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

Enhanced Material Point Method to face dynamic problems:

Local-maximum entropy approximation and explicit predictor-corrector scheme

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

Received: date / Accepted: date

Abstract Material Point Method (MPM) has arisen in the recent years as an alternative to Finite Element Method (FEM) under the large deformation regime. However, the simulation of shock waves propagation and other high frequency problems is still challenging under this approach due the incapability of the standard MPM time integration scheme to filter spurious noises. To overcome this limitation in this paper, an explicit predictor-corrector time integration scheme has been proposed. Its superior performance mitigates the presence of spurious oscillations with minimal dissipation in high frequency problems. Other source of numerical noise in the MPM occurs when to material points cross computational grid boundaries and motivated due to the lack of smoothness of the interpolation functions. This noise results in spurious local variations at the material points, where strain-stress fields are computed. This could invalidate the solution in one case, or damage it in another. To overcome it, this document adopts the *local maximum-entropy approximation schemes* (LME) a robust substitute for the wide range of shape function in the MPM. Local *max-ent* approximation may be regarded as a *thermalization* of Delaunay triangulation which resolves the degenerate cases resulting from the lack or uniqueness of the triangulation. Furthermore, by modifying a regularization parameter they are able to behave finite element like or as a mesh-free method. This capability allows to face a wide range of physics with a single shape function family. Finally this paper demonstrates the performance of both improvements thorough numerical examples.

Keywords LME · MPM · Explicit predictor-corrector · Dynamic problems

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

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

1 Introduction

Since the proposal of the MPM by Sulsky *et al.* (1994) [18] as a generalization to solids of the Fluid Implicit Particle (FLIP) method [5]. It popularity has increased due to its ability to deal with large strain regime without suffer mesh distortion inaccuracies.

However, this method suffers other kind of instabilities, such those when material points crossing cell boundaries. This give rise to the development of other interpolation techniques to overcome this limitation such as the generalized interpolation material point method (GIMP) Bardenhagen & Kober (2004) [4], which has demonstrate to have a good performance in the finite deformation regime. However, in the absence of a regular grid, construction of the weighting functions is only achieved at considerable effort and computational cost. Furthermore, as it is a voxel based discretization technique, it is prone to suffer voxel domains overlap or gaps when the material point mesh becomes irregular, which can introduce severe inaccuracies. This is similar to the difficulty encountered by the finite element methods due to element distortion. A more robust alternative is the dual domain material point method (DDMP) proposed by Zhang *et al.* (2011) [25]. Unfortunately this method shows an unsatisfactory behaviour when particle/cell ratio decreases [8], therefore DDMP requires a large number of particles needed for convergence, this makes the method very expensive. In recent years the employ of spline-lines has gain popularity with the introduction of the B-Spline MPM proposed by Roel Tielen *et al.* (2017) [19], this technique allows the employ of unstructured set of notes and particles. More recently, approximants derived from minimization has been introduced in to the MPM framework with the Conservative Taylor Least Squares (CTLS) reconstruction proposed by Wobbes *et al.* (2018) [22], unfortunately when particles

are spread in a challenging way, the quality of the CTLS approximation decrease locally.

This document adopts the *local maximum-entropy approximation schemes* (LME) as a robust substitute for the wide range of shape function in the MPM. First introduced by Arroyo & Ortiz (2006) [2], it belongs to the class of convex approximation schemes and provides a seamless transition between finite element method (FEM) 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 or the triangulation. Local *max-ent* basis functions possess many desirable properties for mesh-free 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) [16] and Li *et al.* (2012) [14] for Optimal Transportation Meshfree (OTM) method. And more recently under the MPM framework by Wobbes *et al.*(2020) [23] but without exploring the benefits of the regularization parameter β.

This techniques are devoted to mitigate the "grid crossing" error. Nevertheless, in the presence of shock waves spurious numerical noises appears despite of this using techniques [20]. These numerical inaccuracies also known as wiggles are due to inaccuracies in the time discretization technique. A simple approach to face those spurious noises is to add a nonphysical damping source to the equilibrium equations, this approach has been widely employed in this and many other numerical techniques. To avoid introducing this nonphysical sources, many researchers has proposed alternative time integration schemes which reduce the presence of high frequency noises by filtering them or increasing the accuracy of the time integration scheme. One of the most popular is the implicit GIMP (iGIMP) Charlton *et al.* (2017) [6], more recently Tran & Solowski (2019) [20] proposed a generalised-α scheme for the MPM with promising results but at the expense of increasing the computational effort. In this paper a less time consuming and high efficient explicit predictor corrector integration method has been proposed. It consists in an accommodation of the Newmark predictor-corrector or central difference explicit (CD). We have choose this method among other suitable alternatives as those proposed by Wilson *et al.* (1972) [21] or Chung & Hulbert (1993) [7] because it simplicity and it good performance dealing with solids dynamics problems under a mesh-free framework in [16].

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	First order tensor
x	Global coordinates	First order tensor
ξ	Local coordinates	First order tensor
σ	Cauchy stress tensor	Second order tensor
ε	Cauchy strain tensor	Second order tensor
D	Constitutive tensor	Fourth order tensor

The aim of this document is to mitigate the spurious oscillations due inaccuracies in both space and time discretization by the employ of a suitable combination of the maximum-entropy (or local *max-ent*) family shape functions, and the proposal of a explicit predictor-corrector scheme. We illustrate the advantages of this approach thorough to simple but challenging test cases, like the propagation of shock waves in a elastic bar or the response of a block of soil gradually loaded with gravitational forces.

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 applications to prove the numerical accuracy of the proposed approach are presented. Finally, conclusions and future research topics are exposed in Section 7.

2 Brief note concerning the notation

In what follows, we will adopt the following convention. All the physical variables involved in this problem are collected in Table 1. Three kind of subscript or superscript are used within paper. The subscript $\square_p$ is used to define a particle variable. While the subscript $\square_I$ is reserved in this notation for denoting nodal variables. And finally, the superscript $\square^{\psi}$ involves a virtual magnitude. For the operators, the convention is : $\dot{\square}$ and $\ddot{\square}$ for the first and second time derivative, $\otimes$ means the dyadic operator, $(\cdot)$ and $(:)$ means the first and second contraction of a tensor, $div(\square)$ denotes the divergence operator, and finally $grad(\square)$ and $grad^s(\square)$ denotes the gradient and its symmetric part. Einstain subscripts convention is adopted therefore repeated index means addition.

3 Derivation of the MPM

The aim of this section is to provide an overview of the standard explicit MPM algorithm [18]. In consequence it is

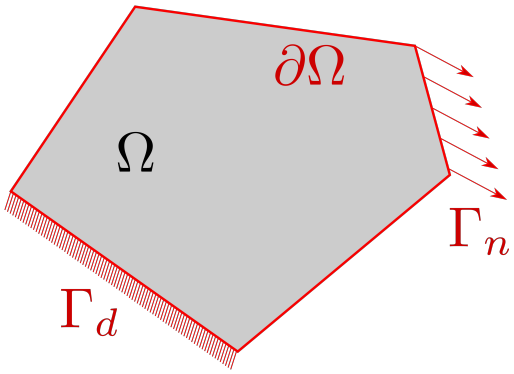


Fig. 1: Description of the boundary-value-problem in a continuum. Red lines represents the closure $\partial\Omega$ of the domain Ω represented in gray.

structured as follows: the governing equations will be introduced in 3.1, later the variational statement of the problem will be presented 3.2, and finally a discretization procedure thorough Galerkin is performed in 3.3. The method has three main steps: (i) a variational recovery process, where particle data is projected to the grid nodes, (ii) an Eulerian step, where balance of momentum equation is expressed as a nodal equilibrium equation thorough a FEM-like procedure, and finally (iii) a Lagrangian advection of the particles. In consequence the MPM can be regarded as a Lagrangian-Eulerian method where particles carries on all the physical information and a set of background nodes is employed to compute the equilibrium equation.

