

## Conservative Taylor least squares reconstruction with application to material point methods

Wobbes, Elizaveta; Möller, Matthias; Galavi, Vahid; Vuik, Cornelis

**DOI**

[10.1002/nme.5956](https://doi.org/10.1002/nme.5956)

**Publication date**

2019

**Document Version**

Final published version

**Published in**

International Journal for Numerical Methods in Engineering

**Citation (APA)**

Wobbes, E., Möller, M., Galavi, V., & Vuik, C. (2019). Conservative Taylor least squares reconstruction with application to material point methods. *International Journal for Numerical Methods in Engineering*, 117(3), 271-290. <https://doi.org/10.1002/nme.5956>

**Important note**

To cite this publication, please use the final published version (if applicable). Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

## RESEARCH ARTICLE

# Conservative Taylor least squares reconstruction with application to material point methods

Elizaveta Wobbes<sup>1</sup>  | Matthias Möller<sup>1</sup> | Vahid Galavi<sup>2</sup> | Cornelis Vuik<sup>1</sup>

<sup>1</sup>Department of Applied Mathematics, Delft University of Technology, Delft, The Netherlands

<sup>2</sup>Geo-Engineering Section, Deltares, Delft, The Netherlands

## Correspondence

Elizaveta Wobbes, Van Mourik Broekmanweg 6, 2628 XE Delft, The Netherlands.

Email: E.D.Wobbes@tudelft.nl

## Funding information

The Netherlands Organisation for Scientific Research (NWO); Deltares; Royal Boskalis Westminster N. V.; Van Oord Dredging and Marine Contractors; Rijkswaterstaat; Stichting FloodControl IJkdijk

## Summary

Within the standard material point method (MPM), the spatial errors are partially caused by the direct mapping of material-point data to the background grid. In order to reduce these errors, we introduced a novel technique that combines the least squares method with the Taylor basis functions, called the Taylor least squares (TLS), to reconstruct functions from scattered data while preserving their integrals. The TLS technique locally approximates quantities of interest such as stress and density, and when used with a suitable quadrature rule, it conserves the total mass and linear momentum after transferring the material-point information to the grid. The integration of the technique into MPM, dual domain MPM, and B-spline MPM significantly improves the results of these methods. For the considered examples, the TLS function reconstruction technique resembles the approximation properties of highly accurate spline reconstruction while preserving the physical properties of the standard algorithm.

## KEYWORDS

B-spline, conservation, function reconstruction, least squares, material point method, Taylor basis

## 1 | INTRODUCTION

The material point method (MPM)<sup>1,2</sup> is a continuum-based numerical tool to simulate problems that involve large deformations. In MPM, the material is represented by a set of Lagrangian particles (material points) that move through a fixed Eulerian background grid. The material points carry the physical properties of the continuum such as the mass, strain, and stress. At the beginning of each time step, this information is projected from the particles to the degrees of freedom (DOFs) of the background grid, where, similarly to the finite element method<sup>3</sup> (FEM), the discretized governing equations are assembled and subsequently solved. The obtained information are then mapped back to update the material points. This conjunction of Lagrangian and Eulerian approaches makes MPM well suited for challenging problems involving large strains.

\*Abbreviations: BSMPPM, B-spline material point method; CPDI, convected particle domain interpolation; DDMPM, dual domain material point method; DOF, degrees Of freedom; FEM, finite element method; GIMP, generalized interpolation material point; NWO, Nederlandse Organisatie voor Wetenschappelijk Onderzoek; MPM, material point method; PPC, particles per cell; RMS, root mean square; TLS, Taylor least squares; ULFEM, updated Lagrangian finite element method.

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2018 The Authors. *International Journal for Numerical Methods in Engineering* Published by John Wiley & Sons, Ltd.

Despite its impressive performance for many engineering problems,<sup>4–6</sup> standard MPM still suffers from several shortcomings. For example, when large deformations are considered, the particular case of Shepard interpolation<sup>7</sup> used by the method to project the scattered material-point data to the DOFs can introduce significant numerical inaccuracies.<sup>8,9</sup> In addition, when material points travel from one cell to another, they generate unphysical oscillations in the forces, frequently referred to as grid-crossing error.<sup>10</sup> This is due to the use of piecewise-linear basis functions, whose gradients are discontinuous on element boundaries. Finally, MPM contains FEM-type errors originating from mass-lumping and interpolation, as well as time-stepping errors.<sup>11</sup>

Much research has been conducted to overcome these shortcomings. On the one hand, methods such as the generalized interpolation material point (GIMP) method,<sup>10</sup> the convected particle domain interpolation (CPDI) method,<sup>12</sup> the dual domain material point method (DDMPM),<sup>13</sup> and the B-spline material point method (BSMPM)<sup>14,15</sup> have been designed to overcome the grid-crossing error. On the other hand, Sulsky and Gong<sup>8,9</sup> have shown that the shortcomings resulting from the mapping of particle information can be decreased by reconstructing functions with a higher-order technique and evaluating these functions at the desired positions. Furthermore, Tielen et al<sup>15</sup> have pointed out that the use of global cubic-spline function reconstruction technique substantially improves the performance of BSMPM. Although modified mapping generally improves the accuracy of the solution, the standard reconstruction techniques such as spline interpolation might lead to the loss of physical properties of the material point methods. In fact, while MPM projection conserves the mass and linear momentum of the system, most standard mapping techniques do not guarantee that. Therefore, more advanced procedures are required for function reconstruction.

In this paper, we propose a novel Taylor least squares (TLS) reconstruction technique, which is based on the least squares<sup>16</sup> approximation constructed from a set of Taylor basis functions.<sup>17</sup> This technique reconstructs quantities of interest such as stress and density locally, within each active cell, and evaluates them at the integration points. After that, a Gauss quadrature is applied to determine the internal forces and velocities. If a sufficient number of Gauss points is defined within each element, the proposed mapping technique preserves the physical qualities of the standard MPM by conserving the mass and linear momentum of the system.

While the TLS reconstruction can be used autonomously within the MPM algorithm, it can also be combined with methods like CPDI or BSMPM to further reduce the spatial errors. Since previous studies have indicated that DDMPM and BSMPM are viable alternatives not only for MPM but also for GIMP and CPDI,<sup>13,18,19</sup> we have applied the TLS reconstruction technique within MPM, DDMPM, and BSMPM. The methods are tested on two one-dimensional benchmarks describing the deformation of a vibrating bar and column compaction under slowly increasing body force, respectively. The conservation property of the TLS reconstruction technique is verified by computing the total mass and momentum before and after projecting the particle information to the background grid. The accuracy of the material point methods with TLS approximation is investigated either qualitatively or based on the spatial errors and convergence rates. In addition, the obtained results are compared with those computed with the cubic-spline reconstruction technique.<sup>15</sup>

We observe that, when material points do not cross element boundaries in the vibrating bar problem, the TLS reconstruction technique has little influence on MPM and DDMPM but significantly improves the convergence of BSMPM. When particles start to cross element boundaries in both examples, the effect of the advanced reconstruction technique becomes clearly evident regardless of the material point method. In fact, the TLS technique smoothens the solutions of MPM, DDMPM, and BSMPM and ensures a close agreement with the reference solutions. Moreover, the TLS approximation outperforms the global spline interpolation in terms of the mass and momentum conservation. Therefore, the TLS reconstruction leads to a conservative projection of the particle data and increases the accuracy of the MPMs.

This paper is structured as follows. Section 2 introduces the governing equations capturing the one-dimensional deformation of one-phase continua. Section 3 summarizes the MPM algorithm for numerical discretization and the solution of these equations and outlines the modifications required to obtain DDMPM. Section 4 focuses on the use of B-splines in MPM. Section 5 describes the main concepts of the TLS reconstruction and proves that its use with a sufficiently accurate Gauss quadrature leads to a conservative data projection. Section 6 discusses the numerical results obtained for vibrating bar and column compaction problems. Finally, Section 7 provides the conclusions.

## 2 | GOVERNING EQUATIONS

In this paper, it is assumed that the considered one-phase continuum occupies a domain  $\Omega_0 \subseteq \mathbb{R}$  at time  $t_0$  and a domain  $\Omega \subseteq \mathbb{R}$  at time  $t > t_0$ . The material does not undergo irreversible deformations. One-dimensional deformations of such a continuum can be described for small strains by a closed coupled system of partial differential equations for velocity and

stress in the following manner<sup>20</sup>:

$$\rho \frac{\partial v}{\partial t} = \frac{\partial \sigma}{\partial x} - \rho g, \quad (1)$$

$$\frac{\partial \sigma}{\partial t} = E \frac{\partial v}{\partial x}. \quad (2)$$

The coordinate  $x$  refers to the spatial direction, whereas  $t$  denotes the time. Moreover,  $\rho$  is the density,  $v$  is the velocity,  $\sigma$  is the stress tensor,  $g$  is the gravitational acceleration, and  $E$  is Young's modulus.

Equation (1) is obtained from the momentum balance equation, whereas Equation (2) describes the linear elastic constitutive relation for small strains. For large deformations, Equation (2) needs to be replaced by<sup>21</sup>

$$\frac{\partial \sigma}{\partial t} = (E - \sigma) \frac{\partial v}{\partial x}. \quad (3)$$

This system can be extended by a relation between the velocity  $v$  and displacement  $u$  given by

$$v = \frac{\partial u}{\partial t}. \quad (4)$$

In order to obtain a unique solution to the above system of equations, initial and boundary conditions have to be prescribed. Therefore, the following initial conditions are required:

$$u(x, 0) = u_0(x), \quad v(x, 0) = v_0(x), \quad \sigma(x, 0) = \sigma_0(x) \quad \forall x \in \Omega_0.$$

In addition, two types of boundary conditions are considered.

- Dirichlet boundary conditions (ie,  $u(x, t) = \hat{u}(t)$  for  $x \in \partial\Omega_u$ );
- Neumann boundary conditions (ie,  $\sigma(x, t) = \tau(t)$  for  $x \in \partial\Omega_\tau$ ).

### 3 | LOW-ORDER MATERIAL POINT METHODS

The momentum balance equation of a continuum (Equation (1)) is solved by MPM<sup>8,9</sup> in its weak formulation:

$$\int_{\Omega} \delta v \rho \frac{\partial v}{\partial t} d\Omega = \delta v \sigma|_{\partial\Omega} - \int_{\Omega} (\nabla \delta v) \sigma d\Omega - \int_{\Omega} \delta v \rho g d\Omega,$$

where  $\delta v$  denotes a test function.

