Preprint ONLY
(submitted for publication)

Computational Error Estimation for The Material Point Method

Martin Berzins[1]

Received: date / Accepted: date

Abstract A common feature of many methods in computational mechanics is that there is often a way of estimating the error in the computed solution. The situation for computational mechanics codes based upon the Material Point Method is very different in that there has been comparatively little work on computable error estimates for these methods. This work is concerned with introducing such an approach for the Material Point Method. Although it has been observed that spatial errors may dominate temporal ones at stable time steps, recent work has made more precise the sources and forms of the different MPM errors. There is then a need to estimate these errors computationally through computable estimates of the different errors in the material point method. Estimates of the different spatial errors in the Material Point Method are constructed based upon nodal derivatives of the different physical variables in MPM. These derivatives are then estimated using standard difference approximations calculated on the background mesh. The use of these estimates of the spatial error makes it possible to measure the growth of errors over time. A number of computational experiments are used to illustrate the performance of the computed error estimates. As the key feature of the approach is the calculation of derivatives on the regularly spaced background mesh, the extension to calculating derivatives and hence to error estimates for higher dimensional problems is clearly possible.

Keywords Material Point Method, Spatial Estimation, Time Integration Error, Particle in Cell Method

1 INTRODUCTION

A notable feature of many computational mechanics codes, particularly finite element codes, since the 1980s has been the availability of computable estimates of the

[1] Scientific Computing and Imaging Institute
University of Utah
Salt Lake City, UT 84112 USA
mb@sci.utah.edu

error and the use of these estimates in accuracy checks, adaptive mesh refinement and element order selection. The Material Point Method (MPM) has proved to be invaluable for many very challenging problems. However in many ways the method is not as far advanced as such finite element methods in terms of accuracy and stability analysis and computable error estimates for errors in both space and their evolution in time. As it is observed that that the spatial error dominates for cases in which the calculation is stable [12], this suggest that emphasis should be on the spatial error.

This work continues the analysis of MPM methods after previous work on gas dynamics [14], null spaces and linearity preservation [7], stability [2], time integration [3], energy conservation [4] and is an extension of a previous conference paper [5] which introduced the main idea used here without demonstrating its effectiveness in a full MPM simulation. The focus here is on a careful prototype demonstration of error estimation in MPM on a well-studied model problem, so as to provide an incentive for further study for more complex problems in higher space dimensions.

Section 2 describes related work on error estimation in the related to MPM. Section 3 describes the MPM method and its application to a simple example problem. A full description of the method and its errors is provided in Section 4. Simple computational estimates of the different mapping errors in MPM are derived in Section 5. In Section 6 these estimates are applied to a simple model problem solution that shows how well they may work. Finally in Section 7 concluding remarks are made together with comments on the extension of these ideas to multiple space dimensions.

2 BACKGROUND EXISTING MPM ERROR WORK

Many of the key features of MPM methods are covered in the two recent surveys. The first of Vaucorbeil et al. [15] references the work by Wallstedt and Guilkey [16], Tran and Berzins [14] , Steffen et al. [10],[12] and Gritton and Berzins [7]. In the area of accuracy Steffen et al. look carefully at space and time errors and what degree of accuracy is seen on example problems, thus paving the way for many of the developments that followed such as the use of higher order methods in both space and time. Improved time integration methods are considered by Wallstedt and Guilkey [16]. The relationship between MPM time integration and symplectic time integration methods is considered by Berzins [3]. Such symplectic methods have good conservation properties and the well known Stormer-Verlet method has third order accuracy locally [4].

The second survey of Solowski et al [9] also describes the same methods but also points put potentially relevant analysis of particle in cell methods such as the book by Grigoriev et al. [6] and one of the first rigorous attempts to provide a solid theoretical basis for particle methods was due to Raviart [8] who provide a mathematical introduction to the vortex numerical method. All these approaches provide insight into the errors of particle methods but none of them provides computable error estimates. Perhaps the closest approach is the adaptive meshing work of Tan and Nairn [13] who use quadratic expansions to calculate second derivatives as an indicator for mesh refinement parameter as well as to estimate a mass lumping error.

This is an extended version of a conference paper [5] that described the approach that is fleshed out here in a very elementary form. The challenge with error estimation of particle methods is that such error estimates either explicitly or implicitly require spatial derivative information. Such particle information is very challenging to compute using only arbitrarily spaced particles. One major advantage of MPM over some other particle methods is its background mesh that allows for the construction of derivative values on that mesh, based upon the solution values that are mapped to that mesh. These derivatives may be used directly in a Taylor series expansion on that mesh or mapped to particle positions to be used in series expansions about these points.

The central idea here is to demonstrate the potential of this approach using a relatively simple and well-studied problem. Nevertheless the basic ideas apply not only to such simple problems but extend relatively easily to more complex cases in higher dimensions. Finally although this error estimation is done in the context of MPM, many of the same ideas also apply to particle in cell methods as they share the similar mappings of particles to grid and grid to particles.

3 MPM MODEL PROBLEM AND METHOD

The description of MPM used here follows [7,4] in that the model problem used here is a pair of equations connecting velocity v, displacement u and density ρ:

$$\frac{Du}{Dt} = v, \tag{1}$$

$$\rho \frac{Dv}{Dt} = \frac{\partial \sigma}{\partial x} + b(x,t), \tag{2}$$

with a linear stress model $\sigma = E\frac{\partial u}{\partial x}$ for which Young's modulus, E, is constant, a body force b, which is initially assumed to be zero, and with appropriate boundary and initial conditions. For convenience a mesh of equally spaced $N+1$ fixed nodes X_i with intervals $I_i = [X_i, X_{i+1}]$, on on the interval $[a,b]$ is used where

$$a = X_0 < X_1 < ... < X_N = b, \tag{3}$$

$$h = X_i - X_{i-1}. \tag{4}$$

These fixed nodes are referred to as the i points. It will also be assumed that periodic boundary conditions exist in that

$$\sigma(a)v(a) = \sigma(b)v(b) \tag{5}$$

together with appropriate initial conditions. While the analysis of MPM for time integration error and energy conservation uses the model problem above it does apply more generally and in multiple space dimensions with a few obvious modifications. The computed solution at the pth particles will be written as $u_p^n = u(x_p^n, t^n)$. Suppose that there are np particles in total. The calculation of the internal forces in MPM at the nodes requires the calculation of the volume integral of the divergence of the stress [16] using

$$f_i^{int} = -\sum_p D_{pi}(x_p^n)\sigma_p V_p \tag{6}$$

The subscript pi represents a mapping from particles p to node i while the subscript ip represents a mapping from nodes i to particles p. The negative sign arises as a result of using integration by parts [7]. The mass at node i is defined by

$$m_i = \sum_p m_p S_{pi}(x_p^n) \tag{7}$$

It is important to note that the coefficients $D_{pi}(x_p^n)$ and $S_{pi}(x_p^n)$ (which here will be abbreviated to D_{pi}^n and S_{pi}^n, depend explicitly on the background mesh and the particle positions and that they also be chosen to reproduce derivatives of constant and linear functions exactly [7]. The initial volume of the particles is uniform for the n_p particles in an interval. The particle volumes are defined using the deformation gradient, F_p^n, and the initial particle volume,V_p^0,

$$V_p^n = F_p^n V_p^0, \text{ where } V_p^0, = \frac{h}{n_p}, \textit{ where } F_p^0 = 1 \tag{8}$$

