Computational Particle Mechanics manuscript No.
(will be inserted by the editor)

Suitability of the local-maximum entropy approximation under the Material Point Method framework for dynamic problems.

Miguel Molinos · **Pedro Navas** · **Manuel Pastor** · **Miguel Martn Stickle**

Received: date / Accepted: date

Abstract This document is devoted to describe the suitability of the Local *maximum-entropy* (LME) meshfree approximation technique under the framework of the Material Point Method for dynamic problems.

Keywords LME · MPM · Dynamic problems

1 Introduction

Since the proposal of the Material Point Method (**MPM**) by Sulsky *et al.* (1994)[16], as a generalization of a particle-in-cell method [10] and a fluid implicit particle (FLIP) method [5], MPM has been used to simulate dynamic problems. However, in the simulations made with the originalMPM, there are numerical noises [17] when it is faced to solve classic dynamic benchmarks like the one proposed by Dyka & Ingel (1995)[8].

The aim of this paper is to mitigate this spurious oscillations by the employ of the maximum-entropy (or local *max-ent*) shape function under the MPM framework. First introduced by [2], it belongs to the class of convex approximation schemes and provides a seamless transition between finite elements (**FE**) and mesh-free interpolations. The approximation scheme is based on a compromise between minimizing the width of the shape function support and maximizing the information entropy of the approximation. The local *max-ent* approximation may be regarded as a regularization, or *thermalization*, of Delaunay triangulation which effectively resolves the degenerate cases resulting from the lack of uniqueness of the triangulation. Local *max-ent* basis functions possess many desirable properties for meshfree algorithms. First of all, they are entirely defined by the nodal set and the domain of analysis. They are also non-negative, satisfy the partition of unity property, and provide an exact approximation for affine functions [2].

This approximation scheme has been proof to have a good performance under the dynamic regime by other researchers like Navas *et al.* (2018)[15] and Li *et al.* (2012)[14] for Optimal Transportation Meshfree (**OTM**) method.

The article is organized as follows. Section 2 briefly reviews the notation here employed. Next, Section 3 is devoted to present briefly the governing equations of the elastic problem, the variational formulation and the Galerkin procedure. In Section 4 an explicit predictor-corrector time integration scheme for the MPM is proposed. Section 5 briefly reviews the local *max-ent* basis functions here employed. In Section 6 a comparison between both methods MPM and OTM employing *max-ent* shape function is performed as well a parametric study for the γ parameter. Finally, conclusions and future research topics are exposed in Section 7.

2 Notation

In what follows, we will adopt the following convention:

Table 1: Physical variables involved in the problem

ρ	Density field	Scalar
a	Acceleration field	First order tensor
v	Velocity field	First order tensor
u	Displacement field	Second order tensor
σ	Cauchy stress tensor	Second order tensor
ε	Cauchy strain tensor	Second order tensor
D	Constitutive tensor	Fourth order tensor

Funding: This study was funded by Agustn de Betancourt Foundation (grant number 262390106114).

Address(es) of author(s) should be given

The subscript p is used to define a particle variable. While the subscript I is reserved in this notation for denoting nodal variables.

The superscript ψ involves a virtual magnitude.

$\dot{\square}$ where $\nabla \cdot \square$ denotes the divergence operator.

where $\nabla^s \square$ denotes the symmetric part of the gradient operation.

Γ_d Γ_n

3 Derivation of the MPM

In this section, we provide an overview of the standard explicit MPM algorithm as presented by [16]. The method has three main steps: a variational recovery process to project particle data to the nodes of the grid, a Eulerian step where balance of momentum is compute by solving internal and external forces in the nodes, and finally the advection of the Lagrangian particles. We will first give a introduction to the governing equations and then follow by an algorithmic description of the process.

3.1 Governing equations

In the MPM approach a continuum is considered. Let define a Ω domain occupied by an elastic body, and $\partial\Omega$ the boundary of the domain, where $\partial\Omega = \Gamma_d \bigcup \Gamma_n$ and $\Gamma_d \cap \Gamma_n = \emptyset$. The behaviour of the continuum can be described by a set of governing equations which are (i) the balance of momentum equation

$$\rho a = \nabla \cdot \sigma + \rho b, \tag{1}$$

next (ii) the compatibility equation with ensure that the velocity field is compatible with the Cauchy strain field,

$$\dot{\varepsilon} = \nabla^s v = \frac{1}{2}\left(\nabla v + \nabla^T v\right), \tag{2}$$

later (iii) the constitutive equation, which relates the rate of Cauchy strain tensor with the rate of Cauchy stress tensor,

$$\dot{\sigma} = \mathsf{D} : \dot{\varepsilon}, \tag{3}$$

finally we introduce (iv) the mass conservation, which if obtained by setting to zero the total derivative of the density field,

$$\frac{D\rho}{Dt} = \dot{\rho} + \rho \nabla v = 0. \tag{4}$$

3.2 Variational formulation

To write the variational statement of the problem, let us define a virtual velocity field such that

$$v^\psi \in \mathscr{H}_0^1(\Omega) = \{v^\psi \in \mathscr{H}^1 \mid v^\psi = 0 \text{ on } \Gamma_d\}. \tag{5}$$

And which satisfies that the Cauchy sequences are convergent in Ω

$$\int_\Omega v^\psi \, d\Omega < \infty \quad \text{and} \quad \int_\Omega \nabla v^\psi \, d\Omega < \infty \tag{6}$$

The principle of virtual work states that the equilibrium solution to the boundary value problem of elasticity is the function $v \in \mathscr{H}_0^1$ such that, for $v^\psi \in \mathscr{H}_0^1$, the following holds:

$$\int_\Omega \rho \left(\frac{\partial \mathbf{v}}{\partial t} - \mathbf{b}\right) \cdot \mathbf{v}^\psi \, d\Omega = \int_{\Gamma_d} \mathbf{t} \cdot \mathbf{v}^\psi \, d\Gamma - \int_\Omega \sigma \cdot \varepsilon^\psi \, d\Omega, \tag{7}$$

therefore (7) is the balance of momentum formulated in its weak form.

