



The momentum-consistent smoothed particle Galerkin (MC-SPG) method for simulating the extreme thread forming in the flow drill screw-driving process

C. T. Wu¹ · Youcai Wu¹ · Dandan Lyu¹ · Xiaofei Pan¹ · Wei Hu¹

Received: 21 January 2019 / Revised: 22 March 2019 / Accepted: 25 March 2019 / Published online: 3 April 2019
© OWZ 2019

Abstract

Flow drill screw (FDS) joining is a modern mechanical fastening technique for connecting metal parts in lightweight car structures. Finite element simulation of FDS joining probably is one of the biggest challenges for CAE engineers in automotive applications. This is mainly because finite element methods inevitably encounter utmost numerical difficulties associated with meshing issues in modeling the extensive plastic deformation and material failure taking place during the thread forming operation in the FDS driving process. This paper presents a FDS thread forming simulation using a new particle method based on authors' recent work (Pan et al. in *Comput Mech*, 2019. <https://doi.org/10.1007/s00466-019-01673-8>). Different from other particle stabilization methods which are obtained by modification of the variational equation using either residual or non-residual stabilization terms, the present method introduces a novel velocity smoothing algorithm to achieve the stabilization effect. It is shown that the semi-discrete equation based on the smoothed velocity field is consistently fulfilling the conservation of linear and angular momentum. Since the new method does not require stabilization terms, it avoids the fundamental difficulty inherent in the stabilization stress computation, thus computationally more efficient. Finally, the stabilized formulation is supplemented with the adaptive anisotropic Lagrangian kernel and bond-based failure criterion to extend the application in severe deformation and material failure analysis. Three numerical benchmarks including one FDS thread forming simulation are utilized to demonstrate the effectiveness of the new approach.

Keywords Particle · Stabilization · FDS · Large deformation · Material failure

1 Introduction

A car structure generally consists of hundreds of fabricated metal panels and frames joined together, using a combination of various joining techniques such as spot welding, riveting, clinching, hinging and screwing. Those joints are often considered the weakest points as regards to structural strength. Therefore, it is very important to understand the manufacturing process of joints and their mechanical behavior in a car design phase. In the vehicle's virtual development process, extensive computer simulations are conducted using commercial finite element codes such as LS-DYNA[®] and

Abaqus to study the performance of joints before a body-in-white structure is ready for production. Thus, it is critical for automotive industry to use the advanced computer aided engineering (CAE) software that enables for various joints modeling and failure prediction in car structure and safety design.

In the meantime, there is an increasing use of lightweight and high-performance materials like aluminum and galvanized or pre-painted steel in energy saving vehicles. As resistance spot welding of those materials is difficult or even impossible, automotive industry is striving toward new joining techniques such as flow drill screw (FDS) [1] and self-piercing rivet (SPR) [2] connections. These new joining methods often exhibit very complicated forming process which is difficult to model by Lagrangian finite element methods. For example, modeling the extreme thread forming procedure in the FDS driving process involves extensive plastic deformation and material separation simulations so traditional finite element methods would therefore be very

✉ C. T. Wu
ctwu@lstc.com

¹ Computational and Multiscale Mechanics Group, Livermore Software Technology Corporation, 7374 Las Positas Road, Livermore, CA 94551, USA

challenging. Specifically, the C^0 -continuity assumption in most finite element methods is unable to describe the kinematic discontinuity of displacement fields for material separation simulation [3]. While the element deletion technique can be applied to reduce excessive straining and mesh tangling problems caused by the C^0 -continuity assumption in the material separation simulation, it gives another instance of numerical instability associated with the loss of conservation properties in mass and linear momentum. Furthermore, due to material erosion, the desired deformed shape such as the threaded boss might not be formed at all. As a consequence, the numerical result could become very problematic and parameter sensitive in the FDS driving simulation.

Although the Eulerian finite element method can be easily applied to solid mechanics applications to overcome the mesh distortion problem, the Eulerian representation of material flow presents other numerical difficulties in tracking the material points and free surfaces in the FDS driving simulation. Unlike the Eulerian finite element method, the arbitrary Lagrangian–Eulerian (ALE) method advances the computational mesh inside the domain arbitrarily while keeps the mesh on the boundary moving along with material flow. Despite this advantage offered by the ALE method to handle the free surface problem, its drawback is the manifestation of numerical oscillations when the convective effect is dominant in the governing equations. This numerical instability often arises when the velocity difference between mesh movement and material flow becomes pronounced. As a generalization of Eulerian approach, the ALE method also has troubles to model the formation of new surfaces in the course of material separation processes [4]. In consequence, extant finite element literatures [5, 6] for modeling FDS thread forming process and its failure characteristics are very limited. It is clear that current finite element modeling strategy for FDS thread forming simulation is not sufficient as it could greatly impede the overall prediction of load-bearing structures in the vehicle’s virtual development process.

Alternatively, Lagrangian particle or meshfree methods [7] have grown in popularity as a practical numerical tool in industrial design and applications. Despite the generic name “meshfree”, not all meshfree methods are truly meshless. Some meshfree methods, particularly those based on the Galerkin method, actually require an auxiliary cell structure called “integration cells” or “background meshes” for the domain integration of weak forms. Although those methods generally are more accurate and stable than truly particle methods, it is difficult for them to deal with the cell distortion issue present in severe deformation and material separation applications. Indeed, Lagrangian particle methods based on background meshes pose significant challenges from both mathematical formulation and programming aspects in dealing with severe deformation problems. Therefore, it

is important to develop a stable and accurate particle method that obviates the inherent limitation of cell-based meshfree methods for FDS thread forming simulations. However, it is widely known that most Lagrangian particle methods are susceptible to several numerical deficiencies in solid mechanics applications. While issues of approximation consistency [8–10], essential boundary conditions [11–13], spurious energy modes [14, 15] and tension instability [11, 16, 17] have been gradually resolved for particle methods, the numerical issues in severe deformation and material separation applications remain to be addressed.

The smoothed particle Galerkin (SPG) method developed by Wu et al. [3, 4, 18, 19] is one of the new stabilized Lagrangian particle methods aims to bypass the need of background mesh for the ductile material failure simulation. The early development of SPG method [3, 4, 18, 19] introduces first-order strain gradients by means of velocity (or displacement) smoothing to achieve the stabilization effect. This SPG stabilization formulation requires two distinct but coinciding integration points [19] per particle for integrating the weak form. Based on Chen’s implicit gradient expansion [20] and Liu’s strain gradient stabilization technique [21], Hillman and Chen [22] proposed another stabilized Lagrangian particle method for high-speed impact applications. Similar strain gradient stabilization technique was also considered by Wu et al. [23] to study the friction drilling application. Those particle methods share a common feature in augmenting the standard quadratic energy functional by several stabilization terms containing first-order strain gradients to stabilize the results in severe deformation simulation. Because the stabilization in those particle methods is accomplished without the use of momentum equation residual, they belong to the non-residual stabilization methods. In order to integrate those non-residual stabilization terms, multiple integration points that are matching at each particle are needed. Although those particle methods can sufficiently control spurious energy modes in nonlinear analysis, they are not computationally efficient. Additionally, since strain gradient stabilization techniques used in those non-residual stabilization methods were initially derived based on the linear elasticity assumption [3], the evaluation of stabilization stress in nonlinear material models is therefore not straightforward. Following the stabilization work of Belytschko and Bindeman [24] in nonlinear finite element methods, a modified shear modulus [3, 4, 19, 22, 23] is usually specified to replace the elastic modulus in those non-residual stabilization terms for the stress update.

In order to improve the computational efficiency as well as to avoid the fundamental complication in the stabilization stress calculation, a new particle stabilization method [25] was recently proposed by the same authors of this manuscript. In this new stabilization method, a special velocity smoothing algorithm was introduced to replace the direct

velocity smoothing in the original SPG stabilization method [3, 4, 18, 19]. This leads to a new stabilization formulation without using residual or non-residual stabilization terms. In other words, the new method only requires one integration point per particle. As a result, the specification of modified shear modulus and the evaluation of stabilization stress can be completely avoided, thus cutting the computational cost. As the new method based on the smoothed velocity field is consistently fulfilling the conservation of linear and angular momentum, we call it the momentum-consistent smoothed particle Galerkin (MC-SPG) method in this paper.

The object of this study is to apply the MC-SPG formulation to study the extreme thread forming procedure that involves large deformation and material separation in the FDS driving process. The reminder of the paper is organized as follows: The velocity smoothing algorithm in MC-SPG formulations is described in Sect. 2. In Sect. 3, additional numerical treatments to simulate the severe deformation and material separation problems are provided. Several numerical benchmarks including one thread forming simulation are given in Sect. 4. Conclusions are drawn in Sect. 5.