3.1 The boundary-value problem

In MPM the continuum mechanics approach is considered. So on, let define a continuum domain Ω occupied by an elastic body like the sketched in 1, and $\partial\Omega$ the boundaries of the domain defined by $\partial\Omega = \Gamma_d \bigcup \Gamma_n$ and $\Gamma_d \cap \Gamma_n = \emptyset$. In this context the field u allows to describe the *global state* of the system. Now the variable $\phi = (\varepsilon, \sigma)$ is defined as the set of *local states* at any point of the continuum which can be derived from the field u through the following set of governing equations and restrictions that must be satisfied. First (i) the *compatibility equation* that extracts from u the strain field is,

$$\varepsilon = grad^s(u), \tag{1}$$

together with essential boundary conditions of Dirichlet type Γ_d. An additional consideration over the strain field is the assumption of inifinitesimal strain, therefore second order terms in the spatial derivatives can be neglected. The corresponding conjugate variable for the strain field if the stress field σ, which satisfies (ii) the *conservation of momentum equation*

$$\rho \frac{Dv}{Dt} = div(\sigma) + \rho b \tag{2}$$

together with the natural boundary conditions of the Neumann type Γ_n. Next (iii) the constitutive equation as a linear application from $\Re^n$ to $\Re^n$, which relates the strain tensor with the stress tensor,

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

The final restriction is (iv) the mass conservation, which can be obtained by setting to zero the total derivative of the density field,

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

3.2 Variational formulation

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

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

And which satisfies that the Cauchy sequences are convergent in Ω

$$\int_\Omega u^\Psi \, d\Omega < \infty \quad \text{and} \quad \int_\Omega \varepsilon^\Psi \, d\Omega < \infty \tag{6}$$

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

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

therefore (7) together with (3) and (4) represents the weak form formulation of the problem.

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 2. 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 displacement field u_p^{ψ}. Therefore, employing the definition of the material integral, where we recover the Riemann integral definition as an addition of a finite set of points, and their volumes are interpreted as quadrature weights. Consequently, individual terms in (7) are solved as follows.

– Acceleration forces :

$$\int_{\Omega} \rho \, \frac{d\mathbf{v}}{dt} \cdot u^{\psi} \, d\Omega = \frac{d\mathbf{v}_p}{dt} \cdot u_p^{\psi} \, m_p. \tag{8}$$

– Internal forces :

$$\int_{\Omega} \sigma : \varepsilon^{\psi} \, d\Omega = \sigma_p : \varepsilon_p^{\psi} \, V_p. \tag{9}$$

– Body forces :

$$\int_{\Omega} \rho \, \mathbf{b} \cdot u^{\psi} \, d\Omega = \mathbf{b}_p \cdot u_p^{\psi} \, m_p. \tag{10}$$

– Loads :

$$\int_{\Gamma_d} \mathbf{t} \, u^{\psi} \, d\Gamma = \int_{\Gamma_d} \rho \, \mathbf{t}^s \cdot u^{\psi} \, d\Gamma = \mathbf{t}_p^s \cdot u_p^{\psi} \, h^{-1} \, m_p, \tag{11}$$

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 displacement field of the particle p as $\mathbf{u}_p = N_{Ip} u_I$, $\mathbf{u}_p^{\psi} = N_{Ip} u_I^{\psi}$, and its gradient as $\varepsilon_p = (u_I \otimes grad(N_{Ip}))^s$, $\varepsilon_p^{\psi} = (u_I^{\psi} \otimes grad(N_{Ip}))^s$. We reach to the nodal balance of forces of the continuum,

$$\dot{\mathbf{p}}_I = \mathsf{m}_{IJ} \dot{\mathbf{v}}_J = \mathbf{f}_I^{int} + \mathbf{f}_I^{ext}, \tag{12}$$

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

$$\mathsf{m}_{IJ} = N_{Ip} m_p N_{Jp}. \tag{13}$$

To improve the computational efficiency and stability, the nodal mass matrix (13) can be substituted by the lumped mass matrix m_{IJ}^{lumped}. Later, internal and external forces are computed as follows,

$$\mathbf{f}_I^{int} = -\sigma_p \cdot grad(N_{Ip}) \frac{m_p}{\rho_p} \tag{14}$$

$$\mathbf{f}_I^{ext} = N_{Ip} \, \mathbf{b}_p \, m_p + N_{Ip} \, \mathbf{t}_p^s \, m_p h^{-1} \tag{15}$$

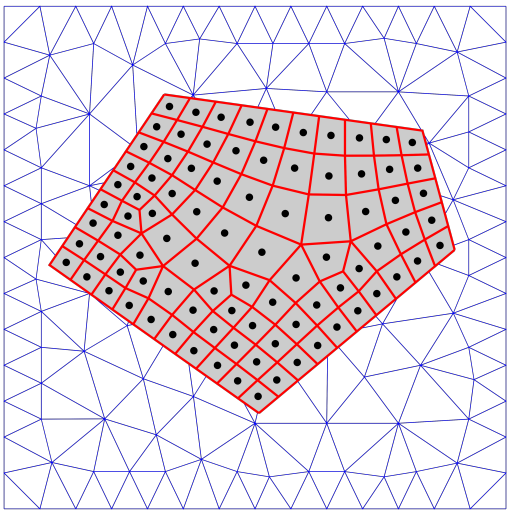


Fig. 2: Description of the spatial discretization for domain presented in the figure 1. Blue mesh represent the background computational support, and the grid mesh conforms the discretized continuum body.

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 (16).

$$\dot{\varepsilon}_p = \frac{\Delta \varepsilon_p}{\Delta t} = \frac{1}{2} \left[grad(N_{Ip}) \otimes \mathbf{v}_I + \mathbf{v}_I \otimes grad(N_{Ip}) \right]. \tag{16}$$

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

$$\dot{\rho} = -\rho \, tra(\dot{\varepsilon}) \tag{17}$$

Finally, to solve the equation (12), 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,

$$\dot{\mathbf{v}}_p = N_{Ip} \, \mathbf{a}_I, \quad and \quad \dot{\mathbf{x}}_p = N_{Ip} \, \mathbf{v}_I \tag{18}$$

