Numerical integration of rate-independent BCC single crystal plasticity models: comparative study of two classes of numerical algorithms

Holanyo K. Akpama, Mohamed Ben Bettaieb, Farid Abed-Meraim

▶ **To cite this version:**

Holanyo K. Akpama, Mohamed Ben Bettaieb, Farid Abed-Meraim. Numerical integration of rate-independent BCC single crystal plasticity models: comparative study of two classes of numerical algorithms. International Journal for Numerical Methods in Engineering, 2016, 108 (5), pp.363-422. ⟨10.1002/nme.5215⟩. ⟨hal-01292713⟩

HAL Id: hal-01292713

https://hal.science/hal-01292713v1

Submitted on 23 Mar 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

HAL Authorization

Numerical integration of rate-independent BCC single crystal plasticity models: comparative study of two classes of numerical algorithms

Holanyo K. Akpama, Mohamed Ben Bettaieb*, Farid Abed-Meraim

LEM3, UMR CNRS 7239 – Arts et Métiers ParisTech, 4 rue Augustin Fresnel, 57078 Metz Cedex 3, France

DAMAS, Laboratory of Excellence on Design of Alloy Metals for low-mAss Structures, Université de Lorraine, France

*Correspondence to: Mohamed Ben Bettaieb; E-mail: Mohamed.BenBettaieb@ensam.eu

Abstract

In an incremental formulation suitable to numerical implementation, the use of rate-independent theory of crystal plasticity essentially leads to four fundamental problems. The first is to determine the set of potentially active slip systems over a time increment. The second is to select the active slip systems among the potentially active ones. The third is to compute the slip rates (or the slip increments) for the active slip systems. And the last problem is the possible non-uniqueness of slip rates. The purpose of this paper is to propose satisfactory responses to the above-mentioned first three issues by presenting and comparing two novel numerical algorithms. The first algorithm is based on the usual return-mapping integration scheme, while the second follows the so-called ultimate scheme. The latter is shown to be more relevant and efficient than the former. These comparative performances are illustrated through various numerical simulations of the mechanical behavior of single crystals and polycrystalline aggregates subjected to monotonic and complex loadings. Although these algorithms are applied in this paper to Body-Centered-Cubic (BCC) crystal structures, they are quite general and suitable for integrating the constitutive equations for other crystal structures (e.g., FCC and HCP).

Keywords: integration algorithm; finite strain; crystal plasticity; rate-independent framework; Schmid's law; multisurface plasticity

1. Introduction

Due to its ability to relate the inelastic behavior of crystalline materials to their microstructure, the modeling of the mechanical response of single crystals has remained an active research topic. The constitutive equations describing the single crystal behavior are now well understood and fairly well established (see, e.g., [1–4]). For a thorough look at various aspects of the constitutive modeling of single crystals, the reader may refer to earlier contributions (see, e.g., [5–9]). In the majority of constitutive models dedicated to single crystals, the inelastic deformation is assumed to arise solely from the slip on the crystallographic slip systems. Accordingly, deformation by diffusion, phase transformation, twinning and grain boundary sliding is not considered herein. Two different modeling frameworks have been proposed in the literature for the evolution of the slip rates: the rate-dependent and rate-independent approach. From a theoretical point of view, it is more appropriate to formulate the problem as rate-independent, especially at low temperature over a substantial range of strain rates, since most metals exhibit weak strain-rate dependency [10]. This rate-independent modeling approach is therefore adopted and studied in this paper. However, several difficulties are related to its numerical integration as will be explained hereafter.

Contrary to its theoretical modeling, the numerical integration of the single crystal behavior is still subject to debate and remains an active research topic, especially with the development of computational codes used to predict the mechanical response of metallic components and structures. These codes are generally based on the finite element (FE) method (e.g. [11], [12]), or some homogenization techniques (e.g. [13–16]), or even a combination of the two strategies, such as the FE^2 method (e.g. [17–19]). In many situations, the number of degrees of freedom of the problem (or the size of structure) is rather large, which leads to a computationally burdensome task, because it requires a great deal of CPU time and memory space. For this reason, it is still of substantial scientific and technical interest to develop robust, efficient and accurate numerical schemes and algorithms to integrate the constitutive equations of the rate-independent theory of crystal plasticity. The purpose of this paper is to propose, after a comparison and an extensive discussion of the different choices, such a scheme.

The constitutive equations of the rate-independent theory of crystal plasticity are incrementally integrated over a typical time increment $I^{\Delta}=[t_0,\ t_0+\Delta t]$. The main tedious task of the numerical integration is to split the increment of the total deformation into an elastic and plastic part. As the plastic deformation is solely due to the slip on the crystallographic systems, the problem of the decomposition of the total strain increment is reduced to the determination of the slip rates (or the slip increments) of the different slip systems. The set of governing equations used to compute the slip rates is known to be generally strongly non-linear, and this non-linearity has two distinct sources. The first comes from the material behavior and is related to the expression of the hardening law (when hardening

is considered). Indeed, the hardening laws used in the literature generally assume a complex, non-linear evolution for the rates of the various critical shear stresses as a function of slip rates. The second source of non-linearity is geometrical and is related to the evolution of the crystal lattice rotation (within the finite strain modeling framework).

