



An efficient and stabilised SPH method for large strain metal plastic deformations

Giorgio Greto¹ · Sivakumar Kulasegaram¹

Received: 27 April 2019 / Revised: 31 July 2019 / Accepted: 12 August 2019
© The Author(s) 2019

Abstract

Due to its simplicity and robustness, smooth particle hydrodynamics (SPH) has been widely used in the modelling of solid and fluid mechanics problems. Through the years, various formulations and stabilisation techniques have been adopted to enhance it. Recently, the authors developed JST–SPH, a mixed formulation based on the SPH method. Originally devised for modelling (nearly) incompressible hyperelasticity, the JST–SPH formulation is mixed in the sense that linear momentum and a number of strain definitions, instead of the displacements, act as main unknowns of the problem. The resulting governing system of conservation laws conveniently enables the application of the Jameson–Schmidt–Turkel (JST) artificial dissipation term, commonly employed in computational fluid dynamics, to solid mechanics. Coupled with meshless SPH discretisation, this novel scheme eliminates the shortcomings encountered when implementing fast dynamics explicit codes using traditional mesh-based methods. This paper focuses on the applicability of the JST–SPH mixed formulation to the simulation of high-rate, large metal elastic–plastic deformations. Three applications—including the simulation of an industry-relevant metal forming process—are examined under different loading conditions, in order to demonstrate the reliability of the method. Results compare favourably with both data from the previous literature, and simulations performed with a commercial finite elements package. Most noticeably, these results demonstrate that the total Lagrangian framework of JST–SPH, fundamental to reduce the computational effort associated with the scheme, retains its accuracy in the presence of large distortions. Moreover, an algorithmic flow chart is included at the end of this document, to facilitate the computer implementation of the scheme.

Keywords Metal plasticity · Equal channel angular extrusion (ECAE) · Solid dynamics · Mixed formulations · SPH · JST

1 Introduction

Displacement-based explicit dynamic codes, implemented using low-order finite element methods, are commonly used for advanced numerical simulations in aerospace, automotive, biomedical, defence and manufacturing applications. However, difficulties arise in modelling high-speed impacts or large material deformations, often leading to simulation failure. In these scenarios, most computational codes prefer to employ the 8-noded underintegrated hexahedral element to model solid components [5]. Nonetheless, for many practical applications (e. g. crashworthiness, ballistics and metal forming), the extremely large deformations still result in severe

mesh distortions. These lead to irregular and entangled elements, unless some form of adaptive remeshing is employed.

In addition, difficulties currently encountered by the widely employed, displacement-based, low-order finite element analysis include lower order of convergence for relevant field variables; excessive element distortion under large deformations, requiring periodic remeshing; locking behaviour in bending scenarios; non-physical pressure instabilities, and high-frequency noise due to Newmark-type time integrators.

Recent developments in computational methods for fast solid dynamics [1,2,7–9,18,19,21,23–25] recommend the representation of motion and deformation of a given body via a system of first-order, mixed formulation conservation laws. The partial differential equations (PDEs) that form this system do not present the displacement as the main unknown to be evaluated, instead yielding a set of other relevant quantities (i. e. velocity, deformation gradient, its cofactor matrix and scalar Jacobian), depending on the number of conser-

✉ Giorgio Greto
gretog@cardiff.ac.uk

¹ School of Engineering, Cardiff University, Queen's Buildings, The Parade, Cardiff CF24 3AA, UK

vation equations being considered. This choice of the field variables will be driven on the one hand, by the need to reproduce certain features of material behaviour (e. g. near or full incompressibility) while ensuring polyconvexity of the strain energy function and, on the other hand, by the computational efficiency of the resulting scheme.

Mixed conservation laws have already been proven [1, 21, 25] to achieve second order of accuracy for stresses and strains. The two-field mixed formulation, composed of linear momentum \vec{p} and deformation gradient \mathbf{F} , was later augmented by including a governing equation to conserve the Jacobian J of the deformation gradient [18]. This enabled the scheme to effectively solve nearly and fully incompressible materials. Further enhancement of this $\{\vec{p}, \mathbf{F}, J\}$ framework has also been recently reported in [7, 19] for modelling compressible, nearly incompressible and fully incompressible materials governed by a polyconvex constitutive law. In particular, this requires the presence of the matrix of co-factors of the deformation gradient \mathbf{F} (denoted here by \mathbf{H}), leading to an extended system of field variables. The $\{\vec{p}, \mathbf{F}, \mathbf{H}, J\}$ system can be reformulated to an alternative description in terms of entropy conjugates, as in [7, 19], due to the existence of a generalised convex entropy function. This readily facilitates the proof of existence of associated real wave speeds, which in turn establishes existence and uniqueness of real solutions.

A novel computational framework was devised to improve the numerical simulation behaviour during large material distortions, and to address the other numerical issues (e. g. locking, spurious oscillations) highlighted earlier. In a total Lagrangian perspective, the proposed methodology combines the use of smooth particle hydrodynamics (SPH), a meshless spatial discretisation technique, with an explicit, two-stage, total variation diminishing (TVD) Runge–Kutta temporal scheme.

To date, the application of mixed formulation has been largely focused on nearly incompressible hyperelasticity [1, 2, 21, 25], where many of the aforementioned shortcomings are encountered while using linear finite element methods. The work presented in this paper aims to investigate the feasibility of using the developed mixed formulation technique in the field of metal plasticity, by exploring some applications involving large material deformations. These will include a high-velocity impact scenario, where material is subjected to a non-uniform deformation and thus develops a large geometric discontinuity; a constrained boundary problem under heavy distortion, and the simulation of severe plastic deformation in a metallurgical process of industrial relevance.

This paper is organised as follows: a description of the mixed hyperbolic system of governing partial differential equations is presented in Sect. 2, while the adopted SPH discretisation methodology, that incorporates a stabilisation procedure based on a Jameson–Schmidt–Turkel (JST) artificial dissipation term, widely used in the field of com-

putational fluid dynamics (CFD), will be detailed in Sect. 3. Numerical applications in metal plasticity are presented in Sect. 4, along with an assessment of the accuracy and stability of the methodology. These numerical examples are mainly chosen to demonstrate the potential and the accuracy of the developed computational strategy. Concluding remarks are then presented in Sect. 5. Finally, for the purpose of completeness, a brief outline of the chosen elasto-plastic constitutive model, including relevant computational procedures for evaluating the material deformation, is provided in “Appendix A”.

2 Conservation laws

In this paper, a vector will be expressed with the notation \vec{a} , and a tensor with \mathbf{A} . The material and spatial coordinates are denoted by \vec{X} and \vec{x} , respectively. The gradient operator that refers to initial spatial coordinates will be denoted by $\vec{\nabla}_0$. The symbol \mathbf{x} represents the tensorial cross product, a key operation utilised in obtaining a concise mathematical representation of the proposed methodology. This notation was introduced in a series of papers devoted to the development of mixed formulation techniques [7–9, 19]. More details on the properties of the tensorial cross product can be found in the cited references.

The system of conservation laws describing the motion and deformation processes in terms of field variables \vec{p} , \mathbf{F} , \mathbf{H} , J can be written as:

$$\frac{\partial \vec{p}}{\partial t} - \vec{\nabla}_0 \cdot \mathbf{P}(\mathbf{F}, \mathbf{H}, J) = \rho \vec{b} \quad (2.1a)$$

$$\frac{\partial \mathbf{F}}{\partial t} - \frac{1}{\rho} \vec{\nabla}_0 \vec{p} = \mathbf{0} \quad (2.1b)$$

$$\frac{\partial \mathbf{H}}{\partial t} - \frac{1}{\rho} \vec{\nabla}_0 \times (\vec{p} \mathbf{x} \mathbf{F}) = \mathbf{0} \quad (2.1c)$$

$$\frac{\partial J}{\partial t} - \frac{1}{\rho} \vec{\nabla}_0 \cdot (\mathbf{H}^T \vec{p}) = 0 \quad (2.1d)$$

In Eq. (2.1a), ρ and \vec{b} , respectively, represent the material density and the body force per unit mass. System (2.1) is solved for \vec{p} , \mathbf{F} , \mathbf{H} , J with respect to time t . It is a set of hyperbolic PDEs similar in structure to those widely used in CFD [42]. This similarity enables one to employ well-established artificial dissipation numerical techniques from CFD to improve the computational stability of Eqs. (2.1a) and (2.1d) [1, 23, 24]. It is worth noting that, in the presence of non-smooth solutions, the above system (2.1) of local conservation laws must be accompanied by suitable jump conditions, as described in [42].

The notation adopted, along with a pictorial representation of the main measures of deformation used in (2.1), is illustrated in Fig. 1.

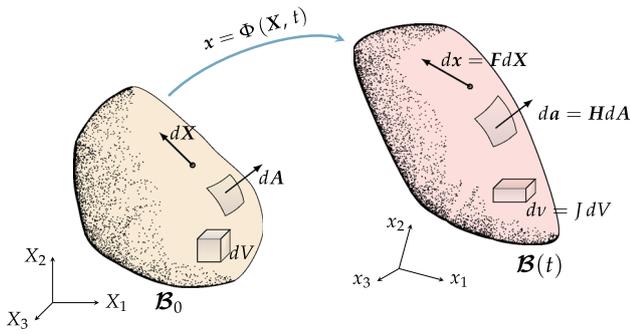


Fig. 1 Finite deformation process on a body \mathcal{B} in the continuum. Shown are measures of fibre, area, and volume strain— \mathbf{F} , \mathbf{H} , J —that define the process from the initial configuration \mathcal{B}_0 , to $\mathcal{B}(t)$ at time t

It is evident from the expressions (2.1b) and (2.1c), respectively, for the \mathbf{F} and \mathbf{H} conservation variables, that, for both physical (required compatibility of strains, [8]) and mathematical (ensure convexity, [17]) reasons, two sets of involutions need to be satisfied by the solution of (2.1):

$$\vec{\nabla}_0 \times \mathbf{F} = \mathbf{0} \tag{2.2a}$$

$$\vec{\nabla}_0 \cdot \mathbf{H} = \vec{0} \tag{2.2b}$$

In the above equations, the components of the material curl of \mathbf{F} in (2.2a), and of the material divergence of \mathbf{H} in (2.2b), can be expanded with the help of the Levi-Civita permutation symbol \mathcal{E}_{ijk} ¹ as, respectively,

$$\left(\vec{\nabla}_0 \times \mathbf{F}\right)_{ij} = \sum_{k=1}^3 \sum_{l=1}^3 \mathcal{E}_{jkl} \frac{\partial F_{il}}{\partial x_k} \quad \text{and}$$

$$\left(\vec{\nabla}_0 \cdot \mathbf{H}\right)_i = \sum_{k=1}^3 \frac{\partial H_{ik}}{\partial x_k} \quad \text{for } i, j = 1, 2, 3.$$

Using the involutions described in (2.2), Eqs. (2.1c) and (2.1d) can be further simplified into [25]:

$$\frac{\partial \mathbf{H}}{\partial t} - \mathbf{F} \times \vec{\nabla}_0 \left(\frac{\vec{p}}{\rho}\right) = \mathbf{0} \tag{2.3a}$$

$$\frac{\partial J}{\partial t} - \mathbf{H} : \vec{\nabla}_0 \left(\frac{\vec{p}}{\rho}\right) = 0 \tag{2.3b}$$

In the place of the full set of conservation variables, denoted here by $\mathcal{U} = \{\vec{p}, \mathbf{F}, \mathbf{H}, J\}$, reduced systems based on $\{\vec{p}, \mathbf{F}, J\}$, or only $\{\vec{p}, \mathbf{F}\}$ formulations have been adopted in the past. The robustness of these reduced systems has been positively ascertained by testing them against a thorough variety of numerical regimes and external conditions [1, 18, 24].