In the MPM literature, this equations (12) and (18), are solved with an explicit forward Euler algorithm. The time integration scheme of the MPM has been described in detail by many researchers [18], [3], [1] and summarized in Figure 3. Other authors have proposed many others time integration alternatives like [10], [20], [6]. In the first publication on the MPM [18], the nodal acceleration was employed to update the particles as

$$v_p^{k+1} = v_p^k + \Delta t \, N_{Ip}^k \, \mathbf{a}_I^k \tag{19}$$

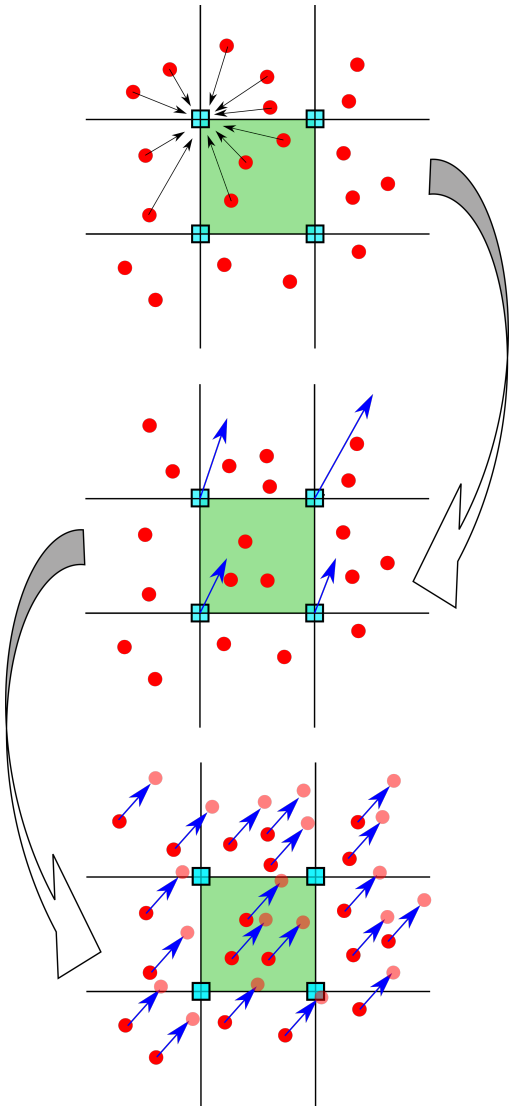


Fig. 3: Description of the MPM standard algorithm.

$$x_p^{k+1} = x_p^k + \Delta t\, N_{Ip}^k\, \mathbf{v}_I^k. \tag{20}$$

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 (20),(19). Hence, a corrected version of this algorithm in shown in Zhang *et al.* (2016)[26]

$$x_p^{k+1} = x_p^k + \Delta t\, \frac{N_{Ip}^k\, \mathbf{p}_I^k}{\mathsf{m}_I}. \tag{21}$$

$$v_p^{k+1} = v_p^k + \Delta t\, \frac{N_{Ip}^k\, \mathbf{f}_I^k}{\mathsf{m}_I}, \tag{22}$$

Later Tran & Solowski (2019)[20] 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.

$$v_p^{k+1} = v_p^k + \Delta t\, N_{Ip}^k \left[(1-\gamma)\, a_I^k + \gamma\, a_I^{k+1}\right], \tag{23}$$

$$x_p^{k+1} = x_p^k + N_{Ip}^k \left[\Delta t\, \mathbf{v}_I^k + \Delta t^2 \left((\frac{1}{2} - \beta)\, \mathbf{a}_I^k + \beta\, \mathbf{a}_I^{k+1}\right)\right] \tag{24}$$

$$a_p^{k+1} = N_{Ip}^k\, \mathbf{a}_I^{k+1}. \tag{25}$$

This scheme has prof to damps out the higher frequency noises [20]. But it can present the same numerical instabilities as in (20),(19) when nodal masses become infinitesimal, and requires extra storage for nodal values of acceleration and previous steps.

4 Explicit predictor-corrector scheme for MPM.

In this section, a explicit predictor-corrector time integration scheme is presented. It is based in the Newmark a-form $\gamma = 0.5$ and $\beta = 0$ which is the central difference explicit. This method is devoted to solve a system of equations of the type

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

As the MPM has a nodal stage, it is possible to apply this methods successfully in the MPM framework as was proved by [20]. Taking the predictor definition from and calculating nodal velocity, and updating particles position employing nodal values of velocity and acceleration. The predictor-corrector algorithm has been described in the classic literature [11], and its computational advantages and stability were widely proof by Liu [15]. The "classic" PCE algorithm starts with a predicted value of the nodal velocity at the $(k+1)$th time step denoted by $\tilde{\mathbf{v}}_I^{k+1}$ given as,

$$\tilde{\mathbf{v}}_I^{k+1} = \mathbf{v}_I^k + (1-\gamma)\, \Delta t\, \mathbf{a}_I^k \tag{26}$$

In (26) arise a *user-defined* parameter $\gamma \geq 0$. This parameter influences both the predictor accuracy and the stability of the algorithm. As point out Liu [15], the truncation error of the predictor formula is $O(\Delta t^3)$ when $\gamma = 0.5$, and is unconditionally stable if $0 < \gamma \leq 0.25$.

To accommodate this step to the MPM framework, it is necessary to get the nodal values of the velocity and acceleration throughout a variational recovery process where particles quantities are transferred to the mesh nodes. This technique arise as a generalization of the super-convergent

recovery procedures described by Zienkiewicz & Zhu [27] (*ZZ*) in the context of FEM. In the MPM Gauss quadrature is not employed as integrals are computed following the Riemann integral definition, where each component of the summation correspond to a particle of the discretization. Also Bardenhagen & Kober [4] proved that thorough this information-transference technique mass and momentum are conserved. So for a general particle variable Φ_p employing the *ZZ* technique is possible to get its nodal homologous Φ_I as,

$$\Phi_I = \frac{m_p N_{Ip} \Phi_p}{m_I} \tag{27}$$

Therefore, to get a analogous expression for (26) in the context of the MPM, the procedure described in the equation (27) is employed reaching to,

$$\tilde{\mathbf{v}}_I^{k+1} = \underbrace{\frac{N_{Ip}^k m_p \mathbf{v}_p^k}{m_I}}_{\mathbf{v}_I^k} + (1-\gamma)\, \Delta t\, \underbrace{\frac{N_{Ip}^k m_p \mathbf{a}_p^k}{m_I}}_{\mathbf{a}_I^k} \tag{28}$$

Nonetheless this way of computing the predictor stage can introduce instabilities due to numerical cancellation likewise the original Sulky algorithm. Thankfully this can be avoided easily by the equivalent formulation (29),

$$\tilde{\mathbf{v}}_I^{k+1} = \frac{N_{Ip}^k m_p (\mathbf{v}_p^k + (1-\gamma)\, \Delta t\, \mathbf{a}_p^k)}{m_I} \tag{29}$$

This way of computing the nodal predictor is both numerically stable and minimize the computational effort. Once nodal velocity are obtained, the essential boundary conditions are imposed. And in the following, the "classic" MPM algorithm continues to reach to the equilibrium equation (12). Here we continue with the *corrector* stage, due to the fact that we already have nodal velocity, this step is computed in the same way as in FEM,

$$\mathbf{v}_I^{k+1} = \mathbf{v}_I^{pred} + \gamma\, \Delta t\, \frac{\mathbf{f}_I^{k+1}}{\mathsf{m}_I^{k+1}} \tag{30}$$

Finally updated particle kinetics are computed using nodal values as,

$$\mathbf{a}_p^{k+1} = \mathbf{a}_p^n + \frac{N_{Ip}^k \mathbf{f}_I^k}{\mathsf{m}_I^k} \tag{31}$$

$$\mathbf{v}_p^{k+1} = \mathbf{v}_p^n + \Delta t\, \frac{N_{Ip}^k\, \mathbf{f}_I^k}{\mathsf{m}_I^k} \tag{32}$$

$$\mathbf{x}_p^{k+1} = \mathbf{x}_p^n + \Delta t\, N_{Ip}^k\, \mathbf{v}_I^k + \frac{1}{2}\Delta t^2\, \frac{N_{Ip}^k\, \mathbf{f}_I^k}{\mathsf{m}_I^k} \tag{33}$$

Notice that particle displacement is computed using the corrected nodal velocities and the accelerations computed with the velocities of the predictor. However, particles velocities and accelerations are computed using the corrected velocities. 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. Notice also that with this approach the calculation of nodal momentum values are not required. Due to its simplicity allows be implemented with minor modifications over a standard forward Euler. It is summarized in shape of pseudo-algorithm A.

