ABAQUS implementation of distortion gradient plasticity

Emilio Martínez-Pañeda[a,*]

[a]*Department of Civil and Environmental Engineering, Imperial College London, London SW7 2AZ, UK*

Abstract

Documentation that accompanies the file UELDGP.f - a user element subroutine (UEL) with an implicit implementation of distortion gradient plasticity (Gurtin 2004), incorporating the role of the plastic spin as well as dissipative and energetic higher order contributions. If using this code for research or industrial purposes, please cite:

S. Fuentes-Alonso, E. Martínez-Pañeda. Fracture in distortion gradient plasticity. International Journal of Engineering Science 156: 103369 (2020)

The files can be downloaded from www.imperial.ac.uk/mechanics-materials and www.empaneda.com, along with many other UEL, UMAT and USDFLD subroutines.

Keywords:
Distortion gradient plasticity, Finite element analysis, Higher-order, ABAQUS, Backward Euler

1. Introduction

Strain gradient plasticity models have enjoyed significant attention in the past 20 years. Constitutive theories accounting for the role of plastic strain gradients, and their associated length scale parameters, have enabled capturing the size effects observed in metals at small scales as well as regularizing

*Corresponding author.
Email address: e.martinez-paneda@imperial.ac.uk (Emilio Martínez-Pañeda)

otherwise ill-posed boundary value problems at the onset of material softening (see, e.g., [1–8] and references therein). Due to its superior modelling capabilities, there is an increasing interest in *distortion* gradient plasticity theory [9], where the role of the plastic spin is accounted for in the free energy and the dissipation [10–14]. Originally proposed by Gurtin [9], this class of gradient plasticity models builds upon the incompatibility of the plastic part of the displacement gradient and the macroscopic characterisation of the Burgers vector [9, 15–17] to rigorously define Nye's tensor as:

$$\alpha_{ij} = \epsilon_{jkl}\gamma^p_{il,k} \qquad (\boldsymbol{\alpha} = \operatorname{curl}\boldsymbol{\gamma}^p) \tag{1}$$

where γ^p_{ij} is the plastic distortion - the plastic part of the displacement gradient.

This report is supplementary to our recent work, Ref. [18], and accompanies UELDGP.f, a user element subroutine for implementing distortion gradient plasticity in ABAQUS. A comprehensive description of the underlying theoretical model is provided in Section 2, while details of the finite element discretisation are given in Section **??**. Some readers might wish to jump directly to Section **??**, where usage instructions are provided together with a simple validation example.

2. Distortion Gradient Plasticity

The equations of this section refer to the mechanical response of a body occupying a space region Ω with an external surface S of outward normal n_i. More details about the higher order theory of distortion gradient plasticity can be found in Ref. [9].

2.1. Variational principles and balance equations

Within a small strain formulation, the displacement gradient $u_{i,j}$ can be decomposed into its elastic and plastic parts:

$$u_{i,j} = \gamma^e_{ij} + \gamma^p_{ij} \tag{2}$$

Where γ^p_{ij}, the plastic distortion, which characterises the evolution of dislocations and other defects through the crystal structure, may in turn be decomposed into its symmetric and skew parts:

$$\gamma^p_{ij} = \varepsilon^p_{ij} + \vartheta^p_{ij} \tag{3}$$

Unlike the plastic strain field ε^p_{ij}, the plastic rotation ϑ^p_{ij} is essentially irrelevant in a conventional theory. However, as pointed out by Gurtin [9], phenomenological models involving Nye's dislocation density tensor α_{ij} as primal higher order kinematic variable,

$$\alpha_{ij} = \epsilon_{jkl}\gamma^p_{il,k} \qquad (\boldsymbol{\alpha} = \operatorname{curl}\boldsymbol{\gamma}^p) \tag{4}$$

must account for the plastic spin since the macroscopic characterisation of the Burgers vector involves both the symmetric and skew parts of the plastic distortion

$$\epsilon_{jkl}\gamma^p_{il,k} = \epsilon_{jkl}\varepsilon^p_{il,k} + \epsilon_{jkl}\vartheta^p_{il,k} \qquad (\operatorname{curl}\boldsymbol{\gamma}^p = \operatorname{curl}\boldsymbol{\varepsilon}^p + \operatorname{curl}\boldsymbol{\vartheta}^p) \tag{5}$$

with ϵ_{jkl} denoting the alternating symbol. The internal virtual work reads:

$$\delta W_i = \int_\Omega \left(\sigma_{ij}\delta\varepsilon^e_{ij} + \zeta_{ij}\delta\alpha_{ij} + S_{ij}\delta\gamma^p_{ij} + \tau_{ijk}\delta\varepsilon^p_{ij,k}\right)\mathrm{d}V \tag{6}$$

where the Cauchy stress is denoted by σ_{ij}. In addition to conventional stresses, the principle of virtual work incorporates the so-called micro-stress tensor, S_{ij} (work conjugate to the plastic distortion, γ^p_{ij}), the defect stress ζ_{ij} (work conjugate to Nye's tensor α_{ij}, the curl of the plastic distortion) and the - here, purely dissipative - higher order stress tensor, τ_{ijk} (work conjugate to the plastic strain gradients $\varepsilon^p_{ij,k}$). By taking into account that the micro-stress tensor can be decomposed into its symmetric and skew parts: $S_{ij} = q_{ij} + \omega_{ij}$, the internal virtual work statement can be expressed as:

$$\delta W_i = \int_\Omega \left(\sigma_{ij}\delta\varepsilon_{ij} + \zeta_{ij}\delta\alpha_{ij} + \left(q_{ij} - \sigma'_{ij}\right)\delta\varepsilon^p_{ij} + \omega_{ij}\delta\vartheta^p_{ij} + \tau_{ijk}\delta\varepsilon^p_{ij,k}\right)\mathrm{d}V \tag{7}$$

with the prime symbol $'$ denoting deviatoric quantities. Applying Gauss' divergence theorem to (7) one can readily derive the strong form equations:

$$\sigma_{ij,j} = 0 \tag{8}$$

$$q_{ij} - \sigma'_{ij} - \tau_{ijk,k} + \eta'_{ij} = 0 \tag{9}$$

$$\omega_{ij} + \varphi_{ij} = 0 \tag{10}$$

and the natural boundary conditions:

$$T_i = \sigma_{ij}n_j \tag{11}$$

$$\Upsilon'_{ij} + \tau_{ijk}n_k = t^\varepsilon_{ij} \tag{12}$$

$$\Delta_{ij} = t^\vartheta_{ij} \tag{13}$$

where η_{ij} and φ_{ij} are, respectively, the symmetric and skew-symmetric parts of the curl of the defect stress $\xi_{ij} = \epsilon_{jkl}\zeta_{il,k} = \eta_{ij}+\varphi_{ij}$; and equivalently, Υ_{ij} and Δ_{ij} respectively denote the symmetric and skew-symmetric parts of the cross product of the defect stress and the outward normal $\Gamma_{ij} = \epsilon_{jkl}\zeta_{il}n_k = \Upsilon_{ij}+\Delta_{ij}$. Also, T_i are the conventional tractions, work conjugate to the displacements, while t^{ε}_{ij} and t^{ϑ}_{ij} denote the higher order tractions work conjugate to plastic strains ε^{p}_{ij} and plastic rotations ϑ^{p}_{ij}, respectively.

2.2. Energetic contributions

In order to account for the influence of GNDs, the free energy is chosen to depend on both the elastic strain ε^{e}_{ij} and Nye's tensor α_{ij}:

$$\Psi = \frac{1}{2}C_{ijkl}\varepsilon^{e}_{ij}\varepsilon^{e}_{kl} + \Phi\left(\alpha_{ij}\right) \tag{14}$$

with C_{ijkl} being the elastic stiffness and $\Phi\left(\alpha_{ij}\right)$ the defect energy that accounts for the recoverable mechanisms associated with the development of GNDs. The widely used quadratic form of the defect energy is adopted

$$\Phi\left(\alpha_{ij}\right) = \frac{1}{2}\mu L_E^2\alpha_{ij}\alpha_{ij} \tag{15}$$

Accordingly, the defect stress equals:

$$\zeta_{ij} = \frac{\partial\Phi\left(\alpha_{ij}\right)}{\partial\alpha_{ij}} = \mu L_E^2\alpha_{ij} \tag{16}$$

with μ being the shear modulus and L_E the energetic material length scale.

2.3. Dissipative contributions

A gradient-enhanced phenomenological effective plastic flow rate is defined,

$$\dot{E}^{p} = \sqrt{\frac{2}{3}\dot{\varepsilon}^{p}_{ij}\dot{\varepsilon}^{p}_{ij} + \chi\dot{\vartheta}^{p}_{ij}\dot{\vartheta}^{p}_{ij} + \frac{2}{3}L_D^2\dot{\varepsilon}^{p}_{ij,k}\dot{\varepsilon}^{p}_{ij,k}} \tag{17}$$

where L_D is a dissipative length parameter and χ is the parameter governing the dissipation due to the plastic spin. Bardella [11] has analytically identified the value of χ that captures the mechanical response of a crystal subjected to multi-slip under simple shear:

$$\chi = \left[\frac{3}{2} + \frac{\sigma_Y}{\mu\varepsilon_Y}\left(\frac{L_D}{L_E}\right)^2\right]^{-1} \tag{18}$$

being σ_Y and ε_Y non-negative material parameters, which implies a value for χ bounded between 0 and 2/3. The flow resistance Σ, work conjugate to $\dot{E}^p$, is given by

$$\Sigma = \sqrt{\frac{3}{2} q_{ij} q_{ij} + \frac{1}{\chi} \omega_{ij} \omega_{ij} + \frac{3}{2L_D^2} \tau_{ijk} \tau_{ijk}} \tag{19}$$

Such that the unrecoverable stresses equal

$$q_{ij} = \frac{2}{3} \frac{\Sigma}{\dot{E}^p} \dot{\varepsilon}^p_{ij}, \qquad \omega_{ij} = \chi \frac{\Sigma}{\dot{E}^p} \dot{\vartheta}^p_{ij}, \qquad \tau_{ijk} = \frac{2}{3} L_D^2 \frac{\Sigma}{\dot{E}^p} \dot{\varepsilon}^p_{ij,k} \tag{20}$$

3. Numerical implementation

The flow theory of distortion gradient plasticity, described in Section 2, is implemented in a robust, backward Euler finite element framework. This is largely facilitated by the definition of a new viscoplastic potential, able to model both rate-dependent and rate-independent behaviour.

3.1. Viscoplastic law

Gradient plasticity theories are commonly implemented within a rate-dependent setting, taking advantage of its well-known computational capabilities and circumventing complications associated with identifying active plastic zones in the corresponding time independent model. In the context of rate-dependent gradient plasticity models, an effective flow resistance Σ is defined,

$$\Sigma\left(\dot{E}^p, E^p\right) = \sigma_F\left(E^p\right) V\left(\dot{E}^p\right) \tag{21}$$

which is work conjugated to the gradient-enhanced effective plastic flow rate $\dot{E}^p$. Here, σ_F is the current flow stress, which depends on the initial yield stress σ_Y and the hardening law. Several viscoplastic laws have been proposed in the literature; the most exploited one is arguably the following, (see, e.g., [14])

$$\mathscr{V}\left(\dot{E}^p, E^p\right) = \frac{\sigma_F\left(E^p\right) \dot{\varepsilon}_0}{m+1} \left(\frac{\dot{E}^p}{\dot{\varepsilon}_0}\right)^{m+1} \tag{22}$$

so that

$$\Sigma\left(\dot{E}^p, E^p\right) = \sigma_F V(\dot{E}^p) = \sigma_F\left(E^p\right) \left(\frac{\dot{E}^p}{\dot{\varepsilon}_0}\right)^m \tag{23}$$

