Boundary representation and boundary condition imposition in the material point method

*Yun Bing[1], Michael Cortis[1], Tim Charlton[1], William Coombs[1] and Charles Augarde[1]

[1]School of Engineering and Computing Sciences, University of Durham, Lower Mountjoy, South Road, Durham, DH1 3LE

*yun.bing@durham.ac.uk

ABSTRACT

Unlike the conventional finite element method (FEM), in which the mesh conforms to the material boundary, the material point method (MPM) [12] does not provide a clear interpretation of the boundary. In the MPM, the problem domain is discretised by a finite number of material points, on which the material properties and history dependent variables are prescribed and carried throughout simulations. The material information is then mapped to a "stationary" background mesh where the equilibrium equations are solved. However, the material boundary and the mesh nodes, in general, do not coincide. This causes difficulties solving boundary-value problems during MPM simulations, in particular, applying traction (Neumann) and non-zero prescribed displacement (inhomogeneous Dirichlet) boundary conditions. However, little attention has been paid to this issue and no literature to date has presented an effective way to model and track boundaries in the MPM. In this paper, a B-spline based boundary approximation method is discussed. A local cubic interpolation technique [10] is employed for boundary representation. Tractions are applied through direct integration over the B-spline boundary and displacements are prescribed via a B-spline based implicit boundary method [3, 8, 9].

Key Words: material point method; B-splines; implicit boundary method; boundary conditions

1. Introduction

The material point method (MPM) was developed by Sulsky *et al.* [12] in 1994 as a solid mechanics extension to the fluid implicit particle method [2] which was an advancement of the particle-in-cell method [6]. The formulation of the MPM is rather similar to the standard finite element method (FEM) as the governing equations are solved on the nodes of the mesh. However, instead of using the mesh elements themselves to describe the problem domain, the MPM discretises the problem domain by a set of material points (MPs). These MPs are similar to the Gauss points in the standard FEM as both methods use these points for stiffness integration except that the MPs follow the deformation of the problem domain whilst the background mesh retains its original shape. In other words, at the end of each MPM load step, the positions of the MPs are updated by the deformed background mesh which is reset for the next load step leaving the MPs at their new positions (see Figure 1). These characteristics of the MPM allow it to model large deformation and fracture problems without going through the expensive processes of re-meshing and re-mapping of state variables.
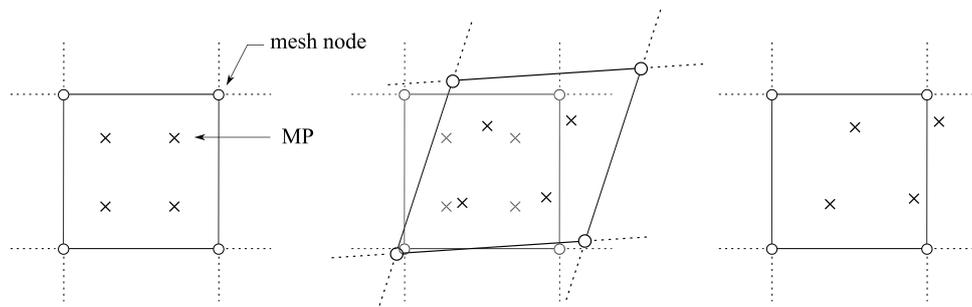
Figure 1: A simple illustration of the MPM simulation procedure: initialisation (left), deformation (middle) and reset mesh with updated MPs position (right).

Nevertheless, because the problem domain is represented by MPs, boundaries are not clearly defined in the MPM. This makes the application of boundary conditions (BCs) rather inaccurate, especially for Neumann and inhomogeneous Dirichlet BCs. However, this problem is rarely discussed in the literature; many have managed to avoid this issue by aligning the problem boundaries with the background mesh or modelling problems that only involve body forces. Inspired by Kim and Youn [7], in this work a B-spline interpolation method is chosen to represent the boundaries in the MPM. Tractions are enforced by direct integration over the B-spline boundaries and displacements are prescribed by incorporating the B-spline boundaries within implicit boundary method (IBM) [3, 4, 8, 9].

2. B-spline basics and boundary representation

A pth-degree B-spline curve defined by a set of n control points, $\{P\}$, and a knot vector, $\{\Xi\} = \{\xi_0, \xi_1, \ldots, \xi_r\}$ with knots ξ_i, $i = 0 \ldots r$, being a sequence of nondecreasing real numbers and $r = n + p + 1$, can be expressed as

$$\{C(\xi)\} = \sum_{i=0}^{n} N_{i,p}(\xi)\{P_i\}, \tag{1}$$

where $N_{i,p}(\xi)$ are the pth-degree B-spline basis functions [10].

A local cubic B-spline interpolation technique has been chosen to represent the boundaries in the MPM. The interpolative nature of this technique allows a greater control on the B-spline represented boundaries and ensures that the key boundary points are included. As a local method, this cubic technique constructs curves in a piecewise fashion. Only local data are used at each step, so a fluctuation in data would only affect the curve locally. Additionally, this method has the ability to deal with sharp corners efficiently [10].