5 Local max-ent approximants

The popularity of the MPM has increase notoriously during the recent years due to its ability to deal with large strain problems without mesh distorsion issues inherent to mesh based methods like FEM, see Zdzislaw [24]. However, in the simulations made with the original MPM, there are numerical noises when particles crossing the cell boundaries. Local maximum-entropy (or local *max-ent*) shape function first introduced by Arroyo & Ortiz (2006)[2] has been recently tested under the MPM framework by Wobbes *et al.* (2020)[23] where they prof that simulations performed with the *max-ent* basis functions show considerably more accurate stress approximations for MPM. Although, in [23] authors does not deep in how the regularization parameter β affects to the accuracy and stability of the solution.

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. This approximation scheme represents a optimal compromise, in the sense of Pareto, between the *unbiased statistical inference* based on the nodal data which leads to the principle of *maximum-entropy* stated by Jaynes [12], and the definition of local shape functions of *least width* the least biased shape functions.

Taking the definition of entropy as a measure of how uncertainty a random variable is averaged on all its possible outcomes. And adopting the Shannon's entropy as the starting point:

$$H(p_1(\mathbf{x}), \ldots, p_n(\mathbf{x})) = -\sum_{I=1}^{N_n} p_I(\mathbf{x}) \log p_I \tag{34}$$

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

$$\text{(LME) Maximize } H(p) = -\sum_{I}^{N_n} p_I(\mathbf{x}) \log p_I$$

$$\text{subject to } \begin{cases} p_I \geq 0, \ \text{I=1, ..., n} \\ \sum_{I=1}^{N_n} p_I = 1 \\ \sum_{I=1}^{N_n} p_I \mathbf{x}_I = \mathbf{x} \end{cases}$$

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

$$\text{(LME)}_\beta \text{ For fixed } \mathbf{x} \text{ minimise } f_\beta(\mathbf{x}, p) = \beta H(\mathbf{x}, p) - H(p)$$

$$\text{subject to } \begin{cases} p_I \geq 0, \ \text{I=1, ..., n} \\ \sum_{I=1}^{N_n} p_I = 1 \\ \sum_{I=1}^{N_n} p_I \mathbf{x}_I = \mathbf{x} \end{cases}$$

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

$$p(\mathbf{x}) = \frac{\exp\left[-\beta \, |\mathbf{x} - \mathbf{x}_I|^2 + \lambda \cdot (\mathbf{x} - \mathbf{x}_I)\right]}{Z(\mathbf{x}, \lambda^*(\mathbf{x}))} \tag{35}$$

where

$$Z(\mathbf{x}, \lambda) = \sum_{I=1}^{N_n} \exp\left[-\beta \, |\mathbf{x} - \mathbf{x}_I|^2 + \lambda \cdot (\mathbf{x} - \mathbf{x}_I)\right] \tag{36}$$

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^*

$$\nabla p_I^* = p_I^* \left(\nabla f_I^* - \sum_{J}^{N_n} p_J^* \nabla f_J^* \right) \tag{37}$$

where

$$f_I^*(\mathbf{x}, \lambda, \beta) = -\beta \, |\mathbf{x} - \mathbf{x}_I|^2 + \lambda \, (\mathbf{x} - \mathbf{x}_I) \tag{38}$$

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

$$\nabla p_I^* = -p_I^* \, (\mathsf{J}^*)^{-1} \, (\mathbf{x} - \mathbf{x}_I) \tag{39}$$

where J is the Hessian matrix, defined by:

$$\mathsf{J}(\mathbf{x}, \lambda, \beta) = \frac{\partial \mathbf{r}}{\partial \lambda} \tag{40}$$

$$\mathbf{r}(\mathbf{x}, \lambda, \beta) \equiv \partial_\lambda \log Z(\mathbf{x}, \lambda) = \sum_{I}^{N_n} p_I(\mathbf{x}, \lambda, \beta) \, (\mathbf{x} - \mathbf{x}_I) \tag{41}$$

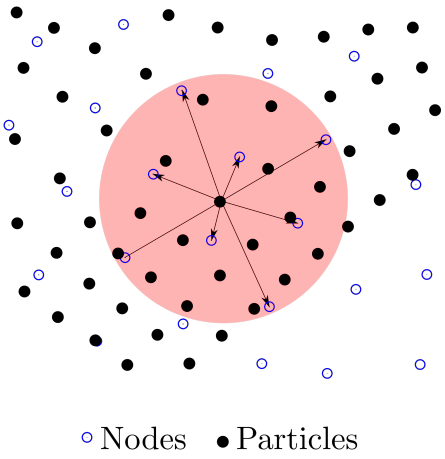


Fig. 4: This tolerance defines the limit values of the influence radius and is used thereafter to find the neighbour nodes of a given integration point.

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. (40) 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. The support size of LME shape functions may be controlled by adjusting a dimensionless parameter, $\gamma = \beta h^2$ [2] (e.g. two dimensional example shown in figure. 5). 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. Regarding this aspect, an crucial consideration for the implementation of the LME shape functions in the MPM is exposed. To avoid the transference of particle information to those nodes fare apart from the particle domain Ω_p, a especial treatment of the neighbour nodes for those particles close to the boundary is required. Here we employ an algorithm inspired in the one proposed by Kumar *et al.* (2019)[13] to deal with interfaces between two materials in the OTM framework. It consists in select only those nodes

employed to compute the solution are those whose position are inside In opposite to the OTM where this approximation technique has been widely. 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. Furthermore, with a conveniently adopted *regularization* parameter it is possible to get a GIMP-like shape function. A proof of this statements is observed in 5.

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 Application to linear elasticity dynamic problems.

This section is devoted to test the ability of both predictor-corrector time integration scheme and the local *max-ent* approximants to overcome spurious oscillations due to the grid crossing and high frequency loads under the context of the MPM.

6.1 Dyka bar

High frequency , the benchmark proposed by Dyka & Ingel (1995)[9] is considered to illustrate the capacity of the proposed time integration algorithm to avoid velocity fields instabilities.

In the one-dimensional bar sketched in the figure 6, 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 $kg/m3$
- Poisson ratio : 0
- Elastic modulus : $200 \cdot 10^9\ Pa$

The boundary conditions are:

$$\sigma|_{x=L} = 0 \quad , \quad v|_{x=0} = 0 \tag{42}$$

The total simulation time is 0.0002 seconds, a direct consequence of that is we let the elastic wave generated travel thorough the bar (from A to C and back to A) at least four times. For the spatial discretization we will set a mesh size of 3.3325 millimeters. The initial mesh will be composed by a single particle in the center of the element.

