outside of the scope of the work presented in this manuscript. However, provided that suitable boundary points can be identified the reconstruction, representation, tracking and boundary enforcement methods presented in this paper can be adopted.

Acknowledgements

This work was supported by the Engineering and Physical Sciences Research Council [Grant Nos. EP/K502832/1, EP/M000397/1 and EP/M000362/1]. All data created during this research are openly available at doi:https://doi.org/10.15128/r2pk02c974d.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.compstruc.2018.11.003.

References

[1] Lucy LB. A numerical approach to the testing of the fission hypothesis. Astron J 1977;82:1013–24.
[2] Belytschko T, Lu YY, Gu L. Element-free Galerkin methods. Int J Numer Meth Eng 1994;37:229–56.
[3] Liu WK, Jun S, Zhang YF. Reproducing kernel particle methods. Int J Numer Meth Fluids 1995;20:1081–106.
[4] Nguyen VP, Rabczuk T, Bordas S, Duflot M. Meshless methods: a review and computer implementation aspects. Math Comput Simul 2008;79:763–813.
[5] Sulsky D, Chen Z, Schreyer H. A particle method for history-dependent materials. Comput Methods Appl Mech Eng 1994;118:179–96.
[6] Sulsky D, Zhou S-J, Schreyer HL. Application of a particle-in-cell method to solid mechanics. Comput Phys Commun 1995;87:236–52.
[7] 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;31:107–27.
[8] Bardenhagen S, Kober E. The generalized interpolation material point method. Comput Model Eng Sci 2004;5:477–95.
[9] Sadeghirad A, Brannon RM, Burghardt J. A convected particle domain interpolation technique to extend applicability of the material point method for problems involving massive deformations. Int J Numer Meth Eng 2011;86:1435–56.
[10] Sadeghirad A, Brannon R, Guilkey J. Second-order convected particle domain interpolation (CPDI2) with enrichment for weak discontinuities at material interfaces. Int J Numer Meth Eng 2013;95:928–52.
[11] York AR, Sulsky D, Schreyer H. The material point method for simulation of thin membranes. Int J Numer Meth Eng 1999;44:1429–56.
[12] York AR, Sulsky D, Schreyer HL. Fluid-membrane interaction based on the material point method. Int J Numer Meth Eng 2000;48:901–24.
[13] Lian Y-P, Liu Y, Zhang X. Coupling of membrane element with material point method for fluid–membrane interaction problems. Int J Mech Mater Des 2014;10:199–211.
[14] Lian Y, Zhang X, Zhou X, Ma S, Zhao Y. Numerical simulation of explosively driven metal by material point method. Int J Impact Eng 2011;38:238–46.
[15] Wang Y, Beom H, Sun M, Lin S. Numerical simulation of explosive welding using the material point method. Int J Impact Eng 2011;38:51–60.
[16] Zhou S, Zhang X, Ma H. Numerical simulation of human head impact using the material point method. Int J Comput Methods 2013;10:312–583.
[17] Liu P, Liu Y, Zhang X, Guan Y. Investigation on high-velocity impact of micron particles using material point method. Int J Impact Eng 2015;75:241–54.
[18] Lee JH, Huang D. Modeling and testing of snow penetration. J Terrramech 2015;59:35–47.
[19] Bardenhagen S, Brackbill J, Sulsky D. The material-point method for granular materials. Comput Methods Appl Mech Eng 2000;187:529–41.
[20] Cummins S, Brackbill J. An implicit particle-in-cell method for granular materials. J Comput Phys 2002;180:506–48.
[21] Solowski W, Sloan SW. Material point method modelling of granular flow in inclined channels. Appl Mech Mater 2014;553:501–6.
[22] Beuth L, Benz T, Vermeer PA. Large deformation analysis using a quasi-static material point method. J Theoret Appl Mech 2008;38:45–60.
[23] Andersen S. Material-point analysis of large-strain problems: modelling of landslides [Ph.D. thesis]. Denmark: Aalborg University; 2009.
[24] Andersen S, Andersen L. Modelling of landslides with the material-point method. Comput Geosci 2010;14:137–47.
[25] Yerro A, Pinyol NM, Alonso EE. Internal progressive failure in deep-seated landslides. Rock Mech Rock Eng 2016;49:2317–32.
[26] Gilabert FA, Cantavella V, Sánchez E, Mallol G. Modelling fracture process in ceramic materials using the material point method. EPL (Europhys Lett) 2011;96:24002.
[27] Li F, Pan J, Cocks A. A new numerical scheme for computer simulation of multiple cracking in ceramic films during constrained sintering. Modell Simul Mater Sci Eng 2012;20:035008.
[28] Beuth L. Formulation and application of a quasi-static material point method [Ph.D. thesis]. University of Stuttgart; 2012.