¹ \mathcal{E}_{ijk} assumes value +1 in case ijk is an even permutation of the sequence [1, 2, 3], and -1 in case is odd.

However, the complete $\{\vec{p}, \mathbf{F}, \mathbf{H}, J\}$ mixed system of conservation laws (2.1) can be provided with mathematical—instead of merely empirical—proof of existence and stability of physically relevant solutions. Such a proof may be obtained by verifying the hyperbolicity of the system, i.e. whether \mathcal{U} can be expressed in wave-like form [18, 27]:

$$\mathcal{U} = f\left(\vec{X} \cdot \vec{Z} - \lambda_i t\right) \vec{R}_i \quad i = 1, 2, 3 \tag{2.4}$$

In (2.4), \vec{Z} is a chosen direction, λ_i is the wave velocity and an eigenvalue of the system matrix, and \vec{R}_i is the wave profile and the eigenvector of the system matrix corresponding to λ_i . As can be seen from (2.4), in order to prove the hyperbolicity of the system, its own matrix should present real and distinct eigenvalues, and eigenvectors orthogonal to each other. Only then the hyperbolicity of a system of conservation laws can be guaranteed by specifically choosing an elastic potential function Ψ , for description of the material in use, able to satisfy certain convexity conditions. This would ensure the existence of minimum values of Ψ , for the solution of the variational problem associated with an elastic system [27].

The most stringent convexity condition is the notion of polyconvexity [3, 27]. More precisely, the polyconvexity condition states that the potential strain function $\Psi(\mathbf{F}, \mathbf{H}, J)$ has to be convex in the function space formed by the components of \mathbf{F} and \mathbf{H} , and by J [3, 8]. This also ensures a one-to-one mapping between the stress \mathbf{P} and strain measures $\{\mathbf{F}, \mathbf{H}, J\}$ [3, 16, 27, 31].

3 Discretisation

The governing systems of Eqs. (2.1) and (2.3) can be discretised by the SPH numerical scheme incorporating JST stabilisation components (JST-SPH) [21, 25].

The JST-SPH methodology consists of three key features:

- spatial discretisation with the SPH scheme;
- a JST-based numerical dissipation tool, adopted from CFD, to improve the accuracy and the stabilisation of the overall discretisation procedure;
- an explicit, two-stage total variation diminishing Runge-Kutta time integrator scheme to follow the time evolution of the solutions during dynamic simulations.

The above main features of the proposed methodology will be briefly elaborated in the following sections.

3.1 The SPH method

Being a meshfree technique, SPH can be effectively employed in the simulation of high-velocity impacts and high strain-rate

deformations [10,15,20,26,28,29,34,39,40]. In fact, beyond the yield point, SPH discretisation becomes attractive because large distortions make standard Lagrangian finite elements analyses difficult to pursue, without resorting to remeshing techniques that increase the computational cost [5,38]. This subsection briefly outlines the background for the SPH method and details the total Lagrangian SPH discretisation of the governing system of equations. The adoption of a total Lagrangian SPH formulation allows to avoid numerical tensile instabilities [4,11,33,40]. However, it was proven in [32] that for simulations involving material discontinuities (such as cracks and fragmentation), stabler results are achieved when switching to Eulerian kernels over the discontinuity regions while retaining Lagrangian kernels elsewhere.

In a discretised domain, the SPH interpolation of a function variable $\phi(\vec{x})$ at any arbitrary point \vec{x} , denoted here by $\langle \phi(\vec{x}) \rangle$, can be approximated by:

$$\langle \phi(\vec{x}) \rangle \approx \sum_b V_b \phi(\vec{x}_b) W(\vec{x}_b - \vec{x}, h) \quad (3.1)$$

where V_b is the fractional volume assigned to a neighbouring particle in \vec{x}_b , h is the smoothing length, and $W(\vec{x}_b - \vec{x}, h)$ is usually a polynomial function with compact support \mathcal{D} of radius $2h$ (that is, W vanishes for $\|\vec{x}_b - \vec{x}\| \geq 2h$). Although studies introducing adaptivity via a variable radius of support h exist in the particle methods literature (see [35]), here h and consequently \mathcal{D} and V for in-field particles are kept constant for simplicity.

For accurate interpolation of any given function, the polynomial kernel should satisfy the following reproducibility conditions:

$$\int_{\mathcal{D}} W(\vec{r}, h) d\vec{r} = 1; \quad (3.2a)$$

$$\lim_{h \rightarrow 0} W(\vec{r}, h) \rightarrow \delta(\vec{r}) \quad (3.2b)$$

In (3.2), $\vec{r} = \vec{x}_b - \vec{x}$, and $\delta(\vec{r})$ represents the Dirac delta function centred at 0. The first derivative of a given function ϕ can be discretised in terms of SPH interpolation as

$$\vec{\nabla} \phi(\vec{x}) = \sum_b V_b \phi(\vec{x}_b) \vec{\nabla} W(\vec{r}, h) \quad (3.3)$$

Several different types of kernel functions have been employed as SPH interpolators. In the present work, to maintain the continuity for spatial derivatives of kernel function up to the fourth order, the following quintic spline polynomial is adopted as the kernel function:

$$W(\vec{r}, h) = \frac{\alpha}{h^d} \begin{cases} (2 - r_h)^5 - 16(1 - r_h)^5 & \text{if } r_h \leq 1 \\ (2 - r_h)^5 & \text{if } 1 < r_h \leq 2, \quad r_h = \frac{\|\vec{r}\|}{h} \\ 0 & \text{if } r_h > 2 \end{cases} \quad (3.4)$$

In (3.4), d is the number of spatial dimensions of the problem, and α is a normalising constant that depends on d as,

$$\begin{cases} \alpha = \frac{1}{32\pi} & \text{if } d = 2 \\ \alpha = \frac{7}{40\pi} & \text{if } d = 3 \end{cases} \quad (3.5)$$

It is widely known that the fundamental discretised form of SPH interpolation in (3.1) suffers from poor accuracies at and in the vicinity of the domain boundaries. More precisely, (3.1) does not completely satisfy the partition of unity condition at or near the boundaries, due to the truncation of kernel functions. To improve the accuracy of the SPH interpolation near boundaries, and to exactly preserve momentum, corrections must be introduced on both the kernel and the gradient of the kernel [10–12]. In the present work, the accuracies of the kernel and the gradient of kernel are improved by using correction methods proposed in [12]. The corrected SPH kernel $\tilde{W}(\vec{r}, h)$ and the corrected gradient of kernel $\vec{\nabla}_0 W(\vec{r}, h)$ defined in [12] exactly fulfil first-order completeness, that is, they allow the SPH representation to exactly reproduce linear fields.

The corrected SPH interpolation of a given function $\vec{f}(\vec{X})$ will thus be expressed as,

$$\vec{\nabla}_0 \vec{f}(\vec{X}_a) = \sum_{b \in \Lambda_a^b} V_b (\vec{f}_b - \vec{f}_a) \otimes \vec{\nabla}_0 W_b(\vec{X}_a). \quad (3.6)$$

In (3.6), Λ_a^b identifies the set of neighbours b of a particle a , at which the gradient is evaluated. The mixed $\{\vec{p}, \mathbf{F}, \mathbf{H}, J\}$ system described by Eqs. (2.1) and (2.3) can now be discretised in space using the corrected SPH interpolation as described above.

Further, to obtain the discretised form of (2.1a), the weak statement for the linear momentum evolution [10–12,43] can be obtained by employing work–conjugate pairs [7,19]. With the help of integration by parts, this yields

$$\int_V \delta \vec{v} \cdot \frac{\partial \vec{p}}{\partial t} dV = \int_V \delta \vec{v} \cdot \rho \vec{b} dV + \int_{\partial V} \delta \vec{v} \cdot \vec{t} dA - \int_V \mathbf{P} : \vec{\nabla}_0 \delta \vec{v} dV \quad (3.7)$$

In (3.7), virtual velocities are denoted by $\delta \vec{v}$. By using particle discretisation on the domain V , the above expression (3.7) is approximated to:

$$\sum_{a=1}^N V_a \delta \vec{v}_a \cdot \frac{d\vec{p}_a}{dt} = \sum_{a=1}^N V_a \delta \vec{v}_a \cdot \rho \vec{b}_a$$

$$+ \sum_{a=1}^N A_a \delta \vec{v}_a \cdot \vec{t}_a - \sum_{a=1}^N V_a \mathbf{P}_a : \vec{\nabla}_0 \delta \vec{v}_a \quad (3.8)$$

In (3.8), A_a and \vec{t}_a are, respectively, associated area and traction force assigned to particles a located at or near the domain boundary. The density is assumed to be homogeneous throughout the domain. Equation (3.8) can now be expressed in terms of SPH discretisation as

$$\frac{\partial \vec{p}_a}{\partial t} = \rho \vec{b}_a + \frac{A_a}{V_a} \vec{t}_a - \sum_{b \in \Lambda_a^b} V_b \left[\mathbf{P}_b \vec{\nabla}_0 W_a(\vec{X}_b) - \mathbf{P}_a \vec{\nabla}_0 W_b(\vec{X}_a) \right] \quad (3.9)$$

The last term on the right-hand side of Eq. (3.9) represents the resultant internal force \vec{T}_a acting on particle a :

$$\vec{T}_a = \sum_{b \in \Lambda_a} V_b \left[\mathbf{P}_b \vec{\nabla}_0 W_a(\vec{X}_b) - \mathbf{P}_a \vec{\nabla}_0 W_b(\vec{X}_a) \right] \quad (3.10)$$

Conservation laws given by Eqs. (2.3) and (2.1d), respectively, for strain measures \mathbf{F} , \mathbf{H} , and J , can also be discretised in a similar manner to what has been done for (3.9), with the linear momentum \vec{p} evaluated through SPH:

$$\frac{\partial \mathbf{F}_a}{\partial t} = \frac{1}{\rho} \sum_{b \in \Lambda_a^b} V_b (\vec{p}_b - \vec{p}_a) \otimes \vec{\nabla}_0 W_b(\vec{X}_a) \quad (3.11a)$$

$$\frac{\partial \mathbf{H}_a}{\partial t} = \mathbf{F}_a \times \frac{1}{\rho} \sum_{b \in \Lambda_a^b} V_b (\vec{p}_b - \vec{p}_a) \otimes \vec{\nabla}_0 W_b(\vec{X}_a) \quad (3.11b)$$

$$\frac{\partial J_a}{\partial t} = \mathbf{H}_a : \frac{1}{\rho} \sum_{b \in \Lambda_a^b} V_b (\vec{p}_b - \vec{p}_a) \otimes \vec{\nabla}_0 W_b(\vec{X}_a) \quad (3.11c)$$

Depending on the chosen materials, the above semi-discrete formulation (3.9)–(3.11) may still suffer from accumulated numerical instabilities over a long span of time response. Therefore, it is necessary to address any such instability via suitable computational procedures.