In the literature, there exists two main classes of algorithms to integrate this set of non-linear equations:

- The first class, which is the most popular, includes the usual return-mapping algorithm for the integration of elastic–plastic constitutive equations, see, e.g., [11], [12], [20]–[22]. The idea behind this class of algorithms is borrowed from the numerical integration of single surface phenomenological plasticity models. It is based on the elastic predictor–plastic corrector scheme. In this case, an elastic trial stress is computed and the corresponding trial resolved shear stresses are determined by projecting the trial stress on the orientation tensor of the different slip systems. A system is said to be potentially active if its trial resolved shear stress is strictly superior to its initial critical shear stress (at t_0). Thus, the set of potentially active slip systems is determined at $t_0+\Delta t$. This set is assumed to be unchanged over the entire time step $[t_0, t_0+\Delta t]$. Accordingly, the return-mapping algorithm is not able to account for intermediate activation and deactivation of slip systems during the time increment. To compute the slip increments of the potentially active slip systems, the discrete Kuhn–Tucker loading condition, resulting from the Schmid criterion [23], is obviously used. This condition is defined for each potentially active slip system by two inequalities and an equality constraint as follows: the slip increment for each slip system is superior or equal to zero, the difference between the critical shear stress and the resolved shear stress is superior or equal to zero, and the product of these two positive quantities is equal to zero. This states that a slip system is active only if its slip increment is strictly positive and the difference between its resolved shear stress and its critical shear stress is equal to zero. Otherwise the system is inactive. Mathematically speaking, this Kuhn–Tucker condition may be considered as a non-smooth complementarity problem and its resolution requires careful attention. To determine the set of active slip systems among the potentially active ones, several search strategies have been employed in the literature. These search strategies are carried out iteratively and, at each iteration, a subset of the set of the potentially active slip systems is selected to be the set of active slip systems. The slip rate of each presumed active slip system is computed by enforcing the equality between its critical shear stress and its resolved shear stress. For the other slip systems, belonging to the set of the potentially active slip systems, their slip rates are assumed to be equal to zero. After this step, the Kuhn–Tucker condition is checked for all the potentially active slip systems. If at least one constraint of this condition is violated, then the assumed set is not an effective set of active slip systems and another set is chosen. Several techniques have been used in the literature to select the set of active slip systems for the next search iteration, such as the intuitive combinatorial strategy developed by Ben Bettaieb et al. [24]

and other less intuitive strategies, see e.g., [11−13]. When material and/or geometrical non-linearities are considered in the modeling of the single crystal behavior, the computation of the slip increments for a given set of active slip systems requires the solution of a system of non-linear differential equations. To solve this system, two main integration schemes can be used. The first is explicit and is generally based on a forward Euler scheme. In this case, the critical shear stresses and the crystal lattice rotation are held fixed over the time increment and are equal to their values at t_0. With this choice, the mathematical system becomes linear and can be solved directly, e.g., [11]. However, the solution is not necessarily unique. In case of non-uniqueness of the solution, some rules are used to determine an optimal solution, such as the pseudo-inversion technique, e.g. [11], or the perturbation technique [12]. Due to its relative simplicity, the use of this explicit scheme allows reducing the CPU time required for each time increment, on the one hand, but, on the other hand, requires the use of a large number of very small strain increments to avoid numerical instabilities and inaccuracies. The second integration scheme used to compute the slip increments is implicit and is based on the backward Euler scheme. In this case, the critical shear stresses and the crystal lattice rotation are evaluated at $t_0+\Delta t$. The resulting system of non-linear differential equations is solved by traditional iterative methods, such as the Newton–Raphson technique [12], [21], [22] or the fixed point method [24]. Contrary to the explicit scheme, the implicit one permits the use of larger time increments in order to reduce the computation time. It must be noted that the non-uniqueness issue may also be encountered in the application of the implicit algorithm. When the rate-independent single crystal constitutive equations are formulated under the small strain assumption, namely without evolution of the crystal lattice rotation and without non-linear hardening (i.e., perfect plasticity or linear hardening), the explicit and the implicit schemes become obviously equivalent. Generally, the use of implicit return-mapping algorithms is rather time consuming. Indeed, such algorithms are based on two separate nested loops: the first loop is used to search the set of active slip systems and the second is employed to compute the slip increments of the active slip systems. Hence, this approach is quite computationally expensive; especially when the number of potentially active slip systems is much larger than the number of active slip systems and/or when complex loading paths are involved (abrupt change in the loading path, elastic unloading...). In order to decrease the CPU time and thus to have an efficient integration scheme, a novel implicit return-mapping algorithm is proposed in the current paper. It is based on the replacement of the two inequalities and equality constraint of the Kuhn–Tucker conditions by a system of equations involving the so-called Fischer–Burmeister complementarity function [25], [26]. In this latter case, the set of active slip systems and the corresponding slip increments are determined iteratively by solving this system, which only requires a single loop. This choice of semi-smooth form of Kuhn–Tucker condition, defined by the Fischer–Burmeister complementarity functions, leads to an alternative

method that is more efficient and robust for determining the active slip systems and their slip rates. Although this strategy is very popular in the field of mathematical optimization [27-29], and is also applied to some problems of solid mechanics, such as contact mechanics, it is unfortunately not sufficiently used to solve the Kuhn–Tucker condition of single crystals. In fact, only few works formulated in the small strain framework can be found in this context [21], [30]. Consequently, the development of this novel algorithm is one of the major contributions of this paper.

- The second class of integration schemes may be built following the so-called ultimate algorithm initially introduced by Borja and Wren [31] and subsequently followed by [13], [24], [32], [33]. In this class, a slip system is considered to be potentially active if its resolved shear stress is equal to its critical shear stress at t_0. Accordingly, the set of potentially active slip systems is evaluated at t_0 and the trial phase is no longer required. Another important difference is that, in this case, the time increment may be divided into several sub-increments with the end of each sub-increment determined by the change in the set of potentially active slip systems (through addition or suppression of slip systems). Contrary to the return-mapping class, the ultimate algorithm provides not only the set of active slip systems at $t_0+\Delta t$, but also their sequence of activation and deactivation over the time increment. This choice allows then accounting for possible changes in the slip activity over $[t_0, t_0+\Delta t]$. For the determination of the set of active slip systems, the iterative search strategies used in the return-mapping algorithms can also be applied to the ultimate algorithms. For example, Knockaert et al. [13] adopted a strategy very similar to that used in [11], [24], [33]. As will be demonstrated later, and unlike the return-mapping class of algorithms, the development of a fully implicit ultimate scheme is conceptually difficult. Only explicit ultimate algorithms have been developed in the literature, see e.g., [13], [31]. In the same way as before, the problem of non-uniqueness of the slip rates for a given set of active slip systems may also be encountered in the case of an ultimate algorithm, and it can be circumvented by using, for example, the pseudo-inversion technique as in [13], or the perturbation technique as in [24], [33]. The development of an explicit/implicit version of the ultimate algorithm class is the second major contribution of this paper. This algorithm is optimal in the determination of the active slip systems and in the computation of the corresponding slip increments.