with m being the material rate sensitivity exponent, $\dot{\varepsilon}_0$ the reference strain rate and $V(\dot{E}^p)$ the viscoplastic function. However, under this choice the

initial tangent is infinite and the derivative $\partial\Sigma/\partial\dot{E}^p$ tends to infinity if $\dot{E}^p \to 0$, making the finite element system ill-conditioned for small values of $\dot{E}^p$. To overcome these numerical issues, Panteghini and Bardella [19] proposed the following viscoplastic function,

$$V(\dot{E}^p) = \begin{cases} \frac{\dot{E}^p}{2\dot{\varepsilon}_0} & \text{if } \dot{E}^p/\dot{\varepsilon}_0 \leq 1 \\ 1 - \frac{\dot{\varepsilon}_0}{2\dot{E}^p} & \text{if } \dot{E}^p/\dot{\varepsilon}_0 > 1 \end{cases} \tag{24}$$

In this way, the contribution of $\partial\Sigma/\partial\dot{E}^p$ will remain bounded when $\dot{E}^p \to 0$. This viscoplastic function is intended to reproduce the rate-independent limit in a robust manner, which is attained when $\dot{\varepsilon}_0 \to 0$. We extend the work by Panteghini and Bardella [19] to develop a viscoplastic algorithm that can overcome the aforementioned numerical issues, and enables modelling both rate-dependent and rate-independent behaviour by recovering the well-known viscoplastic function $V\left(\dot{E}^p\right) = \left(\dot{E}^p/\dot{\varepsilon}_0\right)^m$. For this purpose, a threshold effective plastic strain rate is defined $\dot{E}^p_*$ such that the viscoplastic function reads,

$$V(\dot{E}^p) = \begin{cases} \frac{\dot{E}^p}{\varpi\dot{\varepsilon}_0} & \text{if } \dot{E}^p m/\dot{E}^p_* \leq 1 \\ \left(\frac{\dot{E}^p - \frac{1-m}{m}\dot{E}^p_*}{\dot{\varepsilon}_0}\right)^m & \text{if } \dot{E}^p m/\dot{E}^p_* > 1 \end{cases} \tag{25}$$

where ϖ is a small positive constant ($\varpi << 1$). A smooth transition is obtained by computing the critical $\dot{E}^p_*$ from the relation between the derivatives,

$$\dot{E}^p_* = \dot{\varepsilon}_0 \left(\frac{1}{\varpi m}\right)^{1/(m-1)} \tag{26}$$

and by offsetting the curve a distance $\dot{E}^p_*(1-m)/m$. This distance corresponds to the intersection between the abscissa axis and the tangent line at the critical point. In this way, we are able to reproduce a mechanical response that accurately follows the classic viscoplastic power law while providing a robust numerical framework. Representative curves for the aforementioned viscoplastic functions are shown in Fig. 1; the regularisation proposed here approximates the classic viscoplastic function very well, enabling it to reproduce the rate sensitivity of metals, while retaining the robustness of the proposal by Panteghini and Bardella [19].

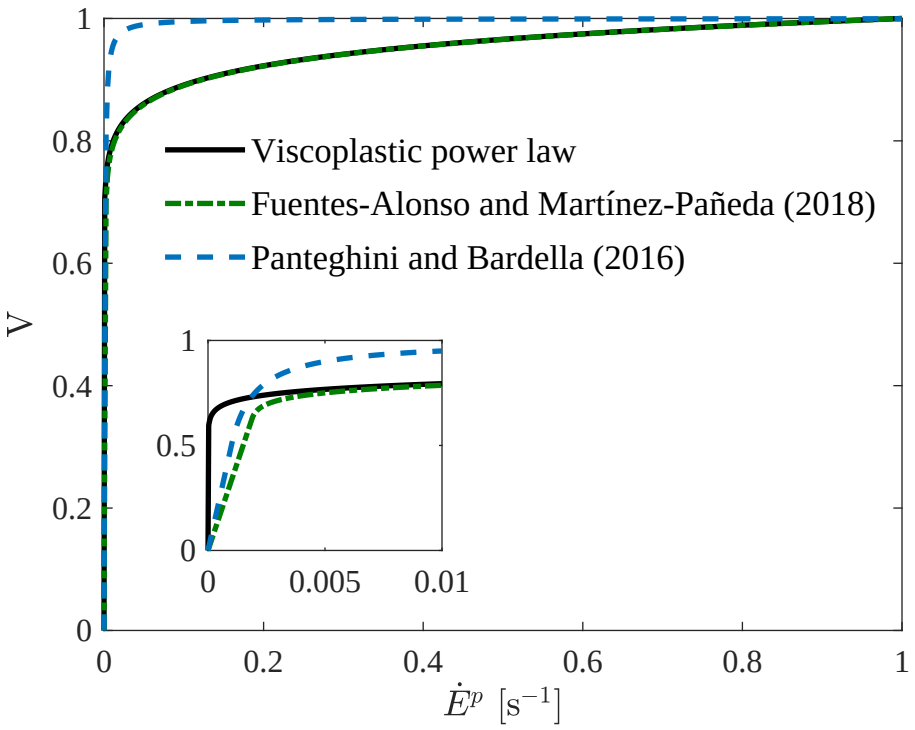

Figure 1: Comparison between the classic viscoplastic power law, the viscoplastic function presented [18] and the one proposed by Panteghini and Bardella [19]. In the function by Panteghini and Bardella [19] the reference strain rate $\dot{\varepsilon}_0$ equals 10^3 s^{-1}, while in the other two cases a rate sensitivity exponent of $m = 0.05$ and a reference strain rate $\dot{\varepsilon}_0 = 1$ are adopted; these choices pertain only to the present graph.