We introduce vector notation for quantities defined at the DOFs to simplify the description. If the domain discretization generates  $N_n$  DOFs, we denote the velocity vector by  $\mathbf{v}(t) = [v_1(t) \ v_2(t) \ \dots \ v_{N_n}(t)]^T$ . The basis function vector is given by  $\boldsymbol{\phi}(x) = [\phi_1(x) \ \phi_2(x) \ \dots \ \phi_{N_n}(x)]^T$ , where  $\phi_i$  represents a basis function associated with the  $i$ th DOF. Thus, the velocity is approximated according to the finite-element approach as follows:

$$v(x, t) \approx \boldsymbol{\phi}(x)^T \mathbf{v}(t).$$

The displacement  $u$  and virtual velocity  $\delta v$  are discretized in a similar way.

The semidiscretized weak formulation can then be written as

$$\int_{\Omega} \rho \boldsymbol{\phi} \boldsymbol{\phi}^T \frac{d\mathbf{v}}{dt} d\Omega = \sigma(x, t) \boldsymbol{\phi}|_{\partial\Omega} - \int_{\Omega} \rho g \boldsymbol{\phi} d\Omega - \int_{\Omega} \sigma(x, t) \nabla \boldsymbol{\phi} d\Omega.$$

Denoting the acceleration vector  $\mathbf{a} = \frac{d\mathbf{v}}{dt}$ , the consistent mass matrix  $M^C = \int_{\Omega} \rho \boldsymbol{\phi} \boldsymbol{\phi}^T d\Omega$ , the external force vector  $\mathbf{f}^{\text{ext}} = \sigma(x, t) \boldsymbol{\phi}|_{\partial\Omega} - \int_{\Omega} \rho g \boldsymbol{\phi} d\Omega$ , and the internal force vector  $\mathbf{f}^{\text{int}} = \int_{\Omega} \sigma(x, t) \nabla \boldsymbol{\phi} d\Omega$ , we obtain

$$M^C \mathbf{a} = \mathbf{f}^{\text{ext}} - \mathbf{f}^{\text{int}}. \quad (5)$$

It should be noted that, in the considered version of MPM, the consistent mass matrix in Equation (5) is typically replaced by the lumped mass matrix  $M$ . The row-sum lumped mass matrix is obtained by summing the off-diagonal entries of each row and adding them to the diagonal entry:  $M_{ii} = \sum_{j=1}^{N_n} M_{ij}^C$ . Alternatively, for bases satisfying the partition

of unity property, the lumping can be performed variationally (ie,  $M = \int_{\Omega} \rho \boldsymbol{\phi} d\Omega$ ). Thus, the mass corresponding to the DOFs is stored in the diagonal  $\mathbf{m}$  of  $M$ .

$$M = \begin{bmatrix} & & \\ & \mathbf{m} & \\ & & \end{bmatrix}.$$

Finally, the stress is updated using the gradient of the basis function vector  $\nabla \boldsymbol{\phi}(x)$  and the constitutive relation. For example, for small deformations, Equation (2) yields

$$\boldsymbol{\sigma}(x, t) \approx E \nabla \boldsymbol{\phi}(x)^T \mathbf{u}(t). \quad (6)$$

In MPM, the material points move within the grid and serve as integration points. If the continuum is discretized by  $N_p$  material points, then for an arbitrary function  $f$ , the integral is computed as

$$\int_{\Omega} f(x) d\Omega \approx \sum_{p=1}^{N_p} V_p f(x_p),$$

where  $V_p$  is the particle volume.

In the considered version of MPM, the Euler-Cromer time-stepping scheme<sup>22</sup> is used for temporal discretization. This method is semi-implicit and applies the forward Euler method to advance the velocity in time and the backward Euler method to advance the displacement. The scheme is energy conservative and conditionally stable.

### 3.1 | Algorithm

When the deformation of a one-phase continuum is studied, each particle carries a certain volume  $V_p$ , density  $\rho_p$ , position  $x_p$ , velocity  $v_p$ , and stress  $\sigma_p$ . All these values are time-dependent, whereas the material-point mass  $m_p$  is constant throughout the simulation. The superscript  $t$  denotes the time level, and  $\Delta t$  is the time step size. At  $t = 0$  s, the material points are initialized (see the work of Al-Kafaji<sup>23</sup> for further details). Assuming that all particle properties are known at time  $t$ , the computation for time  $t + \Delta t$  proceeds as follows.

First, the data from the material points is projected to the DOFs. For example, the diagonal of the lumped mass matrix and the internal force vector are computed as

$$\mathbf{m}^t = \sum_{p=1}^{N_p} m_p \boldsymbol{\phi}(x_p^t), \quad (7)$$

$$(\mathbf{f}^{\text{int}})^t = \sum_{p=1}^{N_p} \sigma_p^t \cdot \nabla \boldsymbol{\phi}(x_p^t) V_p. \quad (8)$$

The accelerations at the DOFs are then obtained by combining the internal forces with any external forces, namely,

$$\mathbf{a}^t = (M^t)^{-1} \left( (\mathbf{f}^{\text{ext}})^t - (\mathbf{f}^{\text{int}})^t \right).$$

Meanwhile, for each DOF, the update-stress-last (USL) scheme<sup>1</sup> calculates the velocities directly from the accelerations, the modified-update-stress-last (MUSL) scheme<sup>2</sup> computes the particle velocities first and then maps the information to the DOFs. Since USL can generate ill-conditioned mass matrices when piecewise-linear basis functions are used, we consider the MUSL scheme. Therefore, in the next step, the velocity of particle  $p$  at time  $t + \Delta t$  is determined as

$$v_p^{t+\Delta t} = v_p^t + \Delta t \boldsymbol{\phi}^T(x_p^t) \mathbf{a}^t \quad \forall p = \{1, 2, \dots, N_p\}.$$

The velocities at the DOFs are subsequently obtained from

$$\mathbf{v}^{t+\Delta t} = (M^t)^{-1} \sum_{p=1}^{N_p} m_p \boldsymbol{\phi}(x_p^t) v_p^{t+\Delta t}. \quad (9)$$

This allows for computing the incremental displacement vector as

$$\Delta \mathbf{u}^{t+\Delta t} = \Delta t \mathbf{v}^{t+\Delta t}.$$

After these steps, the remaining part of the particle properties is updated

$$\begin{aligned} u_p^{t+\Delta t} &= u_p^t + \boldsymbol{\phi}^T(x_p^t) \Delta \mathbf{u}^{t+\Delta t}, \\ x_p^{t+\Delta t} &= x_p^t + \boldsymbol{\phi}^T(x_p^t) \Delta \mathbf{u}^{t+\Delta t}, \\ \Delta \epsilon_p^{t+\Delta t} &= \nabla \boldsymbol{\phi}^T(x_p^t) \Delta \mathbf{u}^{t+\Delta t}, \end{aligned}$$

where  $u_p$  and  $\Delta \epsilon_p$  are the particle displacement and incremental strain, respectively. The particle stress at time  $t + \Delta t$  is computed from  $\sigma^t$  and  $\Delta \epsilon_p^{t+\Delta t}$  using a constitutive law. Since only one-dimensional elastic deformations are considered, it follows from Equations (2), (3), and (6) that

$$\sigma_p^{t+\Delta t} = \begin{cases} \sigma_p^t + E \Delta \epsilon_p^{t+\Delta t} & \forall p = \{1, 2, \dots, N_p\} \text{ for small strains,} \\ \sigma_p^t + (E - \sigma_p^t) \Delta \epsilon_p^{t+\Delta t} & \forall p = \{1, 2, \dots, N_p\} \text{ for large strains.} \end{cases}$$

Finally, the volume and density of each particle are computed from

$$\begin{aligned} V_p^{t+\Delta t} &= (1 + \epsilon_p^{t+\Delta t}) V_p^t, \\ \rho_p^{t+\Delta t} &= \frac{\rho_p^t}{(1 + \Delta \epsilon_p^{t+\Delta t})}. \end{aligned}$$

### 3.2 | Piecewise-linear basis functions

The standard MPM uses piecewise-linear (P1) basis functions illustrated in Figure 1. On the one hand, P1 basis functions have several advantages.

*P.1.* They satisfy the partition of unity property

$$\sum_{i=1}^{N_n} \phi_i(x) = 1 \quad \forall x \in \Omega.$$

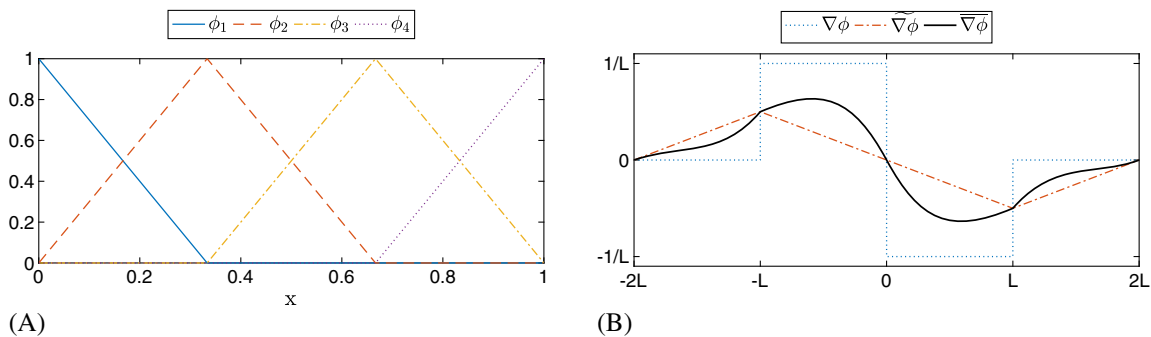
*P.2.* Each  $\phi_i$  has compact support. For  $i = 2, 3, \dots, N_n - 1$ ,  $\phi_i$  is supported by element  $i - 1$  and  $i$ . The compact supports of basis functions  $\phi_1$  and  $\phi_{N_n}$  are contained in element 1 and  $N_n - 1$ , respectively.

*P.3.* Each  $\phi_i$  is nonnegative over its entire support.

*P.4.* They allow for row-sum mass lumping due to *P. 3.*

*P.5.* They can be implemented in a straightforward manner.

On the other hand, the gradients of the piecewise-linear basis functions, just as the gradients of all other  $C^0$ -continuous basis functions, are discontinuous on the element boundaries, which can lead to unphysical oscillations in the internal forces when material points cross those boundaries. The gradient of a P1 function is shown as  $\nabla \phi$  in Figure 1.



**FIGURE 1** Piecewise-linear basis functions (A) and basis function gradients used in dual domain material point method (B). Here,  $L$  represents the element length [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

### 3.3 | Dual domain material point method

Here, DDMPM<sup>13</sup> uses piecewise-linear basis functions but replaces their gradients with smoother ones. For the construction of the new gradients, the method introduces a weight function  $\alpha$  and the gradient  $\widetilde{\nabla}\phi_i$  defined as

$$\widetilde{\nabla}\phi_i(x) = \sum_{j=1}^{N_n} \frac{\phi_j(x)}{V_j} \int_{\Omega} \phi_j \nabla \phi_i d\Omega,$$

where  $V_j$  is the volume associated with node  $j$  and given by  $V_j = \int_{\Omega} \phi_j d\Omega$ . The weight function is required to be zero on the cell boundaries but is not uniquely specified.<sup>13</sup> For the one-dimensional problems considered in this paper, the following expression is adopted:

$$\alpha(x) = 2 \sum_{i=1}^{N_n-1} \phi_i(x) \phi_{i+1}(x).$$

In addition, DDMPM substitutes the gradients of the piecewise-linear basis functions by

$$\overline{\nabla}\phi_i(x) = \alpha(x) \nabla \phi_i(x) + (1 - \alpha(x)) \widetilde{\nabla}\phi_i(x) \quad \forall i = \{1, \dots, N_n\}.$$

The gradients used in this method are depicted in Figure 1. From Section 3.1, it follows that DDMPM modifies the computation of the internal forces and strains.