3.3 Galerkin procedure

In order to arrive to a finite set of equations, in contrast with the FEM, in the MPM a double discretization procedure is performed as we will describe here below. First, the continuum domain Ω is discretized with a finite sum of material points (in the following particles), each one represent a part of the discretized domain $\Omega_p \subset \Omega$ with $p = 1, 2 \ldots, N_p$ where N_p is the number of particles. The material point $\mathbf{x}_p$ is defined at the centroid of each Ω_p, figure 1.

Each material point is assigned with initial values of position, velocity, mass, volume and stress denoted by $\mathbf{x}_p$, $\mathbf{v}_p$, m_p, V_p and σ_p, but also the virtual field v_p^ψ. So employing the definition of the material integral, where we recover the Riemann integral definition as an addition of a finite set of points. Therefore individual terms in (7) are solved as follows.

- Acceleration forces :

$$\int_\Omega \rho \, \mathbf{a} \, v^\psi \, d\Omega = \sum_p^{N_p} \dot{\mathbf{v}}_p \, v_p^\psi \, m_p. \tag{8}$$

- Internal forces :

$$\int_\Omega \sigma : \nabla\psi \, d\Omega = \sum_p^{N_p} \sigma_p : \nabla\psi_p \, V_p. \tag{9}$$

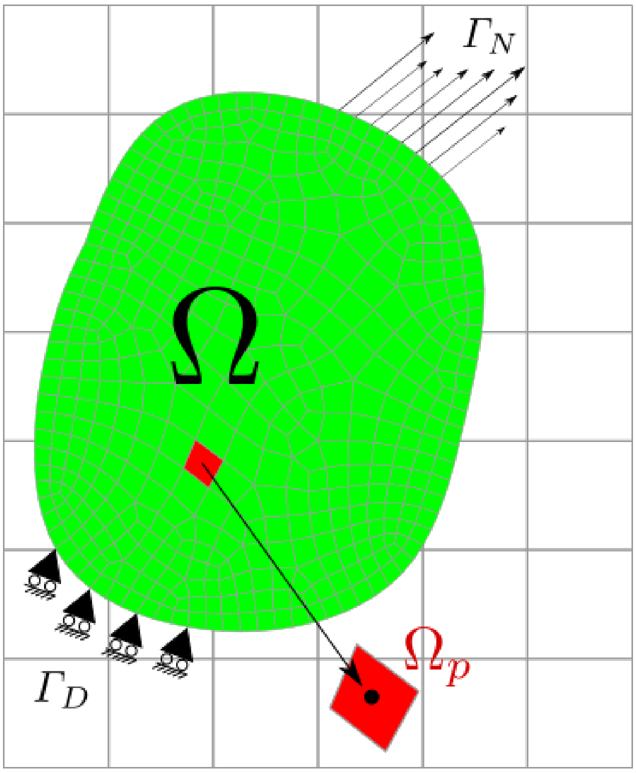


Fig. 1: MPM discretization.

– Body forces :

$$\int_{\Omega} \rho \ \mathbf{b} \ v^{\psi} \ d\Omega = \sum_{p}^{N_p} \mathbf{b}_p \ v_p^{\psi} \ m_p. \tag{10}$$

– Loads :

$$\int_{\Gamma_d} \mathbf{t} \ v^{\psi} \ d\Gamma = \int_{\Gamma_d} \rho \ \mathbf{t}^s \ v^{\psi} \ d\Gamma = \sum_{p}^{N_p} \mathbf{t}_p^s \ v_p^{\psi} \ h^{-1} \ m_p, \tag{11}$$

where h is the thickness of the continuum in a 2D case. Here is where the second discretization procedure appears. A background mesh composed by a finite set of grid points with coordinates $\mathbf{x}_I$, $I = 1,2\ldots,N_n$, is generated. Where N_n is the number of grid nodes. This mesh is employed as a support to compute gradients and divergences.

Introducing (8), (9), (10), and (11) in (7), and approximating the real and virtual velocity field of the particle p as

$$\mathbf{v}_p = \sum_{I}^{Nn} N_I(\mathbf{x}_p) \cdot v_I \quad \text{and} \quad \mathbf{v}_p^{\psi} = \sum_{I}^{Nn} N_I(\mathbf{x}_p) \cdot 1, \tag{12}$$

we reach to the particle balance of forces of the continuum,

$$\dot{\mathbf{p}}_I = \mathsf{m}_{IJ} \dot{\mathbf{v}}_J = \mathbf{f}_I^{int} + \mathbf{f}_I^{ext}, \tag{13}$$

where $\dot{\mathbf{p}}_I$ is the rate of momentum at grid node I, the nodal mass matrix m_{IJ} is,

$$\mathsf{m}_{IJ} = \sum_{p}^{Np} N_{Ip} m_p N_{Jp}. \tag{14}$$

To improve the computational efficiency and stability, the nodal mass matrix (14) can be substituted by the lumped mass matrix m_{IJ}^{lumped}. Later, internal and external forces are computed as follows,

$$\mathbf{f}_I^{int} = -\sum_{p}^{N_p} \sigma_p : \nabla N_{Ip} \frac{m_p}{\rho_p} \tag{15}$$

$$\mathbf{f}_I^{ext} = \sum_{p}^{N_p} \mathbf{b}_p N_{Ip} m_p + \sum_{p}^{N_p} N_{Ip} \mathbf{t}_p^s m_p h^{-1} \tag{16}$$

where $\sigma_p = \sigma_p(\varepsilon_p)$ is the particle p stress field, which can be integrated employing the suitable constitutive model. The strain tensor is updated employing the rate of stress tensor $\dot{\varepsilon}_p$ used to update the strain tensor is as follows (17).

$$\dot{\varepsilon}_p = \frac{\Delta \varepsilon_p}{\Delta t} = \frac{1}{2} \sum_{I}^{Nn} [\nabla N_{Ip} \otimes \mathbf{v}_I + \mathbf{v}_I \otimes \nabla N_{Ip}]. \tag{17}$$

Next, imposing $\frac{D\rho}{Dt} = 0$, we ensures the mass conservation and update the density field.

$$\dot{\rho} = -\rho \ tra(\dot{\varepsilon}) \tag{18}$$

Finally, to solve the equation (13), a second order temporal integration scheme is required. Therefore, time is discretized in to a finite set of time steps $k = 1\ldots,Nt$, where k is the current time step and N_t is the total number of time steps. Once the nodal equilibrium equation it is solved, the values in the nodes are interpolated back in to the particles and each particle it is advected to the new position,

$$\dot{\mathbf{v}}_p = \sum_{I}^{N_n} \mathbf{a}_I \ N_{Ip}, \quad and \quad \dot{\mathbf{x}}_p = \sum_{I}^{N_n} \mathbf{v}_I \ N_{Ip} \tag{19}$$

In the MPM literature, this equations (13) and (19), are solved with a forward Euler algorithm. In the next section, an alternative time integration scheme based on the Newmark a-form is proposed.

