

Simulation of metal cutting using the particle finite-element method and a physically based plasticity model

J. M. Rodríguez¹ · P. Jonsén¹ · A. Svoboda¹

Received: 22 March 2016 / Revised: 30 May 2016 / Accepted: 23 June 2016 / Published online: 3 August 2016
© The Author(s) 2016. This article is published with open access at Springerlink.com

Abstract Metal cutting is one of the most common metal-shaping processes. In this process, specified geometrical and surface properties are obtained through the break-up of material and removal by a cutting edge into a chip. The chip formation is associated with large strains, high strain rates and locally high temperatures due to adiabatic heating. These phenomena together with numerical complications make modeling of metal cutting difficult. Material models, which are crucial in metal-cutting simulations, are usually calibrated based on data from material testing. Nevertheless, the magnitudes of strains and strain rates involved in metal cutting are several orders of magnitude higher than those generated from conventional material testing. Therefore, a highly desirable feature is a material model that can be extrapolated outside the calibration range. In this study, a physically based plasticity model based on dislocation density and vacancy concentration is used to simulate orthogonal metal cutting of AISI 316L. The material model is implemented into an in-house particle finite-element method software. Numerical simulations are in agreement with experimental results, but also with previous results obtained with the finite-element method.

Keywords Particle finite-element method · Dislocation density constitutive models · Metal cutting · Machining

✉ J. M. Rodríguez
rodjua@ltu.se

P. Jonsén
par.jonsen@ltu.se

A. Svoboda
ales.svoboda@ltu.se

¹ Division of Mechanics of Solid Materials, Department of Engineering Sciences and Mathematics, Lulea University of Technology, Luleå, Sweden

1 Introduction

The numerical simulation of metal-cutting processes is complicated mainly by two factors. First one is the material constitutive model. It must adequately represent deformation behavior during high strain rate loading as well as low strain rate loading under a range of temperatures, and account for hardening and softening processes. Material models are usually calibrated based on experimental data from material testing covering the relevant range of loading conditions of the intended application. The magnitudes of strain and strain rate involved in metal cutting are several orders of magnitude higher than those generated from conventional material tension and compression testing. A highly desirable feature is therefore a material model that can be extrapolated outside the calibration range. This is not trivial since materials exhibit different strain hardening and softening characteristics at different strains, strain rates, and temperatures. The advantage of using a physically based plasticity model is an expected larger domain of validity compared with phenomenological models, because physically based models are related to the underlying physics of the deformation coupled to the microstructural evolution. The second challenge is concerned with the modeling and numerical realization of large configuration changes. Numerical simulations of machining process involves large strains and angular distortions, multiple contacts and self-contact, generation of new boundaries, and fracture with multiple cracks and defragmentation. All of them are difficult to handle using standard finite-element methods.

The purpose of this paper is to apply the particle finite-element method (PFEM) [1–3] to solve the problems associated with the large changes in configuration and use a physically based plasticity model [4,5] to extrapolate the material behavior outside the calibration range.

The rest of the paper is organized as follows. Section 2 is devoted to the description of PFEM and the modifications introduced here concerning insertion and removal of particles. In Sect. 3, the basic equations for conservation of linear momentum, mass conservation, and heat transfer for a continuum using a generalized Lagrangian framework is presented. Section 4 deals with the variational formulation of the continuous problem presented in Sect. 3. Then the mixed finite-element discretization using simplicial element with equal order approximation for the displacement, the pressure, and the temperature is presented, and the relevant matrices and vectors of the discretized problem are given in Sect. 5. Section 6 presents the J_2 plasticity model at finite deformation and a description of the physical-based plasticity model. Details of the implicit solution of the Lagrangian FEM equations in time using an updated Lagrangian approach and a Newton–Raphson-type iterative scheme are presented in Sect. 7. Section 8 focuses on a set of representative numerical simulations of metal-cutting processes using PFEM and a physical-based constitutive model. Finally, in Sect. 9, some concluding remarks are presented.

2 The particle finite-element method

The particle finite-element method (PFEM) is a FEM-based particle method [1], initially proposed for the solution of the continuous fluid mechanics equations. The main objectives were, on the one hand, to develop a method to eliminate the convective terms in the governing equations. On the other hand, the introduction of a technology, based on the alpha shape method used in other areas, able to deal with free boundary surfaces is a secondary objective. The interpretation of the method has evolved from a meshless method, in which the nodes are supposed to be particles that move according to simple rules of motion, to a sort of updated Lagrangian approach in which the advantages of the standard FEM formulation for the solution of the incremental problem are used.

Although PFEM was initially applied to problems in the field of fluid mechanics, it is being currently applied to a wide range of simulation problems [6–9]: filling, erosion, mixing processes, thermoviscous processes and thermal diffusion problems, among others. First applications of PFEM to solid mechanics are found in [10–15] to problems involving large strains and rotations, multibody contacts and creation of new surfaces (riveting, powder filling, and machining).

Applications to the response of rockfill dams on overtopping conditions, via a non-Newtonian fluid, can be found in [16].

2.1 The particle finite-element method on fluid mechanics

In the PFEM approach, the continuum material is modeled using an updated formulation, and the finite-element method is used to solve the variational incremental problem. Hence, a mesh, discretizing the domain, is generated in order to solve the governing equations. In this context, discretization nodes can be regarded as material points whose motion is followed during the time stepping solution. The PFEM, typically, consists of the following steps

1. Fill the domain with a set of points referred to as ‘particles’. The accuracy of the numerical solution is clearly dependent on the considered number of particles.
2. Generate a finite-element mesh using the particles as nodes. This is achieved using a Delaunay triangulation [17, 18].
3. Identify the external boundaries to impose the boundary conditions and to compute the domain integrals. In this step, the *Alpha Shape method* [19, 20] is used for the boundary recognition.
4. Solve the nonlinear Lagrangian form of the governing equations finding displacement, velocity and pressure at every node of the mesh.
5. Update the particle positions using the computed values of displacement, velocity and pressure.
6. Go back to step 2 and repeat for the next time step.

The *continuous* reconnection introduced in step 2 is the key strategy to circumvent the typical mesh distortion generated when a description is used with problems involving large strains.

In this solution scheme, not only is the numerical solution of the equations critical from the computational point of view, but also are the generation of a new mesh and the identification of the boundaries.

2.1.1 Identification of the boundaries

In a Lagrangian framework, the external boundary and the reference volume are defined by the position of the material particles. Every time the mesh is regenerated, the particles belonging to the boundary may change, and the new boundary nodes (and therefore the particles) have to be identified. The Delaunay triangulation generates the convex hull of the set of particles. Moreover, the convex hull may not be conformal with the external boundaries. A possibility to overcome this problem is to correct the generated mesh using the so-called α -shape method to remove the unnecessary triangles from the mesh using a criterion based on the mesh distortion. The α -shape method can also be used for the identification of the fluid particles which separate from the rest of the domain.

2.2 The particle finite-element method solid mechanics

The PFEM is a set of numerical strategies combined for the solution of large deformation problems. The standard algorithm of the PFEM for the solution of solid mechanics problems contains the following steps.

1. Definition of the domain(s) Ω_n in the last converged configuration, $t = {}^n t$, keeping existing spatial discretization $\bar{\Omega}_n$.
2. Transference of variables by a smoothing process—from Gauss points to nodes.
3. Discretization of the given domain(s) in a set of particles of infinitesimal size—elimination of existing connectivities $\bar{\Omega}_n$.
4. Reconstruction of the mesh through a triangulation of the domain's convex-hull and the definition of the boundary applying the α -shape method [19,20], defining a new spatial discretization $\bar{\Omega}_n$.
5. A contact method to recognize the multibody interaction.
6. Transference of information, interpolating nodal variables into the Gauss points.
7. Solution of the system of equations for ${}^{n+1}t = {}^n t + \Delta t$.
8. Go back to step 1 and repeat the solution process for the next time step.

2.3 The particle finite-element method in the numerical simulation of metal-cutting processes

The standard PFEM presents some weaknesses when applied in orthogonal cutting simulation. For example, the external surface generated using α -shape may affect the mass conservation, the chip shape, the absence of equilibrium on the boundary due to the introduction of artificial perturbations, and generation of unphysical welding of the workpiece material and the chip.

To deal with this problem, in this work, the use of a constrained Delaunay algorithm is proposed. Furthermore, addition and removal of particles are the principal tools, which are employed for sidestepping the difficulties, associated with deformation-induced element distortion, and for resolving the different scales of the solution.

In the numerical simulation of metal-cutting process, despite the continuous Delaunay triangulation, elements arise with unacceptable aspect ratios; for this reason, the mesh is also subjected to a Laplacian smoothing algorithm to smooth a mesh. For each node in a mesh, a new position is chosen based on the position of neighbors, and the node is moved there. In the case that a mesh is topologically a rectangular grid, then this operation produces the Laplacian of the mesh. These procedures are applied locally and not in every time step. Specific size metrics control node inser-

tion and removal, while the Laplacian smoothing algorithm drives the repositioning of nodes.

In summary, the enhancement of the PFEM takes place in three main areas: the dynamic process for the discretization of the domain into particles, varying the number of them depending on the deformation of the body; transference of the internal variables, from a nodal smoothing through a variable projection; and the boundary recognition, eliminating the geometric criterion of the α -shape method.

2.3.1 Data transfer of internal variables

The transference of internal variables or element information between evolving meshes within the field of PFEM is critical in the numerical simulation of process like machining as is shown in [14,15]. In this work, the authors show that the nodal scheme presented in [10] generates unphysical spring back of the machined surface and sub-estimation of the cutting and feed forces.

Due to the insertion, removal, and relocation of particles through Laplacian smoothing, the transference of gauss point variables is set directly through a mesh projection instead of traditional nodal smoothing [13]. The projection is carried out by a direct search of the position of the integration point of the new connectivity, over the former mesh. The use of this transference scheme give an improved computational efficiency. The use of the projector operator to transfer internal variables guarantees the preservation of the stress free state for the portions of the domains that do not yield. In this zone, there is no insertion, removal or relocation of particles, most of the finite elements of the stress free region remain the same before and after the Delaunay triangulation, getting as a result no diffusion of the internal variables in the stress free zone. More details about the data transfer of internal variables in the numerical simulations are found in [13–15].

3 Governing equations for a Lagrangian Continuum

Consider a domain containing a deformable material which evolves in time due to the external and internal forces and prescribed displacements and thermal conditions from an initial configuration at time $t = 0$ to a current configuration at time $t = t_n$. The volume V and its boundaries Γ at the initial and current configurations are denoted as $({}^0V, {}^0\Gamma)$ and $({}^nV, {}^n\Gamma)$, respectively. The aim is to find a spatial domain that the material occupies, and at same time obtain velocities, strain rates, stresses and temperature in the updated configuration at time ${}^{n+1}t = {}^n t + \Delta t$. In the following, a left super index denotes the configuration where the variable is computed.