## 4 | B-SPLINE MATERIAL POINT METHOD

Here, BSMPM<sup>14,15</sup> replaces the piecewise-linear basis functions from Section 3 by higher-order B-splines, which guarantee at least  $C^0$ -continuity of the gradients. The use of higher-order B-splines reduces not only the grid-crossing error but also the interpolation and time-stepping errors.<sup>11</sup> Similarly to piecewise-linear basis functions, B-splines satisfy properties *P. 1.-P. 4.* but are less straightforward to implement. Therefore, this paper adopts the BSMPM version proposed by Tielen et al<sup>15</sup> that reduces the implementation complexity by defining B-splines recursively following the Cox-de Boor formula.<sup>24</sup>

B-spline basis functions are constructed based on a *knot vector*, a sequence of ordered nondecreasing points in  $\mathbb{R}$  called *knots*. A knot vector is denoted by  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+l+1}\}$  with  $n$  and  $l$  being the number of basis functions and the polynomial order, respectively. If the knots are distributed equidistantly, they are said to be *uniform*. Otherwise, the knots are *nonuniform*. When more than one knot is positioned at the same location, the knots are called *repeated*. In an *open* knot vector, the first and last knots are repeated  $l + 1$  times, which ensures that the resulting basis functions are interpolatory at the domain boundaries. A nonempty knot interval  $[\xi_i, \xi_{i+1})$  is referred to as a *knot span*. For an open uniform knot vector, the number of spans is equal to  $n - l$ .

The Cox-de Boor formula defines B-spline basis functions recursively, starting with piecewise constants (no repeated knots, ie,  $l = 0$ )

$$\phi_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1}, \\ 0 & \text{otherwise.} \end{cases}$$

For  $l > 0$ , the basis functions are given by

$$\phi_{i,l}(\xi) = \frac{\xi - \xi_i}{\xi_{i+l} - \xi_i} \phi_{i,l-1}(\xi) + \frac{\xi_{i+l+1} - \xi}{\xi_{i+l+1} - \xi_{i+1}} \phi_{i+1,l-1}(\xi) \quad \xi \in \hat{\Omega},$$

where  $\hat{\Omega}$  is the parametric domain. The gradients of the B-spline basis functions can be computed from<sup>24</sup>

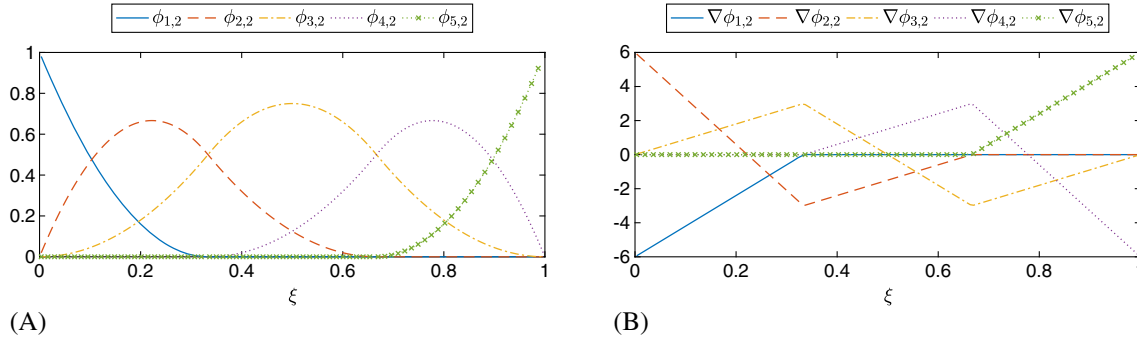
$$\frac{d\phi_{i,l}(\xi)}{d\xi} = \frac{l}{\xi_{i+l} - \xi_i} \phi_{i,l-1}(\xi) - \frac{l}{\xi_{i+l+1} - \xi_{i+1}} \phi_{i+1,l-1}(\xi).$$

B-spline basis functions satisfy the following properties.

*P.1.* They form a partition of unity:

$$\sum_{i=1}^n \phi_{i,l}(\xi) = 1 \quad \forall \xi \in \hat{\Omega}.$$

*P.2.* Each  $\phi_{i,l}$  has compact support  $[\xi_i, \xi_{i+l+1}]$ .



**FIGURE 2** Example of quadratic B-splines for an open, uniform knot vector (A), gradient of a quadratic B-spline basis function (B) [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

P.3. Each  $\phi_{i,l}$  is nonnegative in its support

$$\phi_{i,l}(\xi) \geq 0 \quad \forall \xi \in [\xi_i, \xi_{i+l+1}].$$

P.4. They allow for row-sum mass lumping because of P. 3.

In this paper, we only consider open, uniform knot vectors and restrict the analysis to quadratic B-spline basis functions (ie,  $l = 2$ ). Figure 2 shows quadratic basis functions with  $\Xi = \{0, 0, 0, 1/3, 2/3, 1, 1, 1\}$  and the corresponding gradients.

## 5 | FUNCTION RECONSTRUCTION

An important property of MPM, DDMPM, and BSMPPM is that they preserve the total mass  $\mathcal{M}$  and linear momentum  $\mathcal{P}$  of the system. Equation (7), Equation (9), and the partition of unity property of the basis functions yield

$$\begin{aligned} \mathcal{M} &= \sum_{i=1}^{N_n} m_i = \sum_{i=1}^{N_n} \sum_{p=1}^{N_p} m_p \phi_i(x_p) = \sum_{p=1}^{N_p} m_p \sum_{i=1}^{N_n} \phi_i(x_p) = \sum_{p=1}^{N_p} m_p, \\ \mathcal{P} &= \sum_{i=1}^{N_n} m_i v_i = \sum_{i=1}^{N_n} m_i \frac{1}{m_i} \sum_{p=1}^{N_p} m_p \phi_i(x_p) v_p = \sum_{p=1}^{N_p} m_p v_p \sum_{i=1}^{N_n} \phi_i(x_p) = \sum_{p=1}^{N_p} m_p v_p. \end{aligned}$$

The time index is dropped to simplify the notation.

Moreover, MPM can be viewed as a version of FEM, where the particles provide data for the background grid. In fact, the scattered material-point information is projected to the grid by reconstructing a function of interest and evaluating it at the nodes. Sulsky and Gong<sup>8,9</sup> point out that MPM reconstructs functions using a particular case of Shepard interpolation<sup>7</sup> that can introduce significant errors. They improve the accuracy of the method by adopting higher-order reconstruction techniques. In the modified versions, the velocity, density, and stress fields are reconstructed from the particle data and evaluated either at the nodes or element centres. The MPM-integration is then replaced by a one-point quadrature rule. Moreover, Tielen et al<sup>15,25</sup> introduce an alternative reconstruction technique for BSMPPM. They reconstruct functions using cubic-spline interpolation and integrate with a two-point Gauss rule on the half of each nonzero interval. However, it should be noted that while the adjusted mapping techniques can increase the accuracy of the solution, they do not necessarily conserve the total mass and momentum of the system.

In this section, we introduce a novel reconstruction technique, called Taylor least squares (TLS), which maintains the conservative properties of the standard algorithm. Its fundamental concepts are described in Sections 5.1 to 5.3. The use of the TLS technique for mapping the particle data within the material point methods is outlined in Section 5.4. A mathematical analysis of the conservation properties of the technique is provided in Section 5.5.

### 5.1 | Least squares approximation

Given a set of  $N_p$  distinct data points,  $\{x_p\}_{p=1}^{N_p}$  and the data values of these points,  $\{u(x_p)\}_{p=1}^{N_p}$ , we assume that  $u \in F$ , where  $F$  is a normed function space on  $\mathbb{R}$ , and  $P = \text{span}\{\psi_i\}_{i=1}^{n_b} \subset F$  is a set of  $n_b$  basis functions. The least squares<sup>16</sup>



approximation at a point  $x \in \mathbb{R}$  is the value  $w^* \in P$  that minimizes, among all  $w \in P$ , the least squares error

$$E = \sum_{p=1}^{N_p} (w(x_p) - u(x_p))^2.$$

Using the basis function vector,  $\boldsymbol{\psi}(x) = [\psi_1(x) \psi_2(x) \cdots \psi_{n_b}(x)]^T$  and the vector of unknown coefficients,  $\mathbf{a} = [a_1 \ a_2 \ \cdots \ a_{n_b}]^T$ , the least squares approximation can be written as

$$w^*(x) = \sum_{i=1}^{n_b} a_i \psi_i(x) = \boldsymbol{\psi}^T(x) \mathbf{a}.$$

In order to compute the coefficient vector,  $\frac{\partial E}{\partial a_i}$  is set to zero for  $i = 1, 2, \dots, n$ , leading to the normal equations

$$\sum_{p=1}^{N_p} \boldsymbol{\psi}(x_p) (\boldsymbol{\psi}^T(x_p) \mathbf{a} - u(x_p)) = \mathbf{0}.$$

Therefore, we obtain the following expression:

$$\mathbf{a} = \left( \sum_{p=1}^{N_p} \boldsymbol{\psi}(x_p) \boldsymbol{\psi}^T(x_p) \right)^{-1} \sum_{p=1}^{N_p} \boldsymbol{\psi}(x_p) u(x_p). \quad (10)$$

Defining  $D = \sum_{p=1}^{N_p} \boldsymbol{\psi}(x_p) \boldsymbol{\psi}^T(x_p)$ ,  $B = [\boldsymbol{\psi}(x_1) \ \boldsymbol{\psi}(x_2) \ \dots \ \boldsymbol{\psi}(x_{N_p})]$ , and  $U = [u(x_1) \ u(x_2) \ \dots \ u(x_{N_p})]^T$ , the least squares solution is given by

$$w^*(x) = \boldsymbol{\psi}^T(x) D^{-1} B U. \quad (11)$$

What remains to be specified is the basis for  $P$ .