4 Explicit predictor-corrector scheme for MPM.

The time integration scheme of the MPM has been described in detail by many researchers [16], [3], [1] and summarized in Figure 2. Other authors have proposed many others time integration alternatives like [9], [17], [6]. In the first publication on the MPM [16], the nodal acceleration in employed to update the particles as

$$v_p^{k+1} = v_p^k + \Delta t \sum_I^{N_n} N_{Ip}^k \, \mathbf{a}_I^k \tag{20}$$

$$x_p^{k+1} = x_p^k + \Delta t \sum_I^{N_n} N_{Ip}^k \, \mathbf{v}_I^k. \tag{21}$$

However, as Andersen (2009)[1] point out, this algorithm has been shown to be numerically unstable due to that $f_I^{int,k}$ can be finite for an infinitesimal nodal mass m. This can lead to numerical issues when nodal acceleration is obtained for evaluating (21),(20). Hence, a corrected version of this algorithm in shown in Zhang *et al.* (2016)[22]

$$x_p^{k+1} = x_p^k + \Delta t \sum_I^{N_n} \frac{N_{Ip}^k \, \mathbf{p}_I^k}{\mathsf{m}_I}. \tag{22}$$

$$v_p^{k+1} = v_p^k + \Delta t \sum_I^{N_n} \frac{N_{Ip}^k \, \mathbf{f}_I^k}{\mathsf{m}_I}, \tag{23}$$

Later Tran & Solowski (2019)[17] presented a generalized-α scheme for the MPM inspired in the explicit time integration algorithm proposed by Chung & Hulbert (1993)[7], but with the particularity that the acceleration is evaluated both in the beginning and the end of the time step.

$$v_p^{k+1} = v_p^k + \Delta t \sum_I^{N_n} N_{Ip}^k \, \left[(1-\gamma) \, a_I^k + \gamma \, a_I^{k+1}\right], \tag{24}$$

$$x_p^{k+1} = x_p^k + \sum_I^{N_n} N_{Ip}^k \left[\Delta t \, \mathbf{v}_I^k + \Delta t^2 \left((\frac{1}{2} - \beta) \, \mathbf{a}_I^k + \beta \, \mathbf{a}_I^{k+1} \right)\right] \tag{25}$$

$$a_p^{k+1} = \sum_I^{N_n} N_{Ip}^k \, \mathbf{a}_I^{k+1}. \tag{26}$$

This scheme has prof to damps out the higher frequency noises [17]. But it can present the same numerical instabilities as in (21),(20) when nodal masses become infinitesimal. Here we propose a explicit predictor-corrector scheme for MPM based on the Newmark a-form with $\gamma = 0.5$ and $\beta = 0$ which is the central difference explicit. This method is devoted to solve a system of equations of type

$$\mathbf{M}_{IJ}\ddot{\mathbf{d}}_J + \mathbf{C}_{IJ}\dot{\mathbf{d}}_J + \mathbf{K}_{IJ}\mathbf{d}_J = \mathbf{F}_I$$

Nevertheless it can be applied for an isolated mass, therefore it is possible to apply this methods successfully in the MPM framework as was proved by [17]. Taking the predictor definition from the classic literature [11] and calculating particles velocity and acceleration as described in [22], we get the following expression for the **predictor** stage for both velocity and displacement.

$$\mathbf{x}_p^{k+1} = \mathbf{x}_p^n + \Delta t \underbrace{\sum_I^{N_n} \frac{\mathbf{p}_I^k \, N_{Ip}^k}{\mathsf{m}_I^k}}_{\mathbf{v}_p^k} + \frac{1}{2} \Delta t^2 \underbrace{\sum_I^{N_n} \frac{\mathbf{f}_I^k \, N_{Ip}^k}{\mathsf{m}_I^k}}_{\mathbf{a}_p^k} \tag{27}$$

$$\mathbf{v}_p^{k+1} = \mathbf{v}_p^n + (1-\gamma) \, \Delta t \underbrace{\sum_I^{N_n} \frac{\mathbf{f}_I^k \, N_{Ip}^k}{\mathsf{m}_I^k}}_{\mathbf{a}_p^k} \tag{28}$$

Consequently, the **corrector** stage for the particles velocity is in the following way

$$\mathbf{v}_p^{k+1} = \mathbf{v}_p^{k+1} + \gamma \, \Delta t \underbrace{\sum_I^{N_n} \frac{\mathbf{f}_I^{k+1} \, N_{Ip}^{k+1}}{\mathsf{m}_I^{k+1}}}_{\mathbf{a}_p^{k+1}} \tag{29}$$

Notice that the predictor of the particle displacement is computed using information only from the predictor of the previous step. However, velocities are computed using half information of the past corrector and half of the past predictor. Therefore here we share similarities with the **leapfrog integration** which updates the position at full time step, but updates the velocity at half time steps. Due to its simplicity allows be implemented with minor modifications over a standard forward Euler. It is summarized in shape of pseudo-algorithm **??**.