Moreover, to the authors' best knowledge, the comparison between the two classes of algorithms has not been attempted in the literature. Therefore, the third important objective of this paper is to compare, through several numerical simulations, the accuracy and the efficiency of the two novel algorithms (the return-mapping algorithm and the ultimate one). As will be shown in the sequel, it turns out that the ultimate algorithm is substantially more accurate and more efficient than the return-mapping one. It is also noteworthy that, in contrast to previous literature works, which have mostly dealt with the numerical integration of FCC single crystal constitutive equations, the present work specifically focuses

on BCC crystal structures. From a fundamental perspective, the difference in the numerical treatment of the two crystal structures is not significant. However, in practice, the higher number of slip systems in BCC single crystals, as compared to FCC single crystals, introduces additional difficulties, which thus leads to a more challenging problem.

The paper content is structured in the following way:

- Section 2 outlines the constitutive equations of single crystal plasticity in the framework of finite strain rate-independent theory, by adopting both Eulerian and Lagrangian formulations.

- Section 3 details the algorithmic developments both for the return-mapping and the ultimate algorithm in an incremental formulation. The derivation of a tangent modulus, consistent with the ultimate integration scheme, is given in the end of Section 3.

- The accuracy of the numerical predictions as well as the efficiency of the developed algorithms, in terms of computational cost, are discussed and compared in Section 4 on the basis of simulation results at the single crystal scale.

- The superiority of the ultimate algorithm compared to the return-mapping one, in terms of required computational time, is further highlighted in Section 5, where several numerical tests are carried out on polycrystalline aggregates.

- The consistent tangent modulus derived in Section 3.3 is validated in Section 6 through two numerical tests.

Notations, conventions and abbreviations

The derivations presented in this paper are carried out using classical conventions. Note that the assorted notations can be combined, while additional notations will be clarified as needed following related equations.

Vector and tensor variables are designated by bold letters and symbols.

Scalar variables and parameters are referred to by thin letters and symbols.

Einstein's convention of summation over repeated indices is adopted. The range of the free (resp. dummy) index is indicated before (resp. after) the corresponding equation.

$\dot{\bullet}$	time derivative of $\bullet$
$\bullet^{-1}$	inverse of $\bullet$
$\bullet.$	inner product
$\bullet \wedge \bullet$	cross product
$\bullet:$	double contraction product
$\bullet \otimes$	tensor product
$\bullet^{T}$	transpose of tensor $\bullet$
$\mathrm{tr}(\bullet)$	trace of tensor $\bullet$

R-M and UL will be used, as needed, to specify the return-mapping and ultimate algorithm, respectively.

In some figures of Sections 4, 5 and 6, for the sake of clarity and to avoid clutter, the dots associated with dotted curves are not represented for all time increments (typically only one dot is reproduced at each 5 x 10^{-3} s).

2. Constitutive equations

2.1. Decomposition of the elastic–plastic deformation

The classical treatment of finite deformation plasticity may be traced back to several earlier works, e.g., [1], [3], [6], [7]. It is based on the assumption of the existence of an infinite number of intermediate configurations, also called elastically relaxed configurations, obtained by elastic unloading to a stress free state. Here, only the main theoretical lines are recalled, except in some cases where equations that are essential to subsequent analyses are provided for completeness.

As a starting point, the total deformation gradient $\mathbf{f}$ is taken to be multiplicatively decomposed into an elastic part $\mathbf{f}_e$ and a plastic part $\mathbf{f}_p$

$$\mathbf{f} = \mathbf{f}_e . \mathbf{f}_p . \tag{1}$$

The elastic part $\mathbf{f}_e$ can itself be multiplicatively decomposed into a stretching tensor $\mathbf{v}_e$ and a rotation tensor $\mathbf{r}$

$$\mathbf{f}_e = \mathbf{v} . \mathbf{r} . \tag{2}$$

Rotation $\mathbf{r}$ defines the orientation of the coordinate system related to the intermediate configuration relative to the current one. It can be expressed in terms of the Euler angles $\varphi_1, \varphi_2, \varphi_3$ as

$$\mathbf{r} = \begin{bmatrix} \cos\varphi_1\cos\varphi_2\cos\varphi_3 - \sin\varphi_1\sin\varphi_3 & \cos\varphi_2\cos\varphi_3\sin\varphi_1 & \cos\varphi_1\sin\varphi_3 & -\cos\varphi_3\sin\varphi_2 \\ -\cos\varphi_3\sin\varphi_1 - \cos\varphi_1\cos\varphi_2\sin\varphi_3 & \cos\varphi_1\cos\varphi_3 - \cos\varphi_2\sin\varphi_1\sin\varphi_3 & \sin\varphi_2\sin\varphi_3 & \\ \cos\varphi_1\sin\varphi_2 & \sin\varphi_1\sin\varphi_2 & \cos\varphi_2 & \end{bmatrix}. \quad (3)$$