## 5.2 | Taylor basis functions

A viable choice that leads to an overall conservative reconstruction scheme are the local Taylor basis functions.<sup>17</sup> To define these basis functions, we introduce the concept of the volume average of a function  $f$  over the cell  $e$

$$\bar{f} = \frac{1}{|\Omega_e|} \int_{\Omega_e} f \, d\Omega_e,$$

where  $|\Omega_e|$  is the volume of cell  $e$ . In one dimension,  $\Omega_e = [x_{\min}, x_{\max}]$  with  $x_{\max} > x_{\min}$ , and  $|\Omega_e| = x_{\max} - x_{\min}$ .

The Taylor basis functions are then given by

$$\begin{aligned} \psi_1 &= 1, \\ \psi_2 &= \frac{x - x_c}{\Delta x}, \\ \psi_3 &= \frac{(x - x_c)^2}{2\Delta x^2} - \frac{(x - x_c)^2}{2\Delta x^2}, \text{ etc.} \end{aligned}$$

Here,  $x_c = \frac{x_{\max} + x_{\min}}{2}$  is the cell centroid  $x_c$  of the cell  $e$ , and  $\Delta x = \frac{x_{\max} - x_{\min}}{2}$ .

An important quality of the Taylor basis that will ensure the conservation property of the reconstruction technique is

$$\int_{\Omega_e} \psi_i \, d\Omega_e = \begin{cases} |\Omega_e| & \text{if } i = 1, \\ 0 & \text{if } i \neq 1. \end{cases} \quad (12)$$

## 5.3 | Taylor least squares reconstruction

The TLS approach uses local Taylor basis functions for the least squares approximation of a function  $f$

$$f(x) \approx \hat{f}(x) = \sum_{i=1}^{n_b} a_i \psi_i(x).$$

Suppose that  $\int_{\Omega_e} f(x) d\Omega_e = c$  with  $c \in \mathbb{R}$  should be conserved by the reconstruction. Then, using Equation (12), we obtain

$$\int_{\Omega_e} \hat{f}(x) d\Omega_e = \int_{\Omega_e} \sum_{i=1}^{n_b} a_i \psi_i(x) d\Omega_e = \sum_{i=1}^{n_b} a_i \int_{\Omega_e} \psi_i(x) d\Omega_e = a_1 |\Omega_e|. \tag{13}$$

Therefore, the reconstruction procedure preserves the integral quantity if we set

$$a_1 := \frac{c}{|\Omega_e|}. \tag{14}$$

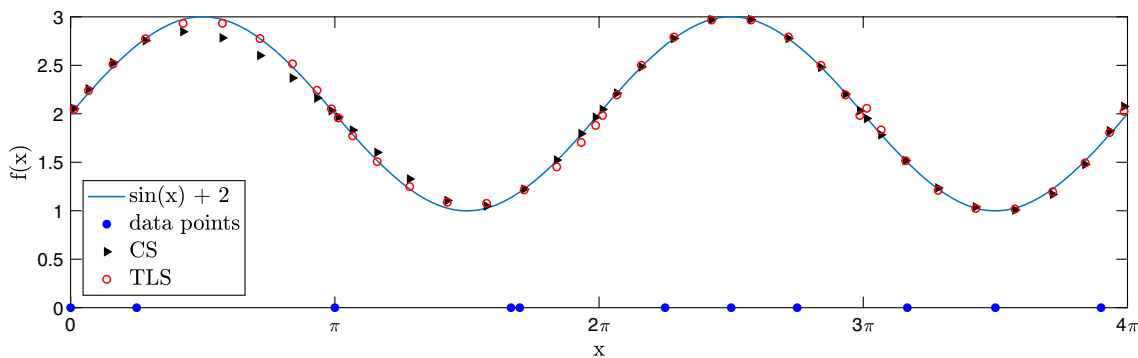
It should be noted that Equation (14) can be enforced explicitly.

We illustrate this property by reconstructing  $f(x) = \sin(x) + 2$  on  $[0, 4\pi]$ . In this case, the integral is equal to  $8\pi$ . The domain is divided into four elements of size  $\pi$  and contains 11 data points. Two data points are located at the boundaries of the first element, (ie, 0 and  $\pi$ ). In  $[2\pi, 3\pi]$ , the data points are distributed uniformly in the interior of the domain. The remaining data points have random positions creating different types of data distribution within an element. It should be noted that the data point at the right boundary of the first element is used for both the first and second elements.

The TLS approximation is obtained using three Taylor basis functions. We compare its performance with that of the cubic-spline reconstruction in terms of the root-mean-square (RMS) error for function  $f$  and the relative error for the integral of  $f$ . The RMS error is computed using 100 Gauss points per element, whereas for the numerical integration, the reconstructed function is evaluated only at two Gauss points within each element. Figure 3 visualizes the data point distribution and the cubic-spline and TLS reconstructions of  $f$  for 10 Gauss points per element, whereas Table 1 provides the corresponding errors.

Table 1 shows that, for the considered example, the TLS technique outperforms the cubic-spline reconstruction when the conservation and accuracy properties are considered. In fact, the TLS approach preserves the integral up to machine precision. However, Figure 3 shows that the performance of the TLS technique depends on the distribution of the data points within each element.

In some rare cases, data distribution can locally decrease the quality of the TLS approximation but has little influence on the cubic-spline interpolation. An example is provided in Figure 4, where  $[\pi, 2\pi]$  contains only two data points located at  $\pi$  and  $5/3\pi$ . This particular data distribution leads to a linear dependence between the columns of matrix  $D$  from

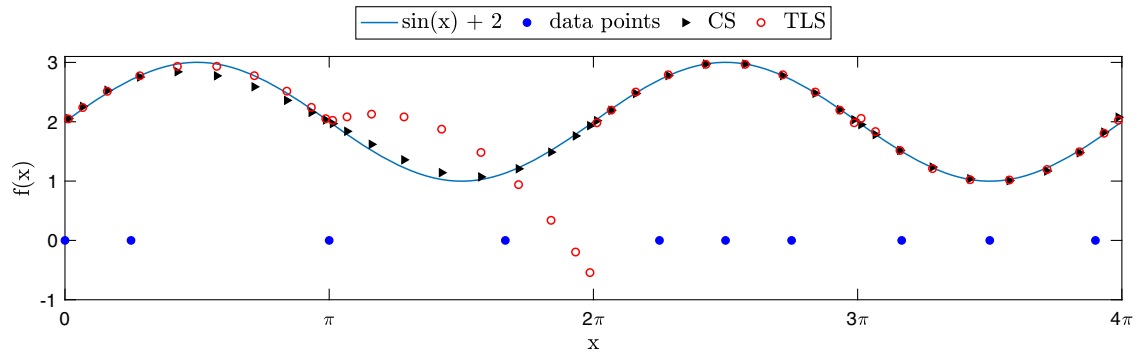


**FIGURE 3** Cubic-spline (CS) and Taylor least squares (TLS) reconstructions of  $f(x) = \sin(x) + 2$  on  $[0, 4\pi]$  for different types of data point distribution within an element [Colour figure can be viewed at wileyonlinelibrary.com]

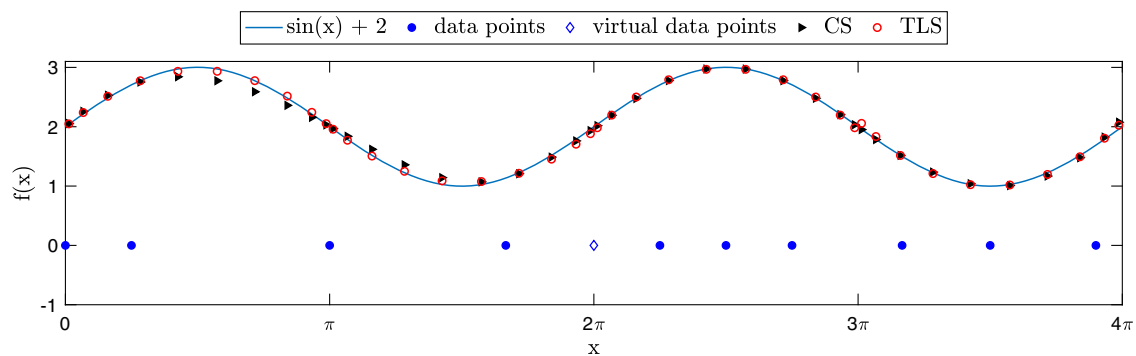
**TABLE 1** Errors made by Taylor least squares (TLS) and cubic-spline reconstruction for  $f(x) = \sin(x) + 2$  and its integral on  $[0, 4\pi]$  with four elements and eleven data points

Error	Cubic-splines	TLS
RMS error for function	$6.0867 \cdot 10^{-2}$	$3.9967 \cdot 10^{-2}$
Relative error for integral	$4.2759 \cdot 10^{-2}$	$2.7903 \cdot 10^{-15}$

Abbreviation: RMS, root mean square.



**FIGURE 4** Comparison of cubic-spline (CS) and Taylor least squares (TLS) reconstructions of  $f(x) = \sin(x) + 2$  on  $[0, 4\pi]$  for a challenging local data distribution [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]



**FIGURE 5** Comparison of cubic-spline (CS) and Taylor least squares (TLS) reconstructions of  $f(x) = \sin(x) + 2$  on  $[0, 4\pi]$  with a virtual data point at  $2\pi$  [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

Equation (11) and hence distorts the TLS approximation within this interval. The condition number of  $D$  can be used to detect the data distributions that decrease the accuracy of the TLS technique. In addition, the quality of the TLS technique in such situations can easily be improved.

First of all, the singularity of  $D$  can be prevented by reducing the number of basis functions used for the reconstruction on  $[\pi, 2\pi]$ . Although this strategy will preserve the conservative properties of the TLS technique and can be implemented in a straightforward manner, it will lower the accuracy of the method. Therefore, we suggest an alternative approach that uses the information from the neighboring intervals to maintain the high quality of the reconstruction scheme. This approach evaluates the TLS approximation of  $f$  associated with  $[2\pi, 3\pi]$  at  $2\pi$  and adds the obtained value to the set of data points, upon which the TLS approximation is based on  $[\pi, 2\pi]$ . However, this so-called *virtual* data point at  $2\pi$  is excluded from the computation of  $a_1$  in Equation (14). Thus, virtual points do not influence the conservative properties of the technique. Figure 5 illustrates the improved approximation.

## 5.4 | Mapping of particle data

When the TLS reconstruction is considered as part of a material point method, particles serve as data points. To conserve the integral of a certain quantity within each element, the coefficient of the first basis function is specified according to Equation (14). The remaining coefficients are calculated from Equation (10) excluding  $\psi_1$ , thereby leaving the integral value unchanged. When the conservation of the reconstructed quantity is not required, a standard least-square approach is followed. This implies that all coefficients are treated as unknowns and their values are obtained from Equation (10).