To construct the boundaries of a MPM simulated problem, the outer layer MPs are firstly identified as the sampling points. A cubic Bézier curve [1] is constructed between every two sampling points. Two inner control points are calculated in addition to the sampling points by computing the tangent at each sampling point via a five-point method [11]. To achieve a uniform parametrisation through out the curve, the first derivatives at the beginning, the middle and the end of the curve segment are set to have the same value. Then, a knot vector is determinated according to the requirement of continuity for the overall curve. Finally, a cubic B-spline curve is constructed by the knot vector and the control points by (1). Detailed calculations for the procedures described above can be found in [10].

3. B-spline based Neumann and Dirichlet boundary conditions

Having described a means of boundary representation, the application of traction to a boundary, Γ, is straightforward. The equivalent nodal forces due to the applied traction can be determined through

$$\{f^t\} = \int_{\Gamma} [M]^T \{t\} \, d\Gamma, \tag{2}$$

where $[M]$ contains the standard finite element basis functions and $\{t\}$ is the prescribed traction.

Numerical integration over a pth degree B-spline represented boundary is performed by using $(p-1)$th order Gauss-Lagrange quadrature [5]. Because the local coordinate of Gauss quadrature has a range of $[-1, 1]$; whereas, the local coordinate of a B-spline curve has positive values only, a two-step mapping between the global coordinates and the B-spline local coordinate ξ is required, that is

$$[J_B] = \left[\frac{d\{C\}}{d\tilde{\xi}}\right] = \left[\frac{d\{C\}}{d\xi}\right]\left[\frac{d\xi}{d\tilde{\xi}}\right] \quad \text{with} \quad \frac{d\xi}{d\tilde{\xi}} = \frac{\xi_{j+1} - \xi_j}{2}, \tag{3}$$

where $[d\{C\}/d\xi]$ is the first derivative of the B-spline approximated boundary [10]. Applying Gauss quadrature to (2), we obtain

$$\{f^t\} = \sum_{i=1}^{n_{gp}} [M_i]^T \{t\} \det([J_B]_i) w_i, \tag{4}$$

where n_{gp} is the number of Gauss points and w_i is the weight associated with Gauss point i.

Dirichlet BCs are imposed by incorporating the B-spline represented boundaries with the IBM which was firstly introduced by Burla and Kumar [3] for FEM problems that use non-conforming meshes. Implementation of fixed Dirichlet BCs with problem boundaries parallel to one of the coordinates has been demonstrated in [3, 8, 9]. This method has later been adopted to the MPM framework and extended to include the imposition of "roller" BCs on

inclined boundaries [4]. The methodology behind the IBM is that essential BCs are enforced by introducing extra penalty stiffness to the system. Dirichlet functions (D-functions) are involved in the solution structure to impose the prescribed displacements directly. In the IBM, the FEM trial solution for an elastostatic problem is expressed as

$$\{u\} = [D][M]\{d^e\} + \{u^a\}, \tag{5}$$

where $\{u\}$ is the displacement within a finite element, $[D]$ contains the D-functions, $\{d^e\}$ are nodal displacements associated with the element and $\{u^a\} = [M]\{u^e\}$ with $\{u^e\}$ being the prescribed nodal displacements. To enforce fully fixed essential BCs, the D-functions are constructed such that they vanish at the degrees of freedoms where non-zero displacements are prescribed and rise to unity when reach the problem domain over a thin band δ. As for roller BCs, the D-functions are assigned to have a value of 1 in the directions that are free to move [4]. Stiffness integration is then performed over δ and along the B-spline represented boundaries where $[J_B]$ (3) is used for mapping between the global and local coordinates.

4. Numerical example

An example of a 10m × 2m cantilever beam is shown here to validate the implementations of the B-spline based boundary method. The cantilever beam was modelled under a plane strain assumption and linear elastic material with a Young's modulus of 1MPa and Poisson's ratio of 0.25 was used. The left hand side boundary was fixed at the mid height of the beam and with rollers above and below. A constant pressure of 1500Pa was applied along the top boundary, which was maintained perpendicular to the boundary through out the analysis.

A background mesh with 1.5m by 1.5m elements was used, and the problem domain was discretised by using 896 uniformly distributed MPs. The outer layer of the MPs were identified as the problem boundaries which were approximated by using B-splines. BCs on the left boundary were applied by the B-spline based IBM and the pressure applied through 5 load steps. The initial discretisation and the final deformed cantilever beam are shown in Figure 2a; however, having boundaries represented by B-splines, the boundaries can be tracked after each load step without plotting out all the MPs (see Figure 2b) and the curvature of the deformed shape has been successfully captured by the B-spline approximation.

(a) Final deformation.