3.2. Finite element discretisation

The finite element framework takes displacements, plastic strains and plastic spin as the primary kinematic variables. Adopting symbolic and Voigt notation, the nodal variables for the displacement field $\hat{\boldsymbol{u}}$, the plastic strains $\hat{\boldsymbol{\varepsilon}}^p$, and the plastic spin $\hat{\boldsymbol{\vartheta}}^p$ are interpolated as,

$$\boldsymbol{u} = \sum_{n=1}^{k} \boldsymbol{N}_n^{\boldsymbol{u}} \hat{\boldsymbol{u}}_n, \qquad \boldsymbol{\varepsilon}^p = \sum_{n=1}^{k} \boldsymbol{N}_n^{\boldsymbol{\varepsilon}^p} \hat{\boldsymbol{\varepsilon}}_n^p, \qquad \boldsymbol{\vartheta}^p = \sum_{n=1}^{k} \boldsymbol{N}_n^{\boldsymbol{\vartheta}^p} \hat{\boldsymbol{\vartheta}}_n^p \tag{27}$$

Here, N_n denotes the shape function associated with node n, for a total number of nodes k. Similarly, the related gradient and curl-based quantities

are discretised as

$$\boldsymbol{\varepsilon} = \sum_{n=1}^{k} \boldsymbol{B}_n^{\boldsymbol{u}} \hat{\boldsymbol{u}}_n, \qquad \nabla \boldsymbol{\varepsilon}^p = \sum_{n=1}^{k} \boldsymbol{B}_n^{\boldsymbol{\varepsilon}^p} \hat{\boldsymbol{\varepsilon}}_n^p, \qquad \boldsymbol{\alpha} = \sum_{n=1}^{k} \left(\boldsymbol{M}_n^{\boldsymbol{\varepsilon}^p} \hat{\boldsymbol{\varepsilon}}_n^p + \boldsymbol{M}_n^{\boldsymbol{\vartheta}^p} \hat{\boldsymbol{\vartheta}}_n^p \right) \tag{28}$$

with the $\boldsymbol{B}$ and $\boldsymbol{M}$ matrices given explicitly below. Assume 2D plane strain conditions; accordingly, for an element with k nodes, the nodal variables read,

$$\hat{\boldsymbol{u}} = \begin{bmatrix} \hat{u}_x^{(1)} & \hat{u}_y^{(1)} & \cdots & \hat{u}_x^{(k)} & \hat{u}_y^{(k)} \end{bmatrix}^T \tag{29}$$

$$\hat{\boldsymbol{\varepsilon}}^p = \begin{bmatrix} \hat{\varepsilon}_x^{p\,(1)} & \hat{\varepsilon}_y^{p\,(1)} & \hat{\gamma}_{xy}^{p\,(1)} & \cdots & \hat{\varepsilon}_x^{p\,(k)} & \hat{\varepsilon}_y^{p\,(k)} & \hat{\gamma}_{xy}^{p\,(k)} \end{bmatrix}^T \tag{30}$$

$$\hat{\vartheta}_{xy}^p = \begin{bmatrix} \hat{\vartheta}_{xy}^{p\,(1)} & \cdots & \hat{\vartheta}_{xy}^{p\,(k)} \end{bmatrix}^T \tag{31}$$

The shape functions matrices for a given node n are then given by,

$$\boldsymbol{N}_n^{\boldsymbol{u}} = \begin{bmatrix} N_n & 0 \\ 0 & N_n \end{bmatrix}; \qquad \boldsymbol{N}_n^{\boldsymbol{\varepsilon}^p} = \begin{bmatrix} N_n & 0 & 0 \\ 0 & N_n & 0 \\ -N_n & -N_n & 0 \\ 0 & 0 & N_n \end{bmatrix} \tag{32}$$

with $\boldsymbol{N}_n^{\boldsymbol{\vartheta}^p}$ being, in plane strain conditions, the scalar N_n for node n. While the interpolation matrices for gradient and curl-based quantities are given by,

$$\boldsymbol{B}_n^{\boldsymbol{u}} = \begin{bmatrix} \frac{\partial N_n}{\partial x} & 0 \\ 0 & \frac{\partial N_n}{\partial y} \\ 0 & 0 \\ \frac{\partial N_n}{\partial y} & \frac{\partial N_n}{\partial x} \end{bmatrix}; \qquad \boldsymbol{B}_n^{\boldsymbol{\varepsilon}^p} = \begin{bmatrix} \frac{\partial N_n}{\partial x} & 0 & 0 \\ \frac{\partial N_n}{\partial y} & 0 & 0 \\ 0 & \frac{\partial N_n}{\partial x} & 0 \\ 0 & \frac{\partial N_n}{\partial y} & 0 \\ -\frac{\partial N_n}{\partial x} & -\frac{\partial N_n}{\partial x} & 0 \\ -\frac{\partial N_n}{\partial y} & -\frac{\partial N_n}{\partial y} & 0 \\ 0 & 0 & \frac{\partial N_n}{\partial x} \\ 0 & 0 & \frac{\partial N_n}{\partial y} \end{bmatrix}; \tag{33}$$

and,

$$\boldsymbol{M}_n^{\boldsymbol{\varepsilon}^p} = \begin{bmatrix} -\frac{\partial N_n}{\partial y} & 0 & \frac{1}{2}\frac{\partial N_n}{\partial x} \\ 0 & \frac{\partial N_n}{\partial x} & -\frac{1}{2}\frac{\partial N_n}{\partial y} \\ -\frac{\partial N_n}{\partial y} & -\frac{\partial N_n}{\partial y} & 0 \\ \frac{\partial N_n}{\partial x} & \frac{\partial N_n}{\partial x} & 0 \end{bmatrix}; \qquad \boldsymbol{M}_n^{\boldsymbol{\vartheta}^p} = \begin{bmatrix} \frac{\partial N_n}{\partial x} \\ \frac{\partial N_n}{\partial y} \\ 0 \\ 0 \end{bmatrix} \tag{34}$$