3.2 JST artificial dissipation

In the context of SPH, various artificial dissipation techniques have been used in the past [28–30] to alleviate the aforementioned numerical instabilities. Alternatively, SPH can be stabilised by introducing stress points in the formulation (see [33]). In this case, however, a background mesh would be needed for the computation and assignment of fractional volumes. In the present work, an adapted nodally conservative JST stabilisation term $\mathcal{D}^{\text{JST}}(\mathbf{U}_a)$ will be incorporated into (3.9), mirroring CFD techniques. The hyperbolic nature of the first-order conservation laws in system (2.1), reflecting

that of the Euler equations in fluid dynamics, makes it possible to introduce the JST term as a dissipative component [1].

The nodally conservative JST stabilisation is additively decomposed into a second-order (harmonic) operator $\mathcal{D}_2(\mathbf{U}_a)$ and a fourth-order (biharmonic) operator $\mathcal{D}_4(\mathbf{U}_a)$ as:

$$\mathcal{D}^{\text{JST}}(\mathbf{U}_a) = \mathcal{D}_2(\mathbf{U}_a) + \mathcal{D}_4(\mathbf{U}_a) \quad (3.12)$$

where

$$\mathcal{D}_2(\mathbf{U}_a) = \kappa^{(2)} c_p \Delta x_{\min} \sum_{b \in \Lambda_a^b} V_b (\mathbf{U}_b - \mathbf{U}_a) \vec{\nabla}_0^2 W_b(\vec{X}_a) \quad (3.13a)$$

$$\mathcal{D}_4(\mathbf{U}_a) = -\kappa^{(4)} c_p \Delta x_{\min}^3 \sum_{b \in \Lambda_a^b} V_b \left(\frac{\mathcal{D}_2(\mathbf{U}_b) - \mathcal{D}_2(\mathbf{U}_a)}{\kappa^{(2)} c_p \Delta x_{\min}} \right) \vec{\nabla}_0^2 W_b(\vec{X}_a) \quad (3.13b)$$

In (3.13), c_p is the pressure wave speed, Δx_{\min} is the particle spacing, $\kappa^{(2)}$ and $\kappa^{(4)}$ are user-defined parameters, while the $\vec{\nabla}_0^2$ symbol represents a corrected Laplacian operator, applied to kernel W as detailed in [10].

3.3 Semi-discrete governing equations

Out of the four unknowns in the vector of state \mathbf{U} , only \vec{p} and J will see JST dissipation terms appear in their respective conservation laws (3.9) and (3.11c). $\mathcal{D}^{\text{JST}}(\mathbf{F}) = \mathcal{D}^{\text{JST}}(\mathbf{H}) = \mathbf{0}$ are imposed in order to respect conditions (2.2), set to ensure these strain measures preserve compatibility with the domain motions. This is because \mathbf{F} and \mathbf{H} are now main independent variables of the problem, and are not associated with displacements \vec{x} . In the context of the standard displacement-based method, it is worth recalling that \mathbf{F} is directly computed from \vec{x} , which acts as main independent variable of the problem.

In the light of the above, the spatial semi-discretisation yielded by the JST–SPH methodology reads as:

$$\left\{ \begin{aligned} \frac{\partial \vec{p}_a}{\partial t} &= \frac{A_a}{V_a} \vec{t}_a + \rho \vec{b}_a - \sum_{b \in \Lambda_a} V_b \left[\mathbf{P}_b \vec{\nabla}_0 W_a(\vec{X}_b) - \mathbf{P}_a \vec{\nabla}_0 W_b(\vec{X}_a) \right] + \mathcal{D}^{\text{JST}}(\vec{p}_a) \\ \frac{\partial \mathbf{F}_a}{\partial t} &= \sum_{b \in \Lambda_a} \frac{V_b}{\rho} (\vec{p}_b - \vec{p}_a) \otimes \vec{\nabla}_0 W_b(\vec{X}_a) \\ \frac{\partial \mathbf{H}_a}{\partial t} &= \mathbf{F}_a \times \sum_{b \in \Lambda_a} \frac{V_b}{\rho} (\vec{p}_b - \vec{p}_a) \otimes \vec{\nabla}_0 W_b(\vec{X}_a) \\ \frac{\partial J_a}{\partial t} &= \mathbf{H}_a : \sum_{b \in \Lambda_a} \frac{V_b}{\rho} (\vec{p}_b - \vec{p}_a) \otimes \vec{\nabla}_0 W_b(\vec{X}_a) + \mathcal{D}^{\text{JST}}(J_a) \end{aligned} \right. \quad (3.14)$$

Due to the incompatibility between strains and displacements introduced by the adoption of the mixed $\{\bar{p}, \mathbf{F}, \mathbf{H}, J\}$ formulation, discussed above, the conservation of angular momentum will not in general be satisfied by the discretised system of Eq. (3.14). In addition, the conservation of linear momentum may also not be respected, due to the lack of radial symmetry introduced by kernel and gradient corrections that are applied on particles located at, or in the vicinity, of domain boundaries.

With the aim to preserve the linear and angular momenta, the internal forces \bar{T} and the JST dissipation terms \mathcal{D}^{JST} have to be amended, in order to enforce the following conditions:

$$\begin{cases} \sum_{a=1}^N V_a \bar{T}_a = \vec{0}; & \sum_{a=1}^N V_a \bar{x}_a \times \bar{T}_a = \vec{0}; \\ \sum_{a=1}^N V_a \mathcal{D}^{\text{JST}}(\bar{p}_a) = \vec{0}; & \sum_{a=1}^N V_a \bar{x}_a \times \mathcal{D}^{\text{JST}}(\bar{p}_a) = \vec{0}; \\ \sum_{a=1}^N V_a \mathcal{D}^{\text{JST}}(J_a) = 0 \end{cases} \quad (3.15)$$

To impose conditions (3.15), a procedure based on the method of Lagrangian multipliers has been implemented (see [25] for more details).

3.4 Two-stage TVD Runge–Kutta temporal integration

To obtain a fully discretised system, that is able to describe the time evolution of the solution, the set of particle equations (3.14) has to be now explicitly integrated between chosen time intervals.

The solution will update from current time step t^n to next step t^{n+1} , until the end of the simulation. An explicit, two-stage TVD Runge–Kutta integrator scheme will be used for the purpose of time integration as follows:

$$\mathbf{u}_a^* = \mathbf{u}_a^n + \Delta t \mathcal{R}_a(\mathbf{u}_a^n, t^n) \quad (3.16a)$$

$$\mathbf{u}_a^{**} = \mathbf{u}_a^* + \Delta t \mathcal{R}_a(\mathbf{u}_a^*, t^{n+1}) \quad (3.16b)$$

$$\mathbf{u}_a^{n+1} = \frac{\mathbf{u}_a^n + \mathbf{u}_a^{**}}{2} \quad (3.16c)$$

The size of the time step, Δt , in (3.16) is obtained using the Courant–Friedrichs–Lewy (CFL) condition [42]:

$$\Delta t = \alpha_{cfl} \frac{h_l}{\max(c_p)_a} \quad a = 1, \dots, N \quad (3.17)$$

In (3.17), h_l is the characteristic length of the problem, here chosen to be equal to the initial distance between two particles, and α_{cfl} is a user-defined constant assuming values

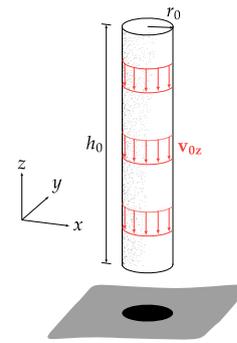


Fig. 2 Taylor bar problem. Initial configuration

between 0 and 1. The lower the value of α_{cfl} , the better the accuracy of the simulation.

The TVD Runge–Kutta scheme as defined in (3.16) has the advantage of being able to control the amount of spurious energy introduced by the explicit time integration, while allowing it to retain its qualities of speed and simplicity, especially when compared to implicit time-stepping methods.

Current particles positions, \bar{x}_a for $a = 1, \dots, N$, are obtained monolithically from the two-stage Runge–Kutta process (3.16), given $\mathbf{u} = \bar{x}$ and $\mathcal{R}(\mathbf{u}) = \frac{\bar{p}}{\rho}$.

4 Applications

In this section, the proposed JST–SPH algorithm will be used to perform three numerical examples. The first amongst these will be the classic Taylor bar test for high-speed-impact plasticity effects: a three-dimensional cylindrical metal bar that impacts a horizontal wall at a high velocity. In the second example, the performance of the method is assessed by examining the elasto-plastic compression of a constrained rectangular body. The last example involves the simulation of the equal channel angular extrusion (ECAE) cold metal forming process, where a workpiece is forced to pass through two 90° turns inside an extrusion die. The localised distortions caused by the substantial shear stresses experienced by the workpiece at these 90° turns should constitute an ideal test for assessing the capability of the developed methodology under large strain, high-velocity conditions.

4.1 Taylor bar high-speed impact

A common benchmark test for plastic deformation at high speed is the Taylor bar impact problem [41], illustrated in Fig. 2.