2 The velocity smoothing algorithm and MC-SPG formulations

Let’s first recall the weak form of the dynamic equation of motion as follows:

$$\int_{\Omega} \rho \ddot{\mathbf{u}}^h \cdot \delta \mathbf{u}^h d\Omega + \int_{\Omega} \boldsymbol{\sigma} \cdot \nabla \delta \mathbf{u}^h d\Omega - \int_{\Omega} \mathbf{b} \cdot \delta \mathbf{u}^h d\Omega - \int_{\partial\Omega^h} \mathbf{h} \cdot \delta \mathbf{u}^h = 0 \tag{1}$$

with initial conditions

$$\mathbf{u}^h(\mathbf{X}, 0) = \mathbf{u}_0(\mathbf{X}) \tag{2}$$

$$\dot{\mathbf{u}}^h(\mathbf{X}, 0) = \dot{\mathbf{u}}_0(\mathbf{X}) \tag{3}$$

where ρ is the current mass density, $\boldsymbol{\sigma}$ is the Cauchy stress, \mathbf{b} is the body force density measured in current configuration and \mathbf{h} is the prescribed traction on the current boundary $\partial\Omega^h$. \mathbf{u}^h is the approximation of the displacement field \mathbf{u} , and $\ddot{\mathbf{u}}^h$ is the corresponding acceleration. For a particle distribution denoted by an index set $Z_I = \{\mathbf{X}_I\}_{I=1}^{NP}$, approximating the displacement field \mathbf{u} at time t using the first-order meshfree approximations [11, 12, 26] gives

$$\mathbf{u}^h(\mathbf{X}, t) = \sum_{I \in Z_I} \phi_I^a(\mathbf{X}) \mathbf{u}_I(\mathbf{X}, t) = \sum_{I \in Z_I} \phi_I^a(\mathbf{X}) \mathbf{u}_I \tag{4}$$

where NP is the total number of particles in the discretization. $\phi_I^a(\mathbf{X})$, $I = 1, \dots, NP$ can be regarded as the Lagrangian shape functions for displacement field \mathbf{u}^h where the superscript “ a ” denotes the support size of particle I . $\dot{\mathbf{u}}_I$ is the velocity at particle I . The introduction of Eq. (4) into

Eq. (1) using the direct nodal integration (DNI) scheme gives the following standard semi-discrete equation to be solved numerically in Lagrangian particle methods.

$$\mathbf{M}^{\text{lump}} \ddot{\mathbf{U}} = \mathbf{F}^{\text{ext}} - \mathbf{F}^{\text{int}} \tag{5}$$

where \mathbf{M}^{lump} is the lumped mass matrix in diagonal form. The construction of mass matrix and the mass lumping scheme follows our previous works in [3, 4, 23]. $\ddot{\mathbf{U}}$ is a vector containing all particle accelerations, and \mathbf{F}^{ext} is the standard external force vector. \mathbf{F}^{int} is the regular internal force term given by

$$\mathbf{F}_I^{\text{int}} = \sum_{J \in Z_I} \mathbf{B}_I^T(\mathbf{X}_J) \boldsymbol{\Xi}(\mathbf{X}_J) V_J \tag{6}$$

where \mathbf{B}_I is the standard displacement gradient matrix, $\boldsymbol{\Xi}(\mathbf{X}_J) = [\sigma_{xx} \ \sigma_{yy} \ \sigma_{zz} \ \sigma_{yz} \ \sigma_{xz} \ \sigma_{xy}]_{X_J}^T$ is the component-wise stress vector of particle J and V_J is volume of particle J .

It is known that the solution of Eq. (5) exhibits spurious energy modes in the displacement field. It is also intuitively evident that the oscillation solution can possibly be stabilized by a smoothing algorithm. These observations prompt applications of velocity smoothing in Lagrangian particle methods as a way to eliminate the spurious energy modes. In the original SPG method [3, 4, 19], this idea is justified by an introduction of a direct velocity smoothing algorithm for the particle velocity $\dot{\mathbf{u}}_I$ as described by

$$\dot{\mathbf{u}}_I = \sum_{J \in Z_I} \phi_J^a(\mathbf{X}_I) \hat{\mathbf{u}}_J \tag{7}$$

where $\hat{\mathbf{u}}_J$ is the unsmoothed (oscillating) velocity at particle J and $\phi_J^a(\mathbf{X}_I)$ is the smoothing function which is same as that in Eq. (4). It is obvious that a blunt use of Eq. (7) to stabilize the solution in Eq. (5) by means of a post-processing procedure may cause a loss of conservation properties for linear and angular momentum. For this reason, a non-residual stabilization term was derived using Eq. (7) to reach the stabilization effect in the original SPG method [3, 4, 19]. This leads to the following new system of equations to solve in the original SPG method

$$\mathbf{M}^{\text{lump}} \ddot{\mathbf{U}} = \mathbf{F}^{\text{ext}} - \mathbf{F}^{\text{int}} - \mathbf{F}^{\text{stab}} \tag{8}$$

where

$$\mathbf{F}_I^{\text{stab}} = \sum_{J \in Z_I} \tilde{\mathbf{B}}_I^T(\mathbf{X}_J) \tilde{\boldsymbol{\sigma}}(\mathbf{X}_J) V_J \tag{9}$$

$\tilde{\mathbf{B}}_I$ is the displacement gradient matrix which contains the first-order strain gradient (see [3, 4, 19]) for computation of the stabilization force \mathbf{F}^{stab} . The stabilization stress $\tilde{\boldsymbol{\sigma}}$ in small strain analysis is formulated using a material response tensor (elasto-plastic tangent modulus) \mathbf{C}^σ as given by [3, 4, 19]

$$\tilde{\boldsymbol{\sigma}} = \mathbf{C}^\sigma : (\nabla \boldsymbol{\varepsilon}(\hat{\mathbf{u}}) \cdot \boldsymbol{\lambda}^b) \tag{10}$$

where λ^b is the coefficient matrix [3, 4, 19] for stabilization. It should be noted that the coefficient matrix does not contain artificial stabilization control parameter in the original SPG method. The stabilization term F^{stab} in Eq. (8) contains the first-order strain gradient [3, 4, 19] which can be considered as suitable gradient jump conditions to stabilize the oscillating solution. Actually, the idea of adding a term penalizing the jump of the gradient to stabilize the solution is not new in the computational fluid dynamics (CFD) community. Pioneering approaches to adopt this idea in solid mechanics applications have been demonstrated by Beissel and Belytschko [27] and Onate et al. [28] for the elasticity analysis.

In practice, however, Eq. (10) cannot be directly applied to material nonlinearity analysis. This is because the numerical evaluation of stabilized stress vector using Eq. (10) involves the elasto-plastic tangent modulus and is computationally unfeasible in the explicit dynamics analysis. For this reason, a modified shear modulus \tilde{G} is often utilized to replace the elasto-plastic tangent modulus according to the suggestion of Belytschko and Bindeman [24] in the finite element stabilization method, which is given by

$$2\tilde{G} = \sqrt{\frac{H_{\Delta\tau}}{H_{\Delta e}}} \tag{11}$$

$$H_{\Delta\tau} = \frac{1}{2} \sum_{i=1}^3 \sum_{j=1}^3 \Delta\tau_{ij} \Delta\tau_{ij}, \quad H_{\Delta e} = \frac{1}{2} \sum_{i=1}^3 \sum_{j=1}^3 \Delta e_{ij} \Delta e_{ij} \tag{12}$$

where $\Delta\tau_{ij}$ and Δe_{ij} are the components of the deviatoric part of the stress and strain increments, respectively.

In contrast to the reconstruction of high-order derivatives in previous gradient stabilization methods, the second type of stabilization methods is the reformulation of the divergence form of governing equations as a first-order system. This, of course, leads to more unknowns and larger problems to solve. Representative CFD approaches in this type of stabilization methods for solid mechanics applications include Taylor–Galerkin stabilization algorithm [29], Jameson–Schmidt–Turkel algorithm [30] and streamline upwind Petrov–Galerkin algorithm [31]. The major disadvantage of these CFD stabilization algorithms is the contradictory demands on an artificial stabilization control parameter placed by the accuracy requirement. It is worth mentioning that improper selection of the stabilization control parameter may lead to the loss of stability properties particularly in the nonlinear material analysis. Moreover, because most of those stabilization methods are formulated in a total Lagrangian framework, their extension to severe deformation and material separation analysis still needs to be developed.

Although the original SPG method [3, 4, 19] eliminates the need of artificial stabilization control parameter, the computation of stabilization stress for material nonlinearity problems is not straightforward and its performance in severe deformation and material failure analysis remains to be investigated. Furthermore, because the stabilization term in Eq. (8) needs to be integrated independently to achieve the stabilization effect, the method also demands additional computational cost. To resolve those issues in the SPG method, a third type of particle stabilization method, the MC-SPG method [25], was developed. As its name indicates, the particle velocity $\dot{\mathbf{u}}_I$ in the MC-SPG method is smoothed via a special algorithm associated with the linear momentum, and this is defined in the following

$$\begin{aligned} \dot{\mathbf{u}}_I = \dot{\mathbf{u}}(X_I) &:= \frac{\mathbf{P}_I}{m_I} = \frac{\sum_{J \in Z_I} \hat{m}_J \phi_I^a(X_J) \hat{\mathbf{u}}_J}{\sum_{J \in Z_I} \hat{m}_J \phi_I^a(X_J)} = \sum_{J \in Z_I} \left(\frac{\hat{m}_J \phi_I^a(X_J)}{\sum_{K \in Z_I} \hat{m}_K \phi_I^a(X_K)} \right) \hat{\mathbf{u}}_J \\ &= \sum_{J \in Z_I} \psi_I^a(X_J) \hat{\mathbf{u}}_J, \quad X_I \in Z_I \end{aligned} \tag{13}$$

where \mathbf{P}_I is the smoothed linear momentum of particle I and will be defined in Eq. (19) and

$$\psi_I^a(X_J) = \frac{\hat{m}_J \phi_I^a(X_J)}{\sum_{K \in Z_I} \hat{m}_K \phi_I^a(X_K)} \tag{14}$$

and $\psi_I^a(X_J)$ can be viewed as a modified smoothing function in contrast to the smoothing function $\phi_J^a(X_I)$ in Eq. (7) in the original SPG method [3, 4, 19]. Equation (13) also defines a standard linear mapping between two particle systems $\hat{\mathbf{U}}$ and \mathbf{U} . Let $\Theta_h : L^2 \mapsto L^2$ be a discrete operator for the velocity smoothing, we have

$$\mathbf{U}(X_I) = \Theta_h(\hat{\mathbf{U}}(X_I)) \quad X_I \in Z_I \tag{15}$$

or in matrix form

$$\mathbf{U} = \mathbf{T} \hat{\mathbf{U}}, \quad T_{IJ} = \psi_I^a(X_J) \mathbf{I} \tag{16}$$

Equation (15) describes the computation of a global L^2 projection. Subsequently, Eq. (16) gives

$$\hat{\mathbf{U}} = \mathbf{T}^{-1} \mathbf{U}, \quad \hat{\mathbf{u}}_I = \sum_{J \in Z_I} T_{IJ}^{-T} \mathbf{U}_J \tag{17}$$

which theoretically should be used for the velocity update. The two particle systems $\hat{\mathbf{U}}$ and \mathbf{U} are collections of particle values $\dot{\mathbf{u}}_I$ and $\hat{\mathbf{u}}_I$, respectively. It is clear that Eq. (17) involves a global matrix inversion which is computationally expansive when the modified smoothing function $\psi_I^a(X_J)$ is updated through time during the large deformation analysis. In what follows we will replace Eq. (17) by the following approximation for the velocity update.

$$\hat{\mathbf{u}}_I \approx \sum_{J \in Z_I} \phi_J^a(X_I) \dot{\mathbf{u}}_J \tag{18}$$

It is important to note that utilization of this modification in the velocity updating procedure will still yield the conservation of linear and angular momentum results as will be shown later.