3.1 Momentum equations

The equation of conservation of linear momentum for a deformable continuum are written in a Lagrangian description as

$$\rho \frac{Dv_i}{Dt} - \frac{\partial^{n+1}\sigma_{ij}}{\partial^{n+1}x_j} - {}^{n+1}b_i = 0, \quad i, j = 1, \dots, n_s \text{ in } {}^{n+1}V \quad (1)$$

In Eq. (1), ${}^{n+1}V$ is the analysis domain in the updated configuration at time ${}^{n+1}t$ with boundary ${}^{n+1}\Gamma$, v_i and b_i are the velocity and the body force components along the Cartesian axis, ρ is the density, n_s is the number of space dimensions, ${}^{n+1}x_j$ is the position of the material point at time ${}^{n+1}t$ and ${}^{n+1}\sigma_{ij}$ are the Cauchy stresses in ${}^{n+1}V$ and $\frac{Dv_i}{Dt}$ is the material derivative of the velocity field.

The Cauchy stresses are split in the deviatoric s_{ij} and pressure p components as

$$\sigma_{ij} = s_{ij} + p\delta_{ij} \quad (2)$$

where δ_{ij} is the Kronecker delta. The pressure is assumed to be positive for a tension state.

3.2 Thermal balance

The thermal balance equation in the current configuration is written in a Lagrangian framework as

$$\rho c \frac{DT}{Dt} - \frac{\partial}{\partial^{n+1}x_i} \left(k \frac{\partial T}{\partial^{n+1}x_i} \right) + {}^{n+1}Q = 0, \quad i, j = 1, \dots, n_s \text{ in } {}^{n+1}V \quad (3)$$

where T is the temperature, c is the thermal capacity, k is the thermal conductivity and Q is the heat source.

3.3 Mass balance

The mass conservation equation can be written for solids domain as

$$-\frac{1}{\kappa} \frac{Dp}{Dt} + \varepsilon_V = 0 \quad (4)$$

where κ is bulk elastic moduli of the solid material, $\frac{Dp}{Dt}$ is the material derivative of the pressure field, and ε_V is the volumetric strain rate defined as the trace of the rate of deformation tensor defined as

$$d_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) \quad (5)$$

For a general time interval $[{}^n t, {}^{n+1} t]$, Eq. (4) is discretized as

$$-\frac{1}{\kappa} ({}^{n+1}p - {}^n p) + \varepsilon_V \Delta t = 0 \quad (6)$$

Equations (1), (3) and (4) are completed by the standard boundary conditions.

3.4 Boundary conditions

3.4.1 Mechanical problem

The boundary conditions at the Dirichlet Γ_v and Neumann Γ_t boundaries with $\Gamma = \Gamma_v \cup \Gamma_t$ are

$$v_i - v_i^P = 0 \quad \text{on } \Gamma_v \quad (7)$$

$$\sigma_{ij} n_j - t_i^P = 0 \quad \text{on } \Gamma_t, i = 1, \dots, n_s \quad (8)$$

where v_i^P and t_i^P are the prescribed velocities and the prescribed tractions, respectively.

3.4.2 Thermal problem

$${}^{n+1}T - {}^{n+1}T^P = 0 \quad \text{on } {}^{n+1}\Gamma_T \quad (9)$$

$$k \frac{\partial T}{\partial n} + {}^{n+1}q_n^P = 0 \quad \text{on } {}^{n+1}\Gamma_q \quad (10)$$

where T^P and q_n^P are the prescribed temperature and the prescribed normal heat flux at the boundaries Γ_T and Γ_q , respectively, and n is a vector in the direction normal to the boundary.

4 Variational formulation

4.1 Momentum equations

Multiplying Eq. (1) by arbitrary test function w_i with dimensions of velocity and integrating over the updated domain ${}^{n+1}V$ gives the weighted residual form of the momentum equations as [21, 22]

$$\int_{{}^{n+1}V} \left(-\frac{\partial^{n+1}\sigma_{ij}}{\partial^{n+1}x_j} - {}^{n+1}b_i \right) w_i = 0 \quad (11)$$

In Eq. (11), the inertial term $\rho \frac{Dv_i}{Dt}$ is neglected because in the problems of interest in this work, this term is much smaller than the other terms appearing in Eq. (11).

Integrating by parts the terms involving σ_{ij} and using the tractions boundary conditions (8) yields the weak form of the

momentum equation as

$$\int_{n+1V} \delta \varepsilon_{ij}^{n+1} \sigma_{ij} dV - \int_{n+1V} w_i^{n+1} b_i dV - \int_{n+1\Gamma_t} w_i^{n+1} t_i^p dV = 0 \tag{12}$$

where $\delta \varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial w_i}{\partial x_j} + \frac{\partial w_j}{\partial x_i} \right)$ is a virtual strain rate field. Equation (12) is the standard form of the principle of virtual power [21, 22].

Using (2), Eq. (12) gives the following expression

$$\int_{n+1V} \delta \varepsilon_{ij}^{n+1} s_{ij} dV + \int_{n+1V} \delta \varepsilon_{ij}^{n+1} p \delta_{ij} dV - \int_{n+1V} w_i^{n+1} b_i dV - \int_{n+1\Gamma_t} w_i^{n+1} t_i^p dV = 0 \tag{13}$$

Introducing, \mathbf{w} , \mathbf{s} and $\delta \mathbf{e}$, the vectors of test function, deviatoric stresses and virtual strain rates respectively; \mathbf{b} and \mathbf{t}^p the body forces and tractions vectors, respectively, and \mathbf{m} an auxiliary vector in Eq. (13), yields

$$\int_{n+1V} \delta \mathbf{e}^T \mathbf{s} dV + \int_{n+1V} \delta \mathbf{e}^T \mathbf{m} p dV - \int_{n+1V} \mathbf{w}^T \mathbf{b} dV - \int_{n+1\Gamma_t} \mathbf{w}^T \mathbf{t}^p dV = 0 \tag{14}$$

where the matrices introduced in Eq. (14) are defined in reference [23, 24].

4.2 Mass conservation equation

To obtain the mass conservation, Eq. (6) is multiplied by an arbitrary test function q , defined over the analysis domain. Integrating over $n+1V$ yields

$$\int_{n+1V} -\frac{q}{\kappa} ({}^{n+1}p - {}^n p) dV + \int_{n+1V} q \varepsilon_V \Delta t dV = 0 \tag{15}$$

4.2.1 Stabilized mass conservation equation: the polynomial pressure projection (PPP)

Mixed formulations have to satisfy additional mathematical conditions, which guarantee its stability. This condition is known as BB-condition, named after its inventors Babuska and Brezzi. In this approach, a stabilized formulation based

on the so-called polynomial pressure projection (PPP) is presented and applied to Stokes equation in [25, 26] is used. The method is obtained by modification of the mixed variational equations using a local L_2 polynomial pressure projection. Unlike other stabilization methods, the PPP does not require the specification of a mesh-dependent stabilization parameter or the calculation of higher-order derivatives. In addition, PPP can be implemented at the elementary level and reduces to a simple modification of the mass conservation equation as follows:

$$\int_{n+1V} -\frac{q}{\kappa} ({}^{n+1}p - {}^n p) dV + \int_{n+1V} q \varepsilon_V \Delta t dV + \int_{n+1V} (q - \tilde{q}) \frac{\alpha}{\mu} (p - \tilde{p}) dV = 0 \tag{16}$$

where α is a stabilization parameter, μ the material shear modulus and \tilde{p} is the best approximation of the pressure in the space of polynomial of order 0.

The best approximation of the pressure field in the space of polynomial of order 0 is given by

$$\int_{n+1V} \tilde{q} (p - \tilde{p}) dV = 0 \tag{17}$$

4.3 Thermal balance equation

Application of the standard weighted residual methods to the thermal balance Eqs. (3), (9) and (10) leads, after standard operations, to [22]

$$\int_{n+1V} \hat{w} \rho c \frac{DT}{Dt} dV + \int_{n+1V} \frac{\partial \hat{w}}{\partial x_i} \left(k \frac{\partial T}{\partial x_i} \right) dV + \int_{n+1V} \hat{w}^{n+1} Q dV + \int_{n+1V} \hat{w}^{n+1} q_n^p d\Gamma = 0 \tag{18}$$

where \hat{w} are the space-weighting functions for the temperature.

5 Finite-element discretization

The analysis domain is discretized into finite elements with n nodes in the standard manner leading to a mesh with a total number of N_e elements and N nodes. In the present work, a simple three-noded linear triangle ($n = 3$) with local linear shape functions N_i^e defined for each node n of the element is used. The displacement, the velocity, the pressure, and the temperature are interpolated over the mesh in terms of their nodal values in the same manner using the global linear

shape function N_j spanning over the nodes sharing node j . In matrix format and 2D problems, we have

$$\mathbf{u} = \mathbf{N}_u \bar{\mathbf{u}}, \quad \mathbf{v} = \mathbf{N}_v \bar{\mathbf{v}}, \quad \mathbf{p} = \mathbf{N}_p \bar{\mathbf{p}}, \quad \mathbf{T} = \mathbf{N}_T \bar{\mathbf{T}} \tag{19}$$

where

$$\bar{\mathbf{u}} = \begin{Bmatrix} \bar{\mathbf{u}}^1 \\ \bar{\mathbf{u}}^2 \\ \vdots \\ \bar{\mathbf{u}}^N \end{Bmatrix} \text{ with } \bar{\mathbf{u}}^i = \begin{Bmatrix} u_1^i \\ u_2^i \end{Bmatrix}, \quad \bar{\mathbf{v}} = \begin{Bmatrix} \bar{\mathbf{v}}^1 \\ \bar{\mathbf{v}}^2 \\ \vdots \\ \bar{\mathbf{v}}^N \end{Bmatrix} \text{ with}$$

$$\bar{\mathbf{v}}^i = \begin{Bmatrix} v_1^i \\ v_2^i \end{Bmatrix}, \quad \bar{\mathbf{p}} = \begin{Bmatrix} \bar{p}^1 \\ \bar{p}^2 \\ \vdots \\ \bar{p}^N \end{Bmatrix}, \quad \bar{\mathbf{T}} = \begin{Bmatrix} \bar{T}^1 \\ \bar{T}^2 \\ \vdots \\ \bar{T}^N \end{Bmatrix}$$

$$\mathbf{N}_u = \mathbf{N}_v = [\mathbf{N}_1 \ \mathbf{N}_2 \ \cdots \ \mathbf{N}_N]$$

$$\mathbf{N}_p = \mathbf{N}_T = [N_1 \ N_2 \ \cdots \ N_N] \tag{20}$$

with $\mathbf{N}_j = N_j \mathbf{I}_2$ where \mathbf{I}_2 is the 2×2 identity matrix.