Elastic behaviour is governed by the material energy function presented in Eq. (A.1). Equation (A.1) is coupled with the plastic yield model described in ‘‘Appendix A’’, using the

Fig. 3 Taylor bar problem. The pictures show the cross-sectional shape of the bar, and the plastic strain contour plot at various instants in the simulation

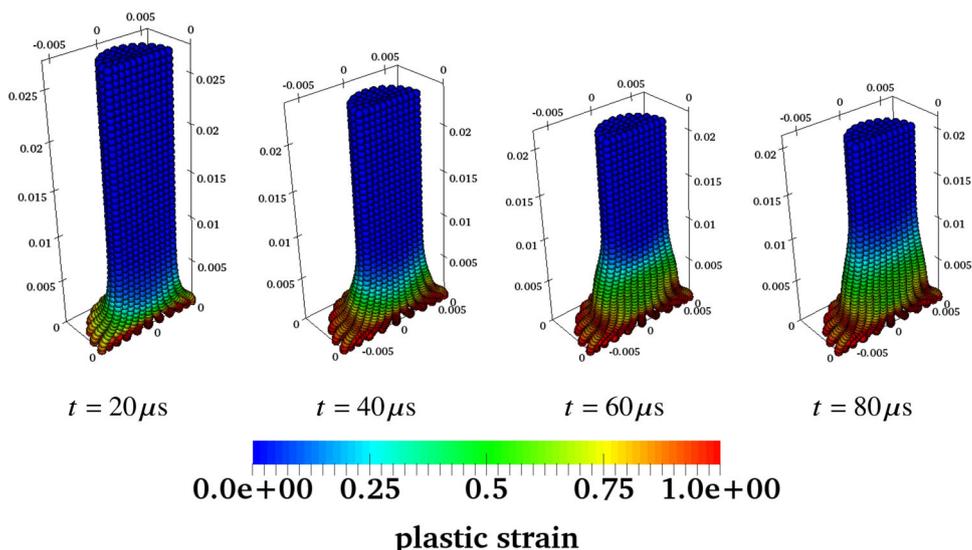
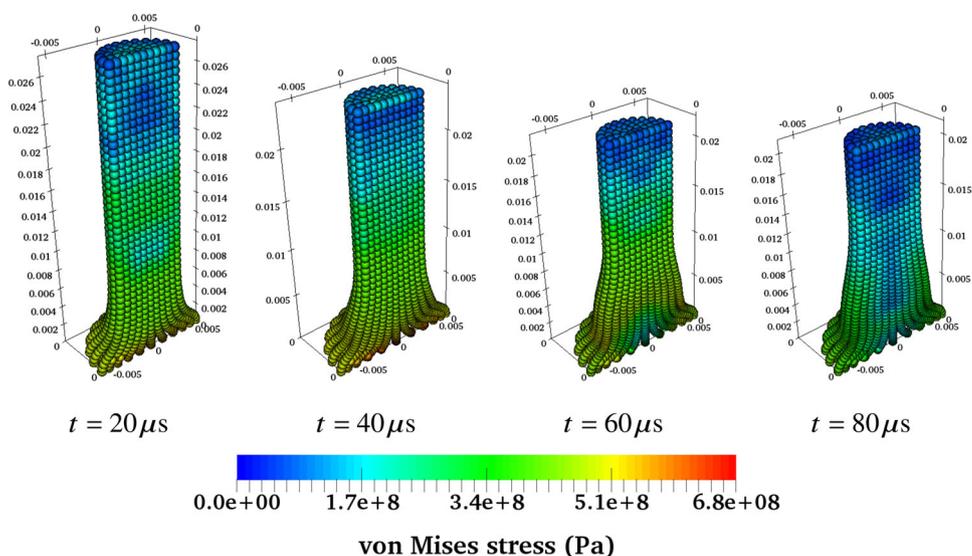


Fig. 4 Taylor bar problem. The pictures show the cross-sectional shape of the bar, and the von Mises stress contour plot at various instants in the simulation



linear isotropic hardening law presented in (A.14). The material is assumed to be copper, with the following properties: Young’s modulus $E = 117$ GPa, Poisson ratio $\nu = 0.35$, yield stress $\sigma_0^Y = 0.4$ GPa, hardening modulus $H = 0.1$ GPa and density $\rho = 8930$ kg/m³. The bar has initial height $h_0 = 32$ mm and radius $r_0 = 3.2$ mm, and is discretised by a set of 4131 particles arranged in a regular pattern. The impact is supposed frictionless and takes place at an initial speed of the bar of $v_{0z} = -227$ m/s in the z direction, normal to the rigid wall. Due to the extensive plastic dissipation, the JST stabilising term is set to a very low value.

Figure 3 presents results of the Taylor bar simulation performed by JST–SPH. As observed in experimental results [44], in Fig. 3 the plastic front appears closer to the bottom wall at the early stages of the simulation (red contour regions). It then slowly moves towards the top of the bar, as more kinetic energy dissipates into plastic strain.

To investigate the stress distribution across the bar, the von Mises equivalent stress is plotted in Fig. 4. A gradual increase in the plastic stress flow over time can be clearly observed.

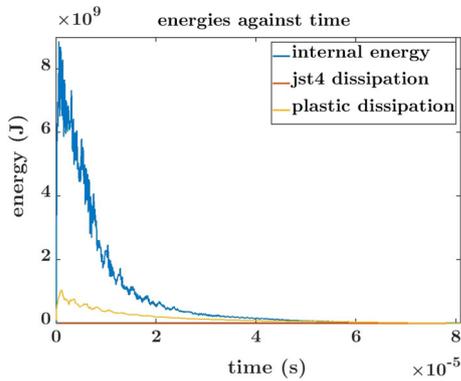
In Table 1, the radius of the bottom face at time $t = 80 \mu s$ is compared to the results of identical tests performed using different numerical methodologies [1]. It can be noted that the mixed formulation techniques, being locking-free, avoid excessive rigid response of the structure often demonstrated by low-order finite elements, displacement-based analyses.

Plots of the total internal energy and of energy dissipation associated with plastic deformation and JST artificial dissipation are reported in Fig. 5.

The equations used to evaluate the total internal energy U_{int} , the total plastic dissipation $W_{(p)}$ and the artificial JST dissipation $W_{(JST)}$ at each time step are given as follows:

Table 1 Taylor bar problem. Radius of bottom face at time $t = 80 \mu\text{s}$, obtained from different numerical methods

Numerical method	Final radius (mm)
Standard finite elements, hexahedra [6]	6.95
Standard finite elements, tetrahedra [6]	5.55
Mixed JST finite volumes, vertex-centred [1]	6.98
Mixed JST–SPH	6.98


Fig. 5 Taylor bar problem. Total values, summed over all particles, of the internal energy (blue line), plastic dissipation (yellow line) and JST artificial dissipation (red line) over simulation time. (Color figure online)

$$U_{\text{int}} = \sum_{a=1}^N \frac{\Delta t}{\rho} \|\bar{p}_a(\vec{X}, t)\| \cdot \|\vec{T}_a\| \quad (4.1a)$$

$$W_{(p)} = \sum_{a=1}^N J \|\sigma_a\| \cdot \Delta \epsilon_{(p)a} \quad (4.1b)$$

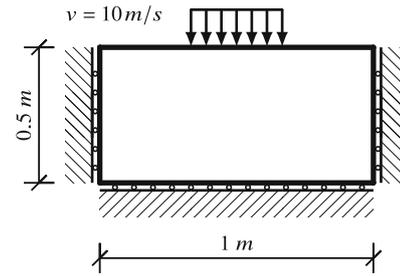
$$W_{(\text{JST})} = \sum_{a=1}^N \frac{\Delta t}{\rho} \mathcal{D}_a^{\text{JST}}(\bar{p}_a) \quad (4.1c)$$

It is shown in Fig. 5 that artificial dissipation introduced by the JST algorithm constitutes a negligible quantity.

4.2 Highly constrained problem with moving discontinuity

An example of the use of JST–SPH in a highly constrained setting is presented in [21,25] for a fully elastic, nearly incompressible material. The effects of plasticity will be studied in this similar example, where a bidimensional rectangular block with dimensions $1 \text{ m} \times 0.5 \text{ m}$ is considered under the assumption of plane strain. The material is copper, with properties chosen to be the same as those in the Taylor bar example in Sect. 4.1.

An external velocity $\vec{v} = (0, -10)^T \text{ m/s}$ is applied throughout the simulation on the top side of the block, as


Fig. 6 Constrained block under external velocity load

shown in Fig. 6. Roller boundary conditions are applied to its left, right and bottom edges.

The rectangular block is discretised by 231 regular particles. Additional numerical parameters used in the simulation are CFL number $\alpha_{cfl} = 0.3$ and JST coefficient $\kappa^{(2)} = 0$ and $\kappa^{(4)} = \frac{1}{1024}$.

These specific values of $\kappa^{(2)}$ and $\kappa^{(4)}$ were chosen in order to provide the simulation with a small amount of JST numerical dissipation, useful to suppress any spurious pressure oscillations at the initial stages of the process.

The purpose of this example is to demonstrate the capability of the total Lagrangian framework of the JST–SPH method, in the presence of high, localised distortions. The evolution of the von Mises equivalent plastic strain $\epsilon_{(p)}$ during the analysis is shown in the top row of Fig. 7. For comparison, the same simulation has been run using a commercial finite element method package, without any remeshing, and the results are presented in the bottom row of Fig. 7.

Results obtained by using increasing numbers of particles in the simulation are presented in Fig. 8.

It is observed from Figs. 7 and 8 that the basic finite element method is not able to handle the excessive distortions, whereas the JST–SPH method is able to simulate severe material deformation without exhibiting any numerical instability.

Plots of the total internal energy and of energy dissipation associated with plastic deformation and JST artificial dissipation are reported in Fig. 9. Energy quantities have been calculated using (4.1). It is shown in Fig. 9 that artificial dissipation introduced by the JST algorithm is a negligible quantity. Roller constraints imposed at the boundaries of the block are at the origin of irregularities appearing in the internal energy plot [calculated as in (4.1a)].

4.3 Simulation of the equal channel angular extrusion (ECAE) process

Numerous metallurgical techniques exploit the mechanics of plastic deformation to attain smaller grain size, and consequently better material properties, for metals and alloys. Amongst them, ECAE [36,37] can offer high levels of shear

Fig. 7 Constrained block problem. Comparison between JST-SPH (top row) and finite elements results obtained using Abaqus/Explicit (bottom row). Configurations shown at subsequent instants during the deformation history; the plastic strain profile is shown through the colour scale

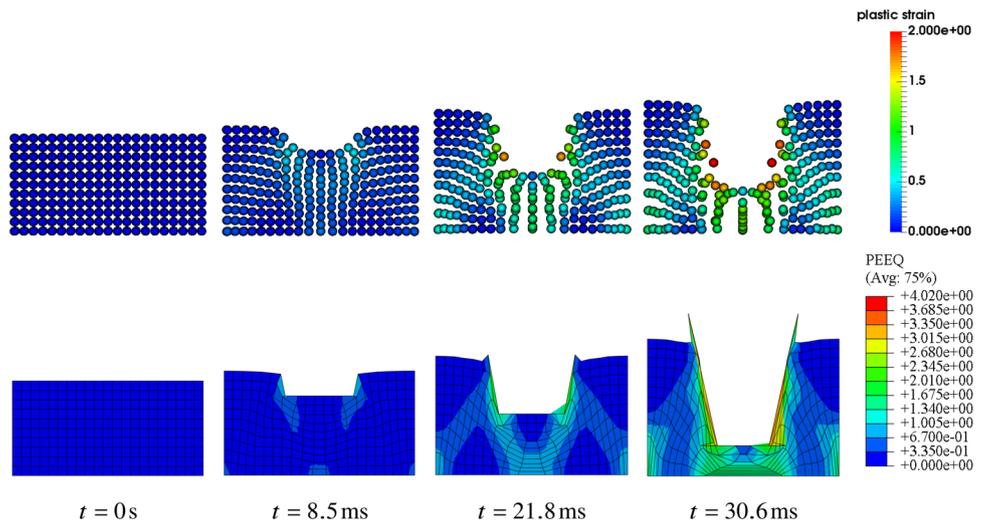


Fig. 8 Constrained block problem. Comparison between JST-SPH (top row) analyses obtained using different particles resolutions at time $t = 15$ ms into the simulation. Contour plot of plastic strain profile shown