From (7) the form of acceleration equation in the MPM method in this case is

$$a_i(t) = \frac{-1}{m_i}\sum_p D_{pi}(x_p(t))\sigma_p(t)F_p(t)V_p^0 \tag{9}$$

The equation to update velocity at the nodes, as denoted by v_i^n is then given by

$$\dot{v}_i = a_i \tag{10}$$

The equation for the update of the particle velocities is then

$$\dot{v}_p = a_p \tag{11}$$

where the value of the acceleration at a point x_p^n is given by interpolation based upon nodal values of acceleration

$$a_p = \sum_i S_{ip}(x_p(t))a_i \tag{12}$$

The equation for the particle position updates is

$$\dot{x}_p = v_p \tag{13}$$

The update of the deformation gradients is given using

$$\frac{\partial v}{\partial x}(x_p(t)) = \sum_i D_{ip}(x_p(t))v_i \tag{14}$$

The deformation update equation is

$$\dot{F}_p = \frac{\partial v}{\partial x}(x_p(t),t)F_p \tag{15}$$

While the stress update equation is, using the appropriate constitutive model and Young's Modulus, E,

$$\dot{\sigma}_p = E\frac{\partial v}{\partial x}(x_p(t)) \tag{16}$$

4 Stress Last MPM and Global Errors in Space and Time

In solving the system of equations defined above by equations (6) to (16) one standard approach used is to order the equations in a certain order and then to solve them in turn using explicit methods. Differences in how the equations are solved corresponds to whether or not the stress is updated first or last in a time step, a choice that is discussed at length by [1]. These two different choices are related to the use of the semi-implicit Euler A or B method [2]. Following Bardenhagen [1] it is preferable to increment stress last. In this case it is assumed that at time t^n a consistent set of particle positions x_p^n, velocities v_p, stresses σ_p^n and deformation gradients F_p^n are available. The description here of the method and the sources of error follows that in [3]. It is also assumed that the GIMP method is used for spatial discretization.

In contrast to the local errors described in [5] the analysis here is concerned with the evolution of the full global error.

The nodal velocity v_i is calculated using

$$v_i^n = \sum_p S_{pi}^n \frac{m_p}{m_i} v_p^n \tag{17}$$

with an associated error $Ev_i^n = v_{i,true}^n - v_i^n$ which is defined in terms of the existing particle errors Ev_p^n and the interpolation error Evi_{pi}^n by

$$Ev_i^n = \sum_p S_{pi}^n \frac{m_p}{m_i} Ev_p^n + EIv_{pi}^n \tag{18}$$

where EIv_{pi}^n is the interpolation or mapping error associated with the coefficients S_{pi}^n which will be defined as follows:

$$EIv_{pi}^n = \tilde{v}_i^n - \sum_p S_{pi}^n \frac{m_p}{m_i} v_p^n \tag{19}$$

where $\tilde{v}_i^n$ is the nodal value consistent with the values v_p^n. This error is estimated in Sect 5. below, equations (44) to (46).

The nodal acceleration is updated by using the stresses and deformation gradients at the current grid points and the body forces

$$a_i^n = \frac{-1}{m_i} \sum_p DS_{pi}^n \sigma_p^n + b(X_i, t^n) \tag{20}$$

where the nodal mass is is defined by equation (7). The equation to update velocity at the nodes is then given by

$$v_i^{n+1} = v_i^n + dt a_i^n \tag{21}$$

The global error in this forward Euler step at time t^{n+1} is given by Ev_i^{n+1} and whose evolution may be approximated by

$$Ev_i^{n+1} = Ev_i^n + \frac{dt^2}{2} \frac{d^2 v_i^n}{dt^2} + dt Ea_i^n \tag{22}$$

where Ea_i^n is the spatial error from using the approximation in equation (20), as defined by

$$Ea_i^n = a_{i,true}^n - a_i^n \tag{23}$$

and which will be estimated as defined in Sect. 5 below. The time derivative term is a simple approximation to the local time error which may be estimated by

$$\frac{dt^2}{2}\frac{d^2v_i^n}{dt^2} \approx \frac{dt}{2}(a_i^n - a_i^{n-1}) \tag{24}$$

The equation for the update of the particle velocity is then

$$v_p^{n+1} = v_p^n + dt\sum_i S_{ip}^n a_i^n \tag{25}$$

The associated global error is defined by

$$Ev_p^{n+1} = v_{p,true}^{n+1} - v_p^{n+1} \tag{26}$$

whose evolution may be approximated by

$$Ev_p^{n+1} = Ev_p^n + \frac{dt^2}{2}\frac{d^2v_p^n}{dt^2} + dt\sum_i S_{ip}^n Ea_i^n + dtEIa_{ip}^n \tag{27}$$

In this case the error EIa_{ip}^n is the approximation error caused by the mapping coefficients S_{ip}^n and is estimated as described in Sect. 5 below using the same approach as in equations (54) and (55). Furthermore the time derivative term is the local time error which may be estimated by

$$\frac{dt^2}{2}\frac{d^2v_p^{n+1}}{dt^2} \approx \frac{dt}{2}(a_p^n - a_p^{n-1}) \tag{28}$$

In the computational experiments the following estimate was used after considerable numerical testing as equation (27) gave computed error estimates that appeared to grow too quickly.

$$Ev_p^{n+1} = \sum_i S_{ip}^n Ev_i^{n+1} + dtEIa_{ip}^n \tag{29}$$

From equations (45) (22) this may be written as

$$Ev_p^{n+1} = \sum_i S_{ip}^n \sum_p S_{pi}^n Ev_p^n + \sum_i S_{ip}^n \left(\frac{dt^2}{2}\frac{d^2v_i^n}{dt^2} + dtEa_i^n + EIv_{pi}^n\right) + dtEIa_{ip}^n \tag{30}$$

The velocity gradients at particles are calculated using the formula

$$\frac{\partial v^{n+1}}{\partial x}(x_p) = \sum_i D_{ip}^n v_i^{n+1} \tag{31}$$

with an associated derivative approximation error as denoted by EIv_{xp}^{i+1}, defined by

$$Ev_{xp}^{i+1} = \frac{\partial v_{true}^{n+1}}{\partial x}(x_p) - \sum_i D_{ip}^n v_i^{n+1} \tag{32}$$