5 Local max-ent approximants

The previous lines describe the main algorithm without defining an interpolation technique. A direct consequence of this, is that it is possible to adopt a width range of them. The MPM has been successfully applied to a wide range of problems due to its ability to deal with large strain problems without mesh distorsion issues inherent to mesh based methods like FE. However, in the simulations made with the original MPM, there are numerical noises when particles crossing the cell boundaries. These numerical inaccuracies give rise to the development of other interpolation techniques. Some of this alternatives techniques are the generalized interpolation material point method (GIMP) Bardenhagen & Kober (2004)[4], the dual domain material point method (DDMP) Zhang *et al.* (2011)[21], the B-Spline MPM Tran *et al.* (2019)[18], the Conservative Taylor Least Squares reconstruction Wobbes *et al.* (2018)[19] and more recently the local maximum-entropy (or local *max-ent*) shape function first

Algorithm Explicit scheme : Newmark central difference

1: **Explicit Newmark Predictor** ($\gamma = 0.5$) :

$$\mathbf{x}_p^{k+1} = \mathbf{x}_p^n + \Delta t \sum_I^{N_n} \frac{\mathbf{p}_I^k \, N_{Ip}^k}{\mathsf{M}_I^k} + \frac{1}{2}\Delta t^2 \sum_I^{N_n} \frac{\mathbf{f}_I^k \, N_{Ip}^k}{\mathsf{M}_I^k}$$

$$\mathbf{v}_p^{k+1} = \mathbf{v}_p^n + (1-\gamma) \sum_I^{N_n} \frac{\mathbf{f}_I^k \, N_{Ip}^k}{\mathsf{M}_I^k} \Delta t$$

2: **Discard the previous nodal values**.

3: **Transference of particles kinetics information to the mesh**:
Calculate the lumped mass matrix M_I, and the nodal momentum $\mathbf{p}_I$ with the predicted values.

$$\mathsf{M}_I = \sum_p^{N_p} m_p \, N_{Ip}^{k+1},$$

$$\mathbf{p}_I^{k+1} = \sum_p^{N_p} m_p \, \mathbf{v}_p^{k+1} \, N_{Ip}^{k+1},$$

4: **Impose essential boundary conditions over momentum nodal values**:
At the fixed boundary, set $\mathbf{p}_I^{k+1} = 0$. Later, get the nodal values of the velocity field $\mathbf{v}_I^{k+1} = \frac{\mathbf{p}_I^{k+1}}{\mathsf{M}_I}$

5: **Deformation tensor increment calculation**.

$$\dot{\varepsilon_p}^{k+1} = \frac{1}{2} \sum_I^{N_n} \nabla N_{Ip}^{k+1} \mathbf{v}_I^{k+1} + (\nabla N_{Ip}^{k+1} \mathbf{v}_I^{k+1})^T$$

$$\Delta \varepsilon_p^{k+1} = \Delta t \; \dot{\varepsilon_p}^{k+1}$$

6: **Update the density field**:

$$\rho_p^{k+1} = \frac{\rho_p^k}{1 + tra\left[\Delta \varepsilon_p^{k+1}\right]}.$$

7: **Balance of forces calculation**:
Calculate the total grid nodal force $\mathbf{f}_I^{k+1} = \mathbf{f}_I^{int,k+1} + \mathbf{f}_I^{ext,k+1}$ evaluating (15) and (16) in the time step $k+1$. In the grid node I is fixed in one of the spatial dimensions, set it to zero to make the grid nodal acceleration zero in that direction.

8: **Integrate the grid nodal momentum equation :**
Calculate the nodal momentum $\mathbf{p}_I$ in $k+1$.

$$\mathbf{p}_I^{k+1} = \mathbf{p}_I^k + \mathbf{f}_I^{k+1} \Delta t$$

9: **Explicit Newmark Predictor**:

$$\mathbf{v}_p^{k+1} = \mathbf{v}_p^{k+1} + \gamma \sum_I^{N_n} \frac{\mathbf{f}_I^{k+1} \, N_{Ip}^{k+1}}{\mathsf{m}_I^{k+1}} \Delta t$$

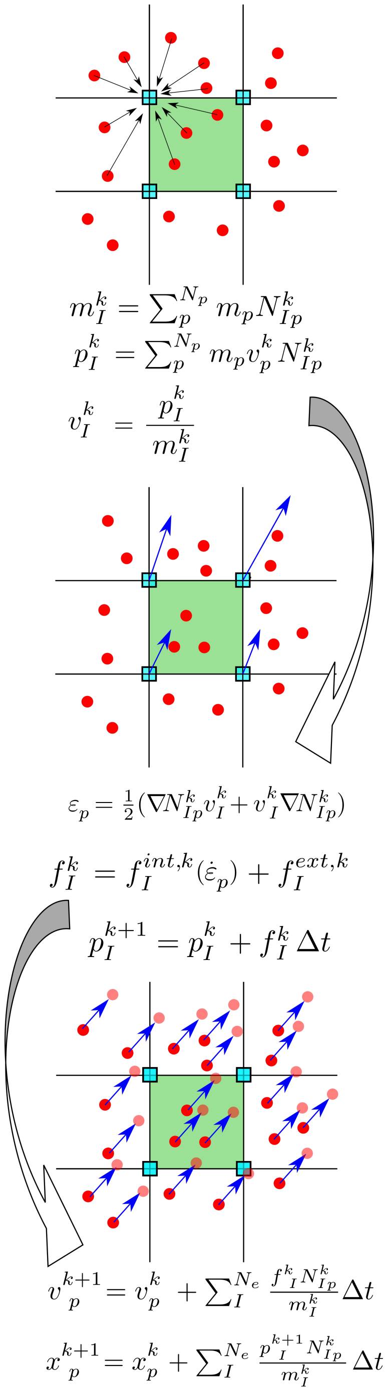


Fig. 2: MPM standard algorithm.

introduced by Arroyo & Ortiz (2006)[2] has been tested under the MPM framework by Wobbes *et al.* (2020)[20] where they prof that simulations performed with the maxent basis functions show considerably more accurate stress approximations for MPM. Although, in [20] authors does not deep in β parameter benefits

The maximum-entropy estimate is defined as the type of statistical inference, which is the least biased possible on the given information [12]. The basic idea of the shape functions based on such an estimate is to interpret the shape function $N_I(\mathbf{x})$ as the probability of $\mathbf{x}$ to obtain the value $\mathbf{x}_I$, $I = 1,\ldots,n$. Here n is the number of nodes in the domain. Taking Shannon's entropy as the starting point:

$$H(p_1(\mathbf{x}),\ldots,p_n(\mathbf{x})) = -\sum_{I=1}^{N_n} p_I(\mathbf{x}) \log p_I \tag{30}$$

where $p_I(\mathbf{x})$ is the probability, equivalent to the mentioned shape function $N_I(\mathbf{x})$, satisfying both the zeroth and first-order consistency. The least-biased approximation scheme is given by

$$\begin{aligned}
\text{(LME)}\ \ \text{Maximize}\ \ & H(p) = -\sum_{I}^{N_n} p_I(\mathbf{x}) \log p_I \\
\text{subject to}\ \ & p_I \geq 0,\ \ \text{I=1, ..., n} \\
& \sum_{I=1}^{N_n} p_I = 1 \\
& \sum_{I=1}^{N_n} p_I \mathbf{x}_I = \mathbf{x}
\end{aligned}$$