The Eulerian velocity gradient $\mathbf{g}$, expressed in the current configuration, is given by the following formula:

$$\begin{aligned} \mathbf{g} &= \dot{\mathbf{f}} . \mathbf{f}^{-1} = \dot{\mathbf{f}}_e . \mathbf{f}_e^{-1} + \mathbf{f}_e . \dot{\mathbf{f}}_p . \mathbf{f}_p^{-1} . \mathbf{f}_e^{-1} \\ &= \dot{\mathbf{v}}_e . \mathbf{v}_e^{-1} + \mathbf{v}_e . \dot{\mathbf{r}} . \mathbf{r}^T . \mathbf{v}_e^{-1} + \mathbf{v}_e . \mathbf{r} . \dot{\mathbf{f}}_p . \mathbf{f}_p^{-1} . \mathbf{r}^T . \mathbf{v}_e^{-1} \end{aligned} \quad (4)$$

As is the case for most metallic materials, the elastic deformation is often assumed to be very small compared to unity. Accordingly, the stretching tensor $\mathbf{v}_e$ is very close to the second-order identity tensor

$$\mathbf{v}_e \approx \mathbf{1} . \quad (5)$$

Combining Eqs. (4) and (5), we obtain

$$\mathbf{g} = \dot{\mathbf{v}}_e + \dot{\mathbf{r}} . \mathbf{r}^T + \mathbf{r} . \dot{\mathbf{f}}_p . \mathbf{f}_p^{-1} . \mathbf{r}^T . \quad (6)$$

The symmetric and anti-symmetric parts of $\mathbf{g}$, denoted as $\mathbf{d}$ and $\mathbf{w}$, respectively, are defined by

$$\mathbf{d} = \frac{1}{2}(\mathbf{g} + \mathbf{g}^T) = \mathbf{d}_e + \mathbf{d}_p \quad ; \quad \mathbf{w} = \frac{1}{2}(\mathbf{g} - \mathbf{g}^T) = \mathbf{w}_e + \mathbf{w}_p \quad (7)$$

where the elastic and plastic parts, $\mathbf{d}_e$ and $\mathbf{d}_p$, of the strain rate tensor $\mathbf{d}$, as well as the elastic and plastic parts, $\mathbf{w}_e$ and $\mathbf{w}_p$, of the rotation rate tensor $\mathbf{w}$ are given by

$$\begin{aligned} \mathbf{d}_e &= \dot{\mathbf{v}}_e \quad ; \quad \mathbf{d}_p = \frac{1}{2}\mathbf{r} . (\dot{\mathbf{f}}_p . \mathbf{f}_p^{-1} + (\dot{\mathbf{f}}_p . \mathbf{f}_p{}^1)^T) . \mathbf{r}^T \\ \mathbf{w}_e &= \dot{\mathbf{r}} . \mathbf{r}^T \quad ; \quad \mathbf{w}_p = \frac{1}{2}\mathbf{r} . (\dot{\mathbf{f}}_p . \mathbf{f}_p^{-1} - (\dot{\mathbf{f}}_p . \mathbf{f}_p{}^1)^T) . \mathbf{r}^T \end{aligned} \quad (8)$$

2.2. Discrete kinematics of single crystal

We recall that the single crystal plastic strain is assumed here to be solely due to the slip on the crystallographic slip systems. Each slip system α is defined, in the deformed configuration, by two orthogonal vectors $\vec{\mathbf{m}}_\alpha$ and $\vec{\mathbf{n}}_\alpha$ representing the slip direction and the normal to the slip plane, respectively. Although this paper specifically focuses on BCC single crystals, we shall assume that each crystal has in general a total of N_s slip systems. N_s is equal to 24 for BCC crystallographic structures. Therefore, the integer α ranges then from 1 to N_s. The symmetric (resp. anti-symmetric) part of the tensor product $\mathbf{M}_\alpha = \vec{\mathbf{m}}_\alpha \otimes \vec{\mathbf{n}}_\alpha$ is denoted by $\mathbf{R}_\alpha$ (resp. $\mathbf{S}_\alpha$) and is called the Schmid tensor associated with the slip system α. The rotation $\mathbf{r}$ between the current configuration and the intermediate one is chosen such that the tensor product $\vec{\mathbf{m}}_\alpha \otimes \vec{\mathbf{n}}_\alpha$ of each slip system α is equal to $\vec{\mathbf{m}} \otimes \vec{\mathbf{n}}_\alpha^0$ when

expressed in the intermediate configuration. With this choice, the slip direction $\vec{\mathbf{m}}_\alpha$ and the normal to the slip plane $\vec{\mathbf{n}}_\alpha$ in the deformed configuration are given by the following relations (Figure 1):

$$\vec{\mathbf{m}}_\alpha = \mathbf{f}_e \,.\vec{\mathbf{m}}_\alpha^0 \quad ; \quad \vec{\mathbf{n}}_\alpha = \vec{\mathbf{n}}_\alpha^0 \,.\mathbf{f}_e^{-1}. \tag{9}$$

Using assumption (5) and the kinematic relations (9), the tensor $\mathbf{M}_\alpha^0$, which is equal to $\vec{\mathbf{m}} \otimes \vec{\mathbf{n}}_\alpha^0$, can be related to $\mathbf{M}_\alpha$ by the following relation:

$$\mathbf{M}_\alpha^0 = \mathbf{r}^T.\mathbf{M}_\alpha.\mathbf{r}\,. \tag{10}$$

The detailed numbering of $\vec{\mathbf{m}}_\alpha^0$ and $\vec{\mathbf{n}}_\alpha^0$ for BCC single crystals is given in Appendix A.