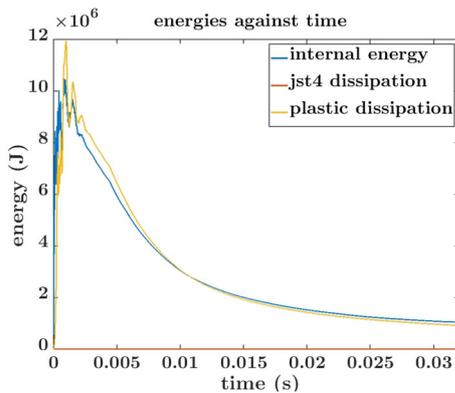
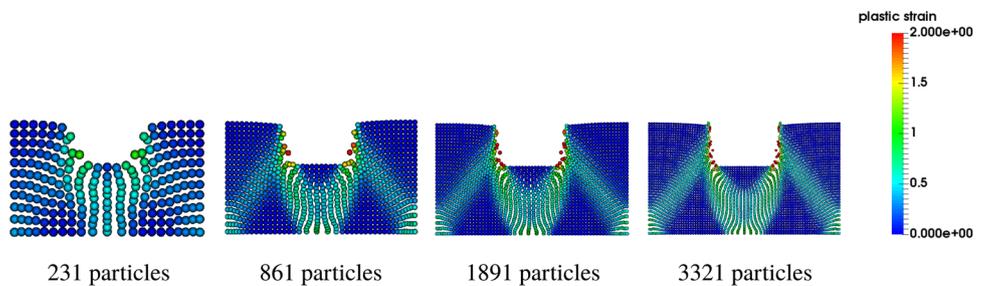


Fig. 9 Constrained block problem. Total values, summed over all particles, of the internal energy (blue line), plastic dissipation (yellow line) and JST artificial dissipation (red line) over the simulation time. (Color figure online)

strain for relatively low levels of external pressure, making it an ideal material processing technique for mass production. In the present setting, the ECAE process will be simulated in a channel with two 90° turns, analogous to the set-up analysed in [36] using a commercial finite element software. This example will demonstrate the robustness of the JST-SPH numerical scheme under a demanding dynamical process, where the material is subjected to continuous large deformation.

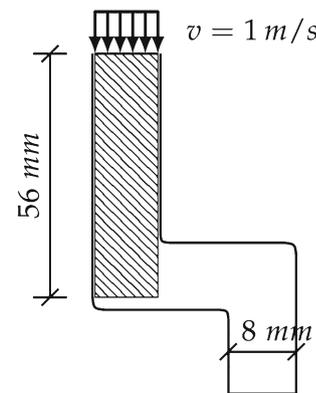


Fig. 10 ECAE simulation, initial set-up

The analysis is performed in plane strain conditions, and the material is made of commercially pure aluminium (Al1100, $E = 69$ GPa, $\nu = 0.33$, $\rho = 2800$ kg/m³, width $l = 8$ mm). The walls of the die are assumed rigid. Its right-angled corners are rounded, with 1.5 mm and 1 mm external and internal fillet radii. The initial set-up of the experiment is presented in Fig. 10.

The simulations are performed with a regular mesh of 400 particles. Contact between workpiece and die is assumed to be lubricated, and considered frictionless during the simulation. For the purpose of simplicity, a contact algorithm based on a reflective, bouncing back procedure [22] is adopted to

Fig. 11 Pressure p at various stages of the ECAE process. Top row, JST–SPH results. Bottom row, finite elements commercial solver Abaqus/Explicit, linear elements

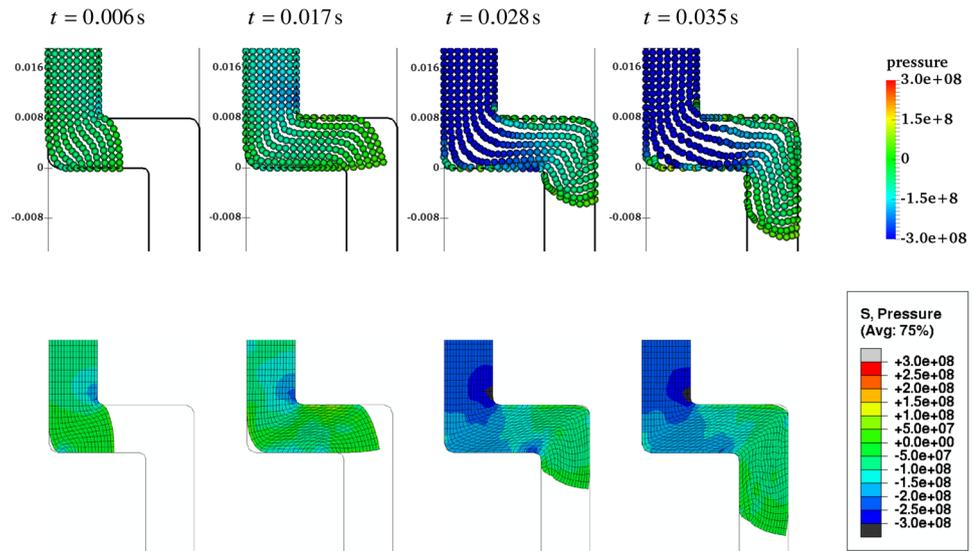
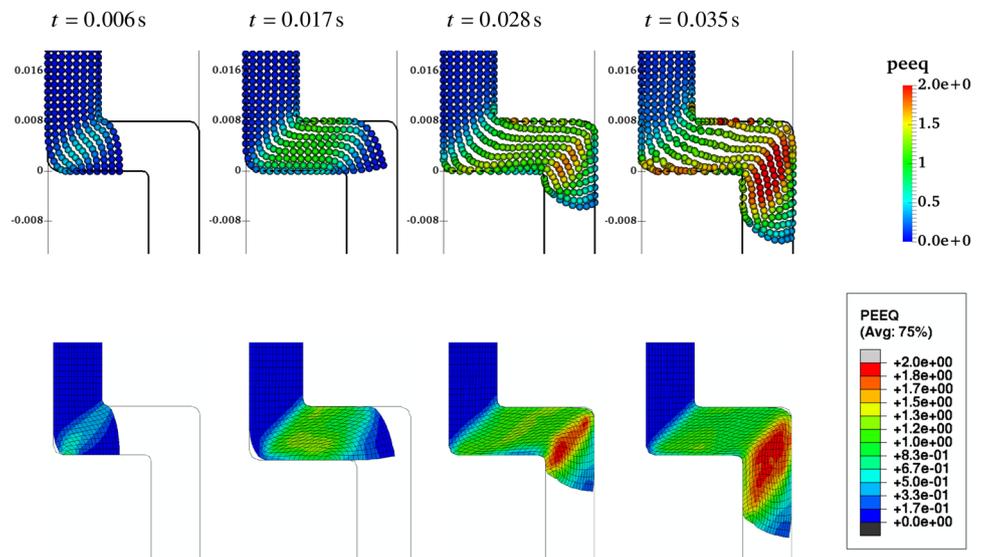


Fig. 12 Equivalent plastic strain (peeq) $\epsilon_{(p)}$ at various stages of the ECAE process. Top row, JST–SPH results. Bottom row, finite elements commercial solver Abaqus/Explicit, linear elements



simulate the interaction between the die walls and the particles representing billet material.

Given that large plastic deformations are expected, it is useful to introduce numerical dissipation at the early stages of the simulation, to prevent pressure instabilities that may arise when the billet initially comes into contact with the bottom wall of the channel. To facilitate this, for each particle, the JST term $k^{(4)}$ was defined as function of the equivalent plastic strain $\epsilon_{(p)}$, to ensure that $k^{(4)}$ linearly decreases, as plastic dissipation gradually sets in. This is achieved by adopting the following formula:

$$\begin{cases} k^{(4)} = \frac{1}{64} - 2\left(\frac{1}{64} - \frac{1}{4096}\right)\epsilon_{(p)} & \text{if } \epsilon_{(p)} \leq 0.5 \\ k^{(4)} = \frac{1}{4096} & \text{if } \epsilon_{(p)} > 0.5 \end{cases} \quad (4.2)$$

The plastic yield stress of the billet material is assumed to follow the nonlinear hardening law given by:

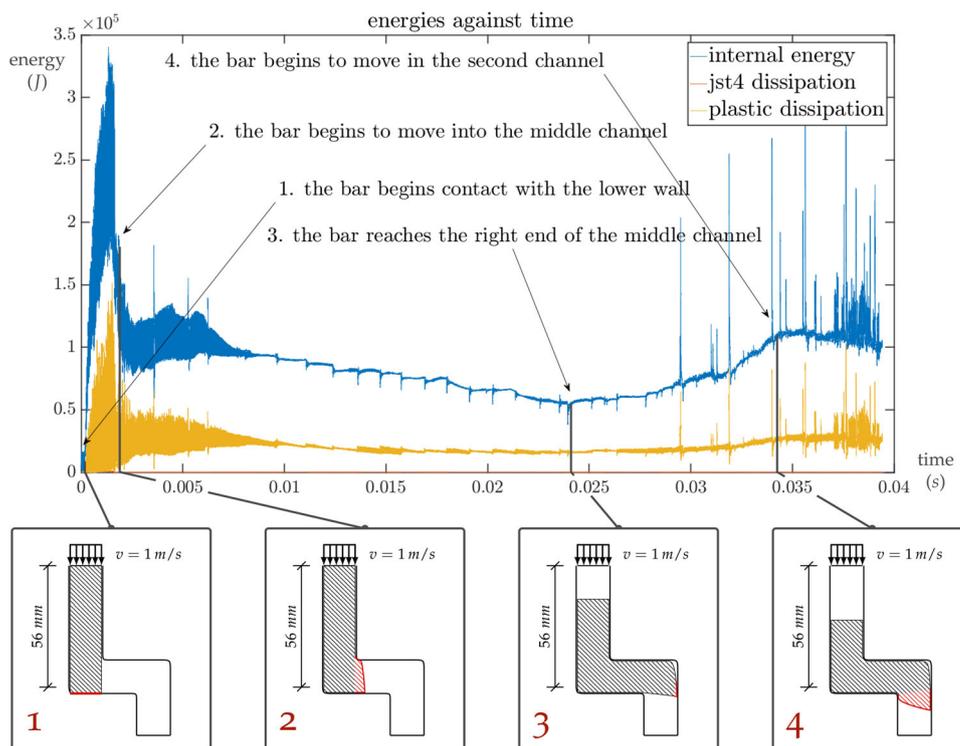
$$\sigma^Y(\epsilon_{(p)}) = A(0.02 + \epsilon_{(p)})^{0.27} \quad \text{where } A = 159 \times 10^6 \quad (4.3)$$

Equation 4.3 is solved for $\epsilon_{(p)}$ numerically by using the Newton–Raphson method.