The local max-ent approximation schemes (LME) as a Pareto set, defined by [2] is as follows

$$\begin{aligned}
\text{(LME)}_\beta\ \ \text{For fixed}\ \mathbf{x}\ \text{minimise}\ \ & f_\beta(\mathbf{x},p) = \beta H(\mathbf{x},p) - H(p) \\
\text{subject to}\ \ & p_I \geq 0,\ \ \text{I=1, ..., n} \\
& \sum_{I=1}^{N_n} p_I = 1 \\
& \sum_{I=1}^{N_n} p_I \mathbf{x}_I = \mathbf{x}
\end{aligned}$$

for $\beta \in (0,\infty)$ is Pareto optimal. The unique solution of the local max-ent problem (LME)_β is:

$$p(\mathbf{x}) = \frac{\exp\left[-\beta\ |\mathbf{x}-\mathbf{x}_I|^2 + \lambda\ (\mathbf{x}-\mathbf{x}_I)\right]}{Z(\mathbf{x},\lambda^*(\mathbf{x}))} \tag{31}$$

where

$$Z(\mathbf{x},\lambda) = \sum_{I=1}^{N_n} \exp\left[-\beta\,|\mathbf{x}-\mathbf{x}_I|^2 + \lambda\,(\mathbf{x}-\mathbf{x}_I)\right] \tag{32}$$

being $\lambda^*(\mathbf{x})$ the unique minimiser for $\log Z(\mathbf{x},\lambda)$

In order to obtain the first derivatives of the shape function, it is also necessary to compute ∇p_I^*

$$\nabla p_I^* = p_I^* \left(\nabla f_I^* - \sum_{J}^{N_n} p_J^* \nabla f_J^*\right) \tag{33}$$

where

$$f_I^*(\mathbf{x},\lambda,\beta) = -\beta\,|\mathbf{x}-\mathbf{x}_I|^2 + \lambda\,(\mathbf{x}-\mathbf{x}_I) \tag{34}$$

Employing the chain rule, rearranging and considering β as a constant, Arroyo and Ortiz [2] obtained the following expression:

$$\nabla p_I^* = -p_I^*\,(\mathsf{J}^*)^{-1}\,(\mathbf{x}-\mathbf{x}_I) \tag{35}$$

where J is the Hessian matrix, defined by:

$$\mathsf{J}(\mathbf{x},\lambda,\beta) = \frac{\partial \mathbf{r}}{\partial \lambda} \tag{36}$$

$$\mathbf{r}(\mathbf{x},\lambda,\beta) \equiv \partial_\lambda \log Z(\mathbf{x},\lambda) = \sum_{I}^{N_n} p_I(\mathbf{x},\lambda,\beta)\,(\mathbf{x}-\mathbf{x}_I) \tag{37}$$

Note that, the objective of the above procedure is to find the λ which minimises $\log Z(\mathbf{x},\lambda)$. The traditional way to obtain such a minimiser is using Eq. (36) to calculate small increments of $\partial\lambda$ in a Newton-Raphson approach. Similar to alternative non-polynomial meshfree basis functions, the LME approximation scheme requires more than $d+1$ nodes to determine the values of the shape functions as well as their derivatives at any point in the convex hull of the nodal set, where d is the dimension of the problem. Due to the FE-compatibility, the LME shape function is degenerated to linear finite element shape function if $d+1$ neighbouring nodes are chosen as the support. The support size of LME shape functions may be controlled by adjusting a dimensionless parameter, $\gamma = \beta h^2$ (e.g. two dimensional example shown in figure. 3)[2].

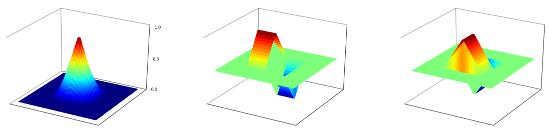
Since p_I is defined in the entire domain, in practice, the function $\exp(-\beta\mathbf{r})$ truncated by a given tolerance, 10^{-6}, for example, would ensure a reasonable range of neighbours, see [2] for details. This tolerance defines the limit values of the influence radius and is used thereafter to find the neighbour nodes of a given integration point. In this research and in [2], is a scalar as the influence area of the shape function is controlled by the Euclidean norm, therefore the search area is geometrically a circle in 2D, or a sphere in 3D. Building upon the idea of anisotropic shape functions, [13] introduced an enhanced version of the original local max-ent scheme, which uses an anisotropic support to deal with tensile inestability. Nonetheless this is out of the scope of the present document but will be incorporated in future research.

6 Numerical results : 1D Elastic bar

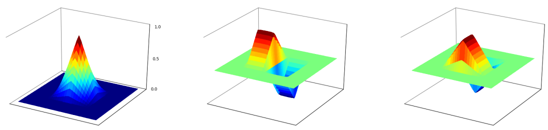
In this section, the benchmark proposed by Dyka & Ingel (1995)[8] is considered to illustrate the capacity of MPM [16] and OTM models to avoid tensile instabilities.

In the one-dimensional bar sketched in the figure 4, the left end of the bar is fixed and the right and an initial velocity $v_0 = 5\ m/s$ is given to the last quarter of it in the x positive direction. The length of it is 0.1333 meters with an unit section. The elastic parameters consider for this test are:

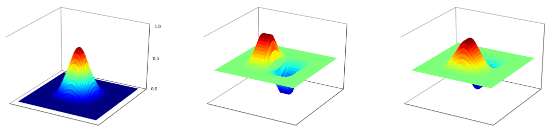
- Density : $7833\ kg/m3$



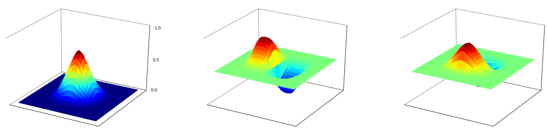
(a) Piecewise linear (Q4).