Accordingly, one can discretise the internal virtual work (7) as,

$$\delta W_i = \int_\Omega \Big\{ \left(\boldsymbol{B}_n^{\boldsymbol{u}}\right)^T \boldsymbol{\sigma}\, \delta\hat{\boldsymbol{u}}_n + \left[\left(\boldsymbol{N}_n^{\vartheta^p}\right)^T \boldsymbol{\omega} + \left(\boldsymbol{M}_n^{\vartheta^p}\right)^T \boldsymbol{\zeta}\right] \delta\hat{\vartheta}_n^p + \left[\left(\boldsymbol{N}_n^{\boldsymbol{\varepsilon}^p}\right)^T (\boldsymbol{q}-\boldsymbol{\sigma}) + \left(\boldsymbol{B}_n^{\boldsymbol{\varepsilon}^p}\right)^T \boldsymbol{\tau} + \left(\boldsymbol{M}_n^{\boldsymbol{\varepsilon}^p}\right)^T \boldsymbol{\zeta}\right] \delta\hat{\boldsymbol{\varepsilon}}_n^p \Big\}\, \mathrm{d}V \tag{35}$$

Differentiating the internal virtual work with respect to the variation of the nodal variables provides the residuals for each kinematic variable as:

$$\boldsymbol{R}_n^{\boldsymbol{u}} = \int_\Omega \left(\boldsymbol{B}_n^{\boldsymbol{u}}\right)^T \boldsymbol{\sigma}\, \mathrm{d}V \tag{36}$$

$$\boldsymbol{R}_n^{\boldsymbol{\varepsilon}^p} = \int_\Omega \left[\left(\boldsymbol{N}_n^{\boldsymbol{\varepsilon}^p}\right)^T (\boldsymbol{q}-\boldsymbol{\sigma}) + \left(\boldsymbol{B}_n^{\boldsymbol{\varepsilon}^p}\right)^T \boldsymbol{\tau} + \left(\boldsymbol{M}_n^{\boldsymbol{\varepsilon}^p}\right)^T \boldsymbol{\zeta}\right] \mathrm{d}V \tag{37}$$

$$\boldsymbol{R}_n^{\vartheta^p} = \int_\Omega \left[\left(\boldsymbol{N}_n^{\vartheta^p}\right)^T \boldsymbol{\omega} + \left(\boldsymbol{M}_n^{\vartheta^p}\right)^T \boldsymbol{\zeta}\right] \mathrm{d}V \tag{38}$$

The components of the consistent tangent stiffness matrices $\boldsymbol{K}_{nm}$ are obtained by considering the constitutive relations and differentiating the residuals with respect to the incremental nodal variables:

$$\boldsymbol{K}_{nm}^{u,u} = \frac{\partial \boldsymbol{R}_n^u}{\partial \boldsymbol{u}_m} = \int_\Omega \left(\boldsymbol{B}_n^{\boldsymbol{u}}\right)^T \boldsymbol{C}\, \boldsymbol{B}_m^{\boldsymbol{u}}\, \mathrm{d}V \tag{39}$$

$$\boldsymbol{K}_{nm}^{u,\varepsilon^p} = \frac{\partial \boldsymbol{R}_n^u}{\partial \boldsymbol{\varepsilon}_m^p} = -\int_\Omega \left(\boldsymbol{B}_n^{\boldsymbol{u}}\right)^T \boldsymbol{C} \boldsymbol{N}_m^{\boldsymbol{\varepsilon}^p}\, \mathrm{d}V \tag{40}$$

$$\boldsymbol{K}_{nm}^{\varepsilon^p,u} = \frac{\partial \boldsymbol{R}_n^{\boldsymbol{\varepsilon}^p}}{\partial \boldsymbol{u}_m} = -\int_\Omega \left(\boldsymbol{N}_n^{\boldsymbol{\varepsilon}^p}\right)^T \boldsymbol{C}\, \boldsymbol{B}_m^{\boldsymbol{u}}\, \mathrm{d}V \tag{41}$$

$$\boldsymbol{K}_{nm}^{\varepsilon^p,\varepsilon^p} = \frac{\partial \boldsymbol{R}_n^{\boldsymbol{\varepsilon}^p}}{\partial \boldsymbol{\varepsilon}_m^p} = \int_\Omega \Bigg\{ \left(\boldsymbol{N}_n^{\boldsymbol{\varepsilon}^p}\right)^T \left[\left(\frac{\partial \boldsymbol{q}}{\partial \varepsilon_m^p} + \boldsymbol{C}\right) \boldsymbol{N}_m^{\boldsymbol{\varepsilon}^p} + \frac{\partial \boldsymbol{q}}{\partial \nabla \varepsilon_m^p} \boldsymbol{B}_m^{\boldsymbol{\varepsilon}^p}\right] + \left(\boldsymbol{B}_n^{\boldsymbol{\varepsilon}^p}\right)^T \left(\frac{\partial \boldsymbol{\tau}}{\partial \varepsilon_m^p} \boldsymbol{N}_m^{\boldsymbol{\varepsilon}^p} + \frac{\partial \boldsymbol{\tau}}{\partial \nabla \varepsilon_m^p} \boldsymbol{B}_m^{\boldsymbol{\varepsilon}^p}\right) + \left(\boldsymbol{M}_n^{\boldsymbol{\varepsilon}^p}\right)^T \frac{\partial \boldsymbol{\zeta}}{\partial \alpha_m} \boldsymbol{M}_m^{\boldsymbol{\varepsilon}^p} \Bigg\}\, \mathrm{d}V \tag{42}$$

$$\boldsymbol{K}_{nm}^{\varepsilon^p,\vartheta^p} = \frac{\partial \boldsymbol{R}_n^{\boldsymbol{\varepsilon}^p}}{\partial \boldsymbol{\vartheta}_m^p} = \int_\Omega \left[\left(\boldsymbol{N}_n^{\boldsymbol{\varepsilon}^p}\right)^T \frac{\partial \boldsymbol{q}}{\partial \vartheta_m^p} \boldsymbol{N}_m^{\vartheta^p} + \left(\boldsymbol{B}_n^{\boldsymbol{\varepsilon}^p}\right)^T \frac{\boldsymbol{\partial \tau}}{\partial \vartheta_m^p} \boldsymbol{N}_m^{\vartheta^p} + \left(\boldsymbol{M}_n^{\boldsymbol{\varepsilon}^p}\right)^T \frac{\partial \zeta}{\partial \alpha_m} \boldsymbol{M}_m^{\vartheta^p}\right] \mathrm{d}V \tag{43}$$