Finally, the smoothed linear momentum \mathbf{P}_I and smoothed mass M_I of particle I in Eq. (13) are defined by

$$\mathbf{P}_I := \sum_{J \in Z_I} \hat{m}_J \phi_I^a(\mathbf{X}_J) \hat{\mathbf{u}}_J \tag{19}$$

$$m_I := \sum_{J \in Z_I} \hat{m}_J \phi_I^a(\mathbf{X}_J) \tag{20}$$

where \hat{m}_J in Eqs. (19) and (20) is the unsmoothed lumped mass of particle J . The value of \hat{m}_J is computed directly using the direct nodal or Gauss integration schemes at $t=0$ as described in [23] and does not change with time.

It is essential that Eq. (13) and Eqs. (19)–(20) employ a “gather” type [32] of smoothing function $\phi_I(\mathbf{X}_J)$ instead of the “scatter” type of smoothing function $\phi_J(\mathbf{X}_I)$ as shown in the original SPG method [see, Eq. (7)]. In general case, $\phi_J(\mathbf{X}_I)$ and $\phi_I(\mathbf{X}_J)$ are not equal. The employment of gather type of smoothing function yields the following equality which is critical to the preservation of linear and angular momentum in the MC-SPG method.

$$\sum_{I \in Z_I} m_I = \sum_{I \in Z_I} \sum_{J \in Z_I} \hat{m}_J \phi_I^a(\mathbf{X}_J) = \sum_{J \in Z_I} \hat{m}_J \left(\sum_{I \in Z_I} \phi_I^a(\mathbf{X}_J) \right) = \sum_{J \in Z_I} \hat{m}_J \tag{21}$$

Equation (21) indicates that the global mass is preserved under the gather type of mass smoothing. It also can be verified that the scatter type of mass smoothing algorithm does not necessarily ensure the conservation of global mass. Now let’s recall the momentum of particle I before velocity smoothing to be

$$\hat{\mathbf{P}}_I = \hat{m}_I \hat{\mathbf{u}}_I \tag{22}$$

Using Eqs. (13), (18)–(22), it is straightforward to prove that the global linear momentum \mathbf{P} is conserved, that is

$$\mathbf{P} = \sum_{I \in Z_I} \mathbf{P}_I = \sum_{I \in Z_I} \hat{\mathbf{P}}_I = \hat{\mathbf{P}} \tag{23}$$

Subsequently, it is also not difficulty to show (see [25] for proof) that the following global conservation of angular moment is valid

$$\sum_{I \in Z_I} \mathbf{x}(\mathbf{X}_I) \times m_I \dot{\mathbf{u}}_I = \sum_{I \in Z_I} \mathbf{x}(\mathbf{X}_I) \times \hat{m}_I \hat{\mathbf{u}}_I \tag{24}$$

When the central difference integration scheme is used for the temporal discretization, Eq. (13) becomes

$$\dot{\mathbf{u}}_I^{n+1/2} = \sum_{J \in Z_I} \psi_I^a(\mathbf{X}_J) \hat{\mathbf{u}}_J^{n+1/2} \tag{25}$$

Using Eq. (25), the particle density can be updated by

$$\rho_I^{n+1} = \rho_I^n \left(1 - \Delta t_{n+1/2} \nabla \cdot \dot{\mathbf{u}}_I^{n+1/2} \right) \tag{26}$$

where $\Delta t_{n+1/2} = t_{n+1} - t_n$. With the updated particle density in Eq. (26), the new particle volume can also be obtained by

$$V_I^{n+1} = \hat{m}_I / \rho_I^{n+1} \tag{27}$$

Consequently, the particle displacements can be updated using Eq. (25) by

$$\mathbf{u}_I^{n+1} = \mathbf{u}_I^n + \Delta t_{n+1/2} \dot{\mathbf{u}}_I^{n+1/2} \tag{28}$$

The particle strain rate $\dot{\epsilon}_I = \nabla^s(\dot{\mathbf{u}}_I)$ also can be computed accordingly. The concept of gather type of velocity smoothing in the MC-SPG method is illustrated in Fig. 1.

Now it is necessary to update the unsmoothed particle velocity in Eq. (25) during the time stepping. Let’s first compute the increment of unsmoothed particle velocity using the modified unsmoothed particle velocity in Eq. (18) and the (smoothed) acceleration from Eq. (5) to yield

$$\Delta \hat{\mathbf{u}}_I^n \approx \Delta t_n \sum_{J \in Z_I} \phi_J^a(\mathbf{X}_I) \ddot{\mathbf{u}}_J^n \tag{29}$$

where $\Delta t_n = t_{n+1/2} - t_{n-1/2}$. The update of unsmoothed particle velocity is accomplished by

$$\begin{aligned} \hat{\mathbf{u}}_I^{n+1/2} &= \hat{\mathbf{u}}_I^{n-1/2} + \Delta \hat{\mathbf{u}}_I^n \\ &= \hat{\mathbf{u}}_I^{n-1/2} + \Delta t_n \sum_{J \in Z_I} T_{IJ}^{-T}(\mathbf{X}_I) \ddot{\mathbf{U}}_J^n \\ &\approx \hat{\mathbf{u}}_I^{n-1/2} + \Delta t_n \sum_{J \in Z_I} \phi_J^a(\mathbf{X}_I) \ddot{\mathbf{u}}_J^n \end{aligned} \tag{30}$$

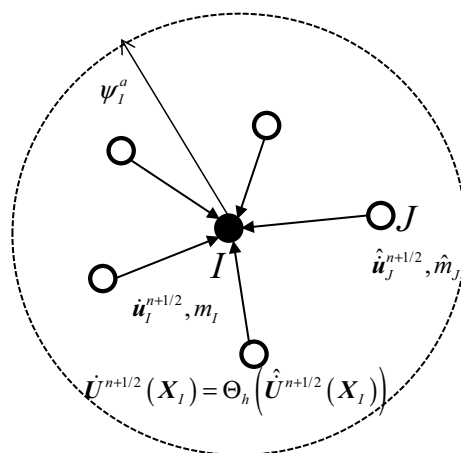


Fig. 1 Illustration of a gather type of velocity smoothing in the MC-SPG method

Using Eqs. (19), (20) and (30), the global linear momentum becomes

$$\begin{aligned}
 \hat{\mathbf{P}}^{n+1/2} &= \sum_{I \in Z_I} \hat{\mathbf{P}}_I^{n+1/2} = \sum_{I \in Z_I} \hat{m}_I \hat{\mathbf{u}}_I^{n+1/2} \\
 &= \sum_{I \in Z_I} \hat{m}_I \hat{\mathbf{u}}_I^{n-1/2} + \Delta t_n \sum_{I \in Z_I} \sum_{J \in Z_I} \hat{m}_I \phi_J^a(X_I) \ddot{\mathbf{u}}_J^n \\
 &= \mathbf{P}^{n-1/2} + \Delta t_n \sum_{J \in Z_I} \left(\sum_{I \in Z_I} \hat{m}_I \phi_J^a(X_I) \right) \ddot{\mathbf{u}}_J^n \quad (31) \\
 &= \mathbf{P}^{n-1/2} + \Delta t_n \sum_{J \in Z_I} m_J \ddot{\mathbf{u}}_J^n \\
 &= \mathbf{P}^{n-1/2} + \Delta \mathbf{p}^n \\
 &= \mathbf{P}^{n+1/2}
 \end{aligned}$$

which verifies that the present velocity smoothing algorithm is consistently preserving the global linear momentum in the explicit time integration scheme. The preservation of global angular momentum also can be shown analogously (see [25] for proof). This smoothed velocity update algorithm is illustrated in Fig. 2.

It is obvious that the stabilization scheme in the MC-SPG method does not involve first or higher order strain gradient terms as needed in the residual and non-residual

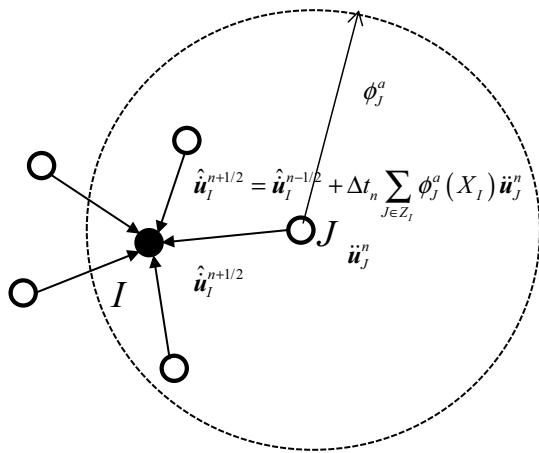


Fig. 2 Illustration of scatter type of velocity update in MC-SPG method

Table 1 The difference between the SPG and MC-SPG methods

	SPG method	MC-SPG method
Velocity smoothing	$\dot{\mathbf{u}}_I = \sum_{J \in Z_I} \phi_J^a(X_I) \hat{\mathbf{u}}_J$	$\dot{\mathbf{u}}_I = \sum_{J \in Z_I} \psi_J^a(X_I) \hat{\mathbf{u}}_J$
Final discrete equation	$\mathbf{M}^{\text{lump}} \ddot{\mathbf{U}} = \mathbf{F}^{\text{ext}} - \mathbf{F}^{\text{int}} - \mathbf{F}^{\text{stab}}$	$\mathbf{M}^{\text{lump}} \ddot{\mathbf{U}} = \mathbf{F}^{\text{ext}} - \mathbf{F}^{\text{int}}$
Stabilization stress	Yes	No
Integration points/particle	Two	One

type of stabilization methods [3, 4, 19]. Because of that, the MC-SPG method only requires a single integration point at each particle, and therefore, the method is computationally more efficient than the original SPG method. As the stress updating procedure can be performed regularly at each particle using the standard material constitutive law, no special specification of elastic modulus for stabilization stress calculation is needed like that in the non-residual type of stabilization methods. The major difference between the SPG and MC-SPG methods is summarized in Table 1.