Substituting Eq. (19) into Eqs. (13),(16) and (18) while choosing a Galerkin formulation with $w_i = q = \hat{w}_i = N_i$ leads to the following system of algebraic equations

$$\mathbf{F}_{\text{res,mech}} = \mathbf{F}_{\mathbf{u},\text{int}}(\bar{\mathbf{u}}, \bar{\mathbf{p}}) - \mathbf{F}_{\mathbf{u},\text{ext}} = 0$$

$$\mathbf{F}_{\text{res,mass}} = \mathbf{F}_{p,\text{press}}(\bar{\mathbf{p}}) - \mathbf{F}_{p,\text{vol}}(\bar{\mathbf{u}}) + \mathbf{F}_{p,\text{stab}}(\bar{\mathbf{p}}) - \mathbf{F}_{p,\text{press},n}({}^n \bar{\mathbf{p}}) = 0$$

$$\mathbf{F}_{\text{res,therm}} = \mathbf{F}_{\theta,\text{dyn}}(\dot{\bar{\mathbf{T}}}) - \mathbf{F}_{\theta,\text{int}}(\bar{\mathbf{T}}) + \mathbf{F}_{\theta,\text{ext}}(\bar{\mathbf{T}}) = 0 \tag{21}$$

where

$$\mathbf{F}_{\mathbf{u},\text{int}}(\bar{\mathbf{u}}, \bar{\mathbf{p}}) = \int_{n+1V} \mathbf{B}_u^{Tn+1} \sigma dV \tag{22}$$

$$\mathbf{F}_{\mathbf{u},\text{ext}} = \int_{n+1V} \mathbf{N}^{Tn+1} \mathbf{b} dV - \int_{n+1\Gamma_t} \mathbf{N}^{Tn+1} \mathbf{t}^p d\Gamma \tag{23}$$

$$\mathbf{F}_{p,\text{press}}(\bar{\mathbf{p}}) = \int_{n+1V} \frac{1}{\kappa} \mathbf{N}^T \mathbf{N} \bar{\mathbf{p}} dV \tag{24}$$

The term $\mathbf{F}_{p,\text{press},n}({}^n \bar{\mathbf{p}})$ is exactly the same term as in Eq. (24), but the nodal pressure is evaluated at time ${}^n t$

$$\mathbf{F}_{p,\text{vol}}(\bar{\mathbf{u}}) = \mathbf{Q}^{Tn+1} \bar{\mathbf{u}} \tag{25}$$

where the element form of the \mathbf{Q} matrix is given by

$$\mathbf{Q}^{(e)} = \int_{n+1V^{(e)}} \mathbf{B}_{ui}^{(e)T} \mathbf{m} N_j^{(e)} dV \tag{26}$$

$$\mathbf{F}_{p,\text{stab}}(\bar{\mathbf{p}}) = \sum_{e=1}^{N_e} \int_{n+1V^{(e)}} \frac{\alpha}{\mu} \left(\mathbf{N}^{(e)} \mathbf{N}^{T(e)} - \tilde{\mathbf{N}}^{(e)} \tilde{\mathbf{N}}^{T(e)} \right) \mathbf{p}^{(e)} dV \tag{27}$$

$$\mathbf{F}_{T,\text{dyn}}(\dot{\bar{\mathbf{T}}}) = \int_{n+1V} \rho c \mathbf{N} \mathbf{N}^T \dot{\bar{\mathbf{T}}} dV \tag{28}$$

$$\mathbf{F}_{\theta,\text{int}}(\bar{\mathbf{T}}) = \int_{n+1V} k \mathbf{B}_\theta^T \mathbf{B}_\theta \bar{\mathbf{T}} dV - \int_{n+1V} \mathbf{N}^T Q dV \tag{29}$$

$$\mathbf{F}_{\theta,\text{ext}}(\bar{\mathbf{T}}) = \int_{n+1\Gamma_q} \mathbf{N}^T q_n^p d\Gamma \tag{30}$$

In Eq. (21), $\dot{\bar{\mathbf{T}}}$ denotes the material time derivative of the nodal temperature.

In finite-element computations, the above force vectors are obtained as the assemblies of element vectors. Given a nodal point, each component of the global force associated with a particular global node is obtained as the sum of the corresponding contributions from the element force vectors of all elements that share the node. In this work, the element force vectors are evaluated using Gaussian quadrature.

Note that Eq. (21) involves the geometry at the updated configuration (${}^{n+1}V$). This geometry is unknown; hence the solution of Eq. (21) has to be found iteratively. The iterative solution scheme proposed in this work is presented in the next section.

6 The constitutive model

6.1 Thermoelastoplasticity model at finite strains

In metal-forming processes like machining, elastic strain are in the order of 10^{-4} , whereas plastic strains can be of the order of 10^{-1} to 10 [27]. In case elastic strains are neglected, the model is not able to predict residual stresses and the springback of the machined surface. For this reason, the constitutive model is developed for small elastic and large plastic deformation, instead of a more complex model that use large elastic large plastic deformation (See. [13–15]). An example of modeling of machining processes using a fluid mechanics approach is presented in [28]. A valuable implication of the small elastic strains is that the rate of deformation tensor $d_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right)$ inherits the additive structure of classical small strain elastoplasticity:

$$\mathbf{d} = \mathbf{d}^e + \mathbf{d}^p \tag{31}$$

where \mathbf{d}^e and \mathbf{d}^p are the elastic and plastic parts of the rate of deformation tensor, respectively.

6.1.1 Elastic response

Let a material with a hypo-elastic constitutive equation like

$$L_v(\tau) = \mathbf{c} : (\mathbf{d} - \mathbf{d}^p) \tag{32}$$

where $L_v(\cdot)$ denotes the Lie objective stress rate, and τ denotes the Kirchhoff stress tensor. It can be assume that the special elasticity \mathbf{c} tensor is given by

$$\mathbf{c} = 2\mu \left(\mathbf{I} - \frac{1}{3} \mathbf{1} \otimes \mathbf{1} \right) + \kappa \mathbf{1} \otimes \mathbf{1} \tag{33}$$

where \mathbf{I} and $\mathbf{1}$, with components $I_{abcd} = (\delta_{ac}\delta_{bd} + \delta_{ad}\delta_{cd})/2$ and $1_{ab} = \delta_{ab}$, are the fourth- and second-order symmetric unit tensors, respectively. The parameters μ and κ represent the shear and the bulk elastic modulus, respectively.

6.1.2 Yield condition

The yield condition used in this work is the von Mises–Huber yield criterion. This criterion is formulated in terms of the second invariant of the Kirchhoff stress tensor $J_2 = \frac{1}{2} \text{dev} \tau : \text{dev} \tau$. Hence, the von Mises yield criterion (henceforth simply called the Mises criterion) can be stated as

$$f(\tau) = \sqrt{2J_2} - \sqrt{\frac{2}{3}} \sigma_y = \|\text{dev} \tau\| - \sqrt{\frac{2}{3}} \sigma_y(\bar{\varepsilon}^p, \rho_i, c_v) = 0 \tag{34}$$

where σ_y denotes the flow stress, $\bar{\varepsilon}^p$ is the hardening parameter or plastic strain, ρ_i denotes the immobile dislocations density, and c_v is the vacancy fraction. The set of nonlinear-coupled differential equations describing the evolution of ρ_i and c_v are presented in Sect. 6.2.

6.1.3 Flow rule

As is customary in the framework of incremental plasticity, the concept of flow rule is applied to obtain the plastic rate of deformation tensor \mathbf{d}^p in terms of the plastic flow direction tensor $\mathbf{n} = \frac{\text{dev} \tau}{\|\text{dev} \tau\|}$ associated to the yield surface:

$$\mathbf{d}^p = \dot{\lambda} \mathbf{n} = \dot{\lambda} \frac{\text{dev} \tau}{\|\text{dev} \tau\|} \tag{35}$$

and the evolution equation for the accumulated effective plastic strain $\bar{\varepsilon}^p$ is governed by

$$\dot{\bar{\varepsilon}}^p = \sqrt{\frac{2}{3}} \dot{\lambda} \tag{36}$$

where $\dot{\lambda}$ is the consistency parameter or plastic multiplier subject to the standard Kuhn–Tucker loading/unloading conditions

$$\dot{\lambda} \geq 0, f(\tau) \leq 0, \dot{\lambda} f(\tau) = 0 \tag{37}$$

Along with the consistency condition [29] complete the formulation of the model.

$$\dot{\lambda} \dot{f}(\tau) = 0. \tag{38}$$

6.2 Physical-based models

Physical-based models are models where the physical mechanisms are underlying the deformation in contrast to empirical models, which are of a more curve-fitting nature. However, due to the need for averaging and also limited knowledge about some of the relations making up the model, physical-based models need also be calibrated. Two different types of physical-based models exist. One option is to explicitly include variables from physics as internal state variables. The other possibility is to determine the format of the constitutive equation based on knowledge about the physical mechanisms causing the deformation. The latter is a so-called “model-based-phenomenology” Some advantages of physical-based models is the possibility to link effects from other scales, e.g. via parameters for grain size or models for microstructural evolution that extends the model validity to a larger domain. This may extrapolate the validity of the model outside their range of calibration. This requires that the physical mechanisms implemented in the models still dominate the deformation in the extended range.

The dislocation density model considers dislocation glide and climb processes contributions to the plastic straining. The yield limit in this approach is separated into two components according to [4, 5, 30–32]

$$\sigma_y = \sigma_G + \sigma^* \tag{39}$$

σ_G and σ^* are the long-range athermal component and the short-range contributions to the flow stress, respectively. The first component σ_G , is the stress needed to overcome the long-range interactions lattice distortions due to the dislocation substructure. The second component, σ^* , is the stress needed for the dislocation to pass through the lattice and to pass short-range obstacles. Thermal vibrations will then also assist the dislocation when passing an obstacle. The long-range stress component is commonly written as;

$$\sigma_G = m \alpha \mu b \sqrt{\rho_i} \tag{40}$$

where m is the Taylor orientation factor, α is a proportionality factor, μ is the temperature dependent shear modulus,

b is the magnitude of Burgers vector and ρ_i is the immobile dislocation density. The short-range stress components may be written as,

$$\sigma^* = \hat{\tau}\mu \left\{ 1 - \left[\frac{kT}{\Delta F b^3 \mu} \ln \left(\frac{\dot{\epsilon}_{ref}}{\dot{\epsilon}^p} \right) \right]^{\frac{1}{q}} \right\}^{\frac{1}{p}} \quad (41)$$

where ΔF denote the required free energy needed to overcome the lattice resistance or obstacles without assistance from external stress, $\hat{\tau}$ denote the athermal flow strength required to move the dislocation past barriers without assistance of thermal energy, $\dot{\epsilon}_{ref}$ denote the reference strain rate. The exponent p and q characterize the barrier profiles and usually have values between $0 \leq p \leq 1$ and $0 \leq q \leq 2$ respectively.