[29] al Kafaji IKJ. Formulation of a dynamic material point method (MPM) for geomechanical problems [Ph.D. thesis]. University of Stuttgart; 2013.
[30] Chen Z, Brannon R. An evaluation of the material point method. Technical report. Albuquerque, New Mexico 87185 and Livermore, California 94550: Sandia National Laboratories; 2002.
[31] Chen Z, Hu W, Shen L, Xin X, Brannon R. An evaluation of the MPM for simulating dynamic failure with damage diffusion. Eng Fract Mech 2002;69:1873–90.
[32] Hamad FM. Formulation of a dynamic material point method and applications to soil-water-geotextile systems [Ph.D. thesis]. University of Stuttgart; 2014.
[33] Cortis M, Coombs WM, Augarde C, Brown M, Brennan A, Robinson S. Imposition of essential boundary conditions in the material point method. Int J Numer Meth Eng 2017:1–23.
[34] Mast C, Mackenzie-Helnwein P, Arduino P, Miller G. Landslide and debris flow-induced static and dynamic loads on protective structures. In: Borja R, editor. Multiscale and multiphysics processes in geomechanics, Springer series in geomechanics and geoengineering. Berlin Heidelberg: Springer; 2011. p. 169–72.
[35] Mast C. Representing arbitrary bounding surfaces in the material point method. In: 6th MPM workshop in Albuquerque, New Mexico; 2010.
[36] Höllig K, Reif U, Wipper J. Weighted extended B-spline approximation of Dirichlet problems. SIAM J Numer Anal 2001;39:442–62.
[37] Höllig K, Reif U. Nonuniform web-splines. Comput Aided Geometr Des 2003;20:277–94.
[38] Höllig K, Apprich C, Streit A. Introduction to the web-method and its applications. Adv Comput Math 2005;23:215–37.
[39] Remmerswaal G. Development and implementation of moving boundary conditions in the Material Point Method [Master's thesis]. Delft University of Technology; 2017.
[40] Burla RK, Kumar AV. Implicit boundary method for analysis using uniform B-spline basis and structured grid. Int J Numer Meth Eng 2008;76:1993–2028.
[41] Kumar AV, Padmanabhan S, Burla R. Implicit boundary method for finite element analysis using non-conforming mesh or grid. Int J Numer Meth Eng 2008;74:1421–47.
[42] Kumar AV, Buria R, Padmanabhan S, Gu L. Finite element analysis using nonconforming mesh. J Comput Inf Sci Eng 2008;8:119–31.
[43] Zhang Z, Kumar AV. Immersed boundary modal analysis and forced vibration simulation using step boundary method. Fin Elem Anal Des 2017.
[44] Lu K, Augarde CE, Coombs WM, Hu Z. Weak impositions of Dirichlet boundary conditions in solid mechanics: a critique of current approaches and extension to partially prescribed boundaries. Comput Methods Appl Mech Eng 2018 [in preparation].
[45] Zhang W, Zhao L, Cai S. Shape optimization of Dirichlet boundaries based on weighted B-spline finite cell method and level-set function. Comput Methods Appl Mech Eng 2015;294:359–83.
[46] Zhang W, Zhao L. Exact imposition of inhomogeneous Dirichlet boundary conditions based on weighted finite cell method and level-set function. Comput Methods Appl Mech Eng 2016;307:316–38.
[47] Kim H-J, Seo Y-D, Youn S-K. Isogeometric analysis for trimmed CAD surfaces. Comput Methods Appl Mech Eng 2009;198:2982–95.
[48] Kim H-J, Seo Y-D, Youn S-K. Isogeometric analysis with trimming technique for problems of arbitrary complex topology. Comput Methods Appl Mech Eng 2010;199:2796–812.
[49] Kim H-J, Youn S-K. Spline-based meshfree method. Int J Numer Meth Eng 2012;92:802–34.
[50] Guilkey JE, Weiss JA. Implicit time integration for the material point method: quantitative and algorithmic comparisons with the finite element method. Int J Numer Meth Eng 2003;57:1323–38.
[51] Sulsky D, Gong M. Improving the material point method. In: Weinberg K, Pandolfi A, editors. Innovative numerical approach for multi-field and multi-scale problems. Springer; 2016. p. 217–40.
[52] Piegl L, Tiller W. The NURBS book. Berlin Heidelberg: Springer-Verlag; 1997.
[53] Bing Y. B-spline based boundary method for the material point method [Master's thesis]. Durham University; 2017.
[54] Cottrell JA, Hughes TJ, Bazilevs Y. Isogeometric analysis: toward integration of CAD and FEA. John Wiley & Sons; 2009.
[55] Hughes T, Cottrell J, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comput Methods Appl Mech Eng 2005;194:4135–95.
[56] Becker A. The boundary element method in engineering: a complete course. McGraw-Hill; 1992.
[57] Timoshenko S, Goodier JN. Theory of elasticity. 2nd ed. McGraw-Hill; 1951.