Regarding this, here we give one consideration to properly modellize this benchmark under the MPM framework. Notice that the free border of the bar has a horizontal displacement of 0.033 millimeters, therefore a computational domain with an extra gap of 0.033 millimeters is required in order to accommodate the unconstrained displacement of the particles in the left border of the bar. Here below the time step is controlled by the courant number, in this case, we adopt a middle value of it (CFL=0.5) as a compromise between stability and numerical accuracy.

The analytical solution in terms of velocity 15, and stress 16 can be found in the appendix B given by the Method of characteristics.

For the convergence analysis, the root-mean-square (RMS) error in the velocity field is computed. RMS error is defined as follows

$$RMS = \sqrt{\frac{1}{N}\sum_{p}^{N}(v_p - \hat{v}_p)^2}, \tag{43}$$

where v_p and $\hat{v}_p$ are respectively the analytical and numerical solutions evaluated in the final time step in the position of each particle.

6.2 Andersen block

In order to test the ability of this interpolation technique to deal with grid crossing instabilities we will simulate the vertical compression of a square block (10 by 10 meters) of soft soil sketched in figure 6.2 and loaded using an incremental gravitation scheme.

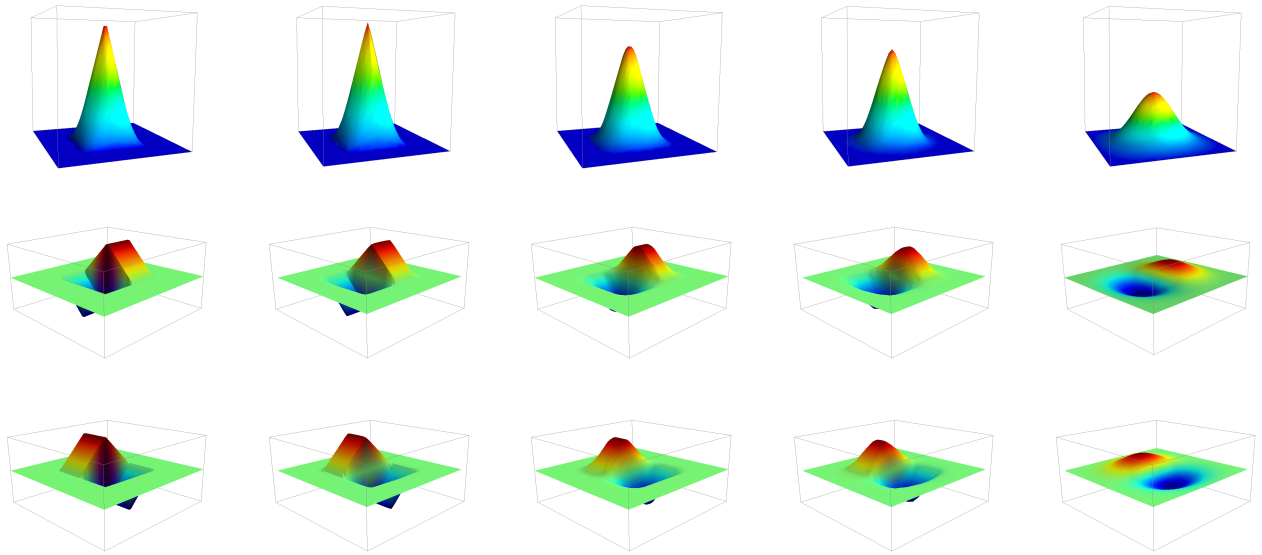
This test was taken from Andersen (2009)[1]. The elastic parameters consider for this test are:

- Initial density : $6 \cdot 10^3\ kg/m^3$
- Poisson ratio : 0
- Elastic modulus : $5\ MPa$

The gravity force is a apply as an external force according to the equations (10), (15). Using a total time period of T (20 seconds) to apply the gravity, it is increased from 0 to $9.81 m/s$ with a sinus function until T/2 and then maintained constant until T in order to arrive to a state of equilibrium,

$$\mathbf{g}(t) = \begin{cases} 0.5\mathbf{g}(\sin(\frac{2t\pi}{T} - \frac{\pi}{2}) + 1) & \text{if } t \leq T/2 \\ \mathbf{g} & \text{if } t > T/2 \end{cases} \tag{44}$$

In order to get a stable solution, we will adopt a time step conducted by a Courant number of 0.1. On the other hand, the explicit predictor-corrector scheme is here employed looking forward getting better results. For the initial spatial discretization we will employ four particles per cell ($\Delta x = 2\ m$). The initial setting of particles inside of the cell changes according to the interpolation technique adopted. For the bi-linear shape functions (Q4) and the LME shape functions, the initial position corresponds to the location of the gauss-points in a standard quadratic finite element. For the uniform-GIMP (uGIMP) shape function the initial position of each particle is located in the center of each voxel, due to the fact that in the initial situation, the voxel domain should not overlap each others, Steffen *et al.* (2008)[17].



(a) Q4 (b) LME_{17} (c) uGIMP (d) LME_{10} (e) LME_5

Fig. 5: Comparative of linear piecewise shape functions (Q4) and uniform GIMP shape functions (uGIMP) *versus* Local max-ent shape functions for a two-dimensional arrangement of nodes, and spatial derivatives for several values of $\gamma = \beta/h^2$.

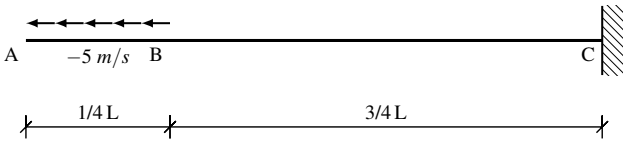


Fig. 6: Geometrical description of the Dyka [9] bar.

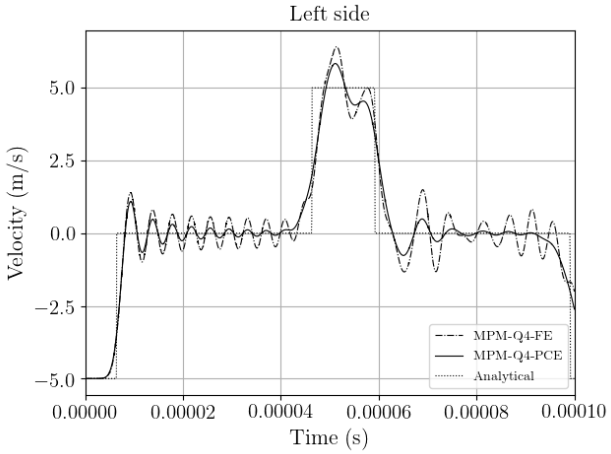


Fig. 7: Velocity evolution at the point in the bar left side.

Fig. 8: Velocity evolution at the point in the bar left side.

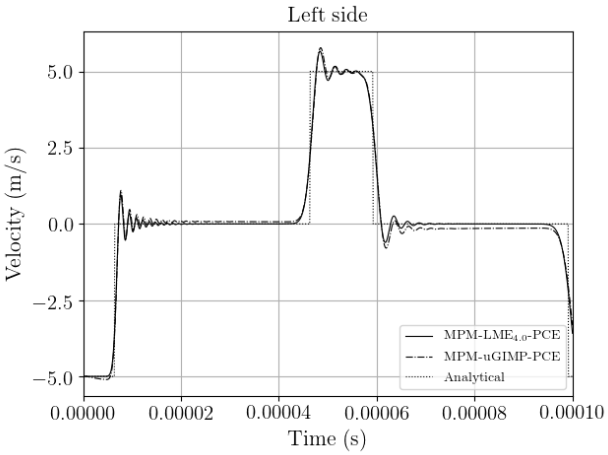


Fig. 9: Velocity evolution at the point in the bar left side.