These velocity gradients are used to update the stress and deformation gradients at particles

$$F_p^{n+1} = F_p^n + dt \frac{\partial v^{n+1}}{\partial x}(x_p^n, t_n) F_p^n dt \tag{33}$$

The associated global error is defined by

$$EF_p^{n+1} = F_{p,true}^{n+1} - F_p^{n+1} \tag{34}$$

For the deformation gradient F_p^n the error evolution may be approximated by

$$EF_p^{n+1} = EF_p^n - \frac{dt^2}{2}\frac{d^2F_p}{dt^2} + dtF_p^n \left(EIv_{xp}^n + \sum_i D_{ip}^n E v_i^{n+1} \right) \tag{35}$$

The second time derivative term corresponds to the local error from a semi-implicit Euler step with updated particle velocity derivatives at t^{n+1}. The semi-implicit method local time error has the opposite sign to the explicit. Stress is updated using the appropriate constitutive model and Young's Modulus, E,

$$\sigma_p^{n+1} = \sigma_p^n + dtE \frac{\partial v^{n+1}}{\partial x}(x_p^n) \tag{36}$$

The associated global error is defined by

$$E\sigma_p^{n+1} = \sigma_{p,true}^{n+1} - \sigma_p^{n+1} \tag{37}$$

In this case the stress global time and space error approximately evolves according to

$$E\sigma_p^{n+1} = E\sigma_p^{n+1} - \frac{dt^2}{2}\frac{d^2\sigma_p^{n+1}}{dt^2} + dtE \left(EIv_{xp}^n + \sum_i D_{ip}^n E v_i^{n+1} \right) \tag{38}$$

in a similar way as for the deformation gradient. In the computational experiments the term $\sum_i D_{ip}^n E v_i^{n+1}$ was not used as it caused the stress error to grow too quickly, in part due to the presence of the constant E. The time derivative term is the local time error which may be estimated by .

$$\frac{dt^2}{2}\frac{d^2\sigma_p^{n+1}}{dt^2} \approx \frac{dtE}{2}\left(\frac{\partial v^{n+1}}{\partial x}(x_p^n) - \frac{\partial v^n}{\partial x}(x_p^{n-1}) \right) \tag{39}$$

However this term is not particularly significant and so was not included in the experiments. The equation for the particle position update is

$$x_p^{n+1} = x_p^n + dtv_p^{n+1} \tag{40}$$

The associated global error is defined by

$$Ex_p^{n+1} = x_{p,true}^{n+1} - x_p^{n+1} \tag{41}$$

For the particle update the error is given by

$$Ex_p^{n+1} = Ex_p^n - \frac{dt^2}{2}\frac{d^2x_p^{n+1}}{dt^2} + dt(Ev_p^n + EIv_p^n) \tag{42}$$

When implementing this it was found that the key error source term was EIv_p^n which is estimated as in equations (54) and (55), and the term Ev_p^n was dropped, as its inclusion resulted in this error being over estimated. The time derivative term is the local time error which may be calculated by

$$\frac{dt^2}{2}\frac{d^2x_p^p}{dt^2} \approx \frac{dt^2}{2}a_p^n \tag{43}$$

Of course, it should be noted that in this analysis the error in evolving the error equations is neglected. As an example if the second time derivatives of the errors in a particle quantity are greater than the second time derivatives of the quantity itself then these errors must also somehow be estimated.

5 ESTIMATING THE SPATIAL ERROR

The above derivations illustrate how the spatial and temporal errors associated with MPM combine to give the overall error. Steffen et al. [12] observed these errors experimentally and arrived at the conclusion that for a stable time step with the methods they considered that temporal errors are dominated by spatial errors. This suggests that the starting point is to focus on the spatial errors in the MPM. Furthermore as it is straightforward to estimate the time local errors in the way already shown above it is now necessary to estimate the spatial errors.

The error framework presented in the previous section makes it possible to derive estimates for the individual parts of MPM associated with the mapping matrix S_{ip} and the differentiation matrix D_{ip} (and their transposes) and to thus provide computable estimates for the error. There are two parts to this process. The first part is is that the error in mapping from particles to the grid nodes. The second part is estimating the error in mapping from the grid nodes back to particles. These are now considered in turn

5.1 Particles to Nodes

In the case of the mapping defined by equation (17),(ignoring the contributions of the masses for the moment)

$$v_p^n = v_i^n + (x_p - X_i)\frac{\partial v}{\partial x}(X_i,t^n) + \frac{(x_p-X_i)^2}{2}\frac{\partial^2 v}{\partial x^2}(X_i,t^n) + \frac{(x_p-X_i)^3}{6}\frac{\partial^3 v}{\partial x^3}(X_i,t^n) + ... \tag{44}$$

Using the approach of [7] it is assumed that the mapping S_{ip} is linearity preserving i.e. $\sum_p S_{pi} = 1$ and $\sum_p S_{pi}x_p = X_i$. Approximating the true value of the error at the nodal velocity

$$EIv_{pi}^n = v_{i,true}^n - \sum_p S_{pi}^n v_p^n \tag{45}$$

by a local solution based upon computed solution values and multiplying (44) by S_{ip} and using linearity preservation gives

$$EIv_{pi}^n \approx -\frac{\partial^2 v}{\partial x^2}(X_i,t^n)\sum_p S_{pi}^n \frac{(x_p-X_i)^2}{2} - \frac{\partial^3 v}{\partial x^3}(X_i,t^n)\sum_p S_{pi}^n \frac{(x_p-X_i)^3}{6} \tag{46}$$

which requires the estimation of second and third derivatives at the nodes. Those derivatives are estimated with the present solution, rather than the true solution.