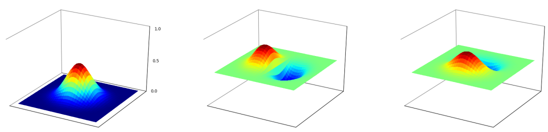
(b) Local *max-ent* $\gamma = 17$.



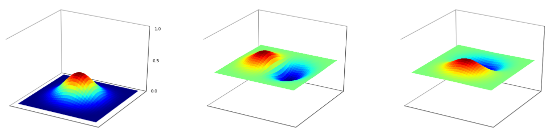
(c) Uniform generalized interpolated material point.



(d) Local *max-ent* $\gamma = 10$.



(e) Local *max-ent* $\gamma = 7$.



(f) Local *max-ent* $\gamma = 5$.

Fig. 3: Local max-ent shape functions for a two-dimensional arrangement of nodes, and spatial derivatives for several values of $\gamma = \beta/h^2$.

- Poisson ratio : 0
- Elastic modulus : $200 \cdot 10^9$ *Pa*

The results are provided for the explicit predictor-corrector version of the MPM scheme (Algorithm **??**) and OTM algorithms. For MPM, piecewise- linear, uGIMP and *max-ent* basis functions are employed. The OTM algorithm is used only with *max-ent* shape functions.

The boundary conditions are:

$$\sigma|_{x=L} = 0 \quad , \quad v|_{x=0} = 0 \tag{38}$$

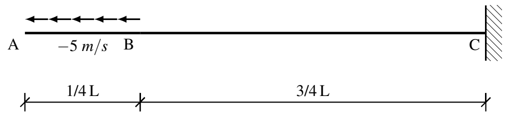


Fig. 4: Geometrical description of the Dyka [8] bar.

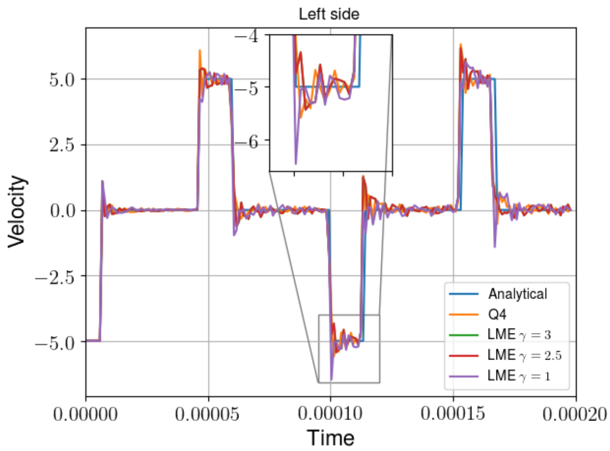


Fig. 5: Velocity evolution at the point in the bar left side.

The analytical solution in terms of velocity 6, and stress 7 can be found in the appendix A given by the Method of characteristics. For the convergence analysis, the root-mean-square (RMS) error in the displacement is computed. RMS error is defined as follows

$$RMS = \sqrt{\frac{1}{N}\sum_{p}^{N}\left(\phi_p - \hat{\phi}_p\right)} \tag{39}$$

6.1 Error analysis MPM vs uGIMP vs MPM-*max-ent*

6.2 MPM versus OTM

7 Conclusions

Conflict of interest

The authors declare that they have no conflict of interest.

A The analytical solution of the 1D Dyka benchmark

For the derivation of this analytical solution we will consider the dynamic behaviour of a 1D elastic bar. The governing equations are the following: (i) The balance of linear momentum,

$$\rho\,\frac{\partial v}{\partial t} = \frac{\partial \sigma}{\partial x} + \rho\, b, \tag{40}$$

where σ is the stress value, ρ is the density, v is the velocity, and b are the body forces. (ii) The constitutive model, which for convenience of the following developments will be written in terms of displacement and velocities as,

$$\frac{\partial \sigma}{\partial t} = E \frac{\partial \varepsilon}{\partial t}, \tag{41}$$

where E is the elastic modulus. (iii) The compatibility equation also in terms of the velocity field,

$$\frac{\partial \varepsilon}{\partial t} = \frac{\partial v}{\partial x}. \tag{42}$$

Next for simplicity, we remove the body forces from (40), and we will introduce (42) in (41), so we get the following system of equations,

$$\frac{\partial v}{\partial t} = \frac{1}{\rho} \frac{\partial \sigma}{\partial x}, \tag{43}$$

$$\frac{\partial \sigma}{\partial t} = E \frac{\partial v}{\partial x}. \tag{44}$$

Introducing (44) in (43) and expressing the remaining equation in terms of the displacement, we reach the 1D wave equation for linear elastic materials,

$$\frac{\partial^2 u}{\partial t^2} = \frac{E}{\rho} \frac{\partial^2 u}{\partial x^2} = c^2 \frac{\partial^2 u}{\partial x^2} \tag{45}$$

where we have introduced the wave celerity c as,

$$c = \sqrt{\frac{E}{\rho}} \tag{46}$$

Alternative, rearranging both equations (43) and (44) it is possible to join them in a single system of equations as,

$$\frac{\partial}{\partial t} \begin{bmatrix} \sigma \\ v \end{bmatrix} + \begin{bmatrix} 0 & -E \\ -1/\rho & 0 \end{bmatrix} \begin{bmatrix} \frac{\partial \sigma}{\partial x} \\ \frac{\partial v}{\partial x} \end{bmatrix} = \mathbf{0}. \tag{47}$$

This expression can be written in a more compact format as,

$$\frac{\partial \phi}{\partial t} + \mathbf{A} \frac{\partial \phi}{\partial x} = \mathbf{0} \tag{48}$$

where both variables are joined in a single vectorial variable ϕ and $\mathbf{A}$ in coupling matrix between both equations,

$$\phi = \begin{bmatrix} \sigma \\ v \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} 0 & -E \\ -1/\rho & 0 \end{bmatrix}.$$

Note that the nature of is still hyperbolic despite the fact it does not have a second order temporal derivative as (45). A proof of this can be easily obtained if we get the zeros of the hypersurface defined by (45). And later the eigenvalues of $\mathbf{A}$ in (48). In both cases, eigenvalues are real and distinct ($\lambda = \pm\sqrt{\frac{E}{\rho}}$), therefore the system is called strictly hyperbolic.