A TLS reconstruction is applied to replace the MPM integration in Equations (8) and (9) by an exact method such as an elementwise Gauss quadrature. We obtain the approximations with a quadratic TLS reconstruction. This implies that

only the first three Taylor basis functions are used (ie,  $n_b = 3$ ). In this case, a two-point Gauss rule within each element leads to an exact integration. The internal forces at the DOFs are computed as follows:

1. Apply a quadratic TLS approach to reconstruct the stress field from the particle data within each active element without specifying the coefficient of the first Taylor basis function:

$$\hat{\sigma}_e = \sum_{i=1}^{n_b} s_i \psi_i,$$

where  $s_i$  is the coefficient corresponding to Taylor basis function  $i$ . Outside of element  $e$ ,  $\hat{\sigma}_e$  is zero. The global approximation of the stress function,  $\hat{\sigma}$ , is then equal to

$$\hat{\sigma} = \sum_{e=1}^{N_e} \hat{\sigma}_e.$$

2. Integrate the stress approximation using a two-point Gauss quadrature

$$\mathbf{f}^{\text{int}} \approx \int_{\Omega} \hat{\sigma}(x, t) \nabla \boldsymbol{\phi} d\Omega = \sum_{g=1}^{N_g} \hat{\sigma}(x_g) \nabla \boldsymbol{\phi}(x_g) \omega_g,$$

where  $N_g$  is the total number of Gauss points,  $x_g$  is the global position of a Gauss point, and  $\omega_g$  is its weight.

The material-point velocities are mapped to the DOFs in the following manner.

1. Apply a quadratic TLS approach to reconstruct the density field and the product of density and velocity from the particle data within each active element, while preserving the mass and momentum of the element:

$$\hat{\rho}_e = \sum_{i=1}^{n_b} r_i \psi_i \quad \text{with} \quad r_1 = \frac{1}{|\Omega_e|} \sum_{\{p|x_p \in \Omega_e\}} m_p$$

$$(\hat{\rho}\hat{v})_e = \sum_{i=1}^{n_b} \gamma_i \psi_i \quad \text{with} \quad \gamma_1 = \frac{1}{|\Omega_e|} \sum_{\{p|x_p \in \Omega_e\}} m_p v_p.$$

where  $r_i$  and  $\gamma_i$  are the coefficients corresponding to Taylor basis function  $i$ . Outside of element  $e$ ,  $\hat{\rho}_e$ , and  $(\hat{\rho}\hat{v})_e$  are zero. The global approximations are then equal to

$$\hat{\rho} = \sum_{e=1}^{N_e} \hat{\rho}_e \quad \text{and} \quad (\hat{\rho}\hat{v}) = \sum_{e=1}^{N_e} (\hat{\rho}\hat{v})_e.$$

2. Integrate the approximations using a two-point Gauss quadrature to obtain the momentum vector  $\mathbf{p}$  and the consistent mass matrix  $M^C$

$$\mathbf{p} = \sum_{g=1}^{N_g} (\hat{\rho}\hat{v})(x_g) \omega_g \boldsymbol{\phi}(x_g),$$

$$M^C = \sum_{g=1}^{N_g} \hat{\rho}(x_g) \omega_g \boldsymbol{\phi}(x_g) (\boldsymbol{\phi}(x_g))^T. \quad (15)$$

It should be noted that  $M^C$  may be replaced by a lumped mass matrix without loss of the conservation property of the algorithm.

3. Compute the velocity vector

$$\mathbf{v} = (M^C)^{-1} \mathbf{p}.$$

From Section 5.3, it follows that the optimal performance of the TLS technique requires at least three particles in each element at the beginning of the simulation. Since this particle distribution is not preserved under large deformations, virtual data points may be used to improve the accuracy of the approximation within elements that contain only one or two material points.

## 5.5 | Conservation of mass and momentum

As mentioned in Section 5.4, the TLS technique reconstructs the density field and the product of density and velocity inside each element in the following way:

$$\rho_e \approx \hat{\rho}_e = \sum_{i=1}^{n_b} r_i \psi_i \quad \text{with } r_1 = \frac{1}{|\Omega_e|} \sum_{\{p|x_p \in \Omega_e\}} m_p,$$

$$(\rho v)_e \approx (\hat{\rho} v)_e = \sum_{i=1}^{n_b} \gamma_i \psi_i \quad \text{with } \gamma_1 = \frac{1}{|\Omega_e|} \sum_{\{p|x_p \in \Omega_e\}} m_p v_p.$$

According to Equations (13) and (14), this preserves the mass  $\mathcal{M}_e$  and momentum  $\mathcal{P}_e$  of element  $e$ . As a result, the total mass and momentum of the system are conserved after the TLS reconstruction

$$\mathcal{M} = \sum_{e=1}^{N_e} \mathcal{M}_e = \sum_{e=1}^{N_e} \sum_{\{p|x_p \in \Omega_e\}} m_p = \sum_{p=1}^{N_p} m_p, \quad (16)$$

$$\mathcal{P} = \sum_{e=1}^{N_e} \mathcal{P}_e = \sum_{e=1}^{N_e} \sum_{\{p|x_p \in \Omega_e\}} m_p v_p = \sum_{p=1}^{N_p} m_p v_p.$$

The mass- and momentum-conservation property of the mapping obtained using TLS reconstruction and Gauss quadrature can be shown as well.

Since the total mass is equal to the sum of the entries in the mass matrix from Equation (15), it can be written as

$$\mathcal{M} = \sum_{i=1}^{N_n} \sum_{j=1}^{N_n} M_{ij}^C = \sum_{i=1}^{N_n} \sum_{j=1}^{N_n} \sum_{g=1}^{N_g} \hat{\rho}(x_g) \omega_g \phi_i(x_g) \phi_j(x_g) = \sum_{g=1}^{N_g} \hat{\rho}(x_g) \omega_g \sum_{i=1}^{N_n} \phi_i(x_g) \sum_{j=1}^{N_n} \phi_j(x_g) = \sum_{g=1}^{N_g} \hat{\rho}(x_g) \omega_g.$$

The last equality is derived using the partition of unity property of piecewise-linear and B-spline basis functions. In the remaining part of the proof, we assume that  $n_b \leq \frac{2N_g}{N_e}$  and that there are  $\frac{N_g}{N_e}$  integration points per element (or knot span), so that the Gauss quadrature is exact. Therefore, the following holds:

$$\mathcal{M} = \sum_{g=1}^{N_g} \hat{\rho}(x_g) \omega_g = \sum_{e=1}^{N_e} \sum_{\{g|x_g \in \Omega_e\}} \hat{\rho}_e(x_g) \omega_g = \sum_{e=1}^{N_e} \int_{\Omega_e} \hat{\rho}_e d\Omega_e = \sum_{e=1}^{N_e} \mathcal{M}_e = \sum_{p=1}^{N_p} m_p.$$

The last two steps emerge from the conservation of mass per element and Equation (16).

For the linear momentum, we also assume that  $n_b \leq \frac{2N_g}{N_e}$  and that there are  $\frac{N_g}{N_e}$  integration points per element (or knot span). Following the above steps, the total momentum after the mapping can be written as

$$\begin{aligned} \mathcal{P} &= \sum_{i=1}^{N_n} p_i = \sum_{i=1}^{N_n} \sum_{g=1}^{N_g} (\hat{\rho} v)(x_g) \omega_g \phi_i(x_g) = \sum_{g=1}^{N_g} (\hat{\rho} v)(x_g) \omega_g \sum_{i=1}^{N_n} \phi_i(x_g) \\ &= \sum_{g=1}^{N_g} (\hat{\rho} v)(x_g) \omega_g = \sum_{e=1}^{N_e} \sum_{\{g|x_g \in \Omega_e\}} (\hat{\rho} v)_e(x_g) \omega_g \\ &= \sum_{e=1}^{N_e} \int_{\Omega_e} (\hat{\rho} v)_e d\Omega_e = \sum_{e=1}^{N_e} \mathcal{P}_e = \sum_{p=1}^{N_p} m_p v_p. \end{aligned}$$

Therefore, we have shown that if the Gauss quadrature is performed using a sufficient number of integration points per element, the mass and momentum balance is satisfied not only by the TLS function reconstruction but also by its combination with the Gauss quadrature.

## 6 | NUMERICAL RESULTS

We study the conservation property of the material point methods by calculating the maximum relative errors in the total mass and momentum over all time steps before and after the computation of the velocity at the DOFs. For MPM, DDMPM, and BSMPM, the errors in the mass and momentum are bounded by  $1 \cdot 10^{-15}$  for the vibrating bar benchmark, and  $1 \cdot 10^{-13}$  for the column compaction example. Therefore, the TLS results are only compared to those obtained with the cubic-spline reconstruction.

### 6.1 | Vibrating bar

This example describes the vibration of a one-phase bar with fixed ends. The motion triggered by an initial velocity that varies along the bar is captured by Equations (1)-(4) with  $g = 0$  (ie, the gravitational force is neglected) and the following initial and boundary conditions:

$$\begin{aligned} u(x, 0) &= 0, \quad v(x, 0) = v_0 \sin\left(\frac{\pi x}{H}\right), \quad \sigma(x, 0) = 0; \\ u(0, t) &= 0, \quad u(H, t) = 0. \end{aligned}$$

For small strains, the analytical solution in terms of displacement, velocity, and stress is given by

$$\begin{aligned} u(x, t) &= \frac{v_0 H}{\pi \sqrt{E/\rho}} \sin\left(\frac{\pi \sqrt{E/\rho} t}{H}\right) \sin\left(\frac{\pi x}{H}\right), \\ v(x, t) &= v_0 \cos\left(\frac{\pi \sqrt{E/\rho} t}{H}\right) \sin\left(\frac{\pi x}{H}\right), \\ \sigma(x, t) &= v_0 \sqrt{E/\rho} \sin\left(\frac{\pi \sqrt{E/\rho} t}{H}\right) \cos\left(\frac{\pi x}{H}\right). \end{aligned}$$

The material-point solutions are considered at the particle positions.

Table 2 provides exemplary parameter values for the vibrating bar benchmark under small deformations. For the spatial convergence analysis, we minimize the contribution of temporal errors by using an artificially small time-step size and a short simulation time. In fact, the time-step size and total simulation time of, respectively,  $1 \cdot 10^{-7}$  s and  $1.9 \cdot 10^{-6}$  s eliminate the contribution of the temporal errors and are significantly increased for the grid-crossing study presented later in this section. Furthermore, the number of elements (knot spans) is varied from 5 to 40, while the initial number of particles per cell (PPC) is fixed to 12. These settings ensure that grid crossing does not occur, and the maximal observed strain is equal to  $5.3 \cdot 10^{-7}$  m.