6.2.1 Structure evolution

The evolution of the structure is considered to consist of a hardening and a recovery process. The used model assumes that the mobile dislocation density is stress and strain independent and much smaller than the immobile ones. Hence the evolution equation is written;

$$\dot{\rho}_i = \dot{\rho}_i^{(+)} - \dot{\rho}_{i(\text{glide})}^{(-)} - \dot{\rho}_{i(\text{climb})}^{(-)} \quad (42)$$

where index i denotes the immobile dislocations. The increase in immobile dislocation density is assumed to be related to the plastic strain rate and may therefore be written according to

$$\dot{\rho}_i^{(+)} = \frac{m}{b} \frac{1}{\Lambda} \dot{\epsilon}^p \quad (43)$$

where m is the Taylor orientation factor, b is the Burgers vector and Λ denote the mean free path. Mobile dislocations in this model are assumed to move an average distance (mean free path), Λ , according to the relation

$$\frac{1}{\Lambda} = \frac{1}{s} + \frac{1}{g_s} \quad (44)$$

g_s and s denote the size of the grains and the subcell, respectively.

The effect of grain size on flow stress, the Petch–Hall effect, is accounted for this way and contributes to the hardening. The size of subcells is related to the immobile DD by the parameter K_c

$$s = K_c \frac{1}{\sqrt{\rho_i}} \quad (45)$$

The recovery may occur by dislocation glide and/or climb. The former is described by

$$\dot{\rho}_{i(\text{glide})}^{(-)} = \Omega \rho_i \dot{\epsilon}^p \quad (46)$$

where Ω is a recovery function which depends on the temperature and strain rate. Recovery by climb is describe by

$$\dot{\rho}_{i(\text{climb})}^{(-)} = 2c_\gamma \frac{D_v \mu b^3}{c_v^{\text{eq}} kT} (\rho_i^2 - \rho_{\text{eq}}^2) c_v \quad (47)$$

where c_v is the vacancy fraction, c_v^{eq} is the thermal equilibrium vacancy concentration, D_v is the diffusivity and c_γ is a calibration parameter.

6.2.2 Vacancy generation and migration

Recovery of dislocations and diffusion of solute atoms are associated with the control generation and motion of vacancies. They affect recovery by climb, as described above. Climb is a nonconservative motion, where a dislocation moves out of the glide plane onto another glide plane. Motion of jogged screw dislocations creates vacancies by nonconservative motion of jogs [4], while annihilation of vacancies may take place at grain boundaries and at dislocations. The mono-vacancy evolution equation presented in [4,5] is used in this study:

$$\dot{c}_v = \left[\chi \frac{\Omega_0}{Q_{vf}} \sigma_y + \zeta \frac{c_j}{4b^2} \frac{\Omega_0}{b} \right] \dot{\epsilon}^p - D_{vm} \left(\frac{1}{K_c^2} \rho_i + \frac{1}{g^2} \right) \times (c_v - c_v^{\text{eq}}) + \dot{c}_v^{\text{eq}} \quad (48)$$

where σ_y is the flow stress, χ is the fraction of mechanical work used for vacancy generation, Q_{vf} is the activation energy for vacancy formation, ζ is the net neutralization effect of vacancy emitting and vacancy-absorbing jogs, c_j is the concentration of thermal jogs, Ω_0 is the atomic volume, and D_{vm} refers to vacancy migration.

6.3 Stress update algorithm

6.3.1 Thermoelastoplasticity model at finite strains

An implicit integration of the constitutive model presented in Sect. 6.1 is summarized in Box 1.

6.3.2 Physical-based models

Given the increment in effective plastic strain and the effective plastic strain rate obtained in the radial-return stress update algorithm (Box 1), the only unknowns of Eq. (42) and (48) are immobile dislocation density and the vacancy concentration. An implicit time discretization of the set of

Box 1 Implicit integration scheme of the thermoelastoplastic model at finite strains

```

Given  $^{n+1}\mathbf{u}, {}^n\boldsymbol{\tau}, {}^n\bar{\varepsilon}^p, {}^n\rho_i, {}^n c_v, \Delta t, \mu$ 
 $\mathbf{f} = \mathbf{1} + \nabla\mathbf{u}_{n+1}$ 
 ${}^{n+1}\mathbf{e} = \frac{1}{2}(\mathbf{1} - {}^{n+1}\mathbf{f}^{-T} \cdot {}^{n+1}\mathbf{f}^{-1})$ 
 $\boldsymbol{\tau}_{n+1}^{\text{trial}} = {}^{n+1}\mathbf{f} \cdot {}^n\boldsymbol{\tau} \cdot {}^{n+1}\mathbf{f}^T + \mathbf{c} : {}^{n+1}\mathbf{e}$ 
Check for plastic loading
 $\text{dev}({}^{n+1}\boldsymbol{\tau}^{\text{trial}}) = {}^{n+1}\boldsymbol{\tau}^{\text{trial}} - \frac{1}{3}\text{tr}({}^{n+1}\boldsymbol{\tau}^{\text{trial}})\mathbf{1} \otimes \mathbf{1}$ 
 $f({}^{n+1}\boldsymbol{\tau}^{\text{trial}}) = \|\text{dev}({}^{n+1}\boldsymbol{\tau}^{\text{trial}})\| - \sqrt{\frac{2}{3}}\sigma_y({}^{n+1}\bar{\varepsilon}^p, {}^n\rho_i, {}^n c_v)$ 
IF  $f({}^{n+1}\boldsymbol{\tau}^{\text{trial}}) < 0$ 
 ${}^{n+1}\boldsymbol{\tau} = {}^{n+1}\boldsymbol{\tau}^{\text{trial}}$ 
 ${}^{n+1}\bar{\varepsilon}^p = {}^{n+1}\bar{\varepsilon}^p$ 
ELSE
Go to return-mapping
ENDIF
The return-mapping
 ${}^{n+1}\mathbf{n} = \frac{\text{dev}({}^{n+1}\boldsymbol{\tau}^{\text{trial}})}{\|\text{dev}({}^{n+1}\boldsymbol{\tau}^{\text{trial}})\|}$ 
FIND  $\Delta\lambda$  from the solution of the yielding equation using
Newton–Raphson
 $\text{dev}({}^{n+1}\boldsymbol{\tau}) = \text{dev}({}^{n+1}\boldsymbol{\tau}^{\text{trial}}) - 2\mu\Delta\lambda{}^{n+1}\mathbf{n}$ 
 ${}^{n+1}\bar{\varepsilon}^p = {}^n\bar{\varepsilon}^p + \sqrt{\frac{2}{3}}\Delta\lambda$ 
    
```

two coupled differential Eqs. (42) and (48) can be obtained as [4,5]:

$$\begin{aligned} \Delta\rho_i - \left(\frac{m}{b} \frac{1}{\Lambda} - \Omega_0\rho_i\right) \dot{\bar{\varepsilon}}^p \Delta t + 2c_\gamma \frac{D_v \mu b^3}{c_v^{\text{eq}} kT} \\ \times (\rho_i^2 - \rho_{\text{eq}}^2) c_v \Delta t = 0 \\ \Delta c_v - \left[\chi \frac{\Omega_0}{Q_{\text{vf}}} \sigma_y + \zeta \frac{c_j}{4b^2} \frac{\Omega_0}{b}\right] \dot{\bar{\varepsilon}}^p \Delta t + D_{\text{vm}} \\ \times \left(\frac{1}{K_c^2} \rho_i + \frac{1}{g^2}\right) (c_v - c_v^{\text{eq}}) \Delta t - \dot{c}_v^{\text{eq}} \Delta t = 0 \end{aligned} \quad (49)$$

where $\Delta\rho_i = {}^{n+1}\rho_i - {}^n\rho_i$, $\Delta c_v = {}^{n+1}c_v - {}^nc_v$ and Δt is the time step increment.

This set can be written in a vector form as

$$\mathbf{H} = \mathbf{0} \quad (50)$$

The linearization of the system gives

$$\mathbf{H}_{(i)} + \frac{\partial \mathbf{H}_{(i)}}{\partial \mathbf{q}} \delta \mathbf{q} = \mathbf{0} \quad (51)$$

where the subscript i is an iteration counter and \mathbf{q} is a vector containing the internal state variables

$$\mathbf{q}^T = (\rho_i, c_v) \quad (52)$$

The iterative changes in the internal state variables can now be computed as

$$\delta \mathbf{q}_{(i)} = - \left[\frac{\partial \mathbf{H}_{(i)}}{\partial \mathbf{q}} \right]^{-1} \mathbf{H}_{(i)} \quad (53)$$

This set is solved by an implicit Newton–Raphson method. When the iterative changes $\delta \mathbf{q}$ are small enough, the iterations are terminated, and the internal state variables are updated as

$${}^{n+1}\mathbf{q} = {}^n\mathbf{q} + \Delta \mathbf{q} \quad (54)$$

$$\Delta \mathbf{q} = \sum_{i=1}^{\text{niter}} \delta \mathbf{q}_{(i)} \quad (55)$$

where $niter$ is the number of iterations necessary to reach the desired tolerance of the Newton–Raphson method. More details about the matrix $\frac{\partial \mathbf{H}_{(i)}}{\partial \mathbf{q}}$ are given in the Appendix and in [4].