Figures 11 and 12, respectively, depict the distributions of pressure and plastic strains during the deformation process of the billet, at various instants in time. For the purpose of comparison, Figs. 11 and 12 also present the corresponding pressure and plastic strain distributions obtained with a commercial finite element software (Abaqus/Explicit) using linear quadrilateral elements.

It can be noted that the results from the JST–SPH method correlate well with those obtained with the finite element method. Results from Figs. 11 and 12 are also consistent with observations reported in [36]. It is evident from the figures

Fig. 13 JST-SPH ECAE. Total values, summed over all particles, of the internal energy (blue line), plastic dissipation (yellow line) and JST artificial dissipation (red line) over the simulation time. The evolution of these quantities discloses information on various stages of the process. (Color figure online)



that the particle distribution in the vicinity of sharp corners of the channel is not smooth at the later stages of the simulation. This uneven distribution of the particles can be attributed to boundary effects. The plastic strain contour illustrated in Fig. 12 compares well with results reported in [36]. Further, it is clearly evident that the plastic strain distribution increases significantly as the billet material passes through each bend of the channel in the die. This feature is what is mainly exploited in the ECAE process to achieve the desired level of plastic strain in the workpiece.

Investigation of the energy patterns throughout the simulation confirms that plastic deformation is the main mechanism of energy dissipation, being on average two orders of magnitude larger than the energy dissipation introduced by activating the JST stabilisation term. Figure 13 compares the energies associated with plastic deformation, JST artificial dissipation, and the total internal energy of the bar during the simulation of the entire ECAE process. Energy quantities in Fig. 13 were calculated using (4.1).

As shown in Fig. 13, the values of total internal energy and of total plastic dissipation follow specific trends that can be linked to the motion of the billet inside the channel of the die. For example, a peak value for both energies was observed around 0.00142 s, corresponding to the instant in which the billet comes into contact with the bottom wall of the middle (horizontal) part of the channel. Following that, the energy values begin to decline steeply during the expansion of the billet material in the mid-channel. A similar, but

less pronounced energy pattern can be observed once the bar reaches, and then begins to flow into the second channel, with a peak value attained around 0.035 s. Moreover, it transpires from Fig. 13 that the energy values are highly oscillatory when the billet initially passes through the angular sections of the channel. However, the frequency and the intensity of these oscillations gradually decrease, as the material moves away from the angular sections.

In addition, Fig. 13 shows the gradual decrease in energy dissipation associated with the JST term at the onset of plastic deformation. This effect follows from the definition of the JST terms in (4.2), and further demonstrates that higher artificial dissipation values are only required at the very early stages of the simulation, that is, before any plastic deformation takes place.

In order to verify the accuracy of the simulation, a convergence study was performed by varying the number of particles in the discretised domain.

The quantity under analysis is the amount of plastic strain at 0.035 s, both in local (the maximum equivalent plastic strain $\max \epsilon_{(p)}$ amongst particles) and in global (the plastic dissipation developed in the single time step leading to instant 0.035 s) terms. Apart from the mesh of 400 particles used so far, alternative meshes with 88, 216 and 640 particles were employed. The results of the convergence analysis are presented in Fig. 14. It can be noted from the figure that the chosen parameters monotonically converge with increasing particle density.

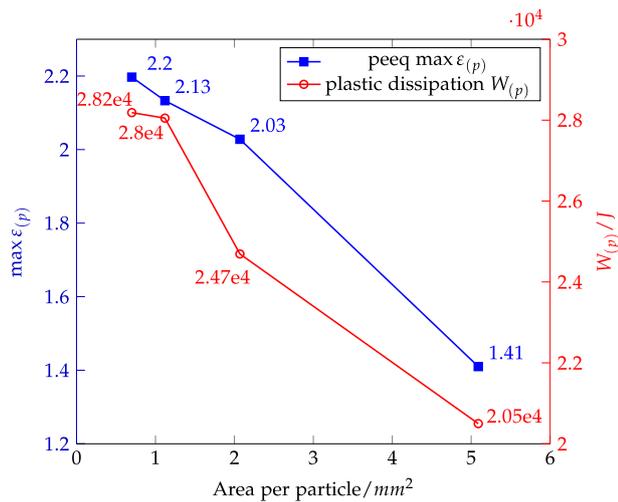


Fig. 14 JST–SPH ECAE. Convergence plot of local maximum plastic strain $\epsilon_{(p)}$ (in blue, left y-axis) and total dissipation $W_{(p)}$ due to plasticity (in red, right y-axis) at the time step entering 0.035 s in simulation time. (Color figure online)

The mesh with 640 particles has also been utilised for closer comparison of the degree of similarity between the JST–SPH ECAE test conducted here with the analysis found in [36]. For this purpose, the strain contour in the direction normal to the billet movement has been investigated towards the end of the test, at simulation time $t = 0.04$ s. The two locations selected are “Section 1” positioned at the centre of the middle channel, and “Section 2” at 5 mm from the second corner of the die. The above locations are consistent with those of the results reported in [36].

Ten equally spaced target positions have been chosen across the two sections of interest. Local values of the plastic strain $\epsilon_{(p)}$ have been computed using SPH interpolated values over neighbouring particles. Results are presented in Fig. 15. Comparison of Fig. 15 with results reported in [36] reveals that plastic strain patterns of the two tests match closely along Section 2, the main difference being that the JST–SPH analysis predictably presents a smoother curve. The same can be observed for Section 1, for which there is qualitative agreement with results from [36], except for the points near the upper wall, where results obtained with JST–SPH yield larger plastic strain. This discrepancy can be attributed to the fact that these points lie in the vicinity of the upper wall, and closer to the first bend. As can be noted in Fig. 12, also the accuracy of the plastic strain at these points is affected by the same boundary effects.

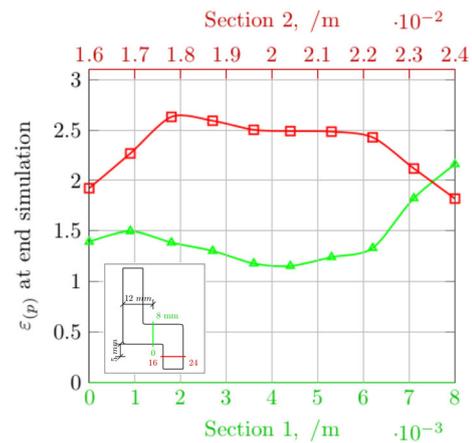


Fig. 15 JST–SPHECAE. Plastic strain $\epsilon_{(p)}$ distribution across the billet sections labelled in colour at margin, at $t = 0.04$ s. Compare to Fig. 5 in [36]

Parametric analyses reported in [36] provided an opportunity for further verification of the validity of the results.

In [36], geometrical properties of the die are modified in order to investigate the possibility of enhancing ECAE metallurgical performances. One of these parameters was the length of the middle channel of the die. In [36], it was noted that interesting deformed shapes develop, when the middle channel is shortened from 24 to 16 mm. The upper part of second bend has a large unfilled area, while the billet experiences more stress due to bending than due to shear. This effect is particularly severe around the inner region of the billet in the vicinity of the bottom part of the second bend. Both of these phenomena could be identified in an analogous simulation performed with JST–SPH, as presented in Fig. 16 at time $t = 0.027$ s. Results obtained with the finite elements commercial solver Abaqus/Explicit, using linear quadrilateral elements, are also shown in Fig. 16.

It is evident from the numerical examples that the JST–SPH scheme eliminates the excessive element distortions usually associated with mesh-based methods during the simulation of large plastic strains. In addition, the total Lagrangian framework adopted here provides an efficient modelling tool to gain better understanding of the physics underlying large material deformations.

An algorithmic diagram, illustrating the JST–SPH solution procedure for simulating elasto-plastic material deformations, is presented next.

JST–SPH algorithm for elasto-plastic simulations

Performs Total Lagrangian dynamics simulations of an elasto-plastic process using $\{\vec{p}, \mathbf{F}, \mathbf{H}, J\}$ mixed formulation: SPH space discretisation + JST dissipation term + 2 stages TVD Runge–Kutta explicit time stepping.

Initialisation

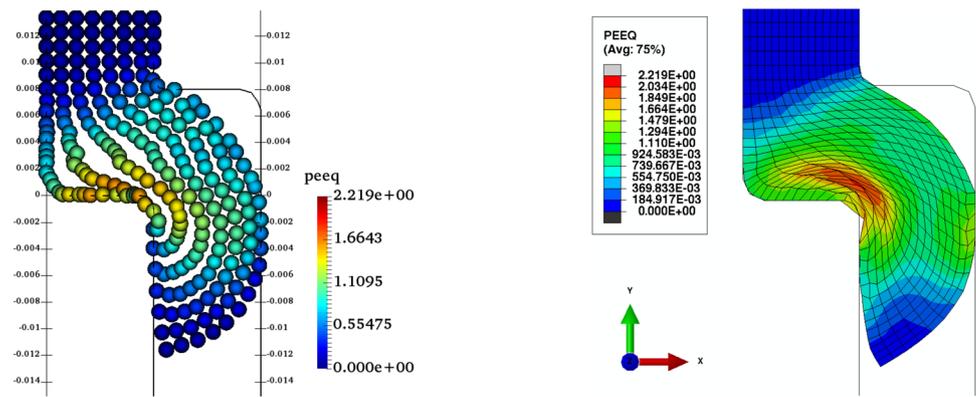
- Initiate all variables (set of particles geometry, material properties, numerical parameters, $\vec{p}_i, \mathbf{F}_i, \mathbf{P}_i, \vec{x}_i$ for each particle i).
- Compute SPH kernel smoothing length $h = \alpha(V/N)^{1/3}$, with N number of particles and α a user-defined input parameter.
- Locate neighbours j for each target particle i , using the *alternating digital tree (ADT)* search algorithm [13].
- Compute corrected kernel \tilde{W} , its gradient $\vec{\nabla}W$ and its Laplacian $\vec{\nabla}^2W$.