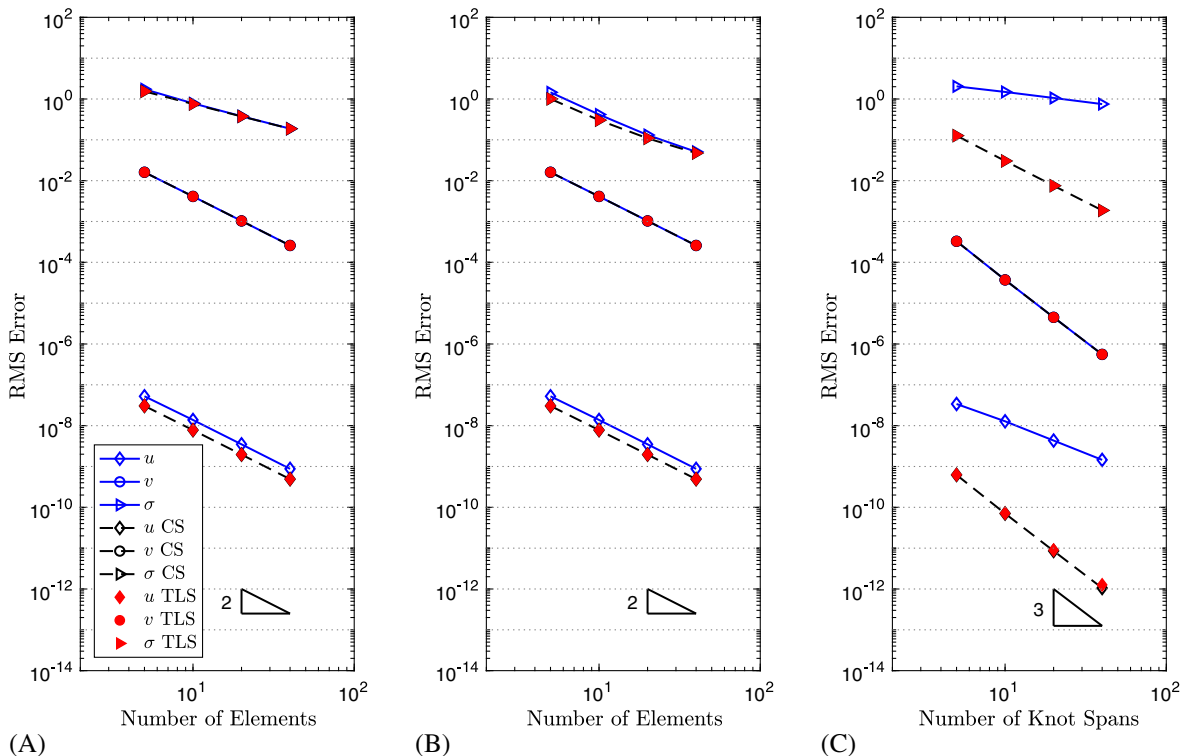
The results in terms of spatial errors are shown in Figure 6. As expected, MPM with piecewise-linear basis functions demonstrates second-order convergence in both the displacement and velocity. Since the stress is not discretized directly but is computed from the displacement by taking the derivative, its convergence rate is one. The application of the TLS reconstruction technique, as well as the cubic-spline interpolation, has almost no influence on the stress but decreases the displacement error by a factor of 1.7. For DDMPM, the application of the reconstruction techniques tends to reduce not only the error in the displacement but also in the stress. However, velocity results are similar to those obtained for MPM.

**TABLE 2** Exemplary parameters allowing for small deformations in the vibrating bar problem

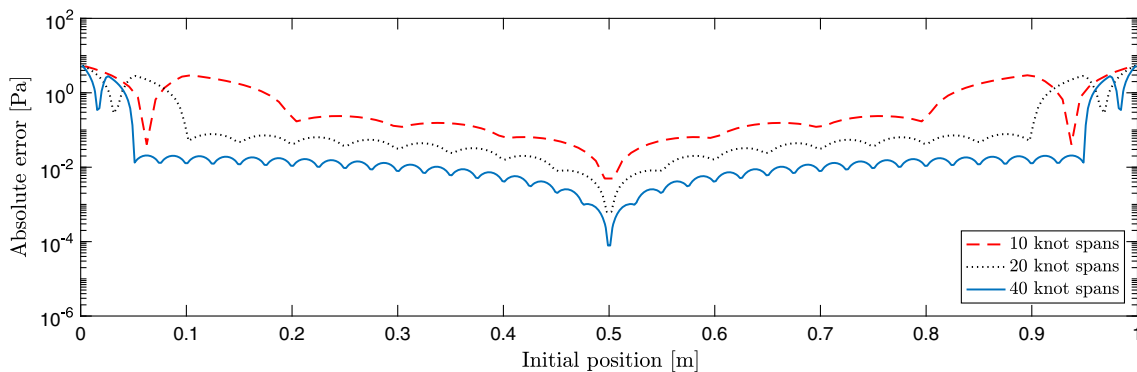
Parameter	Symbol	Value	Unit
Height	$H$	1.00	m
Initial density	$\rho$	$2.00 \cdot 10^3$	kg/m <sup>3</sup>
Young's modulus	$E$	$7.00 \cdot 10^6$	Pa
Max. initial velocity	$v_0$	0.28	m/s <sup>2</sup>

The use of quadratic B-splines as basis functions leads to a significant decrease in the error and a higher convergence order for the velocity but causes problems at the boundaries of the domain for both stress and displacement. The absolute error in the stress over the domain is shown in Figure 7. The large values of the error at the boundaries prevent the reduction of the RMS error and worsen the convergence properties of the method. However, the use of BSMPM with a function reconstruction techniques eliminates the boundary issues. An example is provided in Figure 8. Consequently, third-order convergence is obtained for all the considered quantities. It should also be noted that the integration of the TLS or spline reconstruction in BSMPM produces more accurate results than the other considered methods.

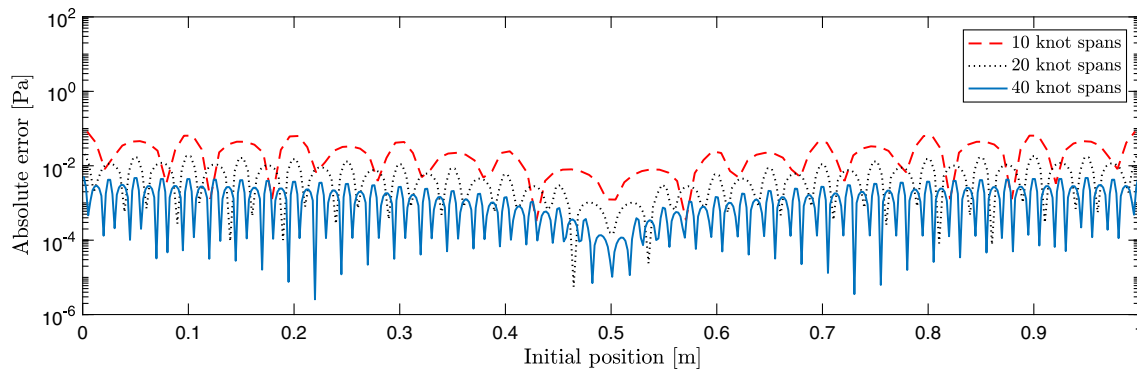
Table 3 compares the relative error in the mass and momentum made using the TLS and cubic-spline reconstructions. The results are provided for MPM, DDMPM, and BSMPM applied to the vibrating bar problem discretized by 40 elements (knot spans) and 12 PPC. They demonstrate that while the cubic-spline interpolation tends to accurately conserve the mass, it produces errors of order  $10^{-6}$  for the linear momentum. The errors produced by the TLS approach consistently



**FIGURE 6** Spatial convergence of material point methods for the vibrating bar problem without grid crossing. The results are shown for the material point methods without reconstruction techniques, with cubic-spline (CS) reconstruction and Taylor least squares (TLS) reconstruction. RMS, root mean square



**FIGURE 7** Absolute error obtained with B-spline material point method for stress distribution in the vibrating bar problem without grid crossing [Colour figure can be viewed at wileyonlinelibrary.com]



**FIGURE 8** Absolute error obtained with B-spline material point method Taylor least squares for stress distribution in the vibrating bar problem without grid crossing [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

**TABLE 3** Maximum relative errors in the total mass  $\mathcal{M}$  and momentum  $\mathcal{P}$  over the simulation run with cubic-spline and Taylor least squares (TLS) reconstructions when grid crossing does not occur. The vibrating bar is discretized using 40 elements (knot spans) and 12 particles per cell

	MPM		DDMPM		BSMPM	
	Error $_{\mathcal{M}}$	Error $_{\mathcal{P}}$	Error $_{\mathcal{M}}$	Error $_{\mathcal{P}}$	Error $_{\mathcal{M}}$	Error $_{\mathcal{P}}$
Cubic-spline	$2.3825 \cdot 10^{-12}$	$3.7053 \cdot 10^{-6}$	$3.9211 \cdot 10^{-13}$	$3.7053 \cdot 10^{-6}$	$7.3896 \cdot 10^{-15}$	$1.7859 \cdot 10^{-6}$
TLS	$7.5033 \cdot 10^{-15}$	$2.1007 \cdot 10^{-16}$	$7.3896 \cdot 10^{-15}$	$2.5208 \cdot 10^{-16}$	$7.5033 \cdot 10^{-15}$	$1.7623 \cdot 10^{-16}$

Abbreviations: BSMPM, B-spline material point method; DDMPM, dual domain material point method; MPM, material point method.

**TABLE 4** Exemplary parameters allowing for large deformations in the vibrating bar problem

Parameter	Symbol	Value	Unit
Height	$H$	1.00	m
Initial density	$\rho$	$2.00 \cdot 10^3$	kg/m <sup>3</sup>
Young's modulus	$E$	$4.00 \cdot 10^4$	Pa
Max. initial velocity	$v_0$	0.80	m/s <sup>2</sup>