Figure 13 shows the vertical displacement evolution of a point in the free surface of the block.

As we can see the simulation using the Q4 interpolation technique turns out to be unstable and in the second 15 it fails. The uGIMP simulation is more stable than the one performed by the Q4. Despite this is still unstable and could trigger severe oscillations if we pretend to simulate non-linear materials. The LME simulation was performed using two kinds of shape functions, one with a low value of the dimensionless parameter, $\gamma = 0.8$, and other with a larger value of it, $\gamma = 3.0$. Notice that the results are both stable, but the larger values of γ give us a very stable solution. This

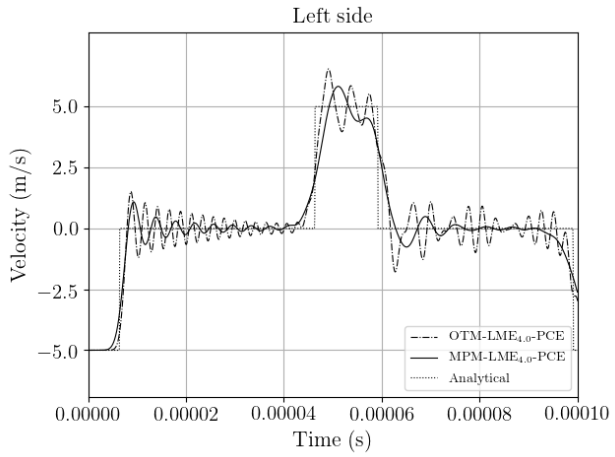


Fig. 10: Velocity evolution at the point in the bar left side.

is due to the fact that with larger value of γ, the shape functions behaves in a similar way to the FEM, which performs very accurate in those cases with a reasonable mesh distorsion, and with a lower value it behaves in a similar way to the uGIMP. This behaviour was noticed previously by [2], were authors highlight how by adjusting the spatial variation of $\beta(x)$, it is possible to select regions of the domain of analysis which are treated by finite elements and regions that are treated in the style of meshfree methods, with seamless transitions between those regions.

Figure 14 shows the evolution of the vertical stress during the loading process. The result is physically realistic as stress increments linearly from the top to the bottom of the specimen, and the value of the vertical stress in a material point located in the bottom of the specimen oscillates centered in $5.2MPa$, which is the analytic value given by $\sigma_{yy} = \rho g h_y$.

7 Conclusions

We have developed a new time integration scheme for the MPM, and proved how local *max-ent* arise as a promising tool to overcome the grid-crossing error.

adapt the value of β to solve the equations fem-like of meshfree-like depending of how behaves the region

This alternative technique overcome the constriction of the GIMP to regular mesh.

The time integration here proposed mitigates the presence of wiggles and overshot

future research : localization of the local max ent

predictor-corrector for high dynamic problems like impact or collisions

Acknowledgements If you'd like to thank anyone, place your comments here and remove the percent signs.

Conflict of interest

The authors declare that they have no conflict of interest.

A Explicit predictor-corrector algorithm

B 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,

$$\rho \frac{\partial v}{\partial t} = \frac{\partial \sigma}{\partial x}, \tag{45}$$

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

$$\frac{\partial \sigma}{\partial t} = E \frac{\partial \varepsilon}{\partial t}, \tag{46}$$

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

$$\frac{\partial \varepsilon}{\partial t} = \frac{\partial v}{\partial x}. \tag{47}$$

Next for simplicity, we will introduce (47) in (46), so we get the following system of equations,

$$\frac{\partial v}{\partial t} = \frac{1}{\rho} \frac{\partial \sigma}{\partial x}, \tag{48}$$

$$\frac{\partial \sigma}{\partial t} = E \frac{\partial v}{\partial x}. \tag{49}$$

Introducing (49) in (48) and expressing the remaining equation in terms of the displacement, we reach the 1D wave equation for linear elastic materials,

$$\frac{\partial^2 u}{\partial t^2} = \frac{E}{\rho} \frac{\partial^2 u}{\partial x^2} = c^2 \frac{\partial^2 u}{\partial x^2} \tag{50}$$

where we have introduced the wave celerity c as,

$$c = \sqrt{\frac{E}{\rho}} \tag{51}$$

Alternative, rearranging both equations (48) and (49) it is possible to join them in a single system of equations as,

$$\frac{\partial}{\partial t} \begin{bmatrix} \sigma \\ v \end{bmatrix} + \begin{bmatrix} 0 & -E \\ -1/\rho & 0 \end{bmatrix} \begin{bmatrix} \frac{\partial \sigma}{\partial x} \\ \frac{\partial v}{\partial x} \end{bmatrix} = \mathbf{0}. \tag{52}$$

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

$$\frac{\partial \phi}{\partial t} + \mathbf{A} \frac{\partial \phi}{\partial x} = \mathbf{0} \tag{53}$$

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

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

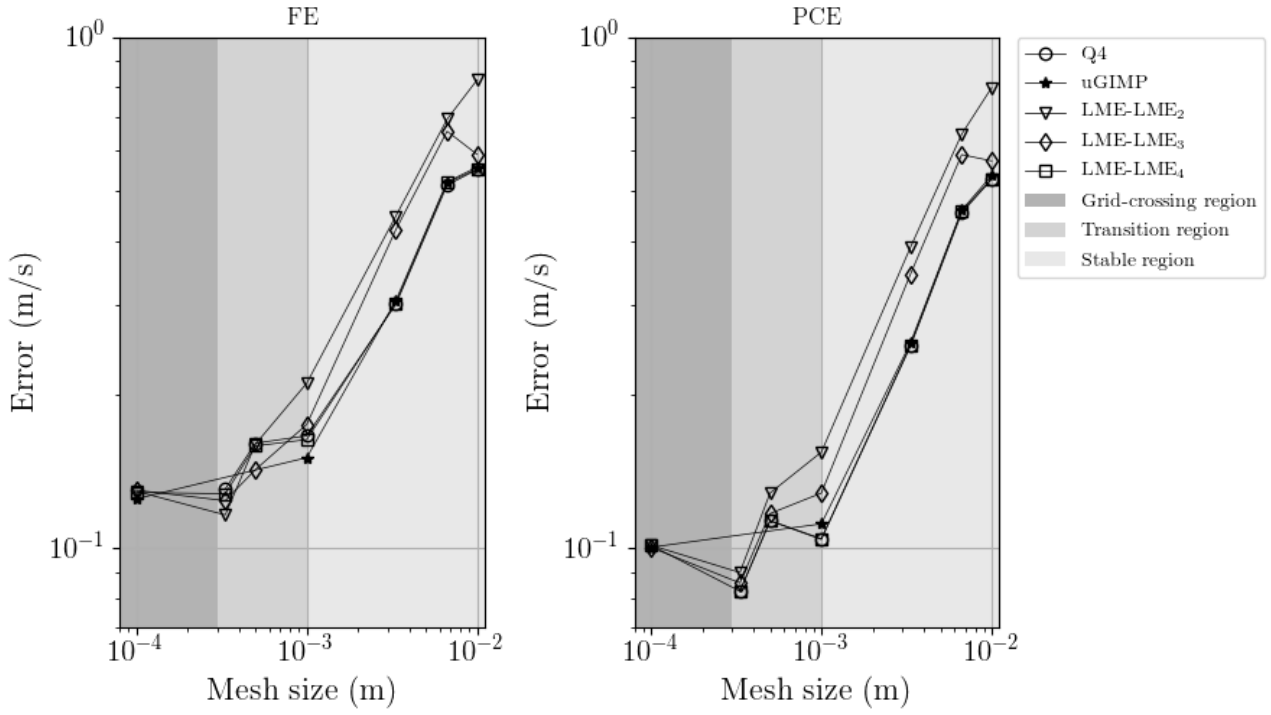


Fig. 11: Velocity evolution at the point in the bar left side.