3 The numerical treatments for severe deformation and material separation analysis

In this section, the stabilization formulation described in the previous section is revised to cover the application in simulating the extensive plastic deformation and material separation problems during the severe FDS thread forming process. This is done by introducing two additional numerical features, the adaptive anisotropic Lagrangian kernel and the bond-based failure criterion originally proposed in the SPG method [3, 4, 19], to the MC-SPG method.

3.1 The adaptive anisotropic Lagrangian kernel for severe deformation analysis

It is recalled that Lagrangian shape functions, $\phi_I^a(\mathbf{X}), I = 1, \dots, \text{NP}$, constructed in Eq. (4) utilize Lagrangian kernels to remove the tension instability [11] in the nonlinear structural analysis. However, particle methods based on Lagrangian shape functions experience the excessive straining problem when the strict use of Lagrangian kernel is no longer applicable. Specifically, the excessive straining during the severe plastic deformation simulation in the FDS thread forming process inevitably causes the numerical breakdown when the deformation gradient computed at the particle ceases to become invertible. It presents the same issue of mesh distortion problems in the finite element method.

In order to handle the excessive straining problem in the MC-SPG method, an adaptive anisotropic Lagrangian kernel is considered [3, 4, 19]. Using the chain rule, the calculation

for the deformation gradient at the particle can be rewritten as

$$\mathbf{F}^{n+m} = \widehat{\mathbf{F}}^{n+m} \mathbf{F}^n \tag{32}$$

where $\widehat{\mathbf{F}}^{n+m}(\widehat{\mathbf{x}})$ is the decomposed deformation gradient, from $t = t_n$ to t_{n+m} , computed based on the new reference configuration and is given by

$$\begin{aligned} \widehat{F}_{ij}^{n+m}(\mathbf{X}_J) &= \frac{\partial \widehat{x}_i}{\partial \widehat{X}_j} = \sum_{I \in \mathcal{Z}_I} \frac{\partial \phi_I^a(\widehat{\mathbf{X}}_J)}{\partial \widehat{X}_j} \widehat{x}_{iI}(\mathbf{X}, t_{n+m}) \\ &= \sum_{I \in \mathcal{Z}_I} \frac{\partial \phi_I^a(\widehat{\mathbf{X}}_J)}{\partial \widehat{X}_j} \left(\widehat{X}_{iI} + \widehat{u}_{iI}(\mathbf{X}, t_{n+m}) \right) \\ &= \delta_{ij} + \sum_{I \in \mathcal{Z}_I} \frac{\partial \phi_I^a(\widehat{\mathbf{X}}_J)}{\partial \widehat{X}_j} \widehat{u}_{iI}(\mathbf{X}, t_{n+m}) \end{aligned} \tag{33}$$

Here, $\widehat{\mathbf{x}} = \widehat{\mathbf{X}} + \widehat{\mathbf{u}}(\mathbf{X}, t_{n+m})$ is a position vector defined in the new reference configuration, i.e., $\widehat{\mathbf{X}} = \mathbf{x}(\mathbf{X}, t_n)$. A local $\widehat{\mathbf{X}}$ -coordinate system in which the axes are parallel to the global Cartesian coordinates whose origin located at $\widehat{\mathbf{X}}_I$ is defined for each particle I . In each new reference configuration, an ellipsoidal nodal support is defined for the neighbor particle searching. The three-dimensional ellipsoidal cubic spline kernel function is defined in another local $\widehat{\mathbf{X}}$ -coordinate system by

$$\varphi_I^a(\widehat{\mathbf{X}}_J) = \phi_1 \left(\frac{\widehat{X}_J}{h_1^n} \right) \phi_1 \left(\frac{\widehat{Y}_J}{h_2^n} \right) \phi_1 \left(\frac{\widehat{Z}_J}{h_3^n} \right) \tag{34}$$

where ϕ_1 is a standard one-dimensional cubic spline kernel function, h_1^n, h_2^n and h_3^n are the current semi-major axes of the ellipsoid. The sizes of semi-major axes can be considered the support sizes of the kernel and are updated according to the deformation [19]. $\widehat{X}_J, \widehat{Y}_J$ and \widehat{Z}_J are the projections of relative position vector $\widehat{\mathbf{X}}_J - \widehat{\mathbf{X}}_I$ on the local $\widehat{\mathbf{X}}$ -coordinate system, respectively. The adaptive anisotropic Lagrangian kernel is updated constantly over a period of time. The spherical shape domain of cubic spline kernel function deforms and rotates according to the Lagrangian motion between each two adaptive Lagrangian kernel steps. We

address the reader to Ref. [19] for a comprehensive description of the approach. For the computational efficiency in explicit time integration, the material derivatives of mesh-free shape functions are always computed and stored at the new reference configuration and reused during the time stepping.

Since the operation of adaptive anisotropic Lagrangian kernel is performed at the particle and does not involve remeshing, the stress-recovery techniques or remapping procedures are not necessary. This unique property of present approach leads to a relatively simple mathematical formulation for simulating the severe plastic deformation problem.

3.2 The bond-based failure criterion for material separation analysis

Excessive straining also appears in the FDS thread forming process when the material starts to fail at the thread forming stage. Precisely, the C^1 -continuity assumption in Lagrangian particle methods is inadequate to describe the kinematic discontinuity of displacement field in a continuous setting for the failure analysis [3]. This makes conventional Lagrangian particle methods very challenging in simulating the extreme thread forming during the FDS driving process.

To further avoid the excessive straining problem due to the assumption of continuous displacement field in the thread forming simulation, a bond-based failure criterion [3, 23] is incorporated with the present stabilization formulation. The origins of this approach can be traced back to the bond failure in the bond-based peridynamics [33, 34] in which material failure is modeled through bond breakage. In Lagrangian particle methods, the bond is a representation of a connection between two particles. Given a length of the bond $\|\mathbf{X}_J - \mathbf{X}_I\|$ for a particle pair consisting of particles I and J in the initial configuration, the stretch ratio e_{IJ} of the bond is defined by

$$e_{IJ} = \frac{\|\mathbf{x}_J - \mathbf{x}_I\|}{\|\mathbf{X}_J - \mathbf{X}_I\|} \tag{35}$$

For the thread forming simulation, we restrict our attention to the material failure in metals. In this bond-based failure criterion, two neighbor particles are considered disconnected during the neighbor particle sorting whenever their averaged effective plastic strain and stretch ratio reach their respective critical values. Accordingly, the three-dimensional ellipsoidal cubic spline kernel function in Eq. (29) for a pair of particles I and J can be modified as:

$$\varphi_I^a(\widehat{X}_J) = \begin{cases} 0, & \text{if } \widehat{X}_J \notin \sup p(\varphi_I^a) \text{ or } (\bar{\varepsilon}_{IJ}^P > \bar{\varepsilon}_{crit}^P \text{ and } e_{IJ} > e_{crit}) \\ \varphi_1\left(\frac{\widehat{X}_J}{h_1^a}\right)\varphi_1\left(\frac{\widehat{Y}_J}{h_2^a}\right)\varphi_1\left(\frac{\widehat{Z}_J}{h_3^a}\right), & \text{otherwise} \end{cases} \tag{36}$$

where $\bar{\varepsilon}_{IJ}^P = \left(\bar{\varepsilon}^P(\widehat{X}_I) + \bar{\varepsilon}^P(\widehat{X}_J) \right) / 2$ represents an averaged effective plastic strain in the bond $\|X_I - X_J\|$ and $\bar{\varepsilon}^P$ is the effective plastic strain. $\bar{\varepsilon}_{crit}^P$ is the critical effective plastic strain for bond failure, and e_{crit} denotes the critical stretch ratio. We consider $e_{crit} \geq 1.0$ in our numerical analysis which implies that the bond failure does not occur under compression. This implication is valid for most metal failure process.

Because the effective plastic strain at each particle is monotonically increasing during the course of deformation, the kinematic disconnection in a particle pair is permanent and irreversible. This is a substantial characteristic for the present bond-based failure mechanism in metal failure analyses since the non-physical material self-healing issues resulting from generic neighbor searching algorithm can also be completely excluded from the material failure simulation.

4 Numerical examples

The present MC-SPG formulation was recently implemented into the commercial software LS-DYNA® [35] by the authors. To demonstrate the accuracy and efficiency of the present method, three examples are studied in this section using the LS-DYNA code. The first example is a small deformation wave propagation problem with comparison to the analytical solution. The second example is a large deformation Taylor bar impact simulation with comparison to the experimental data. The last example is used to demonstrate the numerical ability in the extreme thread forming simulation considering material failure and separation. Convergence study is conducted for all examples to inspect the stability of the present method in the nonlinear analysis.