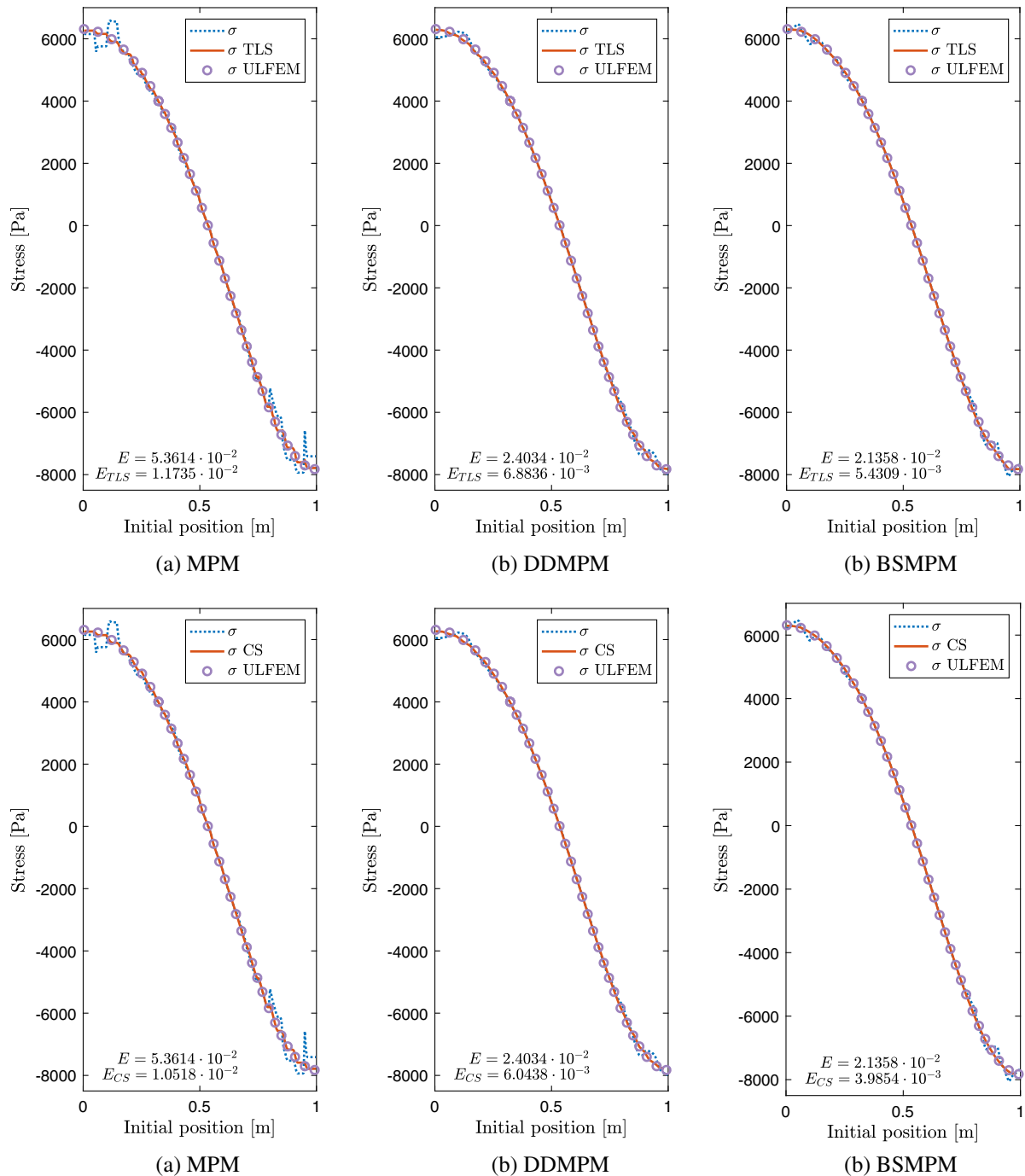
remain close to machine precision and hence are orders of magnitude smaller than those generated by the cubic-spline interpolation.

For large-deformation simulations, the parameters from Table 4 are used. The time-step size and the simulation time are increased to  $1 \cdot 10^{-4}$  s and 0.1 s, respectively. The domain is discretized using 20 elements (knot spans) with initially 8 PPC. The maximal strain that is reached is 0.056 m. Since the analytical solution is not available when the vibrating bar experiences large deformations, the numerical results are compared to the solution obtained with the updated Lagrangian finite element method (ULFEM).<sup>26</sup>

In the standard MPM simulation, material points cross the element boundaries more than 100 times, leading to significant inaccuracies in the results. Although grid crossing influences the computation of the displacement and velocity, its most evident consequences are in the stress distribution. DDMPM and BSMPM reduce the grid-crossing error, but their results still significantly deviate from the solution provided by ULFEM. This is shown in Figure 9.

Figure 9 also illustrates that the application of the TLS approximation or the cubic-spline reconstruction leads to good agreement of the MPM, DDMPM, and BSMPM solutions with that of ULFEM. The maximal reduction of the relative error in  $L_2$ -norm made by standard MPM is achieved when the reconstruction techniques are combined with BSMPM. More precisely, the integration of the spline interpolation or the TLS reconstruction in BSMPM decreases the MPM error by a factor of 13.5 and 9.9, respectively. On the other hand, the conjunction of the reconstruction techniques with DDMPM leads to highly accurate results as well. The spline and TLS reconstruction reduces the MPM error by a factor of 8.9 and 7.8, respectively. As expected, the TLS approximation conserves the total mass and linear momentum significantly more accurately than the spline interpolation, regardless of the material-point method. The conservative properties of the reconstruction techniques are provided in Table 5.





**FIGURE 9** Stress distribution and corresponding relative errors in the  $L_2$ -norm in the vibrating bar problem with grid crossing. The results are obtained for the material point methods without reconstruction technique, with cubic-spline reconstruction (CS) and Taylor least squares (TLS) reconstruction, and are compared to the updated Lagrangian finite element method (ULFEM) results [Colour figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

**TABLE 5** Maximum relative errors in the total mass  $\mathcal{M}$  and momentum  $\mathcal{P}$  over the simulation run with cubic-spline and Taylor least squares (TLS) reconstructions when grid crossing occurs. The vibrating bar is discretized using 20 elements (knot spans) and 8 particles per cell

	MPM		DDMPM		BSMPM	
	$Error_{\mathcal{M}}$	$Error_{\mathcal{P}}$	$Error_{\mathcal{M}}$	$Error_{\mathcal{P}}$	$Error_{\mathcal{M}}$	$Error_{\mathcal{P}}$
Cubic-spline	$8.2084 \cdot 10^{-5}$	$8.0098 \cdot 10^{-5}$	$3.5869 \cdot 10^{-5}$	$3.7566 \cdot 10^{-5}$	$7.3889 \cdot 10^{-7}$	$1.5794 \cdot 10^{-5}$
TLS	$2.9104 \cdot 10^{-14}$	$5.8262 \cdot 10^{-15}$	$2.8990 \cdot 10^{-14}$	$5.7562 \cdot 10^{-15}$	$2.8876 \cdot 10^{-14}$	$5.7186 \cdot 10^{-15}$

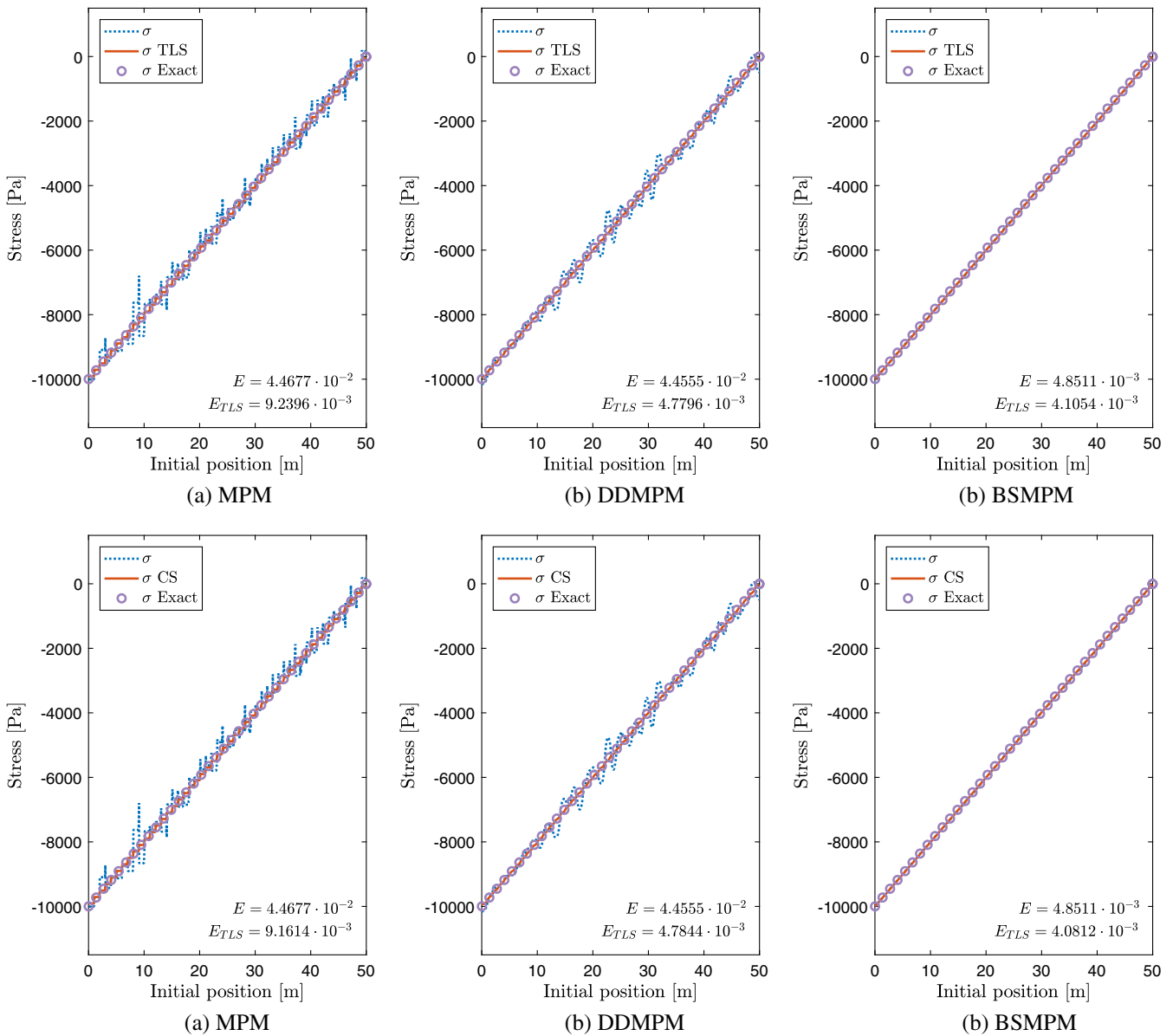
Abbreviations: BSMPM, B-spline material point method; DDMPM, dual domain material point method; MPM, material point method.

### 6.2 | Column compaction

This example considers the response of a column of material to a compressive body force. The body force increases linearly with time during the simulation. A detailed description of the problem that includes the analytical solution is provided by Bardenhagen and Kober.<sup>10</sup>

**TABLE 6** Exemplary parameters for the column compaction problem

Parameter	Symbol	Value	Unit
Initial height	$H$	50.0	m
Initial density	$\rho$	1.00	kg/m <sup>3</sup>
Young's modulus	$E$	$1.00 \cdot 10^6$	Pa
Max. body force	$b_f$	-200	N



**FIGURE 10** Stress distribution and corresponding relative errors in the  $L_2$ -norm in the column compaction problem with grid crossing. The results are obtained for the material point methods without reconstruction technique, with cubic-spline reconstruction (CS) and Taylor least squares (TLS) reconstruction, and are compared to the analytical solution [Colour figure can be viewed at wileyonlinelibrary.com]

**TABLE 7** Maximum relative errors in the total mass  $\mathcal{M}$  and momentum  $\mathcal{P}$  over the simulation run with cubic-spline and Taylor least squares (TLS) reconstructions for column compaction problem

	MPM		DDMPM		BSMPM	
	Error $_{\mathcal{M}}$	Error $_{\mathcal{P}}$	Error $_{\mathcal{M}}$	Error $_{\mathcal{P}}$	Error $_{\mathcal{M}}$	Error $_{\mathcal{P}}$
Cubic-spline	$3.0803 \cdot 10^{-6}$	$3.5400 \cdot 10^{-6}$	$4.7571 \cdot 10^{-7}$	$2.0403 \cdot 10^{-6}$	$1.5112 \cdot 10^{-9}$	$1.4039 \cdot 10^{-7}$
TLS	$1.0526 \cdot 10^{-12}$	$5.6413 \cdot 10^{-13}$	$1.0524 \cdot 10^{-12}$	$5.6464 \cdot 10^{-13}$	$1.0527 \cdot 10^{-12}$	$5.6578 \cdot 10^{-13}$

Abbreviations: BSMPM, B-spline material point method; DDMPM, dual domain material point method; MPM, material point method.