In the case of acceleration matters are more complicated. Consider the mapping defined by equation (20). The first step is to expand the stress at a particle about the node by using a simple Taylor expansion.

$$\sigma_p^n = \sigma_i^n + (x_p - X_i)\frac{\partial \sigma}{\partial x}(X_i,t^n) + \frac{(x_p-X_i)^2}{2}\frac{\partial^2 \sigma}{\partial x^2}(X_i,t^n) + \frac{(x_p-X_i)^3}{6}\frac{\partial^3 \sigma}{\partial x^3}(X_i,t^n) + ... \tag{47}$$

Substituting this in equation (20) gives after assuming that the coefficients D_{pi} exactly differentiate linear functions ($\sum_p DS_{pi} = 0$ and $\sum_p DS_{pi}x_p = 1$), see [7] who also provide a procedure for this. Define the stress derivative error as

$$E\sigma_{xi}^n = \frac{\partial \sigma}{\partial x}(X_i,t^n) + \sum_p D_{pi}^n \sigma_p^n \tag{48}$$

then using a Taylor series gives

$$E\sigma_{xi}^n = \sum_p D_{pi}^n \left[\frac{(x_p-X_i)^2}{2}\frac{\partial^2 \sigma}{\partial x^2}(X_i,t^n) + \frac{(x_p-X_i)^3}{6}\frac{\partial^3 \sigma}{\partial x^3}(X_i,t^n) + ... \right] \tag{49}$$

The stress derivatives at nodes are difficult to estimate and so nodal acceleration derivatives are used instead. This is complicated by the body forces contribution to acceleration so that if

$$\lambda_i^n = \frac{b(x_i,t^n)}{a_i^n} \tag{50}$$

then

$$\frac{\partial \sigma}{\partial x}(X_i,t^n) = (1-\lambda_i^n)a_i^n \tag{51}$$

The same approximation may be used for higher derivatives so that

$$E\sigma_{xi}^n = \frac{-(1-\lambda_i^n)}{\tilde{m}_i}\sum_p DS_{pi}^n \left[\frac{(x_p-X_i)^2}{2}\frac{\partial a}{\partial x}(X_i,t^n) + \frac{(x_p-X_i)^3}{6}\frac{\partial^2 a}{\partial x^2}(X_i,t^n) + ... \right] \tag{52}$$

as $Ea_i = E\sigma_{xi}/(1-\lambda_i^n)$ then

$$Ea_i^n = \frac{-1}{\tilde{m}_i}\sum_p DS_{pi}^n \left[\frac{(x_p-X_i)^2}{2}\frac{\partial a}{\partial x}(X_i,t^n) + \frac{(x_p-X_i)^3}{6}\frac{\partial^2 a}{\partial x^2}(X_i,t^n) + ... \right] \tag{53}$$

The estimation of these spatial acceleration derivatives at the nodes is described below in Sect 5.3.

5.2 Nodes to Particles

In this case we have to consider the mapping from nodal values of accelerations to accelerations at particles, given by equation (25) and the mapping from nodal velocities to velocity derivatives at particles. It is assumed that the transposes of the mapping matrices S and DS (as denoted by switching the subscript pi to ip) satisfy the same equations as above for preserving linearity in the mapping and for differentiating linear functions exactly, e.g. using the procedure of Gritton [7]. In both these cases we expand the nodal values about particles. Consider the mapping equation as used for velocity

$$EIv_p^n = v_{p,true}^n - \sum_i S_{ip}^n v_i^n \tag{54}$$

where the true particle velocity $v_{p,true}^n$ is again approximated using a Taylor series as in equation (44). Using the fact that that the sum $\sum_i S_{ip}^n = 1$ then

$$EIv_p^n \approx \\ -\sum_i S_{ip}^n \left(\frac{(x_p - x_I)^2}{2} \frac{\partial^2 v}{\partial x}(X_i, t^n) + \frac{(x_p - X_I)^3}{6} \frac{\partial^3 v}{\partial x^3}(X_i, t^n) \right) \tag{55}$$

Where higher order than three derivatives are neglected in the Taylor expansion. A similar approach may be used for the approximation error to do with mapping the acceleration from nodes to particles EIa_{ip}^n

In the case of estimating the error in derivatives at particles the approach is similar to estimate the mapping error

$$EIv_{xp}^{n+1} = \frac{\partial v_{true}^{n+1}}{\partial x}(x_p) - \sum_i DS_{ip}^n v_i^{n+1}, \tag{56}$$

Expanding about X_i gives

$$EIv_{xp}^{n+1} \approx -\sum_i DS_{ip}^n \left(\frac{(X_i - x_p)^2}{2} \frac{\partial^2 v}{\partial x^2}(x_p, t^{n+1}) + \frac{(X_i - x_p)^3}{6} \frac{\partial^3 v}{\partial x^3}(x_p, t^{n+1}) \right) \tag{57}$$

This expression requires velocity derivatives at the particles. These values may be approximated by interpolating from the nodal derivatives so that, for instance,

$$\frac{\partial^2 v}{\partial x^2}(x_p, t^{n+1}) \approx \sum_j S_{ip}^n \frac{\partial^2 v}{\partial x^2}(X_j, t^{n+1}) \tag{58}$$

to get

$$EIv_{xp}^{n+1} \approx -\sum_i DS_{ip}^n \left(\frac{(X_i - x_p)^2}{2} \sum_j S_{jp} \frac{\partial^2 v}{\partial x^2}(X_j, t^{n+1}) + \frac{(X_i - x_p)^3}{6} \sum_j S_{jp} \frac{\partial^3 v}{\partial x^3}(X_j, t^{n+1}) \right) \tag{59}$$

Alternatively the expansions

$$\frac{\partial^2 v}{\partial x^2}(x_p, t^{n+1}) = \frac{\partial^2 v}{\partial x^2}(X_i, t^{n+1}) + (x_p - X_i) \frac{\partial^3 v}{\partial x^3}(X_i, t^{n+1}) \tag{60}$$

and

$$\frac{\partial^3 v}{\partial x^3}(x_p,t^{n+1}) = \frac{\partial^3 v}{\partial x^2}(X_i,t^{n+1}) + (x_p - X_i)\frac{\partial^4 v}{\partial x^4}(X_i,t^{n+1} \tag{61}$$

May be truncated and used in equation (57) to get

$$Ev_{xp}^n \approx -\sum_i DS_{ip}^n \left(\frac{(X_i - x_p)^2}{2}\frac{\partial^2 v}{\partial x^2}(X_i,t^{n+1}) + \frac{2(X_i - x_p)^3}{3}\frac{\partial^3 v}{\partial x^3}(X_i,t^n) + \frac{(X_i - x_p)^4}{6}\frac{\partial^4 v}{\partial x^4}(X_i,t^{n+1}) \right) \tag{62}$$

This involves less computation and in the experiments the third and fourth-order terms were not included as in the computational experiments it was sufficient to just use the approximation

$$Ev_{xp}^{n+1} \approx -\sum_i DS_{ip}^n \left(\frac{(X_i - x_p)^2}{2}\frac{\partial^2 v}{\partial x^2}(X_i,t^{n+1}) \right) \tag{63}$$

5.3 Estimating the Spatial Derivatives

The first spatial derivative of the stress or any other quantity at the nodes is straightforwardly estimated using finite differences of nodal stress values.

$$\frac{\partial \sigma}{\partial x}(X_i,t) \approx \frac{\sigma_{i+1} - \sigma_{i-1}}{2h} \tag{64}$$

and the second derivative of the stress is straightforwardly estimated using finite differences of nodal stress values.

$$\frac{\partial^2 \sigma}{\partial x^2}(X_i,t) \approx \frac{\sigma_{i+1} - 2\sigma_i + \sigma_{i-1}}{h^2} \tag{65}$$

and the third derivative similarly as

$$\frac{\partial^3 \sigma}{\partial x^3}(X_i,t) \approx \frac{\sigma_{i+2} - 2\sigma_{i+1} + 2\sigma_{i-1} + \sigma_{i-2}}{h^3} \tag{66}$$

With appropriate modifications at the boundaries.