As the plastic deformation is solely due to the slip on the crystallographic systems, the plastic part $\mathbf{r}.\dot{\mathbf{f}}\,.\mathbf{f}_p^{-1}.\mathbf{r}$ of the velocity gradient (see Eq. (6)) can be written as

$$\mathbf{r}.\dot{\mathbf{f}}\,.\mathbf{f}_p^{-1}.\mathbf{r}^T = \dot{\gamma}_\beta^* \mathbf{M}_\beta \quad ; \quad \beta \quad 1,...,N_s\,, \tag{11}$$

where $\dot{\gamma}_\beta^*$ is the algebraic value of the slip rate of system β. For practical reasons, and for handling only positive values of slip rates, it is more convenient to split each slip system into two opposite oriented slip systems ($\vec{\mathbf{m}}_\beta$, $\vec{\mathbf{n}}_\beta$) and ($-\vec{\mathbf{m}}_\beta$, $\vec{\mathbf{n}}_\beta$). With this new definition, Eq. (11) becomes

$$\mathbf{r}.\dot{\mathbf{f}}\,.\mathbf{f}_p^{-1}.\mathbf{r} \;= \dot{\gamma}_\beta \mathbf{M}_\beta \quad \text{with} \quad \dot{\gamma}_\beta \geq 0 \quad ; \quad \beta = 1,...,2N_s. \tag{12}$$

From Eq. (12), the plastic strain rate $\mathbf{d}_p$ and plastic spin $\mathbf{w}_p$ can be written in terms of the Schmid tensors $\mathbf{R}_\beta$ and $\mathbf{S}_\beta$ as follows:

$$\mathbf{d}_p = \dot{\gamma}_\beta \mathbf{R}_\beta \quad ; \quad \mathbf{w}_p = \dot{\gamma}_\beta \mathbf{S}_\beta \quad ; \quad \beta = 1,...,2N_s\,. \tag{13}$$

The rotation tensor $\mathbf{r}$, which describes the crystallographic orientation of the single crystal and its evolution, is defined by the following relation:

$$\dot{\mathbf{r}}.\mathbf{r}^T = \mathbf{w} - \mathbf{w}_p\,. \tag{14}$$

As in several earlier works, e.g. [24], [33], [34], Eqs. (6), (8), (11) and (13) are expressed in the crystal lattice frame defined by the rotation $\mathbf{r}$ in order to satisfy the objectivity principle [35]. For the sake of clarity, the tensors and vectors evaluated in this frame are denoted by an underbar notation. In this frame, the velocity gradient is expressed as follows:

$$\underline{\mathbf{g}} = \mathbf{r}^T.\mathbf{g}.\mathbf{r} = \underline{\mathbf{d}} + \underline{\mathbf{w}}\,, \tag{15}$$

where $\underline{\mathbf{d}}$ and $\underline{\mathbf{w}}$ are defined by the following relations:

$$\underline{\mathbf{d}} = \underline{\mathbf{d}}_e + \underline{\mathbf{d}}_p \quad , \quad \underline{\mathbf{w}} = \mathbf{r}^T.\dot{\mathbf{r}} + \underline{\mathbf{w}}_p \quad , \quad \underline{\mathbf{d}}_p + \underline{\mathbf{w}}_p = \dot{\gamma}_\beta \underline{\mathbf{M}}_\beta \quad ; \quad \beta = 1,...,2N_s\,. \tag{16}$$

The tensor $\underline{\mathbf{M}}_\beta$ can be determined from $\mathbf{M}_\beta$ by:

$$\underline{\mathbf{M}}_\beta = \mathbf{r}^T . \mathbf{M}_\beta . \mathbf{r} \,. \tag{17}$$

By comparing Eqs. (10) and (17), we can easily conclude that the tensor product $\underline{\mathbf{M}}_\beta$ is constant and equal to $\mathbf{M}_\beta^0$.

2.3. Elastic behavior law

Recognizing that the deformations considered in the present paper are dominated by plastic slip, and that the elastic deformations are small in comparison to those induced by plastic slip, we consider the elasticity as being linear and isotropic in the constitutive equations for simplicity. In a rate form based on Eulerian tensors, the elastic part of the behavior law is classically written as

$$\dot{\underline{\boldsymbol{\sigma}}} = \boldsymbol{\mathcal{L}}_e : \underline{\mathbf{d}}_e = 2\mu\, \underline{\mathbf{d}}_e + \lambda\, \mathrm{tr}(\underline{\mathbf{d}}_e)\mathbf{1} \,. \tag{18}$$

Here, $\boldsymbol{\mathcal{L}}_e$ is the fourth-order elasticity tensor, $\dot{\underline{\boldsymbol{\sigma}}}$ is the Cauchy stress rate tensor, and λ and μ are the Lamé coefficients.

In total form based on Lagrangian tensors, Eq. (18) can be rewritten in the following form:

$$\boldsymbol{\mathcal{L}} \; : \mathbf{e}_e \,, \tag{19}$$

where $\mathbf{T}$ and $\mathbf{e}_e$ are the symmetric second Piola–Kirchhoff stress tensor relative to the intermediate configuration and the elastic Green–Lagrange deformation tensor, respectively, see e.g., [11]. They are defined by the following relations:

$$\mathbf{T} = [\det(\mathbf{f}_e)]\, \mathbf{f}_e^{-1} . \boldsymbol{\sigma} . \mathbf{f}_e \quad ; \quad \mathbf{e}_e = (1/2)(\mathbf{c}_e - \mathbf{1}) \,, \tag{20}$$

with $\mathbf{c}_e$ being the elastic right Cauchy–Green deformation tensor given by

$$\mathbf{c}_e = \mathbf{f}_e^T . \mathbf{f}_e \,. \tag{21}$$

2.4. Plastic behavior law

The plastic part of the behavior law is defined by the Schmid law [23]. This relation states that slip may occur only when the resolved shear stress τ_α on a slip system α becomes equal to a critical value τ_α^c

$$\forall \alpha = 1,...,2\,\mathrm{N_s} : \begin{cases} \tau_\alpha < \tau_\alpha^c \Rightarrow \dot{\gamma}_\alpha = 0 \\ \tau_\alpha = \tau_\alpha^c \Rightarrow \dot{\gamma}_\alpha \geq 0 \end{cases} \tag{22}$$

where the resolved shear stress τ_α, acting on a given slip system α, is defined as the projection of the Cauchy stress on the Schmid tensor $\mathbf{M}_\alpha$ associated with that slip system

$$\tau_\alpha = \boldsymbol{\sigma} : \mathbf{M}_\alpha = (\boldsymbol{\sigma} . \vec{\mathbf{n}}_\alpha) . \vec{\mathbf{m}}_\alpha \,, \tag{23}$$

and the evolution of the critical shear stress τ_α^c is given by the hardening law defined in Section 2.5.