4.1 Elastic wave propagation problem

We will now test how well the present method can capture the wave propagation in a dynamic but small deformation event. To access stability, the convergence of a dimensionless elastic wave propagation problem is considered in the following. The test problem can be stated as:

$$\rho \ddot{u} = E u_{,xx} \quad \text{on }]0, L[\times]0, T[\tag{37}$$

with boundary and initial conditions given as:

$$\begin{aligned} u(0, t) &= 0, \quad t \in]0, T[\\ u_{,x}(L, t) &= 0 \\ u(X, 0) &= 0, \quad X \in]0, L[\\ \dot{u}(X, 0) &= -0.01 \end{aligned} \tag{38}$$

The analytical solution of the displacement and stress fields can be derived and given by:

$$\begin{aligned} u(X, t) &= \sum_{n=1}^{\infty} \left\{ A_n \sin(\omega_n t) \sin \left[(2n-1) \frac{\pi X}{2L} \right] \right\} \\ \sigma(X, t) &= E u_{,x}(X, t) = \sum_{n=1}^{\infty} \left\{ (2n-1) \frac{\pi}{2L} A_n \sin(\omega_n t) \cos \left[(2n-1) \frac{\pi X}{2L} \right] \right\} \end{aligned} \tag{39}$$

with

$$\omega_n = \frac{(2n-1)\pi}{2L} \sqrt{\frac{E}{\rho}} \quad A_n = \frac{8L\dot{u}(X, 0)}{(2n-1)^2 \pi^2} \sqrt{\frac{\rho}{E}} \tag{40}$$

To numerically solve the problem, a three-dimensional square rod is used which has a cross-section of 1.2×1.2 , $E = 100.0$, $\nu = 0.0$, $\rho = 100.0$ and $L = 10.0$. To analyze the convergence behavior of the proposed formulation, three discretizations are used with nodal distance of 0.2 (coarse), 0.1 (medium) and 0.05 (fine), respectively.

Figures 3 and 4 show the axial displacement at the free end and the axial stress at the mid-length of the rod, respectively. Both displacement and stress are seen to match analytical solution nicely. No apparent amplitude and phase errors are observed. Also, it is observed that the numerical solution converges to the analytical solution as discretization is refined.

4.2 Taylor bar impact problem

In this example, we report the stability performance of the present method in a large deformation problem. In this problem [36], a 6061-T6 aluminum alloy cylinder with diameter of 7.82 mm and length of 23.46 mm impacts perpendicularly to a rigid surface at an initial velocity of 373 m/s. In the test, the final cylinder length is deformed to about 16.46 mm. To study the converge performance of the proposed method, the cylinder is discretized into three models with nodal distance of approximately 0.49 mm (coarse, total 11,809 nodes), 0.33 mm (medium, total 38,325 nodes) and 0.25 mm

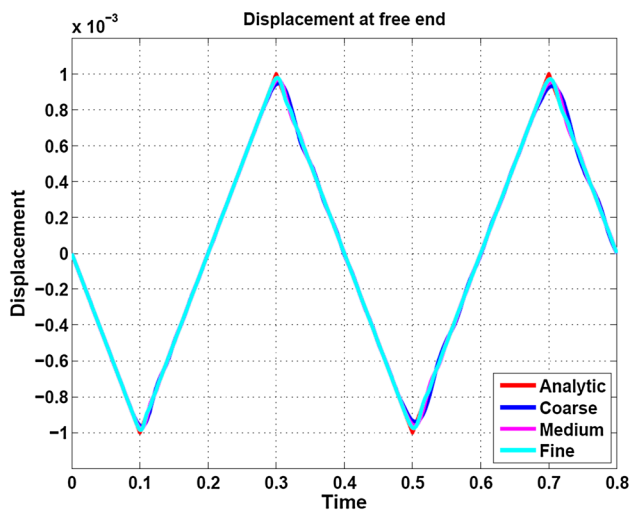


Fig. 3 Elastic wave problem: displacement at free end

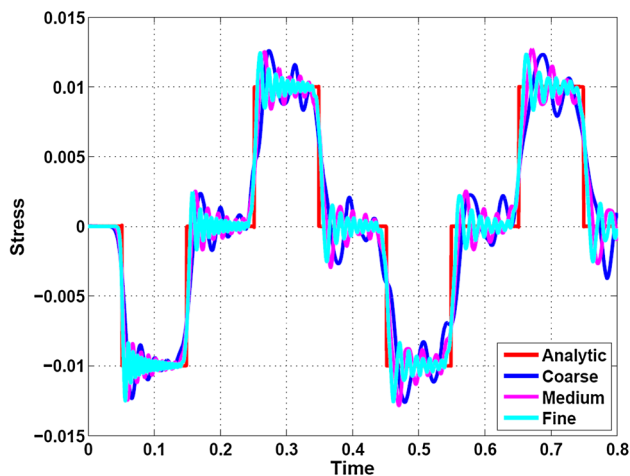


Fig. 4 Elastic wave problem: stress at mid-length

(fine, total 86,621 nodes), respectively. The cross-sectional discretization is shown in Fig. 5 (for clarity, connectivity is shown, but not used in calculation at all). Johnson–Cook material law [37] is employed to model the plastic flow of the material while the parameters are taken from Ref. [38].

Figure 6 shows the displacement history at the free end. It is observed that the numerical solution is converged and close to the experimental result. In fact, the numerical solution does not show too much sensitivity to the discretization, which demonstrates the accuracy of the present method in large deformation analysis.

Figure 7 shows the plastic strain distribution at termination. The contour plots in three different discretization models are in good agreement with each other. Figure 8 shows the final deformed shapes at the contact interface. A background mesh is attached to the deformation plot for

the demonstration purpose. No spurious energy mode is observed in the present solution for all three models. Over all, qualitatively consistent results are obtained.

4.3 FDS thread forming simulation

In this example, we will simulate the extreme thread forming in the FDS driving process. A complete FDS joining process is a 6-step manufacturing operation featured with large plastic flow and thermal mechanical responses. Generally, the FDS joining process includes heating, penetration, extrusion forming, thread forming, screw-driving and tightening steps [39]. As the thermal response is not of interest in the current study, the heating step and thermal effect are ignored in the simulation. The tightening step is not considered either since only one layer of workpiece is used in this thread forming simulation, which is for the purpose of illustration rather than actually joining two or more pieces of workpiece. Our focus in this simulation is the extreme thread forming on the workpiece after the fastener penetrates and drives through the workpiece.

As shown in Fig. 9a, b, a nominal M5x20 fastener is used as the screw in this study. It rotates at $\omega_0 = 5000$ rpm and plunges at $v_0 = 2.0$ m/s (Fig. 9c) during the penetration step and, thereafter, rotates at $\omega_0 = 20,000$ rpm and plunges at $v_0 = 0.2$ m/s until the end of screw-driving step. The pitch distance of the fastener is 0.7 mm. The fastener is modeled by finite element formulation (Fig. 9d) with rigid material.

The workpiece is a 20-mm-diameter and 1.5-mm-thick 6061-T6 aluminum alloy plate. The perimeter of the plate is clamped. To save some computational cost, only the central $\varnothing 7.0$ mm region, where most likely, large plastic flow, material failure and separation will occur, is modeled by the proposed MC-SPG formulation while the remaining majority is modeled by the traditional finite element method since failure is unlikely to occur in this region. The coupling between the MC-SPG and finite element is simply handled by common nodes and the feasibility of this coupling scheme was studied in [25]. To study the convergence behavior of the MC-SPG formulation, four discretizations are used, namely nodal distance (in the central region) 0.15 mm (total 21,131 nodes and 36,000 elements), 0.125 mm (total 35,269 nodes and 63,360 elements), 0.10 mm (total 67,344 nodes and 127,680 elements) and 0.075 mm (total 157,773 nodes and 276,480 elements), respectively. The cross-sectional nodal distribution at the central region is demonstrated in Fig. 10.

The plate is constitutively modeled by Johnson–Cook material law, and the same parameters as the previous example are used. The material failure and separation is handled by the bond failure mechanism described in Sect. 3.2. The critical effective plastic strain and stretch ratio for bond failure is 0.4 and 1.15, respectively. The interaction between the fastener and the metal plate is taken care by the classic

Fig. 5 Taylor impact: discretizations

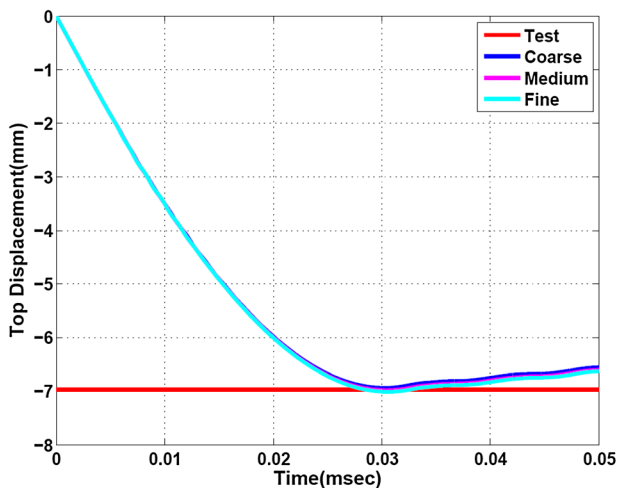
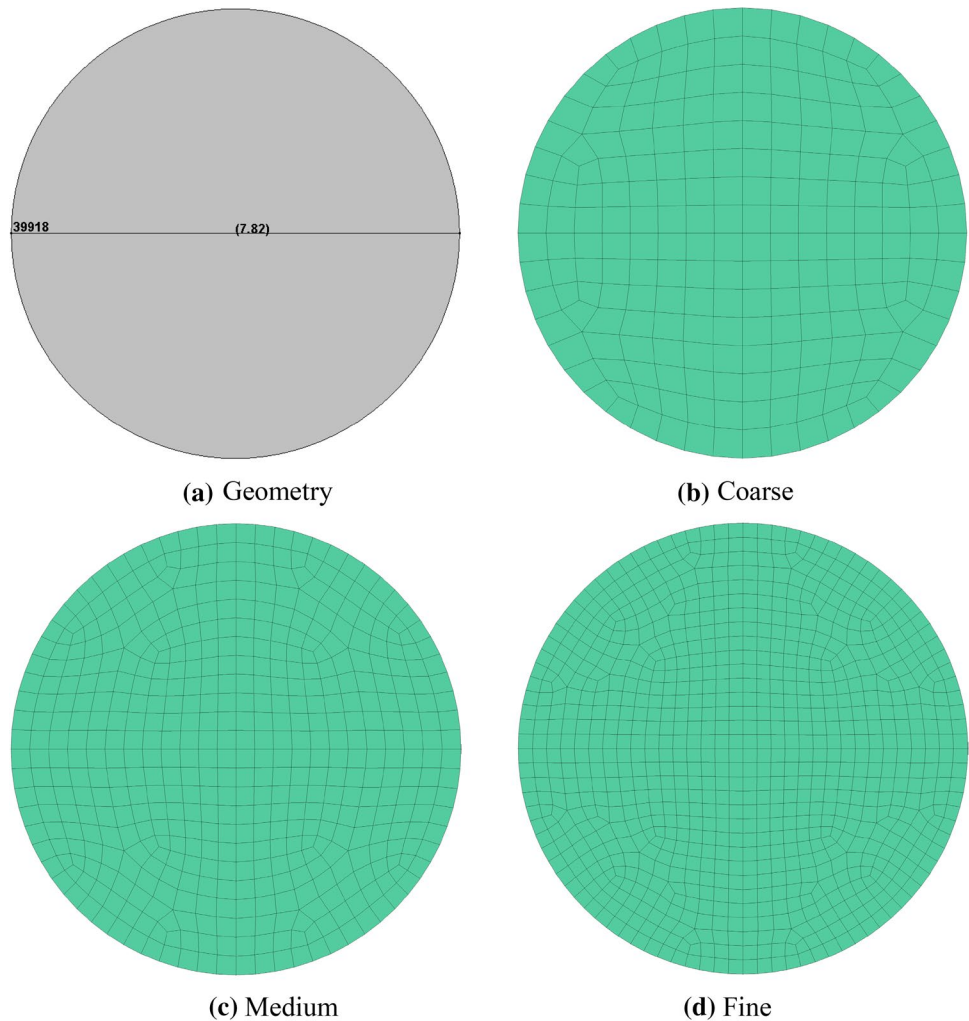