7 Transient solution of the discretized equations

Equation (21) are solved in time with an uncoupled (mechanical–thermal) implicit Newton–Raphson-type iterative scheme. The basic steps within a time increment $[n \cdot n + 1]$ are

- Initialize variables

$$\begin{aligned} ({}^{n+1}\mathbf{x}^1, {}^{n+1}\bar{\mathbf{u}}^1, {}^{n+1}\boldsymbol{\tau}^1, {}^{n+1}\bar{\mathbf{p}}^1, {}^{n+1}\bar{\mathbf{T}}^1, {}^{n+1}\bar{\varepsilon}^p, {}^{n+1}\rho_i, {}^{n+1}c_v) \\ \leftarrow ({}^n\mathbf{x}, {}^n\bar{\mathbf{u}}, {}^n\boldsymbol{\tau}, {}^n\bar{\mathbf{p}}^1, {}^n\bar{\mathbf{T}}, {}^n\bar{\varepsilon}^p, {}^n\rho_i, {}^nc_v) \end{aligned}$$

- In the following, $(\cdot)^i$ denotes a value computed at the i th iteration.
- Iteration loop: $i = 1, \dots, N_{iter}$. For each iteration

Mechanical problem

Step 1 Compute the nodal displacement increments and the nodal pressure, from Eq. (21):

$$\mathbf{K} \begin{bmatrix} \Delta \mathbf{u} \\ \Delta \bar{\mathbf{p}} \end{bmatrix} = - \begin{bmatrix} {}^{n+1}\mathbf{F}_{\text{res,mech}}^i \\ {}^{n+1}\mathbf{F}_{\text{res,mass}}^i \end{bmatrix} \quad (56)$$

The iteration matrix \mathbf{K} is given by

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}_{uu} & \mathbf{K}_{up} \\ \mathbf{K}_{pu} & \mathbf{K}_{pp} \end{bmatrix} \quad (57)$$

where \mathbf{K}_{uu} , \mathbf{K}_{up} , \mathbf{K}_{pu} and \mathbf{K}_{pp} are given by the following expressions

$$\mathbf{K}_{uu,ij} = \int_{n+1V} \mathbf{B}_i^{eT} \left(\mathbf{C}_T^{\text{dev}} - 2p\mathbf{I} \right) \mathbf{B}_j^e dV + \int_{n+1V} \mathbf{G}_i^T \sigma \mathbf{G}_j dV$$

$$\mathbf{K}_{up,ij} = \int_{n+1V} \mathbf{B}_i^{eT} \mathbf{m} N_j^e dV$$

$$\mathbf{K}_{pu} = \mathbf{K}_{up}$$

$$\mathbf{K}_{pp} \int_{n+1V} \frac{1}{\kappa} \mathbf{N}^e T \mathbf{N}^e dV + \int_{n+1V} \frac{\alpha}{\mu} (\mathbf{N}^e \mathbf{N}^e T - \tilde{\mathbf{N}}^e \tilde{\mathbf{N}}^e T) dV \quad (58)$$

where $\mathbf{C}_T^{\text{dev}}$ is the deviatoric part of the consistent algorithmic matrix emanating from the linearization of Eq. (21) (a) with respect to the nodal displacements [21].

Step 2 Update the nodal displacements and nodal pressure:

$$\begin{aligned} n+1 \bar{\mathbf{u}}^{i+1} &= n+1 \bar{\mathbf{u}}^i + \Delta \mathbf{u} \\ n+1 \bar{\mathbf{p}}^{i+1} &= n+1 \bar{\mathbf{p}}^i + \Delta \bar{\mathbf{p}} \end{aligned} \quad (59)$$

Step 3 Update the nodal coordinates and the incremental deformation gradient:

$$\begin{aligned} n+1 \mathbf{x}^{i+1} &= n+1 \mathbf{x}^i + \Delta \mathbf{u} \\ n+1 \mathbf{F}_{ij}^{i+1} &= \frac{\partial n+1 x_i^{i+1}}{\partial^n x_j} \end{aligned} \quad (60)$$

Step 4 Compute the deviatoric Cauchy stresses from Box 1.

Step 5 Compute the stresses:

$$n+1 \sigma_{ij} = \text{dev}(n+1 \tau) + \bar{p} \delta_{ij} \quad (61)$$

Step 6 Check convergence

Verify the following conditions:

$$\begin{aligned} \left\| n+1 \bar{\mathbf{u}}^{i+1} - n+1 \bar{\mathbf{u}}^i \right\| &\leq e_u \left\| n \bar{\mathbf{u}} \right\| \\ \left\| n+1 \bar{\mathbf{p}}^{i+1} - n+1 \bar{\mathbf{p}}^i \right\| &\leq e_p \left\| n \bar{\mathbf{p}} \right\| \end{aligned} \quad (62)$$

where e_u and e_p are the prescribed error norms. In the examples presented in this paper, the error norms are set to $e_u = e_p = 10^{-3}$. If conditions (62) are satisfied, the solution of the thermal problem in the updated configuration $n+1 \mathbf{x}$ is accepted. Otherwise, make the iteration counter $i \leftarrow i + 1$ and repeat Steps 1–6.

Thermal problem

– Iteration loop: $i = 1, \dots, N_{iter}$. For each iteration:

Step 7 Compute the nodal temperatures

$$\left[\int_{n+1V} \frac{\rho c}{\Delta t} \mathbf{N} \mathbf{N}^T dV + \int_{n+1V} k \mathbf{B}_\theta^T \mathbf{B}_\theta \right] \Delta \bar{\mathbf{T}} = -n+1 \mathbf{F}_{\text{res, therm}}^i \quad (63)$$

Step 8 Update the nodal temperatures

$$n+1 \bar{\mathbf{T}}^{i+1} = n+1 \bar{\mathbf{T}}^i + \Delta \bar{\mathbf{T}} \quad (64)$$

Step 9 Check convergence

$$\left\| n+1 \bar{\mathbf{T}}^{i+1} - n+1 \bar{\mathbf{T}}^i \right\| \leq e_T \left\| n \bar{\mathbf{T}} \right\| \quad (65)$$

where e_T is the error norm in the balance of energy. In the examples presented in this work, the error norm is set to $e_T = 10^{-5}$. If condition (65) is satisfied, the solution of the physical-based plasticity model Eq. (49) is accepted. Otherwise, make the iteration counter $i \leftarrow i + 1$ and repeat Steps 7–9.

Physical-based models

– Iteration loop: $i = 1, \dots, N_{iter}$. For each iteration:

Step 10 Compute the gauss point increment of dislocation density ρ_i and vacancy concentration c_v given the value of temperature and strain rate obtained in the solution of the thermal and mechanical problems, respectively.

$$\delta \mathbf{q}_{(i)} = - \left[\frac{\partial \mathbf{H}_{(i)}}{\partial \mathbf{q}} \right]^{-1} \mathbf{H}_{(i)} \quad (66)$$

Step 11 Update the gauss point value of the ρ_i and vacancy concentration c_v

$$n+1 \mathbf{q} = n \mathbf{q} + \Delta \mathbf{q} \quad (67)$$

Step 12 Check convergence of the physical-based constitutive model.

$$\begin{aligned} \|\delta \rho_i(i)\| &\leq e_p \left\| n \rho_i \right\| \\ \|\delta c_v(i)\| &\leq e_p \left\| n c_v \right\| \end{aligned} \quad (68)$$

where e_p is the error norm in the physical-based constitutive model. In the examples presented in this work, a value of $e_p = 10^{-8}$ is used. If conditions (68) are satisfied then make $n \leftarrow n + 1$ and proceed to the next time step. Otherwise, make the iteration counter $i \leftarrow i + 1$ and repeat Steps 10–12.

8 Examples, result and discussion

The ability of PFEM to adaptively insert and remove particles and improve mesh quality is crucial in the problems presented from here on. Then, it is possible to maintain a reasonable shape of elements and also capture gradients of strain, strain rate, and temperature. The PFEM strategy does not require a criterion for modelling of the chip separation from the workpiece.

The friction condition is an important factor that influences chip formation. Friction on the tool–chip interface is a nonconstant function dependent of normal and shear stress

distributions. Normal stresses are largest in the sticking contact region near the tool tip. The stress in the sliding zone along the contact interface from the tool tip to the point where the chip separates from the tool rake face is controlled by frictional shear stress. A variety of complex friction models exists; however, the lack of input data to these models is a limiting factor. The model for tool–chip interface employed in this study is the Coulomb friction model. The friction coefficient $\mu = 0.5$ was selected following the value used in [30,33,34].

The heat generated in metal cutting has a significant effect on the chip formation. The heat generation mechanisms are the plastic work done in the primary and secondary shear zones and the sliding friction in the tool–chip contact interface. Generated heat does not have sufficient time to diffuse away and temperature rise in the work material is mainly due to localized adiabatic conditions. A standard practice in the numerical simulations of mechanical cutting is to assume the fraction of plastic work transformed into heat equal to 0.9, see [30,32,34].

Table 1 Cutting data in simulations

Test no	Cutting speed v_c (m min ⁻¹)	Feed (mm rev ⁻¹)	Cutting depth (mm)
1	180	0.05	3.0
2	180	0.15	3.0
3	240	0.05	3.0
4	240	0.15	3.0

An orthogonal cutting operation was employed to mimic 2D plain strain conditions. The depth of cut, used for all the test cases, was equal to 3 mm. The dimension of the workpiece was 8×1.6 mm. A horizontal velocity corresponding to the cutting speed was applied to the particles at the right side of the tool as is given in Table 1. The particles along the bottom and the left sides of the workpiece were fixed. Material properties for the workpiece material are shown in Appendix 2. Material properties of the tool were assumed as thermoelastic.

Fig. 1 2D plane strain PFEM model of orthogonal cutting:
a initial set of particles,
b initiation of the chip