For a more general description in the following, we will assume that $\mathbf{A}$ has n different eigenvalues $\{\lambda_1, \lambda_2, \ldots, \lambda_i, \ldots \lambda_n\}$ and n eigenvectors $\{\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^i, \ldots \mathbf{x}^n\}$ satisfying that $\mathsf{A}\mathbf{x} = \lambda \mathbf{x}$. Now we introduce the matrix $\mathbf{P}$ whose columns are the n eigenvalues $\mathbf{x}$

$$\mathbf{P} = \{\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3, \ldots \mathbf{x}^n\}. \tag{49}$$

Diagonalizing $\mathbf{A}$ using $\mathbf{P}$ we get

$$\Lambda = \mathbf{P}^{-1} \mathbf{A}\, \mathbf{P}, \tag{50}$$

where $\Lambda_{ii} = \lambda_i$. Next we will define a vector $\Re$ such that:

$$\phi = \mathbf{P}\, \Re \tag{51}$$

we will assume to be integrable. Expanding the above expression with the chain rule and passing the matrix $\mathbf{P}$ to left hand side of the equality we get,

$$d\Re = \frac{\partial \Re}{\partial t} dt + \frac{\partial \Re}{\partial x} dx = \mathsf{P}^{-1} \left(\frac{\partial \phi}{\partial t} dt + \frac{\partial \phi}{\partial x} dx \right) \tag{52}$$

and setting the terms we get,

$$\frac{\partial \Re}{\partial t} = \mathbf{P}^{-1} \frac{\partial \phi}{\partial t}, \quad \frac{\partial \Re}{\partial x} = \mathbf{P}^{-1} \frac{\partial \phi}{\partial x} \tag{53}$$

Next, if we multiply (48) by $\mathbf{P}^{-1}$ we get:

$$\mathbf{P}^{-1} \frac{\partial \phi}{\partial t} + \left(\mathbf{P}^{-1} \mathbf{A} \mathbf{P}\right) \mathbf{P}^{-1} \frac{\partial \phi}{\partial x} = \mathbf{0} \tag{54}$$

finally introducing the expressions (53) we reach to

$$\frac{\partial \Re}{\partial t} + \Lambda \frac{\partial \Re}{\partial x} = \mathbf{0} \tag{55}$$

which consists of n uncoupled equations as Λ is diagonal matrix as we can see in (50). Each of this equations are 1D scalar convective transport equations, with solutions of the form:

$$\Re^{(i)} = F^{(i)} \left(x - \lambda^{(i)} t \right) \tag{56}$$

This uncoupled system, has, therefore, a set of n characteristics. These magnitudes $\Re_i$ which propagate along characteristics are known as "Riemann invariants" of the problem. Here we have a 1D configuration, so the domain is $\Omega : (0,L)\, x\, (0,T)$. For the closure of the problem we require:

- "n" initial conditions of the form $\Re_i(x, t=0) = h_i(x)$, where $i = 0, \ldots, n$, and $h_i(x)$ is a vectorial function given by the physical variables of the problem.
- "n" boundary conditions.

Now particularizing the previous equations for the 1D elastic bar described in [8], we get that the matrix $\mathbf{P}$ is the following:

$$\mathbf{P} = \begin{bmatrix} -\sqrt{E\rho} & \sqrt{E\rho} \\ 1 & 1 \end{bmatrix}$$

and its inverse is:

$$\mathbf{P}^{-1} = \frac{1}{2} \begin{bmatrix} -\frac{1}{\sqrt{E\rho}} & 1 \\ \frac{1}{\sqrt{E\rho}} & 1 \end{bmatrix}$$

And introducing the value of the inverse matrix $\mathbf{P}^1$ in the Riemann definition (51) we get the following system of equations,

$$\Re^I = \frac{1}{2\sqrt{\rho E}} \left(-\sigma + v\, \sqrt{\rho E} \right) \tag{57}$$

$$\Re^{II} = \frac{1}{2\sqrt{\rho E}} \left(\sigma + v\, \sqrt{\rho E} \right) \tag{58}$$

From (57) and (58) we can obtain the values of the stress and the velocity as:

$$v = \Re^I + \Re^{II} \quad , \quad \sigma = \sqrt{E\rho} \left(\Re^{II} - \Re^I \right) \tag{59}$$

The boundary conditions are in both cases of radiation as there is not wave in-going from the exterior. So for the left side (fixed boundary) we get the following conditions:

$$\Re^I = 0 \quad and \quad v_{x=0} = 0$$

Therefore $\sigma_{x=0} = \sqrt{\rho E} \Re^{II}$. And in the left side (free boundary) we get the following conditions:

$$\Re^{II} = 0 \quad and \quad \sigma_{x=L} = 0$$

Therefore $v_{x=L} = \Re^I$. Finally, applying this conditions in the elastic bar sketched in 4, is possible to obtain the velocity history in the right side of the bar 6 and the stress in the last quarter side of the Dyka bar 7 as is demanded in [8].

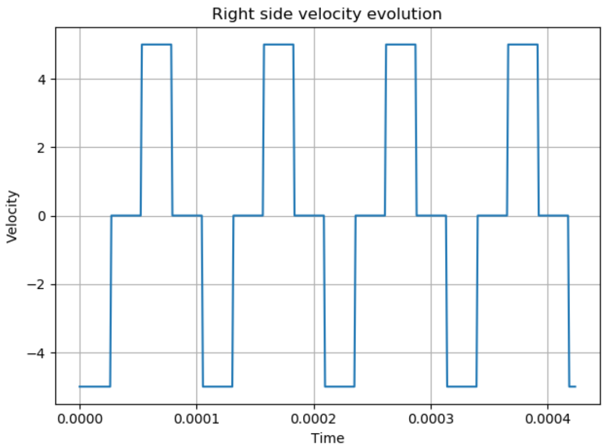


Fig. 6: Analytical solution for the velocity in the right side of the Dyka bar.