Time-stepping the solution

FOR $t < t_f$

- Compute wave speed $c_{ws|i} = \left(\frac{\sqrt{\lambda+2\mu}}{\rho} / \max_i \lambda_{\mathbf{C}|i} \right)$ for particle i .
 - Compute time step Δt (3.17).
 - Runge–Kutta time-stepping (3.16):
 - FOR** $RK = 1, 2$
 - Advance \vec{p} (3.14).
 - Compute JST term (3.13).
 - Apply conservation of ang. momentum on \vec{p} (3.10) and JST (3.13) [25].
 - Advance the other variables \mathbf{F}, \mathbf{H} and J (3.14).
 - Impose boundary conditions
 - Identify particles that have crossed the walls:
 - IF** $|\vec{x}_i| > \vec{x}_W$ **THEN**
 - Determine normal to the wall.
 - Reflect \vec{p}_i around the normal to the wall.
 - Place particle position on the wall, $\vec{x}_i = \vec{x}_W$.
 - Compute first Piola–Kirchhoff stress tensor \mathbf{P}_i as in “Appendix A”:
 - Given initial $\mathbf{C}_{(p)}, J$ and $\nabla\epsilon_{(p)}$.
 - Calculate pressure p .
 - Calculate trial Left Cauchy–Green tensor $\mathbf{b}_{(e)}$.
 - Calculate $\lambda_{(e)a}^2, \vec{n}_a, a = 1, 2, 3$ from $\mathbf{b}_{(e)}$ (eigendecomposition).
 - Calculate $\sigma_a, a = 1, 2, 3$.
 - Calculate $\Delta\epsilon_{(p)}$ and check:
 - IF** $\Delta\epsilon_{(p)} > 0$ **THEN**
 - Correct $\sigma_a, a = 1, 2, 3$ with return map algorithm.
 - Update $\lambda_{(e)a} \rightarrow \lambda_{(e)a}^{\text{upd}}, \mathbf{b}_{(e)} \rightarrow \mathbf{b}_{(e)}^{\text{upd}}$ and $\mathbf{C}_{(p)}$.
 - Compute \mathbf{P}_i from $\sigma_a, a = 1, 2, 3$.
 - Update plastic strain $\epsilon_{(p)}^{n+1} = \epsilon_{(p)}^n + \Delta\epsilon_{(p)}$.
- Obtain conserved variables values and particles positions at time $n + 1$ through averaging Runge–Kutta steps (3.16).

Fig. 16 On the left, contour plot obtained with JST–SPH of the equivalent plastic strain $\epsilon_{(p)}$ distribution over a modified version of the die, with a shortened middle channel length, at time $t = 0.027$ s. On the right, analogous result obtained on Abaqus/Explicit, with linear quadrilateral finite elements and no remeshing



5 Conclusions

This paper has focused on the application of JST–SPH mixed formulation in metal plasticity. This methodology has proved to be a valid tool for computing plastic deformations under dynamic regimes. It has been shown that there is good agreement between results reported in the previous section and data for the same tests found in the literature. The Taylor bar impact case provides an ideal application for JST–SPH, since SPH performs particularly well in high-velocity, large deformation settings. On the other hand, the ECAE test confirmed the robustness of JST–SPH: the total Lagrangian framework was able to withstand high local gradients of stresses, strains and displacements, while the presence of a constant loading in time did not lead to any kind of instability. This robustness was further evidenced in the constrained block simulation, where it can be seen that JST–SPH is able to achieve accurate results while preserving the initial connectivity under very large distortions. The same outcome cannot be obtained with standard mesh-based methods, without remeshing. It was noted, in the case of the ECAE simulation, that the results in the vicinity of the boundaries were slightly affected by sharp corners. These boundary effects can be easily eliminated by using more sophisticated boundary treatments. This will be one of the focal points in future publications.

Acknowledgements On behalf of all authors, the corresponding author states that there is no conflict of interest. The authors would like to acknowledge the financial support provided by the Sêr Cymru National Research Network (Grant No. NRN038). The authors also wish to thank Dr. C.H. Lee, Prof. A. Gil and Prof. J. Bonet for the useful insights that have inspired many of the concepts developed in this work.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix A: Rate-independent plastic constitutive relations

In the following, a stretch-based, elastic energy function will be defined as [14]:

$$\Psi(\lambda_1, \lambda_2, \lambda_3, J) = \mu[(\ln \lambda_1)^2 + (\ln \lambda_2)^2 + (\ln \lambda_3)^2] + \frac{\lambda}{2}(\ln J)^2 \quad (\text{A.1})$$

where μ and λ are the Lamé constants, and $\lambda_1, \lambda_2, \lambda_3$ are the principal stretches, such that the right Cauchy–Green deformation tensor, \mathbf{C} [14], can be expressed as:

$$\mathbf{C} = \sum_{a=1}^3 \lambda_a^2 \vec{N}_a \otimes \vec{N}_a \quad (\text{A.2})$$

In (A.2), \vec{N}_a , $a = 1, 2, 3$, are the principal directions in the reference frame.

From (A.1), it is possible to calculate the first Piola–Kirchhoff stress tensor \mathbf{P} for a polyconvex material in the elastic realm and, in so doing, to close the system of conservation laws (2.1) [7,8,19]:

$$\mathbf{P}(\mathbf{F}, \mathbf{H}, J) = \Sigma_{\mathbf{F}} + \Sigma_{\mathbf{H}} \times \mathbf{F} + \Sigma_J \mathbf{H} \quad (\text{A.3})$$

In (A.3), $\Sigma_{\mathbf{F}}$, $\Sigma_{\mathbf{H}}$ and Σ_J are the entropy conjugates of strain variables \mathbf{F} , \mathbf{H} and J . More detailed definitions of entropy conjugates can be found in [7,19].

Equation (A.3) provides solutions while in the elastic regime. However, once the deformation state progresses beyond the yield point and acquires a plastic component, the first Piola–Kirchhoff stress tensor \mathbf{P} will be hereby computed as a function of the Cauchy stress $\boldsymbol{\sigma}$ in the current configuration, following [14]:

$$\mathbf{P} = J \boldsymbol{\sigma} \mathbf{F}^{-T} \quad (\text{A.4})$$

The stress tensor $\boldsymbol{\sigma}$ in (A.4) can be expressed in spatial principal directions \vec{n}_a , $a = 1, 2, 3$ as

$$\boldsymbol{\sigma} = \sum_{a=1}^3 \sigma_a \vec{n}_a \otimes \vec{n}_a \quad (\text{A.5})$$

where σ_a is given by

$$\sigma_a = \kappa \frac{\ln J}{J} + 2 \frac{\mu}{J} \left(\ln \lambda_{(e)a} - \frac{1}{3} \ln J \right) \quad a = 1, 2, 3 \quad (\text{A.6})$$

In (A.6), the bulk modulus κ for the material was introduced. Operating in principal directions holds the distinct advantage of making quantities computed in the current section automatically frame invariant.

The elastic principal stretches $\lambda_{(e)a}$ for $a = 1, 2, 3$ in (A.6), and the principal directions \vec{n}_a , $a = 1, 2, 3$ in (A.5) are obtained as the eigenvalues and eigenvectors of the elastic part $\mathbf{b}_{(e)}$ of the left Cauchy–Green deformation tensor \mathbf{b} , as

$$\mathbf{b}_{(e)} = \sum_{a=1}^3 \lambda_{(e)a}^2 \vec{n}_a \otimes \vec{n}_a \quad (\text{A.7})$$

The plastic part of the deformation enters the algorithm just described through the preliminary evaluation of $\mathbf{b}_{(e)}$, obtained using the plastic part of the right Cauchy–Green strain tensor in (A.2), $\mathbf{C}_{(p)}$, at the beginning of each time step:

$$\mathbf{b}_{(e)} = \mathbf{F} \mathbf{C}_{(p)}^{-1} \mathbf{F}^T \quad (\text{A.8})$$

Given the deviatoric component $\boldsymbol{\sigma}_{\text{dev}}$ of the stress tensor $\boldsymbol{\sigma}$ found in (A.5), and its principal components $\sigma_{\text{dev},a}$, $a = 1, 2, 3$:

$$\begin{aligned} \boldsymbol{\sigma}_{\text{dev}} &= \boldsymbol{\sigma} - \kappa \frac{\ln J}{J} \mathbf{1}; \\ \sigma_{\text{dev},a} &= 2 \frac{\mu}{J} \left(\ln \lambda_{(e)a} - \frac{1}{3} \ln J \right) \quad a = 1, 2, 3 \end{aligned} \quad (\text{A.9})$$

the plastic domain is entered when the equivalent stress $\sigma_{\text{eq}} = \sqrt{\frac{3}{2}} (\boldsymbol{\sigma}_{\text{dev}} : \boldsymbol{\sigma}_{\text{dev}})$ is past the yield point, as determined by the von Mises yielding criterion:

$$\sigma_{\text{eq}} - \sigma^Y \leq 0 \quad (\text{A.10})$$

In (A.10), $\sigma^Y = \sigma^Y(\epsilon_{(p)})$ is the yield stress, that in general will be a function of the history of accumulated plastic deformation during the process, through a material hardening rule.

In case condition (A.10) does not hold, plastic deformation develops, $\epsilon_{(p)} > 0$, and the yield surface is updated through

a radial return mapping procedure, whereby the normal to the yield surface $\vec{m} = (m_1, m_2, m_3)$ is expressed by

$$m_a = \frac{\sigma_{\text{dev},a}}{\sqrt{\boldsymbol{\sigma}_{\text{dev}} : \boldsymbol{\sigma}_{\text{dev}}}} \quad a = 1, 2, 3 \quad (\text{A.11})$$

and the principal components of the deviatoric stress tensor $\sigma_{\text{dev},a}$ are updated as [14]

$$\sigma_{\text{dev},a}^{\text{upd}} = \sigma_{\text{dev},a} - 2\mu \Delta\epsilon_{(p)} m_a \quad a = 1, 2, 3 \quad (\text{A.12})$$

The resulting increment in equivalent plastic strain $\Delta\epsilon_{(p)}$ will be computed as the strain amount necessary to keep the updated equivalent plastic stress $\sigma_{\text{eq}}^{\text{upd}} = \sqrt{\frac{3}{2}} (\boldsymbol{\sigma}_{\text{dev}}^{\text{upd}} : \boldsymbol{\sigma}_{\text{dev}}^{\text{upd}})$ on the yield surface:

$$\sigma_{\text{eq}}^{\text{upd}}(\Delta\epsilon_{(p)}) - \sigma^Y(\epsilon_{(p)}, \Delta\epsilon_{(p)}) = 0 \quad (\text{A.13})$$

In the case of the classic hardening rule that linearly depends on $\Delta\epsilon_{(p)}$, i. e.

$$\sigma^Y = \sigma_0^Y + H (\epsilon_{(p)} + \Delta\epsilon_{(p)}) \quad (\text{A.14})$$

where σ_0^Y is the initial yield stress value, and H is the hardening modulus, $\Delta\epsilon_{(p)}$ can be explicitly obtained as

$$\Delta\epsilon_{(p)} = \frac{\sigma_{\text{eq}} - \sigma^Y(\epsilon_{(p)})}{3\mu + H} \quad (\text{A.15})$$

If instead a more complicated dependency of the type $\sigma^Y = \sigma^Y(\Delta\epsilon_{(p)})$ is in place, then $\Delta\epsilon_{(p)}$ has to be extracted numerically.