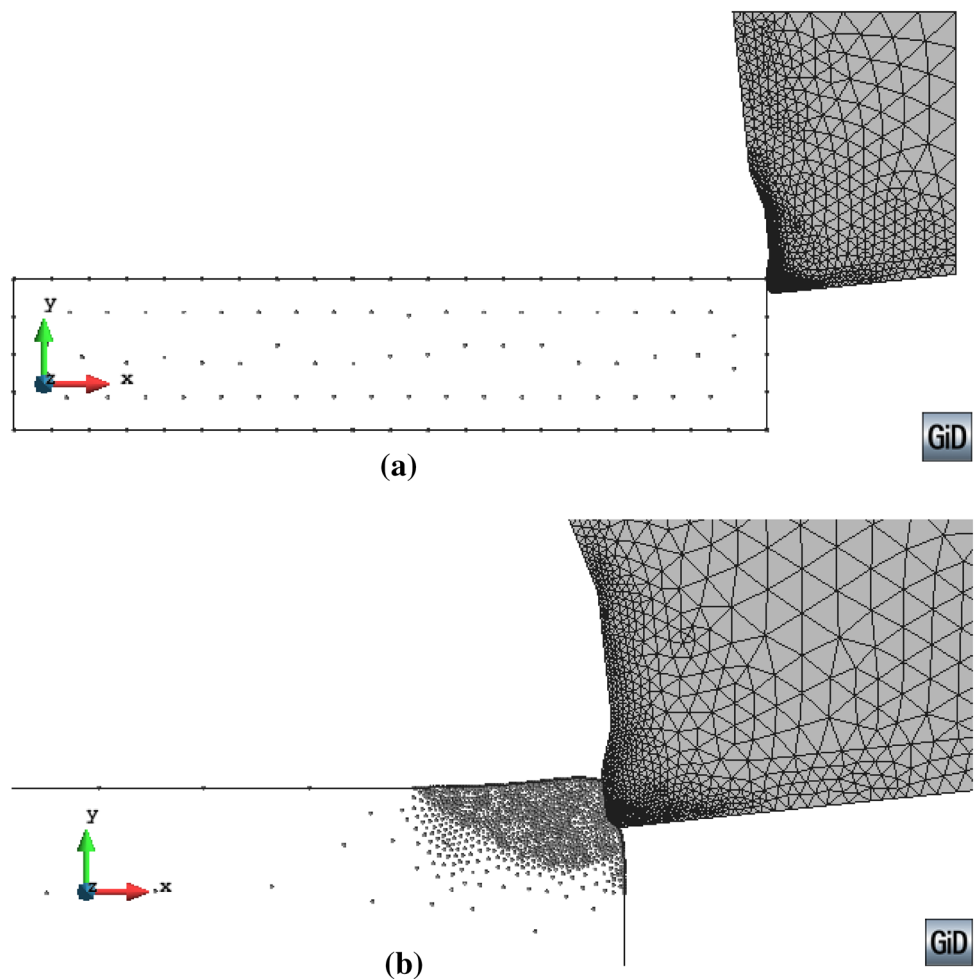


Fig. 2 Intermediate stages of the chip formation: **a** time 3.14×10^{-4} s, **b** time 6.28×10^{-4} s

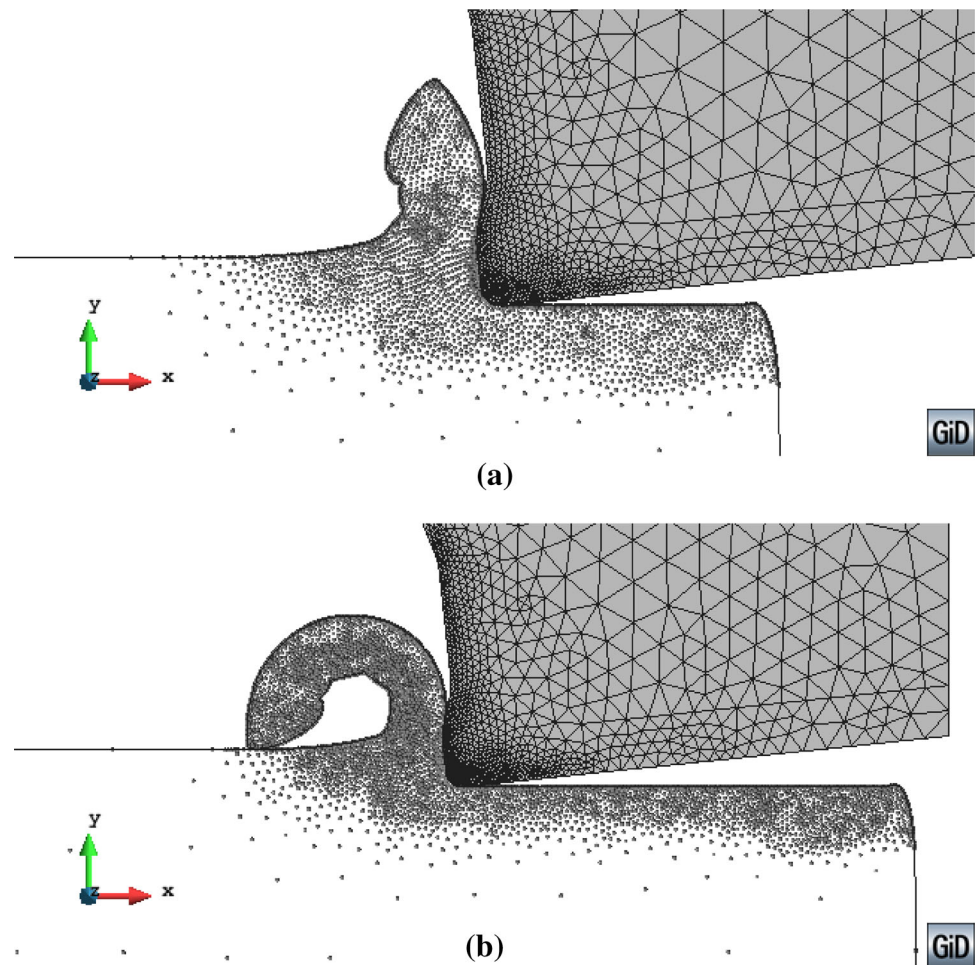


Table 2 Measured and simulated cutting forces

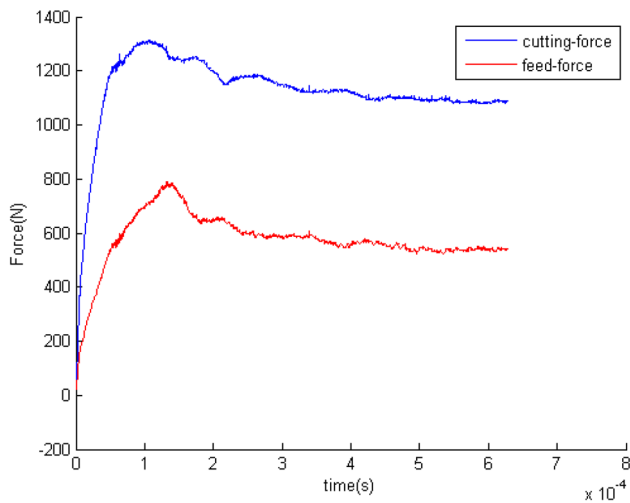
	Measured		Simulated PFEM				Simulated FEM			
	F_c (N)	F_f (N)	F_c (N)	Error (%)	F_f (N)	Error (%)	F_c (N)	Error (%)	F_f (N)	Error (%)
1	446	427	513.6	15.1	427.6	-2.15	485	8.7	400	-8.5
2	960	592	1100	14.5	557	-5.91	1100	14.6	545	-7.9
3	442	439	507	14.7	411	-6.38	480	8.6	390	-11.2
3	929	560	1094	17.7	549	-1.96	1075	15.7	530	-5.4

The workpiece was discretized with 105 particles, see Fig. 1a. The tool geometry was discretized by 2298 tree-node thermomechanical elements. Due to adaptive insertion and removal of particles, the average number of particles increased up to 4800. The effect of insertion of particles near the tool tip is illustrated in Figs. 1b and 2a, b for the test case no. 2. The insertion of particles was controlled by the equidistribution of plastic power.

8.1 Cutting and feed forces

The loading histories of simulated forces for test no 2, see Fig. 6b, were evaluated at the chip tool interface. Average values of the computed forces in the steady state region are compared with the experimental results in Table 2. The error used for the evaluation of the computed results is computed as

$$\text{Error} = \frac{\text{Computed} - \text{Measured}}{\text{Measured}} \times 100 \% \quad (69)$$



The Table 2 shows that the cutting force was overestimated in all tests by about 15 %. Meanwhile, the feed force was underestimated by about 5 %. The results presented in the table above were compared with the results in Svoboda et al [30], showing that PFEM is able to predict results to those obtained using (FEM). The errors in Table 2 must be related to the context where they will be used, namely the cutting tool manufacturing industry. Literature overview [4] show that in the industrial production of nominally identical cutting tool as well as variations in material properties of nominally the same material can cause variations around the 10 % in forces. Estimated errors below 10 % indicate a good model fit for the purpose.