$$\boldsymbol{K}_{nm}^{\vartheta^p,\varepsilon^p} = \frac{\partial \boldsymbol{R}_n^{\vartheta^p}}{\partial \boldsymbol{\varepsilon}_m^p} = \int_\Omega \left[\left(\boldsymbol{N}_n^{\vartheta^p}\right)^T \left(\frac{\partial \boldsymbol{\omega}}{\partial \varepsilon_m^p} \boldsymbol{N}_m^{\boldsymbol{\varepsilon}^p} + \frac{\partial \boldsymbol{\omega}}{\partial \nabla \varepsilon_m^p} \boldsymbol{B}_m^{\boldsymbol{\varepsilon}^p}\right) + \left(\boldsymbol{M}_n^{\vartheta^p}\right)^T \frac{\partial \zeta}{\partial \alpha_m} \boldsymbol{M}_m^{\boldsymbol{\varepsilon}^p}\right] \mathrm{d}V \tag{44}$$

$$\boldsymbol{K}^{\vartheta^p,\vartheta^p}_{nm} = \frac{\partial \boldsymbol{R}^{\vartheta^p}_n}{\partial \boldsymbol{\vartheta}^p_m} = \int_\Omega \left[\left(\boldsymbol{N}^{\vartheta^p}_n \right)^T \frac{\partial \boldsymbol{\omega}}{\partial \vartheta^p_m} \boldsymbol{N}^{\vartheta^p}_m + \left(\boldsymbol{M}^{\vartheta^p}_n \right)^T \frac{\partial \zeta}{\partial \alpha_m} \boldsymbol{M}^{\vartheta^p}_m \right] \mathrm{d}V \tag{45}$$

The non-linear system of equations is solved iteratively from time step t to $(t + \Delta t)$ using the Newton-Raphson method,

$$\begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{\varepsilon}^p \\ \boldsymbol{\vartheta}^p \end{bmatrix}_{t+\Delta t} = \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{\varepsilon}^p \\ \boldsymbol{\vartheta}^p \end{bmatrix}_t - \begin{bmatrix} \boldsymbol{K}^{u,u} & \boldsymbol{K}^{u,\varepsilon^p} & \boldsymbol{0} \\ \boldsymbol{K}^{\varepsilon^p,u} & \boldsymbol{K}^{\varepsilon^p,\varepsilon^p} & \boldsymbol{K}^{\varepsilon^p,\vartheta^p} \\ \boldsymbol{0} & \boldsymbol{K}^{\vartheta^p,\varepsilon^p} & \boldsymbol{K}^{\vartheta^p,\vartheta^p} \end{bmatrix}^{-1}_t \begin{bmatrix} \boldsymbol{R}^{\boldsymbol{u}} \\ \boldsymbol{R}^{\boldsymbol{\varepsilon}^p} \\ \boldsymbol{R}^{\vartheta} \end{bmatrix}_t \tag{46}$$

The present backward Euler time integration scheme follows the work by Panteghini and Bardella [19]; see Ref. [14] for a forward Euler based implementation. The finite element framework is implemented into the commercial package ABAQUS by means of a user element (UEL) subroutine.

4. Usage instructions

A very simple example is provided to ease the use of the subroutine. As in Ref. [12], we will analyse size effects in a long strip of height H free from body forces, with isotropic behaviour and sheared between two bodies in which dislocations cannot penetrate. Hence, the displacement is fully constrained in the lower strip surface, $u_1(x_2 = 0) = u_2(x_2 = 0) = 0$, while the upper strip surface is subjected to uniform horizontal displacement $u_1(x_2 = H) = \Gamma H$ with $u_2(x_2 = H) = 0$. Here, Γ is referred to as the applied strain. Since dislocations pile-up when they reach the strip lower and upper surfaces, the plastic distortion components must be zero at $x_2 = 0$ and $x_2 = H$. The problem is essentially one-dimensional, so that the strip, unbounded along both the shearing direction x_1 and the x_3 direction, is modelled using a single column of 200 plane strain quadrilateral elements along the strip height (H) with appropriate boundary conditions at the sides of the column ($u_2 = \gamma_{11} = \gamma_{22} = 0 \ \forall x_2$).

In order to compare our results with those of Bardella [12], the following hardening law is used:[1]

$$\sigma_F(E^p) = \sigma_Y \left(\frac{E^p}{\varepsilon_Y} \right)^N \tag{47}$$

[1]The code is intended to reproduce the hardening law of Eq. (**??**) but the reader can reproduce these results by uncommenting the two lines that enable to use of the hardening rule below

and we consider the following material properties: $E = 68,380$ MPa, $\nu = 0.3$, $\varepsilon_Y = 0.02$, $\sigma_Y = 200$ MPa, and $N = 0.2$. We approach rate independent behaviour by using the viscoplastic potential by Panteghini and Bardella [19], with the reference strain rate being equal to $\dot{\varepsilon}_0 = 1 \times 10^{-4}$ s^{-1}.

As it can be seen in Fig. 2, the code reproduces very well the results obtained by Bardella [12] by applying the Rayleigh-Ritz method to the Total Complementary Energy functional.