**TABLE 8** Relative error in the  $L_2$ -norm for the stress, comparison with the relative error in the  $L_2$ -norm of dual domain material point method (DDMPM) Taylor least squares, and computational time normalized with respect to DDMPM Taylor least squares obtained using DDMPM with subpoints for the column compaction problem

Number of Subpoints	$E_{\text{subpoints}}$	$E_{\text{subpoints}}/E_{\text{TLS}}$	$t_{\text{subpoints}}/t_{\text{TLS}}$
2	$3.4086 \cdot 10^{-2}$	4.9518	0.3037
4	$2.7207 \cdot 10^{-2}$	3.9524	0.4224
8	$2.1469 \cdot 10^{-2}$	3.1189	0.6793
16	$1.5236 \cdot 10^{-2}$	2.2134	1.2786
32	$8.1219 \cdot 10^{-3}$	1.1799	3.3569
64	$6.2274 \cdot 10^{-3}$	0.9047	6.3747
128	$5.6118 \cdot 10^{-3}$	0.8152	14.422

Parameter values used for the simulation are listed in Table 6. The time-step size and total simulation time are  $1 \cdot 10^{-4}$  s and 0.5 s, respectively. Moreover, the number of elements (knot spans) is equal to 50, while the initial number of particles per cell is set to 20. When the standard MPM is used, material-points cross element boundaries 180 times, and the maximal strain is equal to 0.27 m.

The obtained results are shown in Figure 10. The influence of grid-crossing error on the solution is clearly evident for the standard MPM. While DDMPM only slightly smoothens the oscillations, BSMPM significantly improves the results. In fact, the use of B-spline basis functions reduces the relative error in  $L_2$ -norm by a factor of 10.

Both the TLS reconstruction technique and cubic-spline reconstruction interpolation provide considerably more accurate results than MPM and DDMPM. They also further improve the accuracy of BSMPM, but their influence is less distinct in this case. It should be noted that the error produced with the cubic-spline reconstruction is slightly lower than that achieved with the TLS approach. However, the TLS technique outperforms the former method by accurately conserving the mass and linear momentum. As was mentioned previously in this section, the maximum relative errors of MPM, DDMPM, and BSMPM are bounded by  $1 \cdot 10^{-13}$ . While the TLS approach has little influence on these errors, the spline reconstruction technique increases them to  $2.04 \cdot 10^{-6}$  (Table 7).

Finally, for DDMPM, the TLS mapping is compared with the conservative integration technique proposed by Dhakal and Zhang<sup>18</sup> that is based on subpoints. In Table 8, we present the relative error and computational time for the DDMPM with a varying number of subpoints and compare it with the DDMPM-TLS result from Figure 10.

From the Table, it follows that, for a low number of subpoints, the technique of Dhakal and Zhang effectively decreases the relative error of  $4.4555 \cdot 10^{-2}$  produced by DDMPM and is less computationally intensive than the TLS technique. However, its performance may be limited by computational time and memory when a higher accuracy has to be reached. In fact, the method approaches the accuracy of the TLS reconstruction technique when 128 subpoints per particle are used, leading to a significant (factor of 14.4) increase in the computational time.

## 7 | SUMMARY AND CONCLUSIONS

In this paper, we have introduced a novel TLS reconstruction approach and applied it to MPMs. The proposed technique combines the least squares approximation with local Taylor basis functions to accurately reconstruct the quantities of interest (eg, stress and density fields) from scattered particle data within each element. We have shown that, in contrast to standard reconstruction techniques, the TLS approximation conserves the mass and linear momentum of the system

after the material-point data is mapped to the integration points. More importantly, in conjunction with a sufficiently accurate numerical quadrature method, the technique preserves the total mass and momentum after the information is projected to the degrees of freedom of the grid. This implies that the TLS reconstruction maintains the physical properties of the standard MPM.

The TLS technique was applied to MPM, DDMPM, and BSMPPM within the vibrating bar and column compaction problems. For the vibrating bar problem in the absence of grid crossing, the reconstruction technique had little influence on MPM and DDMPM but was able to improve the convergence properties of BSMPPM. When material points started to cross cell boundaries in both considered examples, the TLS approximation smoothed the solutions of MPM and related methods and brought them closer to the solution computed analytically or by ULFEM.

Furthermore, the novel TLS reconstruction techniques was compared with the global spline reconstruction technique previously used within BSMPPM. In general, the differences in spatial accuracy of the techniques were negligible, but the error in total mass and linear momentum was consistently much lower for the TLS reconstruction. For the column compaction problem, it was also shown that the TLS technique integrated into DDMPM reaches a higher accuracy than the conservative DDMPM method with subpoints while requiring lower computational costs. Therefore, this study has demonstrated that the combination of MPM and related methods with the TLS technique leads to a higher accuracy of the material point methods and preserves their fundamental physical properties.

## ACKNOWLEDGEMENTS

This research was carried out as a part of the MPM-Flow project “Understanding flow slides in flood defences.” It has received funding from the Netherlands Organisation for Scientific Research (NWO), Deltares, Royal Boskalis Westminster N.V., Van Oord Dredging and Marine Contractors, Rijkswaterstaat, and Stichting FloodControl IJkdijk. The authors are also grateful to Roel Tielen for his excellent work on BSMPPM.

## ORCID

Elizaveta Wobbes  <http://orcid.org/0000-0001-5567-0886>

## REFERENCES

1. Sulsky D, Chen Z, Schreyer HL. A particle method for history-dependent materials. *Comput Methods Appl Mech Eng*. 1994;118(1-2):179-196.
2. Sulsky D, Zhou S-J, Schreyer HL. Application of a particle-in-cell method to solid mechanics. *Comput Phys Commun*. 1995;87(1-2):236-252.
3. Brenner SC, Scott LR. *The Mathematical Theory of Finite Element Methods*. 3rd ed. New York, NY: Springer Science+Business Media LLC; 2008.
4. Martinelli M, Rohe A. Modelling fluidisation and sedimentation using material point method. Paper presented at: 1st Pan-American Congress on Computational Mechanics; 2015; Buenos Aires, Argentina.
5. Ceccato F, Beuth L, Vermeer PA, Simonini P. Two-phase material point method applied to the study of cone penetration. *Comput Geotech*. 2016;80:440-452.
6. Galavi V, Tehrani FS, Martinelli M, Elkadi AS, Luger D. Axisymmetric formulation of the material point method for geotechnical engineering applications. In: *Numerical Methods in Geotechnical Engineering IX*. Boca Raton, FL: CRC Press; 2018:427-434.
7. Shepard D. A two-dimensional interpolation function for irregularly-spaced data. In: *Proceedings of the 1968 23rd ACM National Conference*; 1968; Las Vegas, NV.
8. Sulsky D, Gong M. Improving the material-point method. In: *Innovative Numerical Approaches for Multi-Field and Multi-Scale Problems: In Honor of Michael Ortiz's 60th Birthday*. Basel, Switzerland: Springer International Publishing Switzerland; 2016:217-240.
9. Gong M. *Improving the Material Point Method* [dissertation]. Albuquerque, NM: The University of New Mexico; 2015.
10. Bardenhagen SG, Kober EM. The generalized interpolation material point method. *Comput Model Eng Sci*. 2004;5(6):477-496.
11. Steffen M, Kirby RM, Berzins M. Decoupling and balancing of space and time errors in the material point method (MPM). *Int J Numer Methods Eng*. 2010;82(10):1207-1243.
12. 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 Methods Eng*. 2011;86(12):1435-1456.
13. Zhang DZ, Ma X, Giguere PT. Material point method enhanced by modified gradient of shape function. *J Comput Phys*. 2011;230(16):6379-6398.
14. Steffen M, Kirby RM, Berzins M. Analysis and reduction of quadrature errors in the material point method (MPM). *Int J Numer Methods Eng*. 2008;76(6):922-948.
15. Tielen R, Wobbes E, Möller M, Beuth L. A high order material point method. *Procedia Eng*. 2017;175:265-272.

16. Legendre AM. *Nouvelles Méthodes Pour La Détermination Des Orbites Des Comètes*. Paris, France: F. Didot; 1805.
17. Luo H, Baum JD, Löhner R. A discontinuous Galerkin method based on a Taylor basis for the compressible flows on arbitrary grids. *J Comput Phys*. 2008;227(20):8875-8893.
18. Dhakal TR, Zhang DZ. Material point methods applied to one-dimensional shock waves and dual domain material point method with sub-points. *J Comput Phys*. 2016;325:301-313.
19. Gan Y, Sun Z, Chen Z, Zhang X, Liu Y. Enhancement of the material point method using B-spline basis functions. *Int J Numer Methods Eng*. 2018;113(3):411-431.
20. Malvern LE. *Introduction to the Mechanics of a Continuous Medium*. Englewood Cliffs, NJ: Prentice-Hall; 1969.
21. Hill R. Some basic principles in the mechanics of solids without a natural time. *J Mech Phys Solids*. 1959;7(3):209-225.
22. Cromer A. Stable solutions using the Euler approximation. *Am J Phys*. 1981;49(5):455-459.
23. Al-Kafaji IKJ. *Formulation of a Dynamic Material Point Method (MPM) for Geomechanical Problems* [dissertation]. Stuttgart, Germany: Institut für Geotechnik der Universität Stuttgart; 2013.
24. De Boor C. *A Practical Guide to Splines*. New York, NY: Springer; 2001. *Applied Mathematical Sciences*; vol. 27.
25. Tielen RPWM. *High-Order Material Point Method* [thesis]. Delft, The Netherlands: Delft University of Technology; 2016.
26. Bathe K-J, Ramm E, Wilson E-L. Finite element formulations for large deformation dynamic analysis. *Int J Numer Methods Eng*. 1975;9(2):353-386.

**How to cite this article:** Wobbes E, Möller M, Galavi V, Vuik C. Conservative Taylor least squares reconstruction with application to material point methods. *Int J Numer Methods Eng*. 2019;117:271–290. <https://doi.org/10.1002/nme.5956>