Fig. 6 Taylor impact: longitudinal displacement

pin-ball contact algorithm. Frictional contact with coefficient of friction 0.3 is considered.

Figure 11 shows the applied force (on the fastener) for the FDS driving process. The abscissa is the longitudinal distance that the fastener moved. The legend indicates the nodal distance in the MC-SPG discretization. It is observed that the difference between the MC-SPG solutions from different discretizations is marginal; therefore, we can conclude that the MC-SPG formulation converges in this simulation. In other words, we can also claim that the MC-SPG solution does not have high sensitivity to discretization, which is dramatically different from the traditional FEM with element erosion technique for failure analysis. It is also observed that the MC-SPG solution matches the original SPG solution in Fig. 11 for most of the forming stages. There is about 15% difference at the screw-driving stage, which is still acceptable for this type of extreme deformation with material failure problems. The higher SPG force response at the final screw-driving stage may be due to the modification of

Fig. 7 Taylor impact: plastic strain at termination

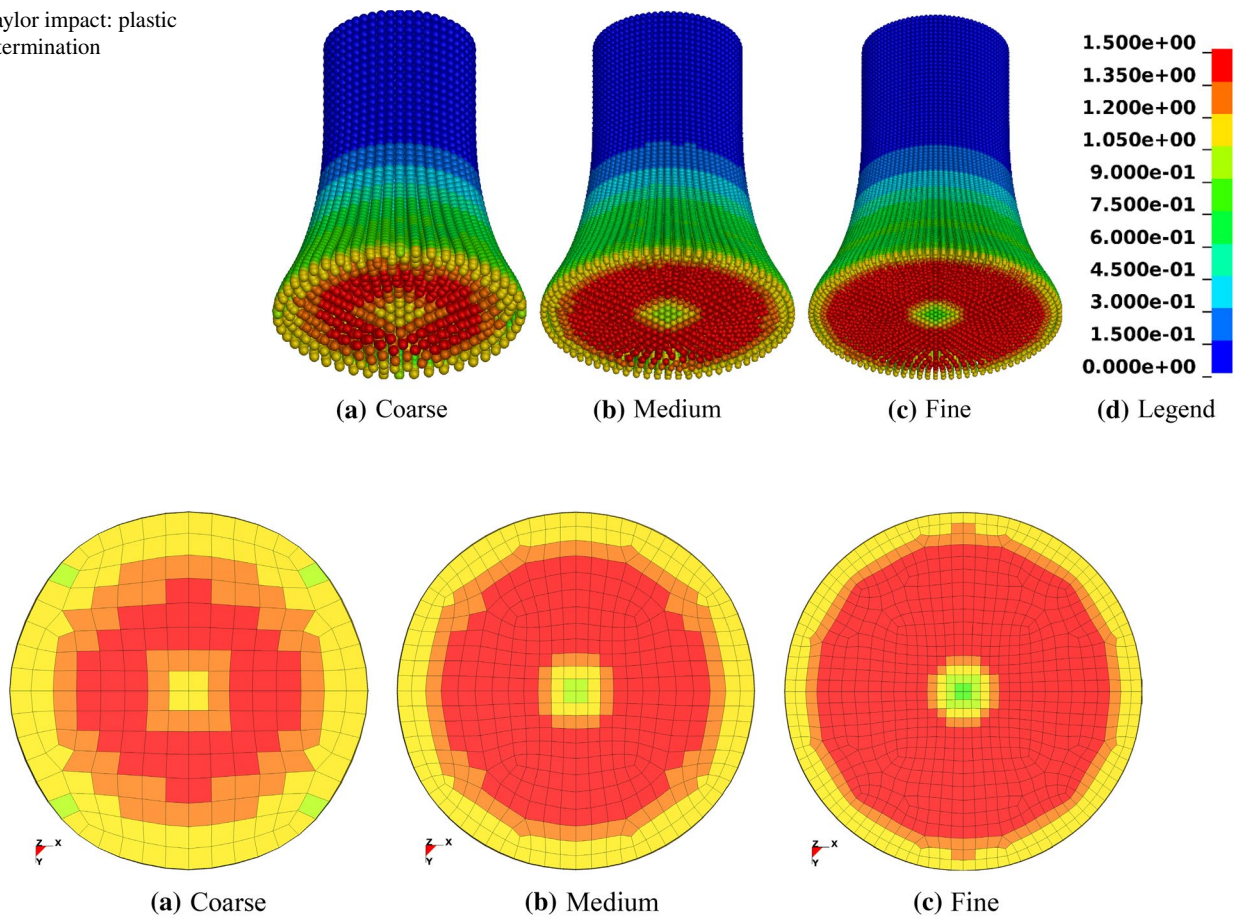


Fig. 8 Taylor impact: deformed shapes at the bottom

shear modulus in the stabilized stress computation for the original SPG method. On the other hand, if the element erosion technique is used in the finite element method to mimic the material failure phenomenon, the force is dramatically underestimated, which is unphysical. This is clearly demonstrated in Fig. 11 (green curve, obtained by finite element with element erosion at effective plastic strain of 0.4).

While the accuracy of the MC-SPG formulation is close to the SPG formulation, its efficiency is higher. In this case, for the nodal distance 0.125 mm discretization, the CPU time for SPG formulation is about 2.4 times of that of the MC-SPG (refer to Table 2 for CPU time comparison). The force response shows that the compressive thrust builds up until the tip of the fastener penetrate through the plate. Then, as more and more bond failure occurs, the thrust cannot be build up any longer. Once extrusion forming starts, a direction reversed force is resulted due to material extruding up (along the thread on the fastener) and the compressive pulse can eventually turn into a tensile one. This is consistent with the physics described in [39]. In fact, if the FDS rotational

and translational speeds are such that the plunge distance is exactly one pitch after each round of rotation, the applied force during the thread forming and screw-driving stages might just be zero because there is no material extrudes up or down along the fastener threads.

Figure 12 shows the new formed threads on the workpiece at termination for different discretizations obtained using the MC-SPG formulation. Cleaner and clearer threads are observed as the discretization is refined, which indicates that the critical physics of the FDS thread forming process is well simulated by the MC-SPG algorithm. This observation implies that to form clear threads, the discretization needs to be fine enough such that there are enough nodes in the pitch distance (0.15 mm—5 nodes, not clear; 0.125 mm—6 nodes, not clear; 0.10 mm—8 nodes, clear; 0.075 mm—10 nodes, best). This once again confirms that the MC-SPG formulation converges in this simulation. Had this process been analyzed by FEM with element erosion technique, no thread would be observed on the workpiece due to erosion upon failure, which is verified in Fig. 13b.