6 COMPUTATIONAL EXPERIMENTS

While it is straightforward to write down the equations for evolution of the errors there are many subtleties and challenges in implementing such an approach. The first major challenge comes from the coupled nature of all the errors in the system of error equations defined by MPM. In particular this means that over or under estimation of a particular error propagates through the whole system. Thus with over estimation

the error equations at increase at an unrealistic rate. After considerable experimentation the stated algorithm given above appears to a good compromise with regard to estimating the errors in all the components.

In order to test the estimates derived above the vibrating bar example that is often a standard MPM benchmark problem is used, e.g. [7]. The problem considered is a 1D bar problem, following similar examples in [11,7,2].

$$\sigma = P = E\frac{\partial u}{\partial X} = E(F-1), \tag{67}$$

where E is the Young's modulus. The rate of change of stress is then computed as,

$$\dot{\sigma} = E(\dot{F}), \tag{68}$$
$$= E(lF), \tag{69}$$

where l is the velocity gradient in the spatial description. The analytic solutions for displacement and velocity defined in the material description are:

$$u(X,t) = Asin(2\pi X)sin(c\pi t), \tag{70}$$
$$v(X,t) = = Ac\pi sin(2\pi X)cos(c\pi t), \tag{71}$$

where $c = \sqrt{E}$ for a uniform density and A is the maximum displacement. The constitutive model is defined in Equation 67 and the body force is,

$$b(X,t) = 3A(c\pi)^2 u(X,t). \tag{72}$$

The initial spatial discretization is on the spatial domain of $[0,1]$. The periodic nature of the analytic solution means that both periodic boundary conditions and zero Dirichlet boundary conditions are both appropriate. The initial conditions for the updated Lagrangian description of the particles are:

$$F = 1, \tag{73}$$
$$x_p = X_p^0, \tag{74}$$
$$V_p = V_p^0. \tag{75}$$

For this problem two cases are considered. Case 1 uses a cell width is $h = 10^{-2}$ and two evenly spaced particles per cell. Case 2 uses a cell width is $h = 0.510^{-2}$ and four evenly spaced particles per cell. The material density is $\rho_0 = 1$, and Young's modulus is varied from $E = 4$ to $E = 1000$, maximum displacement is $A = 0.1$ and the timestep is varied. It should be noted that with the use of the above parameters there are many particle crossings, ranging from 1000 with E=4 to 8000 with E= 1000. Suppose that the error in the stress at a spatial point x is given by $e\sigma(x)$, then the following definitions are needed.

$$Ex_p^n = \left[Ex_1^n, ..., Exp_{np}^n\right]^T \tag{76}$$

$$Ev_N^n = [Ev_1^n, ..., Ev_N^n]^T \tag{77}$$

$$Ev_p^n = \left[Evp_1^n, ..., Evp_{np}^n\right]^T \tag{78}$$

Table 1 Case 1 Average Error Indices for Particles Displacement, Velocity, Acceleration and Stress (*=failure)

E	dt	Err X IAv	Err V IAv	Err A IAv	Err σ IAv
1.0e+3	1.0e-3	0.621	0.886	0.882	0.75
	1.0e-4	0.604	1.380	1.151	1.89
	1.0e-5	0.403	1.280	1.302	1.63
64	1.0e-3	0.844	1.53	1.01	1.45
	1.0e-4	0.411	2.01	1.27	1.22
	1.0e-5	0.392	2.05	1.38	1.19
4	1.0e-3	0.470	2.09	1.139	1.20
	1.0e-4	0.418	2.32	1.266	1.18
	1.0e-5	0.416	2.42	1.283	1.18

$$EA_N^n = [EA_1^n, ..., EA_N^n]^T \tag{79}$$

$$E\sigma_p = \left[E\sigma_1^n, ..., E\sigma_{np}^n\right]^T \tag{80}$$

Suppose that the estimated error is similarly denoted by $e_{st}\sigma_p$. The L_2 vector norm divided by the number of sample points is used. The error index of the estimated error norm is given by

$$ErrXIAv = \frac{\sum_{k=1}^{nsteps} ||Ex_p^k||_2}{\sum_{k=1}^{nsteps} ||E_{true}x_P^k||_2} \tag{81}$$

$$ErrVIAv = \frac{\sum_{k=1}^{nsteps} ||Ev_p^k||_2}{\sum_{k=1}^{nsteps} ||E_{true}v_P^k||_2} \tag{82}$$

$$ErrAIAv = \frac{\sum_{k=1}^{nsteps} ||EA_N^k||_2}{\sum_{k=1}^{nsteps} ||E_{true}A_N^k||_2} \tag{83}$$

$$Err\sigma IAv = \frac{\sum_{k=1}^{nsteps} ||E\sigma_p^k||_2}{\sum_{k=1}^{nsteps} ||E_{true}\sigma_P^k||_2} \tag{84}$$

Where the index k refers to the time at which particular error quantity that is being estimated. These results show that the error estimators we have developed initially appear to do a good job of estimating the errors as is shown by the error indices. While the error indices described above show the average behavior of the error estimates it is good to also have more detailed information on the time evolution of the errors. Three cases are considered, of E=4 E = 64 and E =1000. Figures 1 to 3 show the evolution of actual and estimated errors in particle displacement and velocity and nodal acceleration for Case 1. Figures 4 to 6 show the evolution of actual and estimated errors in particle displacement and velocity and nodal acceleration for Case 2.

These plots illustrate the challenges of estimating the errors in the complex system of MPM equations and also show that the error estimation approach does a good job overall of estimating the error.