By using Eqs. (9), $(20)_{(1)}$, (21) and (23), τ_α can be expressed as follows:

$$\tau_\alpha = \frac{\mathbf{1}}{\det(\mathbf{f}_e)} \left(\mathbf{c}_e.\mathbf{T} \right). \vec{\mathbf{n}}_\alpha^0 \Big). \vec{\mathbf{m}}_\alpha \,, \tag{24}$$

where $\mathbf{c}_e.\mathbf{T}$ is the inner product between $\mathbf{c}_e$ and $\mathbf{T}$.

For infinitesimal elastic stretches ($\det(\mathbf{f}_e) \approx 1$ and $\mathbf{c}_e \approx \mathbf{1}$), the resolved shear stress τ_α can be approximated by

$$\tau_\alpha = (\mathbf{T}. \vec{\mathbf{n}}_\alpha^0). \vec{\mathbf{m}}_\alpha^0 = \mathbf{T} : \mathbf{M}_\alpha^0 \quad \text{with} \quad \mathbf{M}_\alpha^0 = \vec{\mathbf{m}}_\alpha^0 \otimes \vec{\mathbf{n}}_\alpha^0 \,. \tag{25}$$

Considering Eq. (22), the yield function $f_\alpha = (\tau_\alpha^c - \tau_\alpha)$ corresponding to slip system α then reads

$$\forall\, \alpha = 1,...,2\,\mathrm{N_s} : \; f_\alpha = (\tau_\alpha^c - \tau_\alpha) \geq 0 \quad ; \quad \dot{\gamma}_\alpha \geq 0 \quad ; \quad \dot{\gamma}_\alpha f_\alpha = 0 \,. \tag{26}$$

Eq. (26) may be viewed as a non-smooth complementarity problem and can therefore be replaced by an equivalent system involving the semi-smooth Fischer–Burmeister function [25], [26]

$$\forall\, \alpha = 1,...,2\,\mathrm{N_s} : \; \varphi_\alpha = \sqrt{(f_\alpha)^{} + (\dot{\gamma}_\alpha)^2} - (f_\alpha + \dot{\gamma}_\alpha) = 0 \,. \tag{27}$$

2.5. Hardening description

The hardening law describes the evolution of the critical shear stress τ_α^c during the loading history. The literature provides numerous models with various expressions for the hardening law, which are generally motivated by the crystal physical microstructure and are dependent, via a hardening modulus $\mathbf{h}$, on the slip rates of the different slip systems. These hardening laws have traditionally a rate form and can be expressed as

$$\forall\, \alpha = 1,..., \mathrm{N_s} : \; \dot{\tau}_{\alpha+\mathrm{N_s}}^c = \dot{\tau}_\alpha = \mathrm{h}_{\alpha\beta} (\dot{\gamma}_\beta + \dot{\gamma}_{\beta+\mathrm{N}}) \quad ; \quad \beta = 1,..., \mathrm{N_s} \,. \tag{28}$$

The hardening modulus $\mathbf{h}$ has been defined in the literature by various forms: diagonal, isotropic, anisotropic, symmetric or asymmetric. An overview of the different hardening laws adopted in the literature is given in [7], [36]. In the present paper, the integration scheme is general enough to be independent of the choice of the hardening law.

3. Numerical integration

We now elaborate on the algorithmic treatment of the above-described constitutive equations. In this aim, two new algorithms are developed. The first is based on the return-mapping class of algorithms, while the second relies on the so-called ultimate algorithm. For convenience, the return-mapping algorithm is formulated within a Lagrangian hyperelastic approach, whereas the ultimate algorithm is developed within an Eulerian hypoelastic framework.

The time integration of the constitutive equations, described by Eqs. (1)–(28), proceeds by discretizing the deformation history in time and numerically integrating these equations over each time increment $I^{\Delta}=[t_0, t_0+\Delta t]$. For this purpose, we assume that the mechanical quantities: $\mathbf{f}_e$, $\mathbf{f}_p$, $\boldsymbol{\sigma}$, $\mathbf{r}$, γ_α and τ_α^c (for $\alpha = 1,...,2N_s$) are known at t_0. The Eulerian velocity gradient $\mathbf{g}$ is assumed to be constant and known over the time increment I^{Δ}. The aim of both incremental algorithms is to compute $\boldsymbol{\sigma}$, $\mathbf{r}$, γ_α and τ_α^c (for $\alpha = 1,...,2N_s$) at $t_0+\Delta t$. In what follows, a variable $\mathbf{x}$ evaluated at t_0 (resp. $t_0+\Delta t$) is denoted by $\mathbf{x}(t_0)$ (resp. $\mathbf{x}(t_0+\Delta t)$).

3.1. Return-mapping algorithm

This algorithm is defined by three main steps, which will be detailed in Sections 3.1.1, 3.1.2 and 3.1.3.