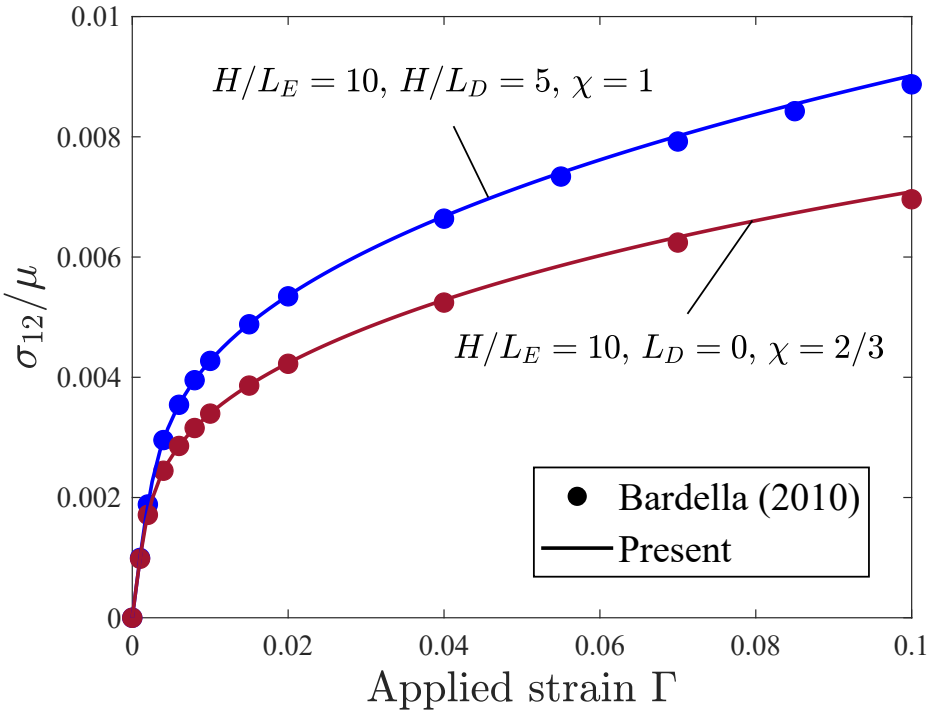

Figure 2: Validation: simple shear of a constrained strip. Comparison of the numerical results of the present model (lines) with the predictions of Bardella [12] (symbols) for different values of L, ℓ and χ. Other material parameters: $\mu = 26300$ MPa, $N = 0$, $\varepsilon_Y = 0.02$, $\sigma_Y = 200$ MPa.

The steps followed in creating the model are described below. First, we create the model in Abaqus/CAE following the usual procedure and meshing with CPE8 elements. The input file is then edited to accommodate the file for the user element (UEL) subroutine; by, e.g., using Notepad++ or the

specific scripts provided in Abaqus2Matlab [20]. The first change involves the definition of the user element, one should replace:

```
*Element, type=CPE8
```

by,

```
*USER ELEMENT,TYPE=U1,NODES=8,COORDINATES=2,PROPERTIES=10,VAR=126
1,2,3,4,5,6
*ELEMENT, TYPE=U1, ELSET=SOLID
```

where we specify the number of nodes, coordinates, properties and state variables, along with the degrees of freedom (from 1 to 6: $\Delta\hat{u}_x$, $\Delta\hat{u}_y$, $\Delta\hat{\varepsilon}^p_{xx}$, $\Delta\hat{\varepsilon}^p_{yy}$, $\Delta\hat{\gamma}^p_{xy}$, $\Delta\hat{\vartheta}^p_{xy}$). After the element connectivity listing we introduce the 10 element properties outlined before [2],

```
*UEL PROPERTY, ELSET=SOLID
68380.,0.3,200.,20.,0.,0.667,1.e-4,0.2,
0.05, 3
```

in this way the variables of the analysis can be defined without modifying the Fortran code. The 10 user-defined properties employed in the UEL subroutine are described in Table 1.

Table 1: Equivalence between the properties and the corresponding variables.

PROPS	Variable
1	E - Young's modulus
2	ν - Poisson's ratio
3	σ_Y - Initial yield stress
4	L_E - Energetic length scale parameter
5	L_D - Dissipative length scale parameter
6	χ - Plastic spin dissipation coefficient
7	$\dot{\varepsilon}_0$ - Reference strain rate
8	N - Strain hardening exponent
9	m - Rate sensitivity exponent
10	Viscoplastic flag variable (1 - [18], 2- [14], 3 - [19])

[2]The choices of L_D and L_E are given relative to H, which equals here 200 mm.

Optionally, we can define a "phantom mesh" so as to visualize results in Abaqus/Viewer as this is not possible when using user elements. By writing a very simple user material (UMAT) subroutine we can define elements with a stiffness matrix equal to zero, avoiding any influence on the simulation, and at the same time store the results as solution dependent variables that we can request as output. We include with the files a Matlab script (VirtualMesh.m), which is part of the Abaqus2Matlab package [20], that reads the original (unmodified) input file created in Abaqus/CAE (named `Job-1.inp`) and automatically creates a new file (`Job-1a.inp`) with the listing of the dummy mesh by increasing the number of elements by one order of magnitude. We copy and paste this information in the input file and add a name for the set of elements:

```
*Element, type=cpe8, elset=output
1001,    1,    2,    4,    3,  403,  404,  405,  406
1002,    3,    4,    6,    5,  405,  407,  408,  409
...
```

Then we edit the definition of our solid section to refer to this set,

```
*Solid Section, elset=output, material=user
```

Accordingly we define our material as a user material, so as to transfer the output from the UEL subroutine to the UMAT subroutine and visualize it in Abaqus/Viewer.

```
*Material, name=user
*Depvar
     22
1, S11, S11
2, S22, S22
3, S33, S33
4, S12, S12
5, E11, E11
6, E22, E22
7, E33, E33
8, E12, E12
9, EP11, EP11
10, EP22, EP22
11, EP33, EP33
```

```
12, EP12, EP12
13, Alph13, Alph13
14, Alph23, Alph23
15, Alph31, Alph31
16, Alph32, Alph32
17, Z13, Z13
18, Z23, Z23
19, Z31, Z31
20, Z32, Z32
21, PS12, PS12
22, EP, EP
*User Material, constants=1
200.,
```

The only user property defined in the material is the number of elements of the model (200). Optionally, we assign names to the user dependent variables to ease the interpretation of the results in the code (in this way, the equivalence of each state variable is self-explanatory).