Fig. 3 Effective plastic strain rate

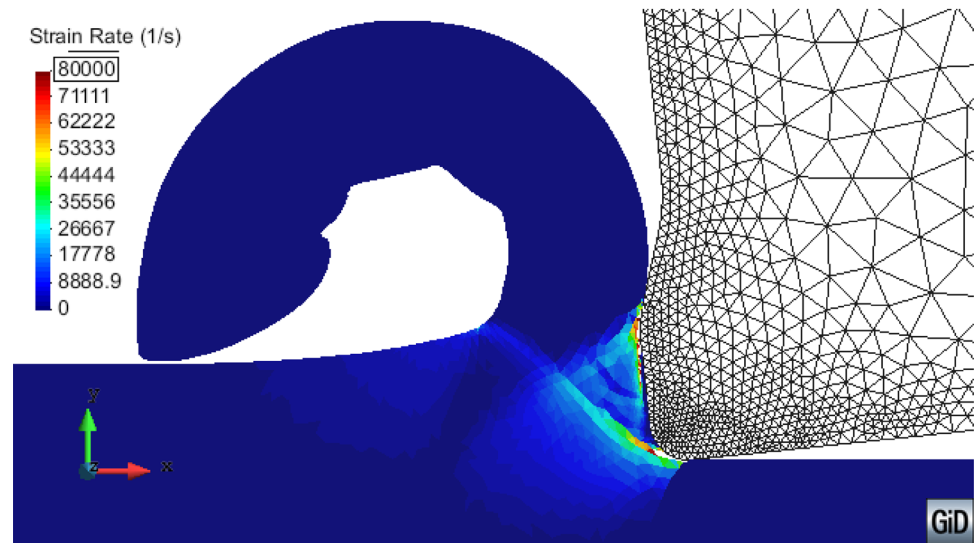


Fig. 4 Temperature distribution

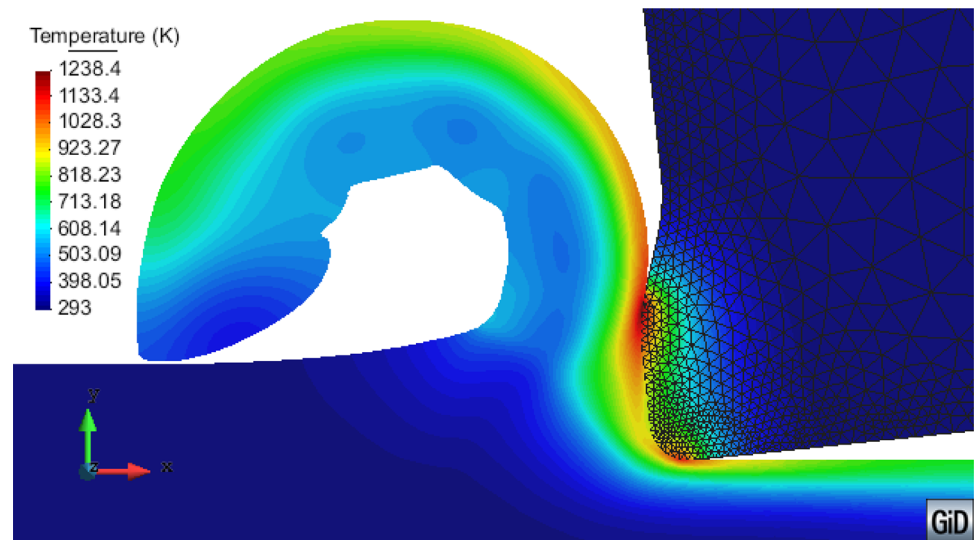


Fig. 5 Dislocation density distribution

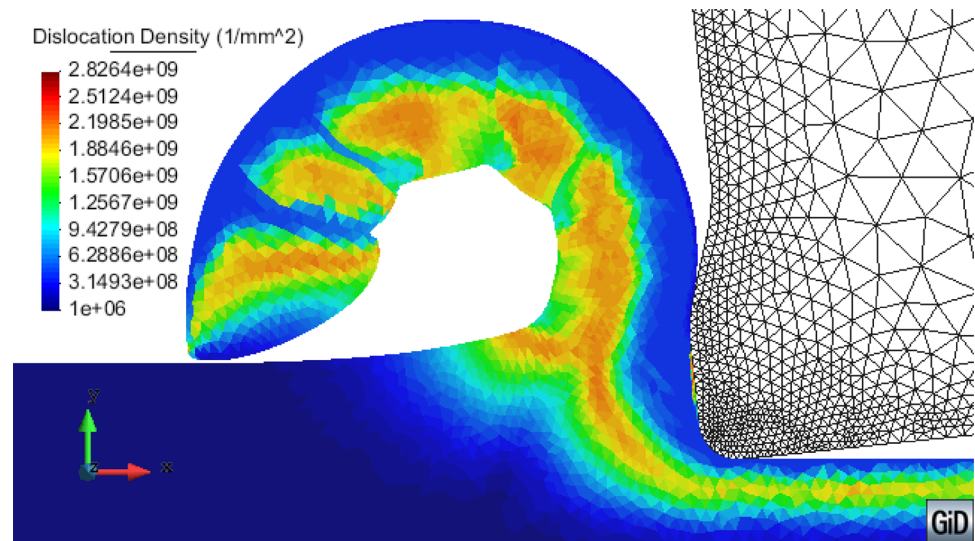
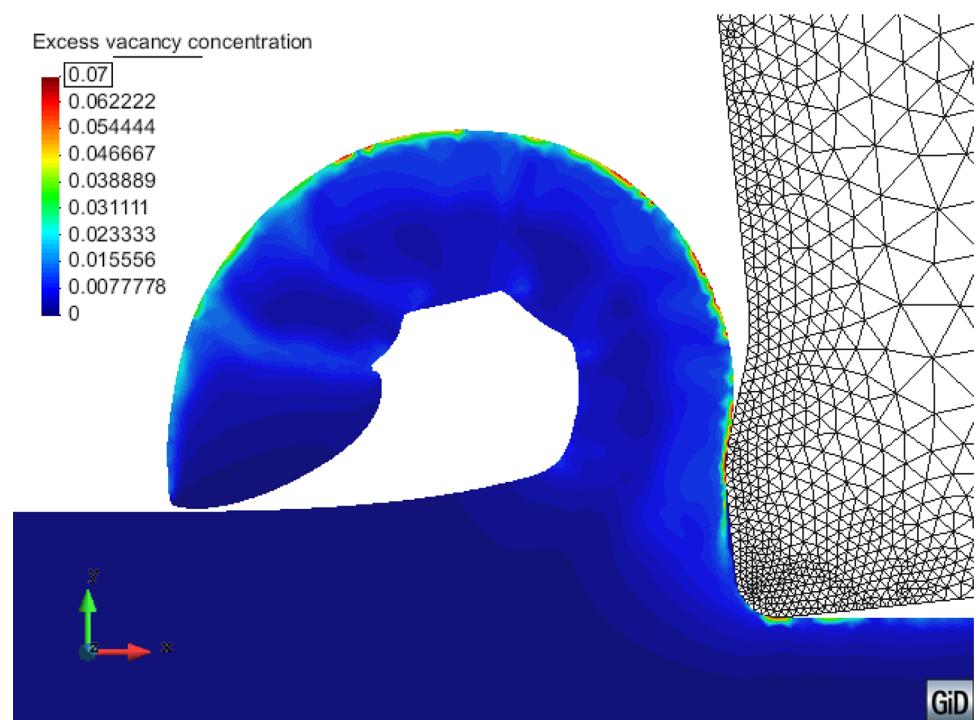


Fig. 6 Excess of vacancy concentration distribution



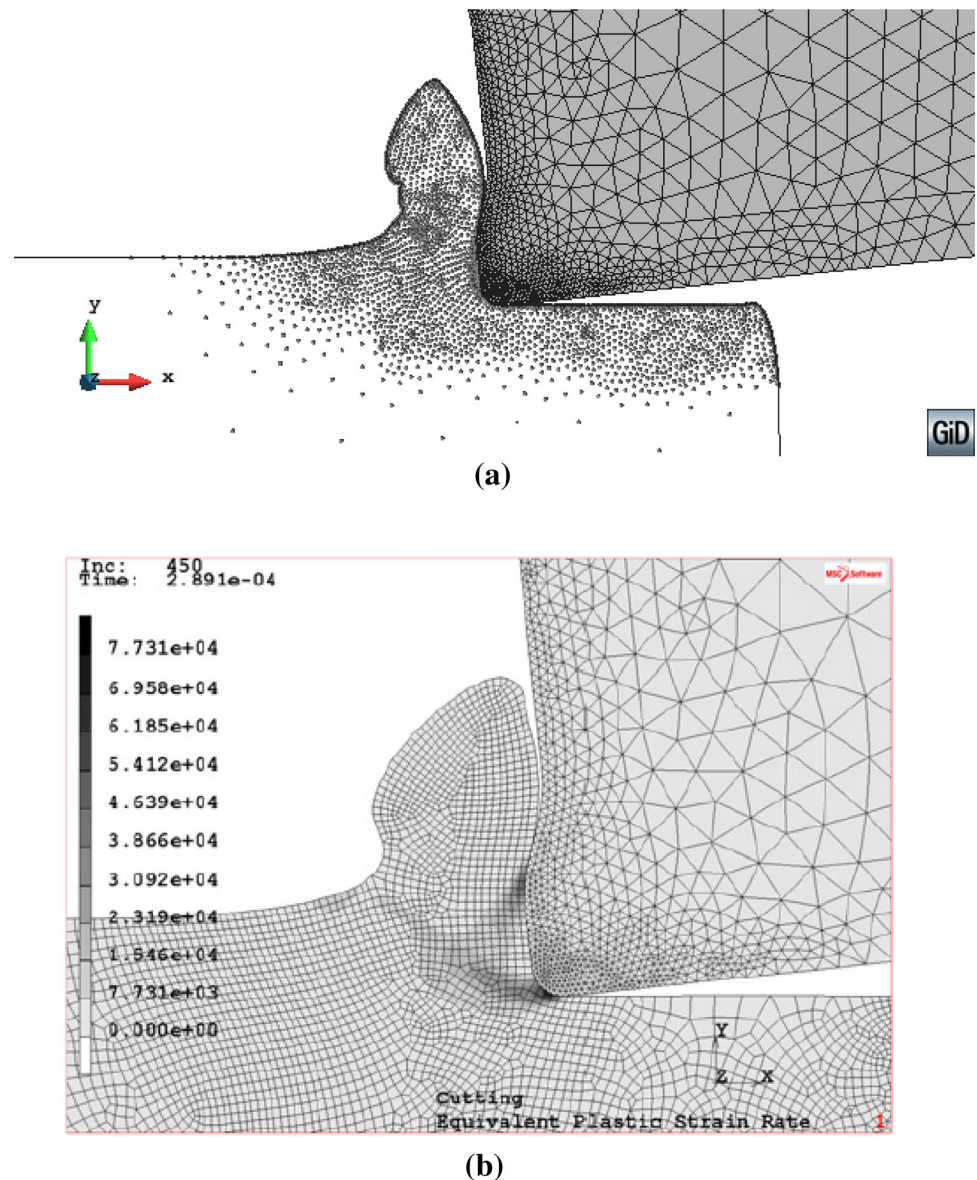
8.2 Material response

All figures presented in this section correspond to the steady state conditions. The results shown are for the cutting velocity of 180 m min^{-1} and feed of 0.15 mm . Figure 3 illustrates distribution of plastic strain rates in the primary and the secondary shear zones. Figure 3 presents a maximum plastic strain rate value of $80,000 \text{ s}^{-1}$.

Temperature fields are presented in Fig. 4. Maximum temperature was generated in the contact between the chip and rake face of the tool.

The dislocation density (DD) and vacancy concentration are shown in Figs. 5 and 6, respectively. In the area close to the outer surface of the formed chip with lower-temperature level, the increased dislocation density controls the hardening, Fig. 5. Initial value of DD 10^6 mm^{-2} was increased up to 10^9 mm^{-3} . In the domains of the highest temperature concentrated close to the tool rake face, Fig. 6, the significant generation of vacancies coupled with the dislocation recovery is present, see Fig. 5.

Fig. 7 A comparison of the predicted chip shape. **a** PFEM, **b** FEM



8.3 A comparison between FEM and PFEM

In order to validate PFEM as an effective strategy to deal with machining problems, we present a small comparison with MSC.Marc based on the FEM. The comparison is performed with test no. 2 presented in Sect. 8 of this work. It is important to mention that there are some differences and similarities between the formulations used in MSC.Marc and our in-house PFEM code. They use distinct formulation, finite elements, convergence criteria, and contact laws. Main differences between the numerical models are listed below:

- MSC.Marc uses a quadrilateral finite element with reduced integration, whereas PFEM uses a linear displacement linear pressure-stabilized finite element.
- MSC.Marc uses Coulomb friction law, while PFEM uses a regularized Coulomb friction law.
- MSC.Marc uses a convergence criterion of relative displacement and forces of 0.01, whereas PFEM uses a convergence criterion of relative displacement of 0.001.
- MSC.Marc is a FEM-based code implemented in non-interpreting languages, while PFEM is implemented in a vectorized Matlab-based code.
- MSC.Marc uses adaptive mesh refinement schemes, whereas PFEM uses insertion, removal, and relocation of particles for avoiding the difficulties, associated with deformation-induced element distortion, and for resolving the different scales of the solution.