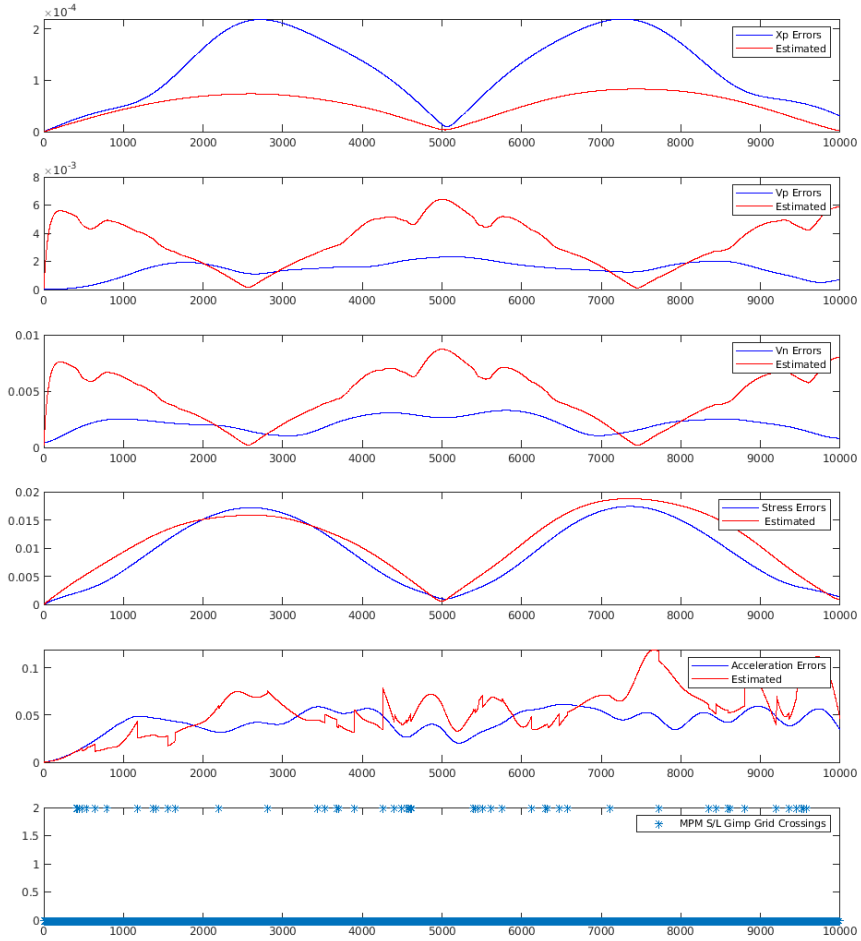

Fig. 1 1d Bar, Case 1:Plots of Errors for E=4

7 CONCLUSIONS AND FUTURE WORK

The result of the approach presented here is that a decomposition of the different errors in MPM has been used to derive estimates for the mappings inherent in MPM between particles and nodes and vice versa. These simple estimates have been shown to work well in the simple demonstration case used here. There are differences in that reliability of the error estimates for the different solution components and one topic for future study is why the error estimate may grow much more quickly than

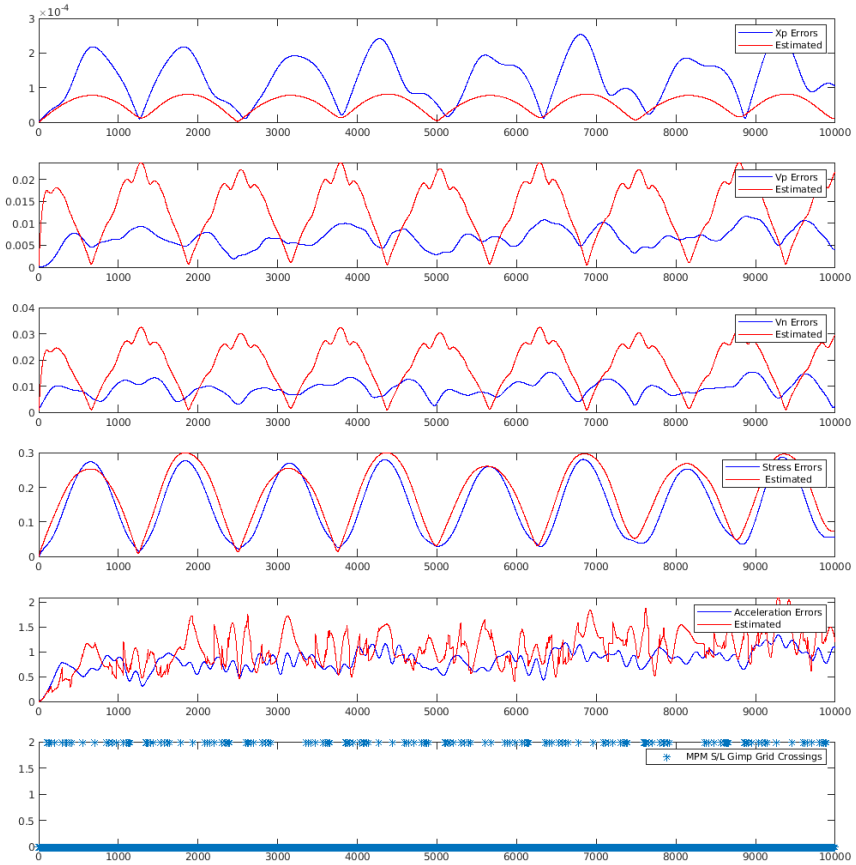

Fig. 2 1d Bar,Case 1 Plots of Errors for E=64

Table 2 Case 2 Average Error Indices for Particles Displacement, Velocity, Acceleration and Stress (*=failure)

E	dt	Err X IAv	Err V IAv	Err A IAv	Err σ IAv
1.0e+3	*	*	*	*	*
	1.0e-4	1.16	1.17	1.363	2.89
	1.0e-5	0.52	1.85	1.404	2.55
64	1.0e-3	2.52	1.78	1.43	2.83
	1.0e-4	0.59	2.83	1.73	2.52
	1.0e-5	0.50	3.32	1.55	2.49
4	1.0e-3	0.80	1.78	1.44	2.83
	1.0e-4	0.52	2.83	1.37	2.52
	1.0e-5	0.51	3.32	1.55	2.49