(b) Visualisation after ecah load step.
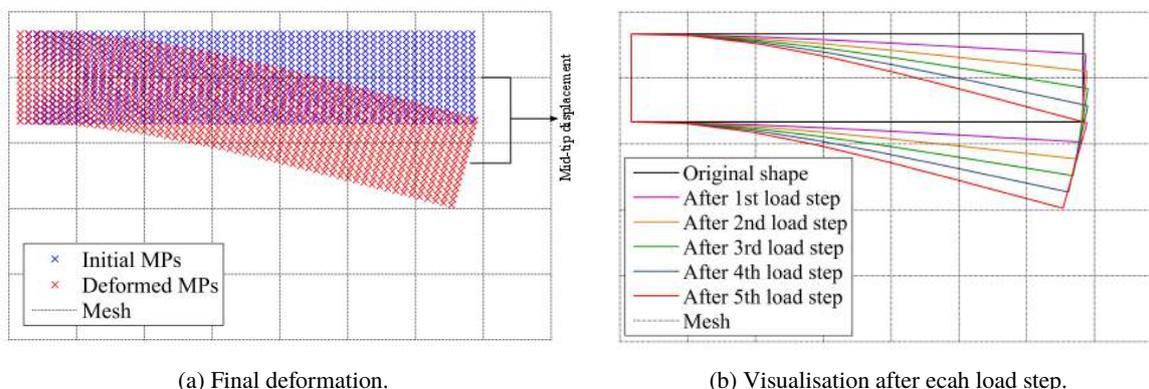
Figure 2: Illustrations of the deformed cantilever beam.

As there is no analytical solution for this problem, a convergence test on the displacement at the centre of the beam tip (see Figure 2a) was carried out by performing h-refinement on the mesh and increasing the number of MPs. As shown in Figure 3, for all three mesh configurations, the mid-tip displacement converges when more MPs are introduced. Reduction of the change in displacement with mesh refinement indicates that the mid-tip displacement also converges when the mesh size is decreased.

5. Conclusions

This paper has presented a general method of representing boundaries in the MPM by using cubic B-splines and which is also used as a means of enforcing both Neumann and Dirichlet BCs. This B-spline based boundary method not only allows easy visualisation of the deformed problem domain, but has made the MPM capable of modelling problems that were previously not possible. These has been demonstrated through an example of cantilever beam with applied traction. Although this approach of boundary representation and BC impositions has been developed in 2D, the same theory is also applicable to 3D. The formulation of local bicubic B-spline surface interpolation is

straightforward to implement and extending the IBM to 3D is equivalent to introducing penalty stiffness to the 3D MPM.
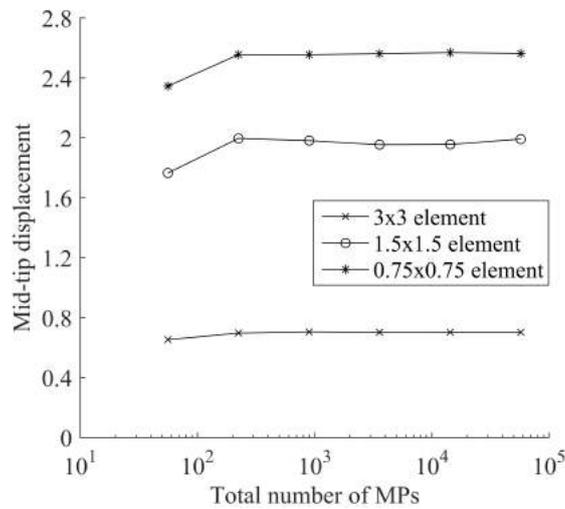
Figure 3: Convergence of mid-tip displacement.

References

[1] P. Bézier. *Numerical Control: Mathematics and Applications*, John Wiley, 1972.

[2] J.U. Brackbill and H.M. Ruppel. FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions. *Journal of Computational Physics*, 65(2), 314–343, 1986.

[3] R.K. Burla and A.V. Kumar. Implicit boundary method for analysis using uniform B-spline basis and structured grid. *International Journal for Numerical Methods in Engineering*, 76(13), 1993–2028, 2008.

[4] M. Cortis, W.M. Coombs, C.E. Augarde, M.J.Z. Brown, A. Brennan and S. Robinson. Imposition of essential boundary conditions in the material point method. *International Journal for Numerical Methods in Engineering*, 2017. Under review.

[5] J.A. Cottrell, T.J.R. Hughes and Y. Bazilevs. *Isogeometric analysis: Toward Integration of CAD and FEA*, John Wiley & Sons, 2009.

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

[7] H. Kim and S. Youn. Spline-based meshfree method. *International Journal for Numerical Methods in Engineering*, 92(9), 802–834, 2012.

[8] A.V. Kumar, S. Padmanabhan and R.K. Burla. Implicit boundary method for finite element analysis using non-conforming mesh or grid. *International Journal for Numerical Methods in Engineering*, 74(9), 1421–1447, 2008.

[9] A.V. Kumar, R.K. Burla, S. Padmanabhan and L. Gu. Finite Element Analysis Using Nonconforming Mesh. *Journal of Computing and Information Science in Engineering*, 8, 2008.

[10] L. Piegl and W. Tiller. *The NURBS Book*, Springer-Verlag Berlin Heidelberg, 1997.

[11] G. Renner. A Method of Shape Description for Mechanical Engineering Practice. *Computers in Industry*, 3(1), 137–142, 1982.

[12] D. Sulsky, Z. Chen and H.L. Schreyer. A particle method for history-dependent materials. *Computer Methods in Applied Mechanics and Engineering*, 118(1), 179–196, 1994.