Finally, when using higher order gradient plasticity models we may want to take advantage of one of their strengths: the capability of prescribing boundary conditions on the components of the plastic strain and spin. In this boundary value problem we choose to define $\varepsilon^p_{xx} = \varepsilon^p_{yy} = \varepsilon^p_{xy} = \vartheta^p_{xy} = 0$ at the bottom and top, and $\varepsilon^p_{xx} = \varepsilon^p_{yy} = 0$ at the sides. We do that by adding boundary conditions related to the degrees of freedom 3, 4, 5 and 6 as follows:

```
** Name: Bot Type: Displacement/Rotation
*Boundary
Set-1, 1, 1
Set-1, 2, 2
Set-1, 3, 3
Set-1, 4, 4
Set-1, 5, 5
Set-1, 6, 6
** Name: Sides Type: Displacement/Rotation
*Boundary
Set-2, 2, 2
Set-2, 3, 3
Set-2, 4, 4
```

and,

```
** Name: Top Type: Displacement/Rotation
*Boundary
Set-3, 1, 1, 20.
Set-3, 2, 2
Set-3, 3, 3
Set-3, 4, 4
Set-3, 5, 5
Set-3, 6, 6
```

Additionally, one should remember to request the variable `SDV` in the field output request to visualize results,

```
*Element Output, directions=YES
SDV,
```

5. Conclusions

If the code and the documentation provided here are useful please cite:

S. Fuentes-Alonso, E. Martínez-Pañeda. Fracture in distortion gradient plasticity. International Journal of Engineering Science 156: 103369 (2020)

Do not hesitate to contact for further clarifications.

6. Acknowledgments

E. Martínez-Pañeda acknowledges financial support from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme (FP7/2007-2013) under REA grant agreement n° 609405 (COFUNDPostdocDTU).

References

[1] E. Martínez-Pañeda, C. Betegón, Modeling damage and fracture within strain-gradient plasticity, International Journal of Solids and Structures 59 (2015) 208–215.

[2] E. Martínez-Pañeda, C. F. Niordson, R. P. Gangloff, Strain gradient plasticity-based modeling of hydrogen environment assisted cracking, Acta Materialia 117 (2016) 321–332.

[3] E. Martínez-Pañeda, C. F. Niordson, On fracture in finite strain gradient plasticity, International Journal of Plasticity 80 (2016) 154–167.

[4] E. Martínez-Pañeda, S. del Busto, C. Betegón, Non-local plasticity effects on notch fracture mechanics, Theoretical and Applied Fracture Mechanics 92 (2017) 276–287.

[5] E. Martínez-Pañeda, S. Natarajan, S. Bordas, Gradient plasticity crack tip characterization by means of the extended finite element method, Computational Mechanics 59 (2017) 831–842.

[6] T. V. Mathew, S. Natarajan, E. Martínez-Pañeda, Size effects in elastic-plastic functionally graded materials, Composite Structures 204 (2018) 43–51.

[7] E. Martínez-Pañeda, V. S. Deshpande, C. F. Niordson, N. A. Fleck, The role of plastic strain gradients in the crack growth resistance of metals, Journal of the Mechanics and Physics of Solids 126 (2019) 136–150.

[8] P. K. Kristensen, C. F. Niordson, E. Martínez-Pañeda, A phase field model for elastic-gradient-plastic solids undergoing hydrogen embrittlement, Journal of the Mechanics and Physics of Solids 143 (2020) 104093.

[9] M. E. Gurtin, A gradient theory of small-deformation isotropic plasticity that accounts for the Burgers vector and for dissipation due to plastic spin, Journal of the Mechanics and Physics of Solids 52 (11) (2004) 2545–2568.

[10] L. Bardella, A. Giacomini, Influence of material parameters and crystallography on the size effects describable by means of strain gradient plasticity, Journal of the Mechanics and Physics of Solids 56 (9) (2008) 2906–2934.

[11] L. Bardella, A comparison between crystal and isotropic strain gradient plasticity theories with accent on the role of the plastic spin, European Journal of Mechanics, A/Solids 28 (3) (2009) 638–646.

[12] L. Bardella, Size effects in phenomenological strain gradient plasticity constitutively involving the plastic spin, International Journal of Engineering Science 48 (5) (2010) 550–568.

[13] L. H. Poh, R. H. Peerlings, The plastic rotation effect in an isotropic gradient plasticity model for applications at the meso scale, International Journal of Solids and Structures 78-79 (2016) 57–69.

[14] E. Martínez-Pañeda, C. F. Niordson, L. Bardella, A finite element framework for distortion gradient plasticity with applications to bending of thin foils, International Journal of Solids and Structures 96 (2016) 288–299.

[15] J. M. Burgers, Some considerations on the fields of stresses connected with dislocations in a regular crystal lattice, Koninklijke Nederlandse Akademie van Wetenschappen 42 (1931) (1939) 293–325.

[16] J. F. Nye, Some geometrical relations in dislocated crystals, Acta Metallurgica 1 (2) (1953) 153–162.

[17] N. A. Fleck, J. W. Hutchinson, Strain gradient plasticity, Advances in Applied Mechanics 33 (1997) 295–361.

[18] S. Fuentes-Alonso, E. Martínez-Pañeda, Fracture in distortion gradient plasticity, International Journal of Engineering Science 156 (2020) 103369.

[19] A. Panteghini, L. Bardella, On the Finite Element implementation of higher-order gradient plasticity, with focus on theories based on plastic distortion incompatibility, Computer Methods in Applied Mechanics and Engineering 310 (2016) 840–865.

[20] G. Papazafeiropoulos, M. Muñiz-Calvente, E. Martínez-Pañeda, Abaqus2Matlab: A suitable tool for finite element post-processing, Advances in Engineering Software 105 (2017) 9–16.