Once $\Delta\epsilon_{(p)}$ is known, the elastic principal stretches $\lambda_{(e)a}$ can be corrected to take the plastic dissipation into account

$$\lambda_{(e)a}^{\text{upd}} = \lambda_{(e)a} e^{-\Delta\epsilon_{(p)} m_a} \quad a = 1, 2, 3 \quad (\text{A.16})$$

and the updated Cauchy stress components in principal directions $\sigma_{\text{dev},a}^{\text{upd}}$ from (A.12) are returned as

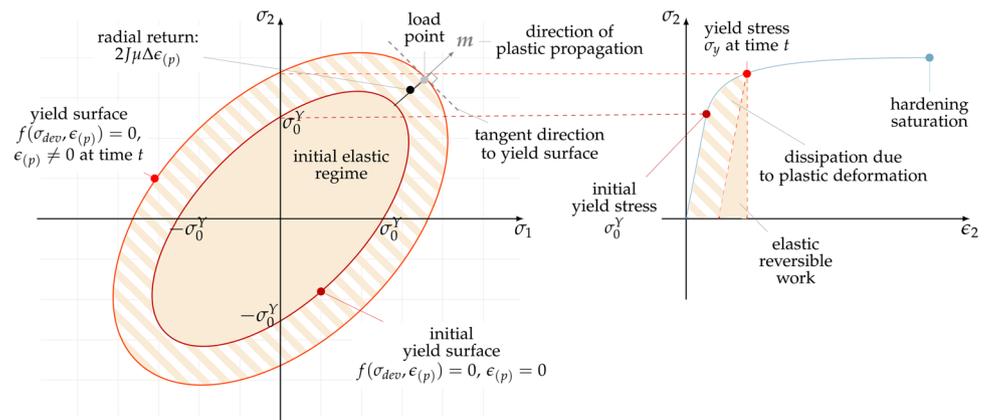
$$\sigma_{\text{dev},a}^{\text{upd}} = 2 \frac{\mu}{J} \left(\ln \lambda_{(e)a} - \frac{1}{3} \ln J \right) \left(1 - \frac{2\mu \Delta\epsilon_{(p)}}{\sqrt{\boldsymbol{\sigma}_{\text{dev}} : \boldsymbol{\sigma}_{\text{dev}}}} \right) \quad (\text{A.17})$$

Equation (A.16) allows updating the elastic deformation tensor $\mathbf{b}_{(e)}^{\text{upd}}$ accordingly:

$$\mathbf{b}_{(e)}^{\text{upd}} = \sum_{a=1}^3 (\lambda_{(e)a})^2 \vec{n}_a \otimes \vec{n}_a \quad (\text{A.18})$$

Equation (A.18) exploits the known fact that plastic dissipation does not alter principal directions \vec{n}_a , $a = 1, 2, 3$.

Fig. 17 A typical von Mises-defined yield surface in plain stress and in the presence of isotropic hardening. Radial return mapping also shown



What is left in order to complete the algorithm is then to substitute in Eq. (A.5) σ_a with $\sigma_{\text{dev},a}^{\text{upd}}$ obtained from (A.17), and to add the dilatation pressure term $\kappa \frac{\ln J}{J}$. The plastic deformation state is also updated as

$$\mathbf{C}_{(p)}^{-1} = \mathbf{F}^{-1} \mathbf{b}_{(e)} \mathbf{F}^{-T} \quad (\text{A.19a})$$

$$\epsilon_{(p)}^{\text{upd}} = \epsilon_{(p)} + \Delta \epsilon_{(p)} \quad (\text{A.19b})$$

The von Mises plastic algorithm described in this section is depicted in Fig. 17 in the case of isotropic hardening.

References

- Aguirre M, Gil A, Bonet J, Carreño A (2014) A vertex centred finite volume Jameson–Schmidt–Tukel algorithm for a mixed conservation formulation in solid dynamics. *J Comput Phys* 259:672–699
- Aguirre M, Gil A, Bonet J, Lee C (2015) An upwind vertex centred finite volume solver for Lagrangian solid dynamics. *J Comput Phys* 300:387–422
- Ball J (1977) Convexity conditions and existence theorems in nonlinear elasticity. *Arch Ration Mech Anal* 63(4):337–403
- Belytschko T, Guo Y, Liu W, Xiao P (2000) A unified stability analysis of meshless particle methods. *Int J Numer Methods Eng* 48(9):1359–1400
- Belytschko T, Liu W, Moran B (2000) *Nonlinear finite elements for continua and structures*. Wiley, Hoboken
- Bonet J, Burton A (1998) A simple average nodal pressure tetrahedral element for incompressible and nearly incompressible dynamic explicit applications. *Commun Numer Methods Eng* 14(5):437–449
- Bonet J, Gil A, Lee C, Aguirre M, Ortigosa R (2015) A first order hyperbolic framework for large strain computational solid dynamics. Part I: total Lagrangian isothermal elasticity. *Comput Methods Appl Mech Eng* 283:689–732
- Bonet J, Gil A, Ortigosa R (2015) A computational framework for polyconvex large strain elasticity. *Comput Methods Appl Mech Eng* 283:1061–1094
- Bonet J, Gil A, Ortigosa R (2016) On a tensor cross product based formulation of large strain solid mechanics. *Int J Solids Struct* 84:49–63
- Bonet J, Kulasegaram S (2000) Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations. *Int J Numer Methods Eng* 47(6):1189–1214
- Bonet J, Kulasegaram S (2001) Remarks on tension instability of Eulerian and Lagrangian corrected smooth particle hydrodynamics (CSPH) methods. *Int J Numer Methods Eng* 52(11):1203–1220
- Bonet J, Lok TS (1999) Variational and momentum preservation aspects of smooth particle hydrodynamic formulations. *Comput Methods Appl Mech Eng* 180(1–2):97–115
- Bonet J, Peraire J (1991) An alternating digital tree (ADT) algorithm for 3D geometric searching and intersection problems. *Int J Numer Methods Eng* 31(1):1–17
- Bonet J, Wood R (2008) *Nonlinear continuum mechanics for finite element analysis*, 2nd edn. Cambridge University Press, Cambridge
- Cleary P, Prakash M, Ha J (2006) Novel applications of smoothed particle hydrodynamics (SPH) in metal forming. *J Mater Process Technol* 177(1–3):41–48
- Coleman B, Noll W (1959) On the thermostatics of continuous media. *Arch Ration Mech Anal* 4:97–128
- Dafermos C (2013) Non-convex entropies for conservation laws with involutions. *Philos Trans R Soc Lond A Math Phys Eng Sci* 371(2005):20120344
- Gil A, Lee C, Bonet J, Aguirre M (2014) A stabilised Petrov–Galerkin formulation for linear tetrahedral elements in compressible, nearly incompressible and truly incompressible fast dynamics. *Comput Methods Appl Mech Eng* 276:659–690
- Gil A, Lee C, Bonet J, Ortigosa R (2016) A first order hyperbolic framework for large strain computational solid dynamics. Part II: total Lagrangian compressible, nearly incompressible and truly incompressible elasticity. *Comput Methods Appl Mech Eng* 300:146–181
- Gray J, Monaghan J, Swift R (2001) SPH elastic dynamics. *Comput Methods Appl Mech Eng* 190(49–50):6641–6662
- Greto G, Kulasegaram S, Lee C, Gil A, Bonet J. A stabilised total Lagrangian corrected smooth particle hydrodynamics technique in large strain explicit fast solid dynamics. In: *ECCOMAS congress 2016—proceedings of the 7th European congress on computational methods in applied sciences and engineering*, pp 8231–8240
- Hoomans B, Kuipers J, Briels W, van Swaaij W (1996) Discrete particle simulation of bubble and slug formation in a two-dimensional gas–fluidised bed: a hard-sphere approach. *Chem Eng Sci* 51(1):99–118
- Lee C, Gil A, Bonet J (2013) Development of a cell centred upwind finite volume algorithm for a new conservation law formulation in structural dynamics. *Comput Struct* 118:13–38
- Lee C, Gil A, Bonet J (2014) Development of a stabilised Petrov–Galerkin formulation for conservation laws in Lagrangian fast solid dynamics. *Comput Methods Appl Mech Eng* 268:40–64

25. Lee C, Gil A, Greto G, Kulasegaram S, Bonet J (2016) A new JST smooth particle hydrodynamics algorithm for large strain explicit fast dynamics. *Comput Methods Appl Mech Eng* 311:71–111
26. Libersky L, Petschek A, Carney T, Hipp J, Allahdadi F (1993) High strain Lagrangian hydrodynamics a three-dimensional SPH code for dynamic material response. *J Comput Phys* 109(1):67–75
27. Marsden J, Hughes T (1983) *Mathematical foundations of elasticity*. Dover civil and mechanical engineering. Dover Publications, Mineola (reprint)
28. Monaghan J (1988) An introduction to SPH. *Comput Phys Commun* 48(1):89–96
29. Monaghan J (1992) Smoothed particle hydrodynamics. *Ann Rev Astron Astrophys* 30(1):543–574
30. Monaghan J, Pongracic H (1985) Artificial viscosity for particle methods. *Appl Numer Math* 1(3):187–194
31. Ogden R (1970) Compressible isotropic elastic solids under finite strain constitutive inequalities. *Q J Mech Appl Mech* 23(4):457–468
32. Rabczuk T, Belytschko T (2007) A three-dimensional large deformation meshfree method for arbitrary evolving cracks. *Comput Methods Appl Mech Eng* 196(29):2777–2799
33. Rabczuk T, Belytschko T, Xiao S (2004) Stable particle methods based on Lagrangian kernels. *Comput Methods Appl Mech Eng* 193(12):1035–1063
34. Randles P, Libersky L (1996) Smoothed particle hydrodynamics: some recent improvements and applications. *Comput Methods Appl Mech Eng* 139(1–4):375–408
35. Ren H, Zhuang X, Cai Y, Rabczuk T (2016) Dual-horizon peridynamics. *Int J Numer Methods Eng* 108(12):1451–1476
36. Rosochowski A, Olejnik L (2002) Numerical and physical modelling of plastic deformation in 2-turn equal channel angular extrusion. *J Mater Process Technol* 125–126:309–316
37. Segal V (1995) Materials processing by simple shear. *Mater Sci Eng A* 197(2):157–164
38. Simo J, Hughes T (2000) *Computational inelasticity*. Interdisciplinary applied mathematics. Springer, New York
39. Springel V (2010) Smoothed particle hydrodynamics in astrophysics. *Ann Rev Astron Astrophys* 48:391–430
40. Swegle J, Hicks D, Attaway S (1995) Smoothed particle hydrodynamics stability analysis. *J Comput Phys* 116(1):123–134
41. Taylor G (1948) The use of flat-ended projectiles for determining dynamic yield stress. *Proc R Soc Lond A Math Phys Eng Sci* 194:289–299
42. Toro E (1999) *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Applied mechanics: researchers and students, 2nd edn. Springer, Berlin
43. Vidal Y, Bonet J, Huerta A (2007) Stabilized updated Lagrangian corrected SPH for explicit dynamic problems. *Int J Numer Methods Eng* 69(13):2687–2710
44. Wilkins M, Guinan M (1973) Impact of cylinders on a rigid boundary. *J Appl Phys* 44(3):1200–1206

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.