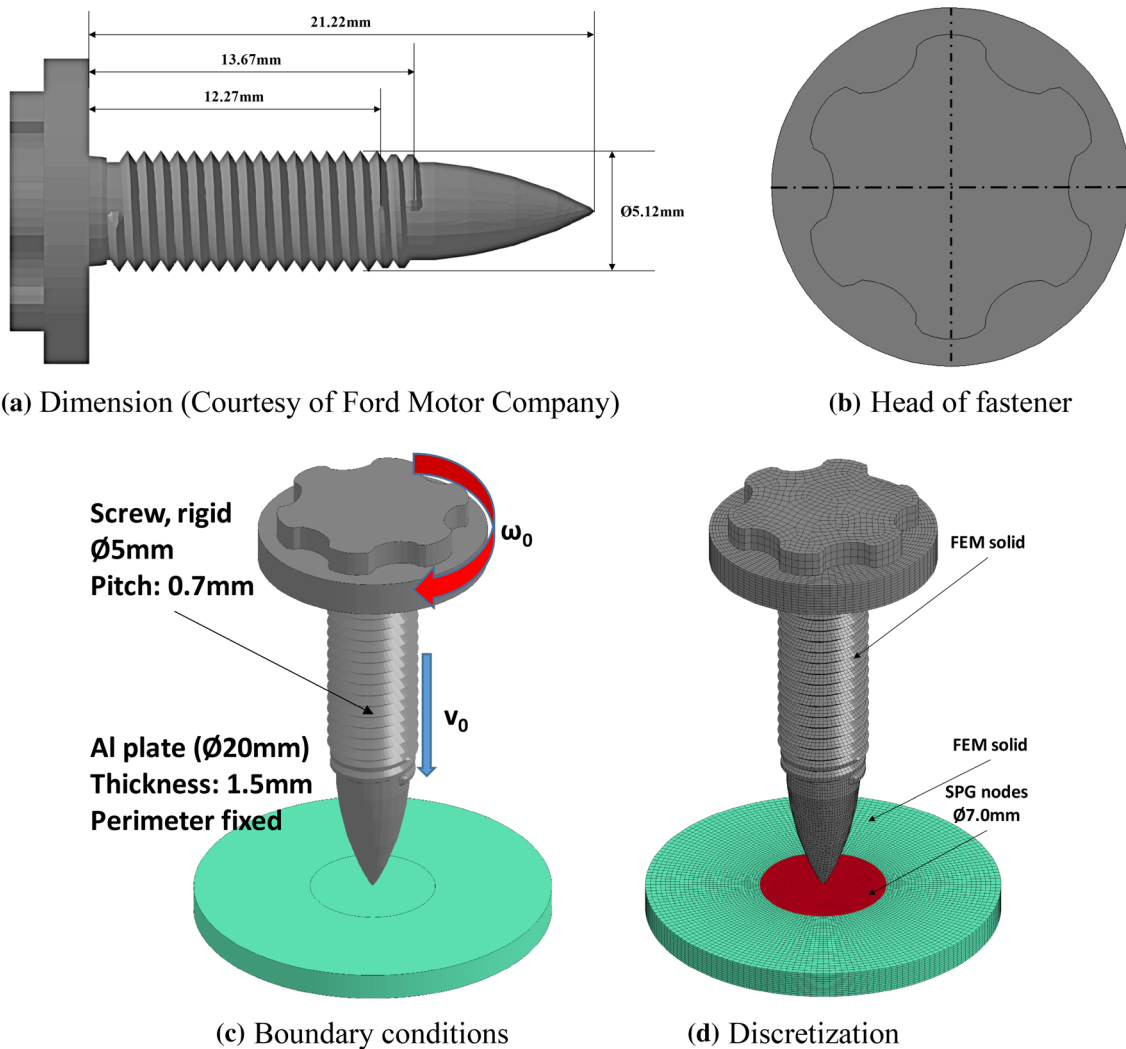


Fig. 9 Flow drill screw model: geometry and discretization

5 Conclusions

Flow drill screw (FDS) joining is a new mechanical fastening technique widely used to connect metal parts in modern lightweight car structures. Numerical simulations of FDS joining that well represents the physical behavior are important for the automotive industry. In particular, the ability to model extensive plastic deformation and material separation taking place in the extreme thread forming process is vital to the success of the FDS joining simulation. Thanks for the characteristics of discretization flexibilities and customized approximations, particle or meshfree methods have attracted significant attention from scientists and engineers in the last two decades to model challenging scientific and engineering problems. This paper presents a new particle method, the MC-SPG method, for simulating the extreme thread forming in the FDS driving process.

The starting point for the extreme thread forming simulation using this new particle method is the stabilization of formulation by a novel velocity smoothing algorithm. We show that this velocity smoothing algorithm consistently fulfills the conservation of momentum. The stabilized formulation is further supplemented with the adaptive anisotropic Lagrangian kernel and bond-based failure criterion to handle the severe deformation and material failure in the analysis. The present method differs from existing residual and non-residual stabilization methods in several important aspects. Most notably, the method does not require additional stabilization terms; thus, modifications of elastic modulus for stabilization stress and stabilization control parameters are not necessary. Furthermore, the present method is efficient and simple. Its implementation relies on velocity smoothing operators whose action is evaluated at the particle level using standard meshfree approximations. As a result, an

Fig. 10 Flow drill screw model: discretization of central $\varnothing 7.0$ mm region

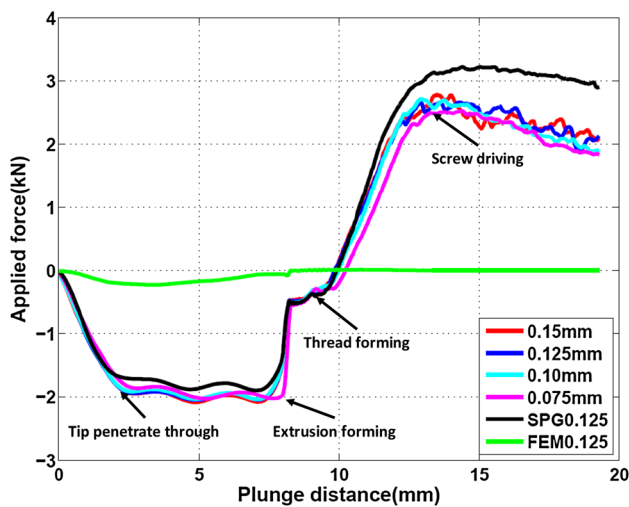
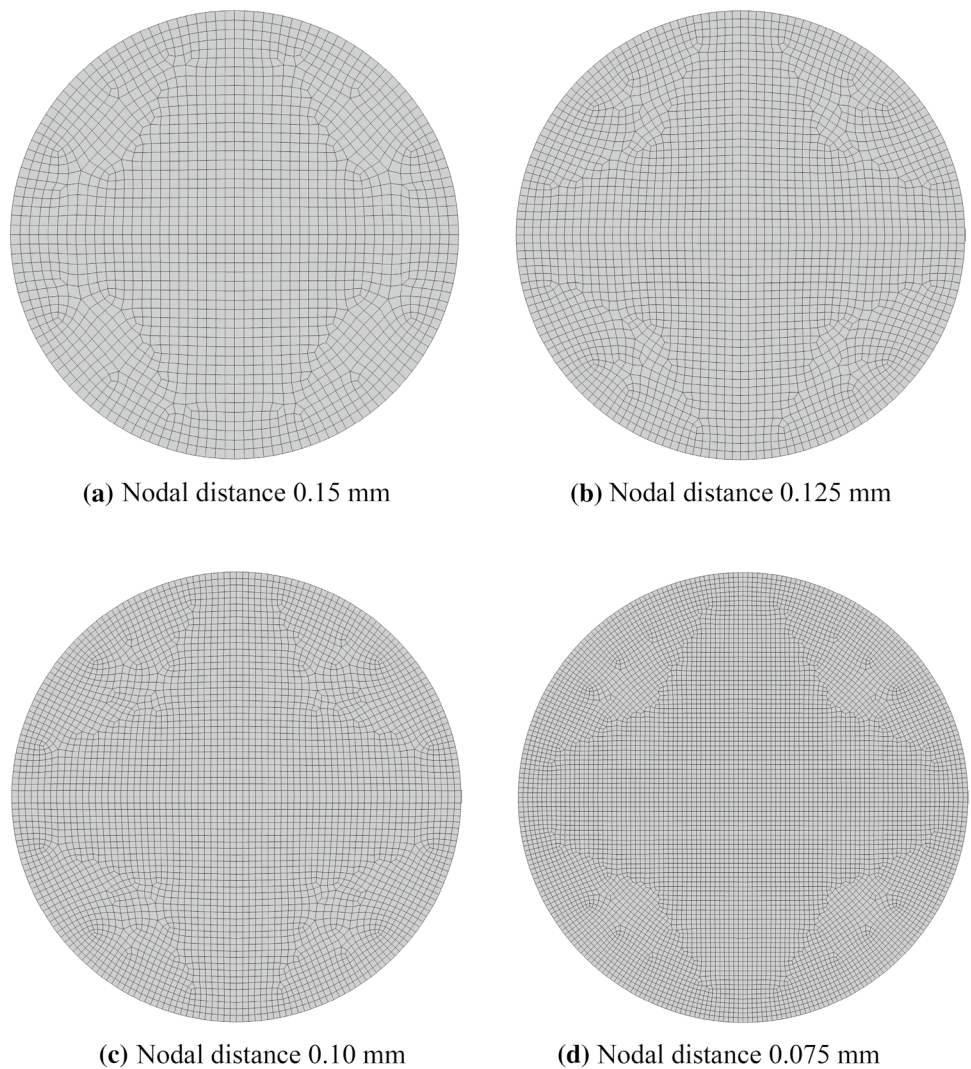


Fig. 11 Flow drill screw-driving simulation: plunge force

Table 2 Normalized CPU time

Discretization	MC-SPG			SPG
	0.15 mm	0.125 mm	0.10 mm	0.125 mm
Normalized CPU	0.61	1.0	2.03	2.39

existing meshfree code can be modified easily to handle the new stabilization procedure.