3.1.1. Determination of the set of potentially active slip systems

A direct time integration of Eq. (4) over I^{Δ} gives the following update equation:

$$\mathbf{f}(t + \Delta t) = e^{\Delta t \mathbf{g}} . \mathbf{f}(t_0) . \tag{29}$$

When $\Delta t\,\mathbf{g}$ is very small ($\|\Delta t\,\mathbf{g}\| << 1$), Eq. (29) can be approximated by

$$\mathbf{f}(t_0 + \Delta t) \approx (\mathbf{1} + \Delta t\,\mathbf{g}) . \mathbf{f}(t) . \tag{30}$$

Despite the accuracy of the kinematic approximation (30) (especially for small time increments), the exact expression (29) is used in this algorithm. Note that approximation (30) is widely used in the literature (see. e.g. [11]).

The trial elastic deformation gradient $\mathbf{f}^{tr}(t_0 + \Delta t)$ is computed as

$$\mathbf{f}_e^{tr}(t_0 + \Delta t) = \mathbf{f}(t_0 + \Delta t) . \mathbf{f}_p^{-1}(t_0), \tag{31}$$

in terms of which we express the trial elastic Green–Lagrange strain tensor by

$$\mathbf{e}_e^{tr}(t_0 + \Delta t) = (1/2)(\mathbf{c}_e^{tr}(t_0 + \Delta t) - \mathbf{1}) \quad ; \quad \mathbf{c}_e^{tr}(t_0 + \Delta t) = \mathbf{f}_e^{trT}(t_0 + \Delta t) . \mathbf{f}_e^{tr}(t_0 + \Delta t) . \tag{32}$$

The trial second Piola–Kirchhoff stress tensor associated with the trial elastic strain $\mathbf{e}_e^{tr}$ is defined by the following relation:

$$\mathbf{T}^{tr}(t_0 + \Delta t) = \boldsymbol{\mathcal{L}}_e : {}^{tr}_{e}(t_0 + \Delta t) . \tag{33}$$

And the trial resolved shear stresses are given by (see Eq. (25))

$$\forall \alpha = 1,...,2N_s : \quad \tau_\alpha^{tr}(t_0 + \Delta t) = \mathbf{T}^{tr}(t_0 + \Delta t) : \mathbf{M}_\alpha^0 . \tag{34}$$

A slip system α is potentially active if it satisfies the following inequality:

$$\tau_\alpha^{tr}(t_0 + \Delta t) - \tau_\alpha^c(t_0) > 0 . \tag{35}$$

Accordingly, the set of potentially active slip systems $\mathcal{P}$ is given by

$$\mathcal{P} = \left\{ \alpha = 1,...,2N_s ; \quad \tau_\alpha^{tr}(t_0+\Delta t) - \tau_\alpha^{c}(t_0) > 0 \right\}. \tag{36}$$

At this stage, two possibilities may occur [37]:

- If $\mathcal{P} = \varnothing$, then the mechanical behavior is purely elastic over I^Δ (i.e., there is no plastic flow) and the slip rates of all the slip systems are equal to 0. The algorithm described in Section 3.1.2 is then skipped.
- If $\mathcal{P} \neq \varnothing$, then the mechanical behavior is elastic–plastic over I^Δ and there is at least one system that is active among the set of potentially active slip systems. To compute the slip rates of the potentially active slip systems, the algorithm detailed in Section 3.1.2 is followed.

3.1.2. Determination of the slip increments of the potentially active slip systems

Eq. (12) can be written in the frame of the crystal lattice

$$\dot{\mathbf{f}}_p . \mathbf{f}^{-1} = \dot{\gamma}_\beta \underline{\mathbf{M}}_\beta = \dot{\gamma}_\beta \mathbf{M}_\beta \quad ; \quad \beta = 1,...,2N_s . \tag{37}$$

If a slip system α is not potentially active, then its slip rate $\dot{\gamma}_\alpha$ is obviously equal to 0. Hence, Eq. (37) can be reduced to the following form:

$$\dot{\mathbf{f}} . \mathbf{f}_p^{-1} = \dot{\gamma}_\beta \mathbf{M} \quad ; \quad \beta \in \mathcal{P} . \tag{38}$$

A backward integration of Eq. (38) yields

$$\mathbf{f}_p(t_0+\Delta t) - \mathbf{f}_p(t_0) = \Delta\gamma_\beta \mathbf{M}_\beta^0 . \mathbf{f}_p(t_0+\Delta t) \quad ; \quad \beta \in \mathcal{P}, \tag{39}$$

which can be written equivalently as

$$\mathbf{f}_p(t_0) . \mathbf{f}_p^{-1}(t_0+\Delta t) = (\mathbf{1} - \Delta\gamma_\beta \mathbf{M}) \quad ; \quad \beta \in \mathcal{P} . \tag{40}$$

On the other hand, the elastic Green–Lagrange strain tensor can be written as

$$\mathbf{e}_e(t_0+\Delta t) = (1/2)(\mathbf{c}_e(t_0+\Delta t) - \mathbf{1}) . \tag{41}$$

The deformation gradient $\mathbf{f}(t_0+\Delta t)$ can be written in the following two equivalent forms:

$$\mathbf{f}(t_0+\Delta t) = \mathbf{f}_e(t_0+\Delta t) . \mathbf{f}_p(t_0+\Delta t) = \mathbf{f}_e^{tr}(t_0+\Delta t) . \mathbf{f}_p(t_0) , \tag{42}$$

which implies that

$$\mathbf{f}_e(t_0+\Delta t) = \mathbf{f}_e^{tr}(t_0+\Delta t) . \mathbf{f}_p(t_0) . \mathbf{f}_p^{-1}(t_0+\Delta t) . \tag{43}$$