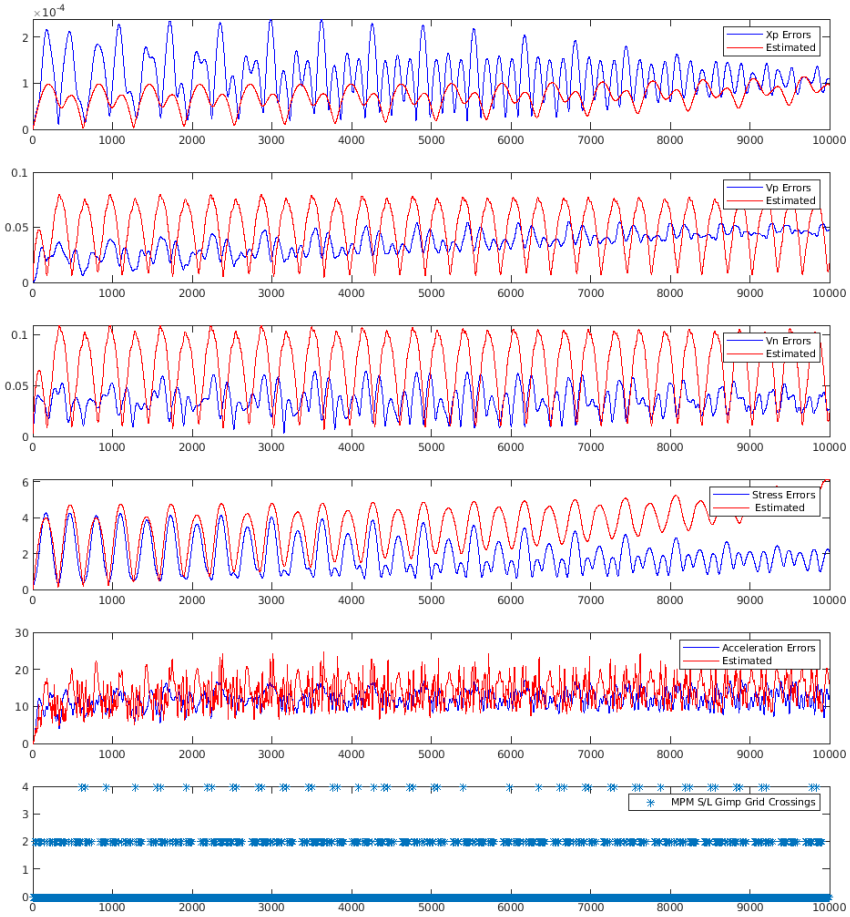

Fig. 3 1d Bar, Case 1 Plots of Errors for E=1000

the actual error. Future work involves applying this approach in a full MPM simulation and for two and three space dimensions. As the estimates derived here involve approximating derivatives of solution values on a regular mesh, this would seem to be entirely possible and is ongoing work using the method of manufactured solutions problems in [16].

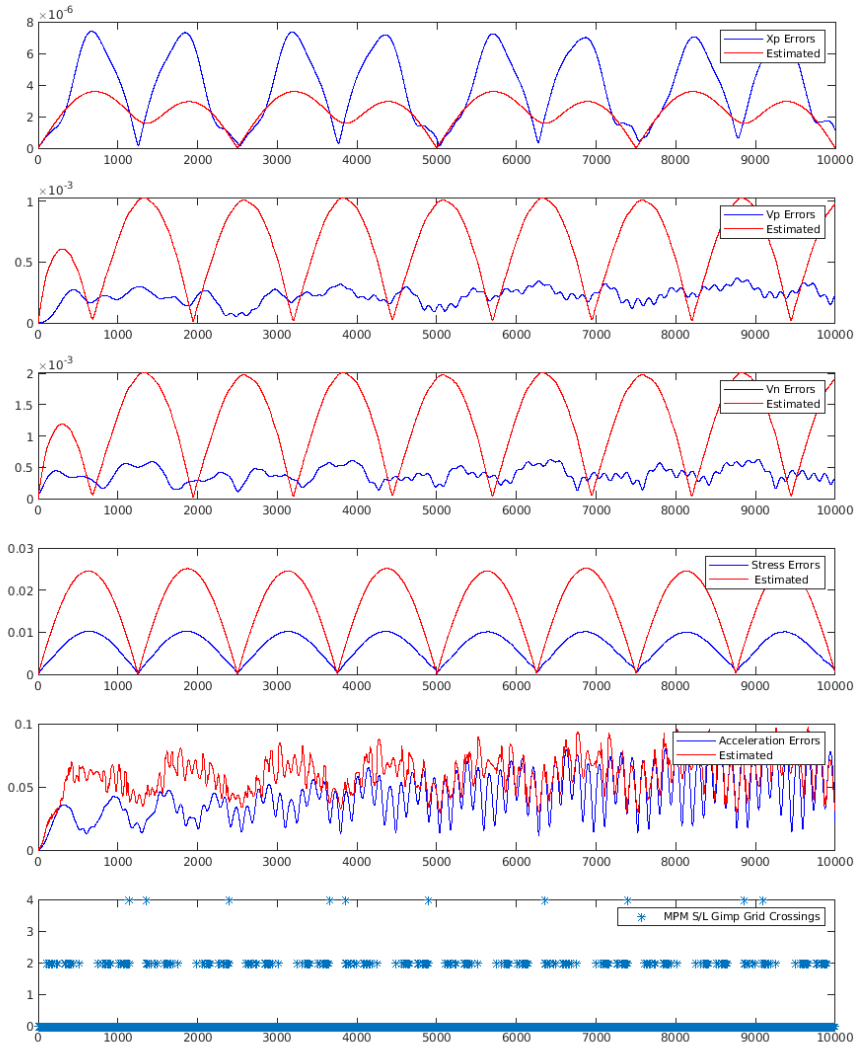

Fig. 4 1d Bar, Case 2:Plots of Errors for E=4

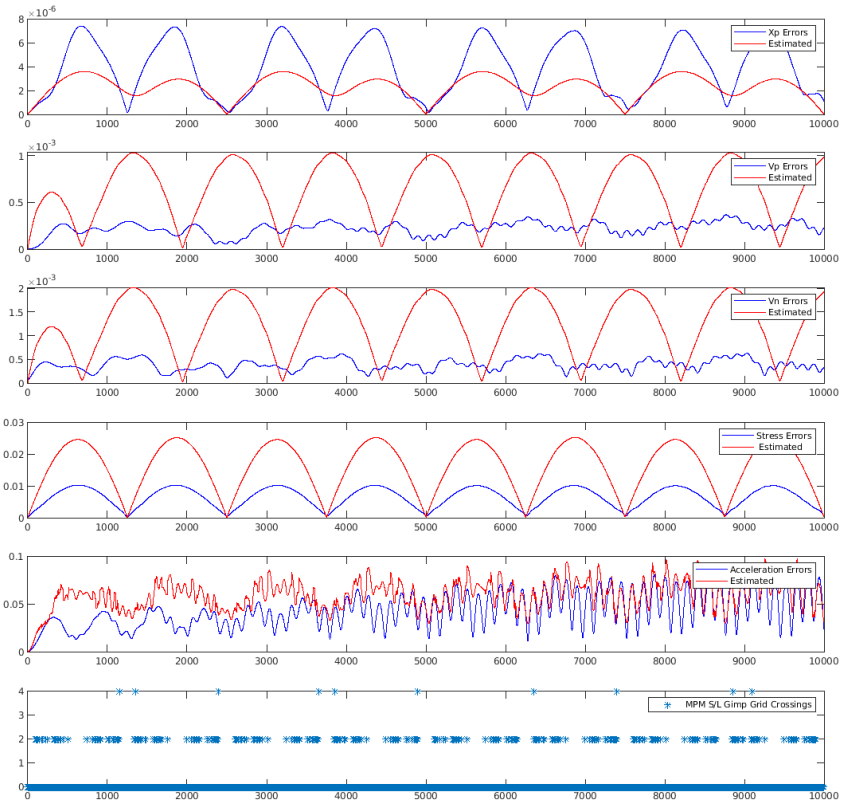

Fig. 5 1d Bar,Case 2 Plots of Errors for E=64

ACKNOWLEDGEMENTS

This research was partially sponsored by the Army Research Laboratory under Cooperative Agreement Number W911NF-12-2-0023. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government.