Figure 7 compares the predicted chip shapes using PFEM and FEM showing a reasonable agreement. In our previous

work [30], the FEM predicted chip shape was compared with experimental results showing similar shapes in the experimental and the numerical approach.

For the reasons mentioned above, the numerical model setup with PFEM is considered to be accurate enough to compete with current commercial softwares based in the FEM method (Abaqus, AdvantEdge, Deform, MSC.Marc) [13,15]. Concerning with the computational cost, a MATLAB code with the PFEM implementation was used for this comparison. The calculation time in a serial execution was similar to the commercial ones. We guess that, with an implementation in a more optimized programming language, the PFEM would be clearly faster.

9 Conclusions

A Lagrangian formulation for analysis of metal-cutting processes involving thermally coupled interactions between deformable continua is presented. The governing equations for the generalized continuum are discretized using elements with equal linear interpolation for the displacement and the temperature. The merits of the formulation in terms of its general applicability have been demonstrated in the solutions of four representative numerical simulations of orthogonal cutting using the PFEM.

In this work, a dislocation density model has been used together with PFEM to virtually reproduce orthogonal cutting of AISI 316L steel. Numerical results obtained using PFEM have been compared with both experimental results and numerical results. Also, results of the numerical model developed within this work are in agreement with experimental results and can predict forces close to the required precision.

A comparison with FEM results show that the PFEM model gives similar results and is accurate enough to compete with the commercial software in the market.

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 1: Linearization of the physical-based model presented in Section 6.2

Derivatives of the set of nonlinear Eq. (49) can be written as follows

$$\frac{\partial \mathbf{H}}{\partial \mathbf{q}_{11}} = 1 - \left(\frac{m}{b} \frac{1}{K_c} \frac{1}{2\sqrt{\rho_i^{n+1}}} - \Omega(T) \right) \times \dot{\epsilon}_p \Delta t + 4c_\gamma(T) \frac{D_v}{c_v} \frac{\mu b^3}{kT} \rho_i c_v \Delta t$$

$$\frac{\partial \mathbf{H}}{\partial \mathbf{q}_{21}} = -\chi \frac{\Omega_0}{Q_{vf}} m \alpha \mu b \frac{1}{2\sqrt{\rho_i}} \dot{\epsilon}_p \Delta t + D_{vm} \frac{1}{K_c^2} (c_v - c_v^{eq}) \Delta t$$

$$\frac{\partial \mathbf{H}}{\partial \mathbf{q}_{12}} = 2c_\gamma(T) \frac{D_v}{c_v} \frac{\mu b^3}{kT} (\rho_i^2 - \rho_{eq}^2) \Delta t$$

$$\frac{\partial \mathbf{H}}{\partial \mathbf{q}_{22}} = 1 + D_{vm} \left(\frac{1}{K_c^2} \rho_i + \frac{1}{g^2} \right) \Delta t$$

Appendix 2

See Tables 3, 4 and 5.

Table 3 SANMAC 316L—material parameters

Young’s modulus E (GPa)	Poisson’s ratio ν	Heat capacity C_p (J kg ⁻¹ °C ⁻¹)	Thermal expansion α (1/°C)	Thermal conduct k (W °C ⁻¹ m ⁻¹)	Density ρ (kg m ⁻³)
201	0.3	445	15.5e-6	14	7900

Table 4 Known or assumed parameters

k (JK ⁻¹)	m	b (m)	$\dot{\epsilon}_{ref}$ (s ⁻¹)	Q_{vf} (J)	χ	Ω_0 (m ³)	ξ	g (m)
1.38×10^{-23}	3.06	2.58×10^{-10}	1×10^6	2.46×10^{-19}	0.1	1.21×10^{-29}	2/3	90×10^{-6}
α	ρ_{i0} (mm ⁻³)	p	q	c_γ	Q_{vm} (J)	D_{vm0} (m ² s ⁻¹)	c_{v0}	D_{v0} (m ² s ⁻¹)
0.4	1×10^{12}	0.333	1.902	0.0275	2.38×10^{-19}	1.37×10^{-5}	2.7183	37×10^{-6}

Table 5 Temperature dependent parameters

Temperature (°C)	K_c	$\hat{\tau}$	ΔF	Ω
20	59.73	0.010390	0.5056	4.499
200	32.64	0.010000	0.8609	7.329
400	56.09	0.007392	0.9117	10.58
600	48.15	0.005643	0.9913	10.29
700	45.69	0.006256	0.9297	13.68
800	45.69	0.0007939	0.7789	13.54
900	55.13	0.005409	0.6795	13.05
1100	121.1	0.032410	0.6242	14.73
1300	200.0	0.050000	0.6615	28.62

References

- Idelsohn SR, Oñate E, Pin FD (2004) The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves. *Int J Numer Methods Eng* 61(7):964–989
- Oñate E, Idelsohn SR, del Pin F, Aubry R (2004) The particle finite element method—an overview. *Int J Comput Methods* 01(02):267–307. doi:10.1142/S0219876204000204
- Idelsohn SR, Oñate E, Pin FD, Calvo N (2006) Fluid-structure interaction using the particle finite element method. *Comput Methods Appl Mech Eng* 195(17–18):2100–2123. doi:10.1016/j.cma.2005.02.026
- Lindgren L-E, Domkin K, Hansson S (2008) Dislocations, vacancies and solute diffusion in physical based plasticity model for AISI 316L. *Mech Mater* 40(11):907–919. doi:10.1016/j.mechmat.2008.05.005
- Wedberg D, Lindgren L-E (2015) Modelling flow stress of AISI 316L at high strain rates. *Mech Mater* 1:194–207. doi:10.1016/j.mechmat.2015.07.005
- Oñate E, Rossi R, Idelsohn SR, Butler KM (2010) Melting and spread of polymers in fire with the particle finite element method. *Int J Numer Methods Eng* 81(8):1046–1072. doi:10.1002/nme.2731
- Oñate E, Idelsohn SR, Celigueta MA, Rossi R (2008) Advances in the particle finite element method for the analysis of fluid-multibody interaction and bed erosion in free surface flows. *Comput Methods Appl Mech Eng* 197(19–20):1777–1800. doi:10.1016/j.cma.2007.06.005
- Carbonell J, Oñate E, Suárez B (2010) Modeling of ground excavation with the particle finite-element method. *J Eng Mech* 136(4):455–463. doi:10.1061/(ASCE)EM.1943-7889.0000086
- Cante JC, Riera MD, Oliver J, Prado JM, Isturiz A, Gonzalez C (2011) Flow regime analyses during the filling stage in powder metallurgy processes: experimental study and numerical modelling. *Granul Matter* 13(1):79–92. doi:10.1007/s10035-010-0225-4
- Oliver J, Cante JC, Weyler R, González C, Hernandez J (2007) Particle finite element methods in solid mechanics problems. *Comput Methods Appl Sci* 7:87–103
- Sabel M, Sator C, Müller R (2014) A particle finite element method for machining simulations. *Comput Mech* 54(1):123–131. doi:10.1007/s00466-014-1025-1
- Rodríguez J, Arrazola P, Cante J, Kortabarria A, Oliver J (2013) A sensibility analysis to geometric and cutting conditions using the particle finite element method (PFEM). *Procedia CIRP* 8:105–110. doi:10.1016/j.procir.2013.06.073
- Rodríguez JM, Carbonell JM, Cante JC, Oliver J (2015) The particle finite element method (PFEM) in thermo-mechanical problems. *Int J Numer Methods Eng*. doi:10.1002/nme.5186
- Rodríguez JM (2014) Numerical modeling of metal cutting processes using the particle finite element method (PFEM). *Universitat Politècnica de Catalunya (UPC), Barcelona*
- Rodríguez JM, Cante JC, Oliver X (2015) On the numerical modelling of machining processes via the particle finite element method (PFEM), vol 156. CIMNE, Barcelona
- Laese A, Rossi R, Oñate E, Idelsohn SR (2012) A coupled PFEM-Eulerian approach for the solution of porous FSI problems. *Comput Mech* 50(6):805–819. doi:10.1007/s00466-012-0768-9
- Lee DT, Schachter BJ (1980) Two algorithms for constructing a Delaunay triangulation. *Int J Comput Inf Sci*, 219–242. doi:10.1007/bf00977785
- George PL, Borouchaki H (1998) Delaunay triangulation and meshing: application to finite elements. HERMES, Paris
- Xu X, Harada K (2003) Automatic surface reconstruction with alpha-shape method. *Vis Comput* 19(7–8):431–443
- Edelsbrunner H, Mücke EP (1994) Three-dimensional alpha shapes. *ACM Trans Graph* 13(1):43–72. doi:10.1145/174462.156635
- Belytschko T, Liu WK, Moran B (2000) *Nonlinear finite element for continua and structures*. Wiley, Chichester
- Fish J, Belytschko T (2007) *A first course in finite elements*. Wiley, Chichester
- Zienkiewicz OC, Taylor RL, Zhu JZ (2013) *The finite element method: its basis and fundamentals*, 7th edn. Elsevier, Oxford
- Zienkiewicz OC, Taylor RL, Fox DD (2014) *The finite element method for solid and structural mechanics*. Elsevier, Oxford
- Dohrmann CR, Bochev PB (2004) A stabilized finite element method for the Stokes problem based on polynomial pressure projections. *Int J Numer Methods Fluids* 46(2):183–201
- Bochev PB, Dohrmann CR, Gunzburger MD (2008) Stabilization of low-order mixed finite elements for the Stokes equations. *SIAM J Numer Anal* 44(1):82–101
- Lal GK (2009) *Introduction to machining science*, 3rd edn. New Age, New Delhi
- Sekhon GS, Chenot JL (1993) Numerical simulation of continuous chip formation during non-steady orthogonal cutting simulation. *Eng Comput* 10(1):31–48
- Simo JC, Hughes TJR (1998) *Computational inelasticity*. Springer-Verlag, New York
- Svoboda A, Wedberg D, Lindgren L-E (2010) Simulation of metal cutting using a physically based plasticity model. *Model Simul Mater Sci Eng* 18(7):075005
- Kalhor V, Wedberg D, Lindgren L-E (2010) Simulation of mechanical cutting using a physical based material model. *Int J Mater Form* 3(1):511–514. doi:10.1007/s12289-010-0819-8
- Wedberg D, Svoboda A, Lindgren L-E (2012) Modelling high strain rate phenomena in metal cutting simulation. *Model Simul Mater Sci Eng* 20(8):085006
- Arrazola PJ, Özel T (2010) Investigations on the effects of friction modeling in finite element simulation of machining. *Int J Mech Sci* 52(1):31–42
- Özel T (2006) The influence of friction models on finite element simulations of machining. *Int J Mach Tools Manuf* 46(5):518–530