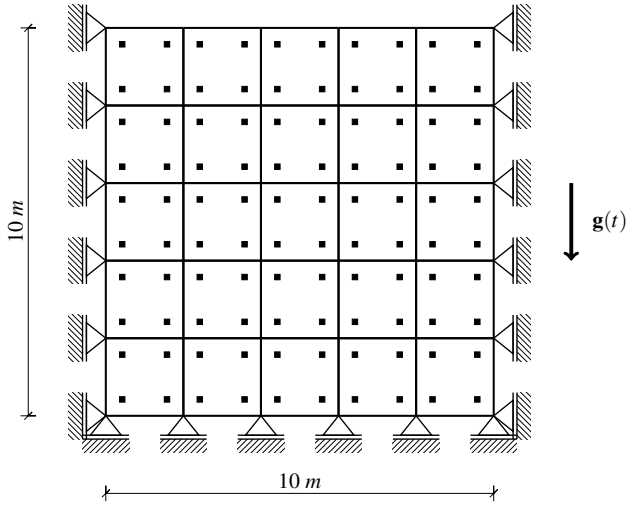


Fig. 12: Geometrical description of a soil block

.

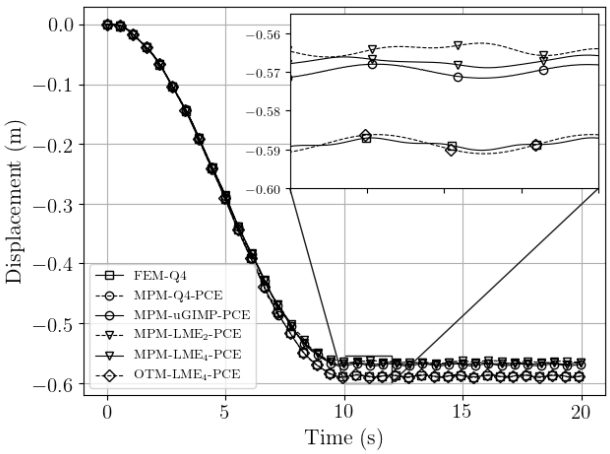


Fig. 13: Comparative of the vertical displacement evolution in a point located in the free surface employing different interpolation schemes and numerical techniques.

Note that the nature of is still hyperbolic despite the fact it does not have a second order temporal derivative as (50). A proof of this can be easily obtained if we get the zeros of the hypersurface defined by (50). And later the eigenvalues of $\mathbf{A}$ in (53). 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 $\mathbf{x}$

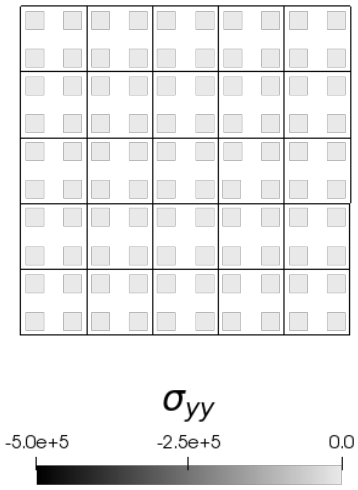
$$\mathbf{P} = \{\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3, \ldots \mathbf{x}^n\}. \tag{54}$$

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

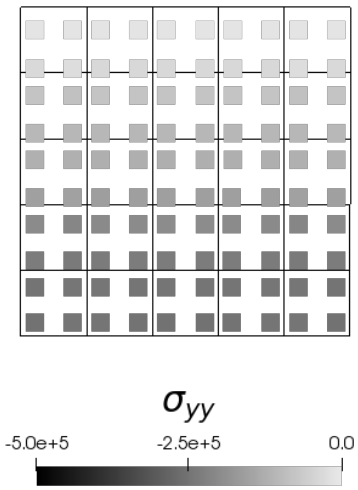
$$\Lambda = \mathbf{P}^{-1}\mathbf{A}\,\mathbf{P}, \tag{55}$$

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

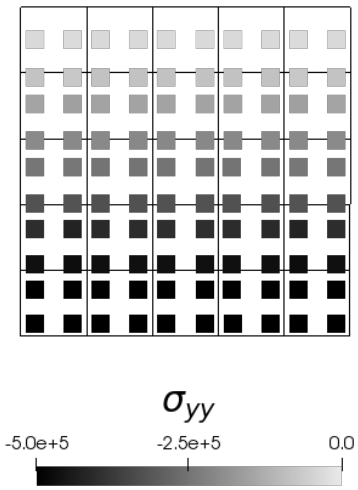
$$\phi = \mathbf{P}\,\Re \tag{56}$$



((a)) t = 0 seconds.



((b)) t = 5 seconds.



Algorithm Explicit Predictor-Corrector scheme

1: **Update mass matrix**:

$$\mathsf{m}_I = N_{Ip}^k \, m_p,$$

2: **Explicit Newmark Predictor**:

$$\mathbf{v}_I^{pred} = \frac{N_{Ip}^k m_p (\mathbf{v}_p^k + (1-\gamma)\,\Delta t\,\mathbf{a}_p^k)}{m_I}$$

3: **Impose essential boundary conditions**:
At the fixed boundary, set $\mathbf{v}_I^{pred} = 0$.

4: **Deformation tensor increment calculation**.

$$\dot{\varepsilon_p}^{k+1} = \left[\mathbf{v}_I^{pred} \otimes grad(N_{Ip}^{k+1})\right]^s$$
$$\Delta\varepsilon_p^{k+1} = \Delta t\ \dot{\varepsilon_p}^{k+1}$$

5: **Update the density field**:

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

6: **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 (14) and (15) 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.