References

1. S. Bardenhagen, Energy conservation error in the material point method for solid mechanics, *Journal of Computational Physics*, 180, 2002, 383-403.
2. Berzins M. Nonlinear Stability and time step selection in the material point method. *Computational Particle Mechanics* January 2018.

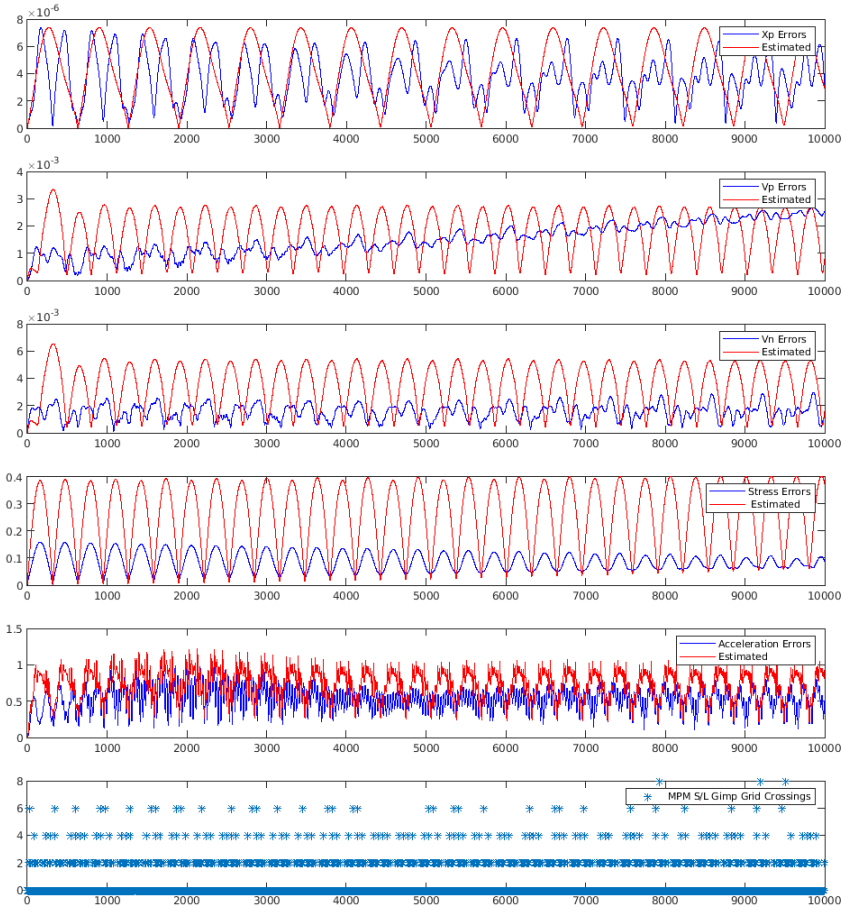

Fig. 6 1d Bar, Case 2 Plots of Errors for E=1000

3. Berzins M. Symplectic Time Integration Methods for the Material Point Method, Experiments, Analysis and Order Reduction, *WCCM-ECCOMAS2020 virtual Conference* proceedings January, 2021.
4. Berzins M. Energy Conservation and Accuracy of Some MPM Methods. *Computational Particle Mechanices* Published February 04 2022.
5. Berzins M. Time Stepping with Space and Time Errors and Stability of the Material Point Method. *VII International Conference on Particle-Based Methods PARTICLES 2021* (Eds) P. Wriggers, M. Bischoff, E. O nate, M. Bischoff, A. Duster and T. Zohdi proceedings (to appear) 2021.
6. Grigoryev, Y. N., Vshivkov, V. A., and Fedoruk, M. P. "Numerical Particle-in-cell methods: Theory and applications." **2012** De Gruyter https://doi.org/10.1515/9783110916706
7. Gritton C. and Berzins M. Improving accuracy in the MPM method using a null space filter, *Computational Particle Mechanics*, 4, pp. 131142 2017.

8. Raviart, P. A. (1985). "An analysis of particle methods". In F. Brezzi (Ed.), i *Numerical Methods in Fluid Dynamics. Lecture Notes in Mathematics, vol 1127*. 1985 243324. Springer Berlin Heidelberg. https://doi.org/10.1007/BFb0074532
9. Wojciech T. Solowski , Martin Berzins, William M. Coombs, James E. Guilkey, Matthias Moller, Quoc Anh Tran, Tito Adibaskoro, Seyedmohammadjavad Seyedan, Roel Tielen, and Kenichi Soga Material point method: Overview and challenges ahead Chapter 2 of *Advances in Applied* Vol. 54. Eds. Stephane P.A. Bordas and Daniel S, Balint. November 23 2021, ISBN: 9780323885195.
10. Steffen M., Kirby R.M., Berzins M. Analysis and Reduction of Quadrature Errors in the Material Point Method (MPM), *Int. J. for Numer. Meths. in Engnng*,**76**, 6, 922–948, 2008.
11. Steffen M., Wallstedt P.C., Guilkey J.E. , Kirby R.M. and Berzins M., Examination and analysis of implementation choices within the Material Point Method (MPM), *Computer Modeling in Engineering & Sciences*, 2008, **31**, 2, 107-127.
12. Steffen M., Kirby R.M., Berzins M. Decoupling and Balancing of Space and Time Errors in the Material Point Method (MPM), *Int. J. for Numer. Meths. in Engnng*, **82**, No. 10, pp. 1207–1243. 2010.
13. Honglai Tan, John A. Nairn "Hierarchical, adaptive, material point method for dynamic energy release rate calculations" *Computer Methods in Applied Mechanics and Engineering* **191**, 2123-2137.
14. Tran, L. T., Kim, J., and Berzins, M. (2010). "Solving time-dependent PDEs using the material point method, a case study from gas dynamics." *International Journal for Numerical Methods in Fluids*. 23 March 2009 https://doi.org/10.1002/fld.2031
15. Alban de Vaucorbeil, Vinh Phu Nguyen, Sina Sinaie, Jian Ying Wu, Chapter Two - Material point method after 25 years: Theory, implementation, and applications, Editor(s): Stphane P.A. Bordas, Daniel S. Balint, *Advances in Applied Mechanics,* Elsevier, 2020 **53**, 2020, 185-398.
16. Wallstedt P.C. and Guilkey J.E., An evaluation of explicit time integration schemes for use with the generalized interpolation material point method. *Journal of Computational Physics* 2008, **227**, 22, 9628-9642.