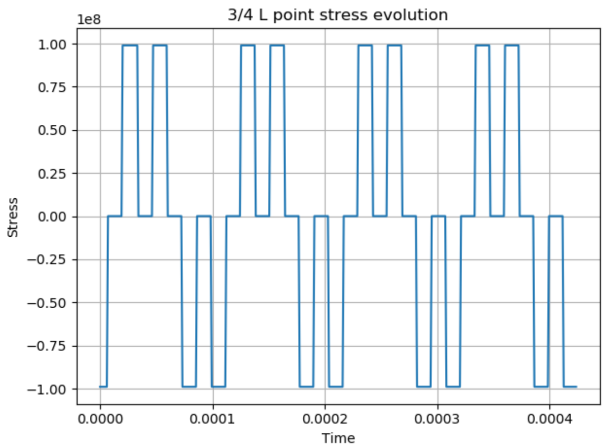


Fig. 7: Analytical solution for the stress in the last quarter of the Dyka bar.

References

1. Andersen, S.M.: Material-Point Analysis of Large-Strain Problems: modelling of landslides. Ph.D. thesis, Department of Civil Engineering, Aalborg University (2009)
2. Arroyo, M., Ortiz, M.: Local maximum-entropy approximation schemes: A seamless bridge between finite elements and meshfree methods. International Journal for Numerical Methods in Engineering (2006). DOI 10.1002/nme.1534
3. Bardenhagen, S.G.S.: Energy Conservation Error in the Material Point Method for Solid Mechanics. Journal of Computational Physics **180**(1), 383–403 (2002). DOI 10.1006/JCPH.2002.7103. URL `https://www.sciencedirect.com/science/article/pii/S0021999102971032`
4. Bardenhagen, S.G.S.G., Kober, E.M.: The generalized interpolation material point method. CMES - Computer Modeling in Engineering and Sciences **5**(6), 477–495 (2004)
5. Brackbill, J.U., Ruppel, H.M.: FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions. Journal of Computational Physics (1986). DOI 10.1016/0021-9991(86)90211-1
6. Charlton, T.J., Coombs, W.M., Augarde, C.E.: iGIMP: An implicit generalised interpolation material point method for large deformations. Computers & Structures **190**, 108–125 (2017). DOI 10.1016/j.compstruc.2017.05.004
7. Chung, J., Hulbert, G.M.: A Time Integration Algorithm for Structural Dynamics With Improved Numerical Dissipation: The Generalized-alpha Method. J. Appl. Mech. **60**(2), 371 (1993). DOI 10.1115/1.2900803. URL `http://appliedmechanics.asmedigitalcollection.asme.org/article.aspx?articleid=1410995`
8. Dyka, C., Ingel, R.: An approach for tension instability in smoothed particle hydrodynamics (SPH). Computers & Structures **57**(4), 573–580 (1995). DOI 10.1016/0045-7949(95)00059-P. URL `https://www.sciencedirect.com/science/article/pii/004579499500059P`
9. Guilkey, J., Weiss, J.A.: Implicit time integration for the material point method : Quantitative and algorithmic comparisons with the nite element method. International Journal for Numerical Methods in Engineering (2003). URL `https://www.semanticscholar.org/paper/Implicit-time-integration-for-the-material-point-{%}3A-Guilkey-Weiss/f31f3e5c2653ebaa0b8e723e8bc0ba4ab4d8f7a4`
10. Harlow, F.H., Evans, M.W.: A Machine Calculation Method for Hydrodynamic Problems. Los Alamos Scientific Laboratory (1956)
11. Hughes, T.J.R.: The finite element method : linear static and dynamic finite element analysis. Dover Publications (2000)
12. Jaynes, E.: Information Theory and Statistical Mechanics. The Physical Review **106**(4), 620–630 (1957)
13. Kumar, S., Danas, K., Kochmann, D.M.: Enhanced local maximum-entropy approximation for stable meshfree simulations. Computer Methods in Applied Mechanics and Engineering (2019). DOI 10.1016/j.cma.2018.10.030
14. Li, B., Kidane, A., Ravichandran, G., Ortiz, M.: Verification and validation of the Optimal Transportation Meshfree (OTM) simulation of terminal ballistics. International Journal of Impact Engineering (2012). DOI 10.1016/j.ijimpeng.2011.11.003
15. Navas, P., López-Querol, S., Yu, R.C., Pastor, M.: Optimal transportation meshfree method in geotechnical engineering problems under large deformation regime. International Journal for Numerical Methods in Engineering (2018). DOI 10.1002/nme.5841
16. Sulsky, D.L., Schreyer, H., Chen, Z.: A particle method for history-dependent materials. Computer Methods in Applied Mechanics and Engineering **118**(1), 179–196 (1994). DOI 10.1016/0045-7825(94)90112-0
17. Tran, Q.A., Sołowski, W.: Temporal and null-space filter for the material point method. International Journal for Numerical Methods in Engineering (2019). DOI 10.1002/nme.6138
18. Tran, Q.A.A., Wobbes, E., Solowski, W.T., Möller, M., Vuik, C.: Moving least squares reconstruction for B-spline Material Point Method. In: 2nd International Conference on the Material Point Method for Modelling Soil-Water-Structure Interaction Anura3D, January, pp. 35–41 (2019). URL `https://aaltodoc.aalto.fi/handle/123456789/36294`
19. Wobbes, E., Moller, M., Galavi, V., Vuik, C., Möller, M., Galavi, V., Vuik, C., Moller, M., Galavi, V., Vuik, C.: Conservative Taylor Least Squares reconstruction with application to material point methods: Conservative Taylor Least Squares reconstruction. International Journal for Numerical Methods in Engineering **117**(3), 271–290 (2018). DOI 10.1002/nme.5956
20. Wobbes, E., Tielen, R., Möller, M., Vuik, C.: Comparison and unification of material-point and optimal transportation meshfree methods. Computational Particle Mechanics (2020). DOI 10.1007/s40571-020-00316-7
21. Zhang, D.Z., Ma, X., Giguere, P.T.: Material point method enhanced by modified gradient of shape function. Journal of Computational Physics **230**(16), 6379–6398 (2011). DOI 10.1016/J.

JCP.2011.04.032. URL https://www.sciencedirect.com/science/article/pii/S0021999111002804
22. Zhang, X., Chen, Z., Liu, Y.: The Material Point Method: A Continuum-Based Particle Method for Extreme Loading Cases. Elsevier (2016). DOI 10.1016/b978-0-12-407716-4.00003-x