7: **Explicit Newmark Corrector**:

$$\mathbf{v}_I^{k+1} = \mathbf{v}_I^{pred} + \gamma\,\Delta t\,\frac{\mathbf{f}_I^{k+1}}{\mathsf{m}_I^{k+1}}$$

8: **Update particles lagrangian quantities**:

$$\mathbf{a}_p^{k+1} = \mathbf{a}_p^n + \frac{N_{Ip}^k \mathbf{f}_I^k}{\mathsf{m}_I^k}$$
$$\mathbf{v}_p^{k+1} = \mathbf{v}_p^n + \Delta t\,\frac{N_{Ip}^k\,\mathbf{f}_I^k}{\mathsf{m}_I^k}$$
$$\mathbf{x}_p^{k+1} = \mathbf{x}_p^n + \Delta t\,N_{Ip}^k\,\mathbf{v}_I^k + \frac{1}{2}\Delta t^2\,\frac{N_{Ip}^k\,\mathbf{f}_I^k}{\mathsf{m}_I^k}$$

9: **Reset nodal values**

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,

$$d\mathfrak{R} = \frac{\partial \mathfrak{R}}{\partial t}dt + \frac{\partial \mathfrak{R}}{\partial x}dx = \mathsf{P}^{-1}\left(\frac{\partial \phi}{\partial t}dt + \frac{\partial \phi}{\partial x}dx\right) \tag{57}$$

and setting the terms we get,

$$\frac{\partial \mathfrak{R}}{\partial t} = \mathbf{P}^{-1}\frac{\partial \phi}{\partial t}, \quad \frac{\partial \mathfrak{R}}{\partial x} = \mathbf{P}^{-1}\frac{\partial \phi}{\partial x} \tag{58}$$

Next, if we multiply (53) by $\mathbf{P}^{-1}$ we get:

$$\mathbf{P}^{-1}\frac{\partial \phi}{\partial t} + \left(\mathbf{P}^{-1}\mathbf{A}\mathbf{P}\right)\mathbf{P}^{-1}\frac{\partial \phi}{\partial x} = \mathbf{0} \tag{59}$$

finally introducing the expressions (58) we reach to

$$\frac{\partial \mathfrak{R}}{\partial t} + \Lambda\frac{\partial \mathfrak{R}}{\partial x} = \mathbf{0} \tag{60}$$

which consists of n uncoupled equations as Λ is diagonal matrix as we can see in (55). Each of this equations are 1D scalar convective transport equations, with solutions of the form:

$$\mathfrak{R}^{(i)} = F^{(i)}\left(x - \lambda^{(i)}t\right) \tag{61}$$

This uncoupled system, has, therefore, a set of n characteristics. These magnitudes $\mathfrak{R}_i$ 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 $\mathfrak{R}_i(x,t=0) = h_i(x)$, 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 [9], 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:

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

And introducing the value of the inverse matrix $\mathbf{P}^1$ in the Riemann definition (56) we get the following system of equations,

$$\mathfrak{R}^I = \frac{1}{2\sqrt{\rho E}}\left(-\sigma + v\sqrt{\rho E}\right) \tag{62}$$

$$\mathfrak{R}^{II} = \frac{1}{2\sqrt{\rho E}}\left(\sigma + v\sqrt{\rho E}\right) \tag{63}$$

From (62) and (63) we can obtain the values of the stress and the velocity as:

$$v = \mathfrak{R}^I + \mathfrak{R}^{II} \quad , \quad \sigma = \sqrt{E\rho}\left(\mathfrak{R}^{II} - \mathfrak{R}^I\right) \tag{64}$$

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

$$\mathfrak{R}^{II} = 0 \quad and \quad v_{x=L} = 0$$

Therefore $\sigma_{x=L} = -2\sqrt{\rho E}\,\mathfrak{R}^I$. And in the left side (free boundary) we get the following conditions:

$$\mathfrak{R}^I = 0 \quad and \quad \sigma_{x=0} = 0$$

Therefore $v_{x=0} = 2\mathfrak{R}^{II}$. Finally, applying this conditions in the elastic bar sketched in 6, is possible to obtain the velocity history in the right side of the bar 15 and the stress in the last quarter side of the Dyka bar 16 as is demanded in [9].

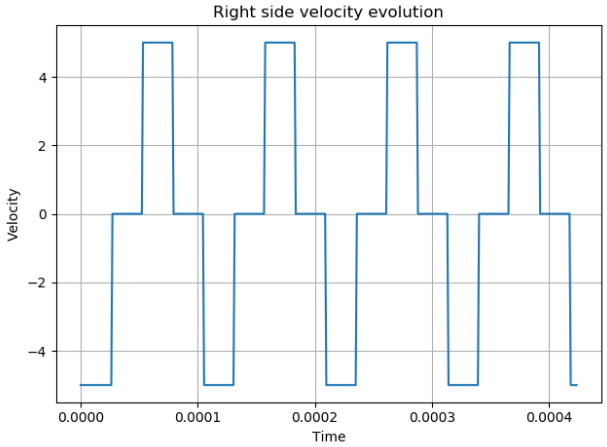


Fig. 15: Analytical solution for the velocity in the right side of the Dyka bar.

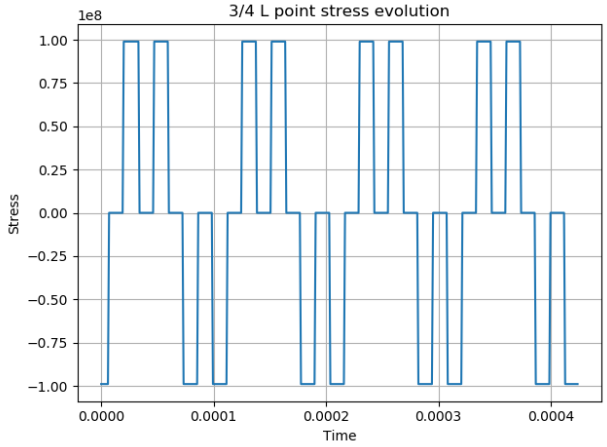


Fig. 16: Analytical solution for the stress in the last quarter of the Dyka bar.

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
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 defor-

mations. 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
8. Dhakal, T.R., Zhang, D.Z.: Material point methods applied to one-dimensional shock waves and dual domain material point method with sub-points. Journal of Computational Physics **325**, 301 – 313 (2016). DOI https://doi.org/10.1016/j.jcp.2016.08.033. URL http://www.sciencedirect.com/science/article/pii/S0021999116303904
9. 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
10. 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)
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. Liu, X.: On the stability of a newmark's scheme-based predictor-corrector algorithm. Computers Structures - COMPUT STRUCT **53**, 27–33 (1994). DOI 10.1016/0045-7949(94)90126-0
16. 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
17. Steffen, M., Kirby, R.M., Berzins, M.: Analysis and reduction of quadrature errors in the material point method (MPM). International Journal for Numerical Methods in Engineering **76**(6), 922–948 (2008). DOI 10.1002/nme.2360. URL http://doi.wiley.com/10.1002/nme.2360
18. 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
19. Tielen, R., Wobbes, E., Mller, M., Beuth, L.: A high order material point method. Procedia Engineering **175**, 265 – 272 (2017). DOI https://doi.org/10.1016/j.proeng.2017.01.022. URL http://www.sciencedirect.com/science/article/pii/S187770581730022X. Proceedings of the 1st International Conference on the Material Point Method (MPM 2017)
20. 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
21. Wilson, E.L., Farhoomand, I., Bathe, K.J.: Nonlinear dynamic analysis of complex structures. Earthquake Engineering & Structural Dynamics **1**(3), 241–252 (1972). DOI 10.1002/eqe.4290010305. URL http://doi.wiley.com/10.1002/eqe.4290010305
22. 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
23. 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
24. Zdzislaw, W.: The material point method in large strain engineering problems. Computer Methods in Applied Mechanics and Engineering **193**(39-41 SPEC. ISS.), 4417–4438 (2004). DOI 10.1016/j.cma.2004.01.035
25. 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
26. 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
27. Zienkiewicz, O.C., Zhu, J.Z.: The superconvergent patch recovery and a posteriori error estimates. part 1: The recovery technique. International Journal for Numerical Methods in Engineering **33**(7), 1331–1364 (1992). DOI 10.1002/nme.1620330702