Numerical results illustrate the performance of the MC-SPG method. Convergence studies in small, large and severe deformation problems suggest the stability of the present method and its applicability in the nonlinear analysis. In particular, the present method has shown to offer a unique numerical capability in modeling the extreme thread forming operation. To the authors’ best knowledge, this is the first time in the literature a numerical method is successfully applied to model the forming of threads during the FDS driving process. In the future, the applications of the

Fig. 12 Flow drill screw-driving simulation: threads at termination by MC-SPG

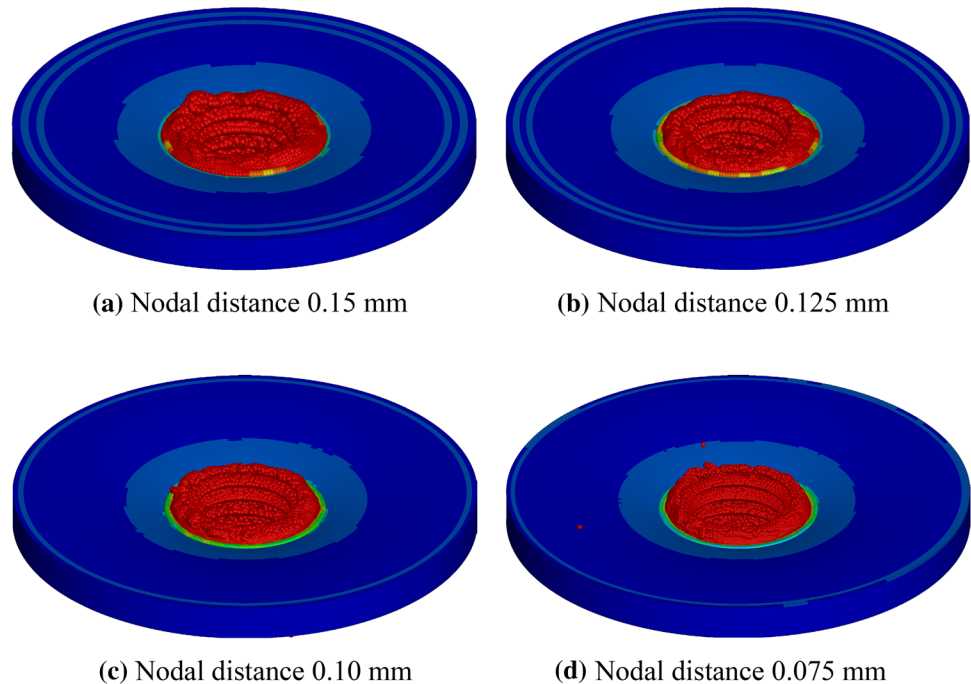
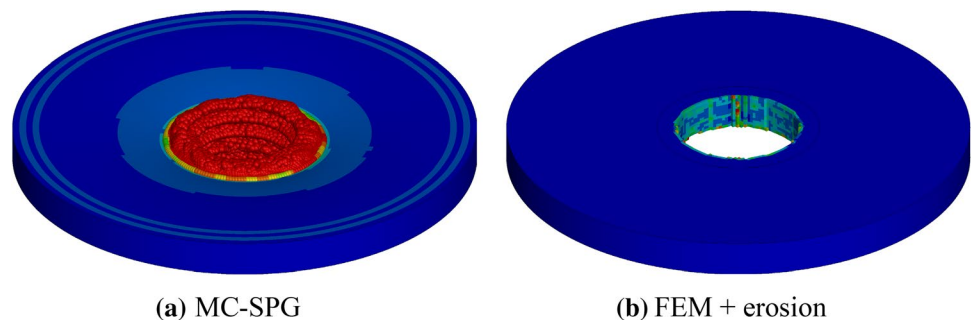


Fig. 13 Comparison of thread forming at termination by MC-SPG versus FEM



MC-SPG method to the simulation of other advanced joining techniques such as SPR connections will be studied.

Acknowledgements The authors wish to thank Dr. John O. Hallquist of LSTC for his support to this research. The support from Ford Motor Company is also gratefully acknowledged.

References

1. Sønstabø JK, Morin D, Langseth M (2016) Macroscopic modeling of flow-drill screw connections in thin-walled aluminum structures. *Thin Wall Struct* 105:185–206
2. Porcaro R, Hassen AG, Aalberg A, Langseth M (2010) Jointing of aluminum using self-piercing riveting: testing, modeling and analysis. *Int J Crashworthiness* 13:141–154
3. Wu CT, Bui TQ, Wu YC, Luo TL, Wang M, Liao CC, Chen PY, Lai YS (2018) Numerical and experimental validation of a particle Galerkin method for metal grinding simulation. *Comput Mech* 61:365–383
4. Wu CT, Wu YC, Crawford JE, Magallanes JM (2017) Three-dimensional concrete impact and penetration simulations using the smoothed particle Galerkin method. *Int J Impact Eng* 106:1–17
5. Grujicic M, Snipes J, Ramaswami S (2016) Process and product-performance modeling for mechanical fastening by flow drilling screws. *Int J Struct Integr* 7:370–396
6. Sønstabø JK, Morin D, Langseth M (2018) Static and dynamics testing and modeling of aluminum joints with flow-drill connections. *Int J Impact Eng* 115:58–75
7. Li S, Liu WK (2004) *Meshfree particle method*. Springer, Berlin
8. Belytschko T, Lu YY, Gu L (1994) Element-free Galerkin methods. *Int J Numer Meth Eng* 37:229–256
9. Liu WK, Jun S, Li S, Adee J, Belytschko T (1995) Reproducing kernel particle methods for structural dynamics. *Int J Numer Methods Eng* 38:1655–1679
10. Li B, Habbal F, Ortiz M (2010) Optimal transportation meshfree approximation schemes for fluid and plastic flows. *Int J Numer Methods Eng* 83:1541–1579
11. Chen JS, Pan C, Wu CT, Liu WK (1996) Reproducing kernel particle methods for large deformation analysis of non-linear structures. *Comput Methods Appl Mech Eng* 139:195–227
12. Wu CT, Park CK, Chen JS (2011) A generalized approximation for the meshfree analysis of solids. *Int J Numer Methods Eng* 85:693–722

13. Wang DD, Chen P (2014) Quasi-convex reproducing kernel mesh-free method. *Comput Mech* 54:689–709
14. Chen JS, Wu CT, Yoon S, You YC (2001) A stabilized conforming nodal integration for Galerkin meshfree methods. *Int J Numer Methods Eng* 50:435–466
15. Hillman M, Chen JS, Chi SW (2014) Stabilized and variationally consistent nodal integration for meshfree modeling of impact problems. *Comput Part Mech* 1:245–256
16. Rabczuk T, Belytschko T, Xiao SP (2004) Stable particle methods based on Lagrangian kernels. *Comput Methods Appl Mech Eng* 193:1035–1063
17. Guan PC, Chen JS, Wu YC, Tang H, Gaidos J, Hofstetter K, Alsaleh M (2009) Semi-Lagrangian reproducing kernel formulation and application to modeling earth moving operations. *Mech Mater* 41:670–683
18. Wu CT, Koishi M, Hu W (2015) A displacement smoothing induced strain gradient stabilization for the meshfree Galerkin nodal integration method. *Comput Mech* 56:19–37
19. Wu CT, Chi SW, Koishi M, Wu YC (2016) Strain gradient stabilization with dual stress points for the meshfree nodal integration method in inelastic analysis. *Int J Numer Methods Eng* 107:3–30
20. Chen JS, Zhang X, Belytschko T (2015) An implicit gradient model by a reproducing kernel strain regularization in strain localization problems. *Comput Methods Appl Mech Eng* 193:2827–2844
21. Liu WK, Ong JS, Uras RA (1985) Finite element stabilization matrices: a unification approach. *Comput Methods Appl Mech Eng* 53:13–46
22. Hillman M, Chen JS (2016) An accelerated, convergent, and stable nodal integration in Galerkin meshfree methods for linear and nonlinear mechanics. *Int J Numer Methods Eng* 107:603–630
23. Wu CT, Wu YC, Liu Z, Wang D (2018) A stable and convergent Lagrangian particle method with multiple nodal stress points for large strain and material failure analyses in manufacturing processes. *Finite Elem Anal Des* 146:96–106
24. Belytschko T, Bindeman LP (1993) Assumed strain stabilization of the eight node hexahedral element. *Comput Methods Appl Mech Eng* 105:225–260
25. Pan X, Wu CT, Hu W, Wu YC (2019) A momentum-consistent stabilization algorithm for Lagrangian particle methods in the thermo-mechanical friction drilling analysis. *Comput Mech*. <https://doi.org/10.1007/s00466-019-01673-8> (in early view)
26. Wu CT, Koishi M (2012) Three-dimensional meshfree-enriched finite element formulation for micromechanical hyperelastic modeling of particulate rubber composites. *Int J Numer Methods Eng* 91:1137–1157
27. Beissel S, Belytschko T (1996) Nodal integration of the element-free Galerkin method. *Comput Methods Appl Mech Eng* 139:49–74
28. Onate E, Perazzo F, Miquel J (2001) A finite point method for elasticity problems. *Comput Struct* 79:2151–2163
29. Herreros MI, Mabssout M (2011) A two-steps time discretization scheme using the SPH method for shock wave propagation. *Comput Methods Appl Mech Eng* 200:1833–1845
30. Lee CH, Gil AJ, Greto G, Kulasegaram S, Bonet J (2016) A new Jameson–Schmidt–Turkel smoothed particle hydrodynamics algorithm for large strain explicit fast dynamics. *Comput Methods Appl Mech Eng* 311:71–111
31. Lee CH, Gil AJ, Hassan OI, Bonet J, Kulasegaram S (2017) A variationally consistent streamline upwind Petrov–Galerkin smoothed particle hydrodynamics algorithm for large strain solid dynamics. *Comput Methods Appl Mech Eng* 318:514–536
32. Liu GR, Liu MB (2003) *Smoothed particle hydrodynamics*. World Scientific, Singapore
33. Silling SA, Askari E (2005) A meshfree method based on the peridynamic model of solid mechanics. *Comput Struct* 83:1526–1535
34. Ren B, Wu CT, Askari E (2017) A 3D discontinuous Galerkin finite element method with the bond-based peridynamics model for dynamics brittle failure analysis. *Int J Impact Eng* 99:14–25
35. Hallquist JO (2006) *LS-DYNA® keyword user's manual*. Livermore Software Technology Corporation, Livermore
36. Wilkins ML, Guinan MW (1973) Impact of cylinders on a rigid boundary. *J Appl Phys* 44:1200–1206
37. Johnson GR, Cook WH (1983) A constitutive model and data for metals subjected to large strains, high strain rates, and high temperatures. In: *Proceedings 7th international symposium on ballistics* pp 541–547; The Hague, Netherlands, 19–21 April 1983
38. Schwer LE (2009) Aluminum plate perforation: a comparative case study using Lagrange with erosion, multi-material ALE, and smooth particle hydrodynamics. In: *7th European LS-DYNA conference*, Salzburg, Austria, 14–15 May 2009
39. Skovron J, Mears L, Ulatan D, Detwiler D, Paolini D, Baeumler B, Claus L (2015) Characterization of flow drill screwdriving process parameters on joint quality. *SAE Int J Mater Manf* 8:35–44

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