By using Eq. (43), expression (41) of the elastic Green–Lagrange strain tensor becomes

$$\begin{aligned} \mathbf{e}_e(t_0+\Delta t) &= (1/2)\left[\left(\mathbf{f}_p(t_0) . \mathbf{f}_p^{-1}(t_0+\Delta t)\right)^T . \left(\mathbf{f}_e^{trT}(t_0+\Delta t) . \mathbf{f}_e^{tr}(t_0+\Delta t)\right) . \left(\mathbf{f}_p(t_0) . \mathbf{f}_p^{-1}(t_0+\Delta t)\right) - \mathbf{1} \right] \\ &= (1/2)\left[\left(\mathbf{f}_p(t_0) . \mathbf{f}_p^{-1}(t_0+\Delta t)\right)^T . \left(\mathbf{f}_e^{trT}(t_0+\Delta t) . \mathbf{f}_e^{tr}(t_0+\Delta t) - \mathbf{1}\right) . \left(\mathbf{f}_p(t_0) . \mathbf{f}_p^{-1}(t_0+\Delta t)\right) \right. \\ &\left. + (1/2)\left[\left(\mathbf{f}_p(t_0) . \mathbf{f}_p^{-1}(t_0 \quad \Delta t)\right)^T . \left(\mathbf{f}_p(t_0) . \mathbf{f}_p^{-} (t_0 \quad \Delta t)\right) - \mathbf{1} \right] . \end{aligned} \tag{44}$$

By substituting Eq. (40) in Eq. (44), $\mathbf{e}_e(t_0+\Delta t)$ can be rewritten after some lengthy but straightforward algebraic manipulations as

$$\mathbf{e}_e(t_0+\Delta t)=\mathbf{e}_e^{tr}(t_0+\Delta t)-\Delta\gamma_\beta\,\mathrm{sym}\left(\mathbf{c}_e^{tr}(t_0+\Delta t).\mathbf{M}_\beta^0\right)\quad;\quad\beta\in\mathcal{P}\,, \tag{45}$$

where sym(**x**) is the symmetric part of tensor **x**. Note that the slip increments $\Delta\gamma_\beta$ are assumed to be small so that, in the expression (45) above, their second-order terms have been disregarded.

The resolved shear stress τ_α is determined on the basis of Eqs. (19), (25) and (45)

$$\begin{aligned}\tau_\alpha(t_0+\Delta t)&=\mathbf{M}_\alpha^0:\boldsymbol{\mathcal{L}}_e:\left[\mathbf{e}_e^{tr}(t_0+\Delta t)-\Delta\gamma_\beta\,\mathrm{sym}\left(\mathbf{c}_e^{tr}(t_0+\Delta t).\mathbf{M}_\beta^0\right)\right.\\&=\mathbf{M}_\alpha^0:\boldsymbol{\mathcal{L}}_e:\mathbf{e}_e^{tr}(t_0+\Delta t)-\Delta\gamma_\beta\,\mathbf{M}_\alpha^0:\boldsymbol{\mathcal{L}}_e:\mathrm{sym}\left(\mathbf{c}_e^{tr}(t_0+\Delta t).\mathbf{M}_\beta^0\right)\quad;\quad\beta\in\mathcal{P}\end{aligned}\,. \tag{46}$$

Finally, τ_α can be rearranged in the following simpler form:

$$\tau_\alpha(t_0+\Delta t)=\tau_\alpha^{tr}(t_0+\Delta t)-\Delta\gamma_\beta\,\mathbf{M}_\alpha^0:\boldsymbol{\mathcal{L}}_e:\mathrm{sym}\left(\mathbf{c}_e^{tr}(t_0+\Delta t).\mathbf{M}_\beta^0\right)\quad;\quad\beta\in\mathcal{P}\,, \tag{47}$$

where the trial resolved shear stresses have already been expressed through Eqs. (33) and (34).

On the other hand, via backward integration of the evolution equation (28) for the critical shear stresses, the following relations are derived:

$$\forall\,\alpha=1,...,2N_s:\quad\tau_\alpha^c(t_0+\Delta t)=\tau_\alpha(t_0)+h_{\alpha\beta}(t_0+\Delta t)\,\Delta\gamma_\beta\quad;\quad\beta\in\mathcal{P}\,. \tag{48}$$

By combining Eqs. (47) and (48), we can obtain the following expression for the yield function f_α (see Eq. (26)):

$$\forall\,\alpha\in\mathcal{P}:\quad f_\alpha=\tau_\alpha^c(t_0+\Delta t)-\tau_\alpha(t_0+\Delta t)=A_{\alpha\beta}\Delta\gamma_\beta-b_\alpha\quad;\quad\beta\in\mathcal{P}\,, \tag{49}$$

where

$$\forall\,\alpha,\beta\in\mathcal{P}:\begin{cases}A_{\alpha\beta}=h_{\alpha\beta}(t_0+\Delta t)+\mathbf{M}_\alpha^0:\boldsymbol{\mathcal{L}}_e:\mathrm{sym}\left(\mathbf{c}_e^{tr}(t_0+\Delta t).\mathbf{M}_\beta^0\right)\\ b_\alpha=\tau_\alpha^c(t_0)-\tau_\alpha^{tr}(t_0+\Delta t)\end{cases}. \tag{50}$$

The semi-smooth form (27) of the Schmid law can therefore be written as

$$\forall\,\alpha\in\mathcal{P}:\ \varphi_\alpha=\sqrt{(A_{\alpha\beta}\Delta\gamma_\beta-b_\alpha)\ +(\Delta\gamma_\alpha)^2}-(A_{\alpha\beta}\Delta\gamma_\beta-b_\alpha+\Delta\gamma_\alpha)=0\quad;\quad\beta\in\mathcal{P}\,. \tag{51}$$

The slip increments $\Delta\gamma_\beta$ (and then the set of active slip systems $\mathcal{A}$) can be determined by solving the system of non-linear equations (51) using a global Newton–Raphson method. The development of this global Newton–Raphson method is extensively detailed in Appendix B. It must be noted that, during the global Newton–Raphson iterations, no ad hoc assumptions concerning the determination of the active set are necessary, as the indices α and β in Eq. (B.3) cover all (active and inactive) slip systems. Indeed, the use of the semi-smooth function (51) permits to avoid the typical problems related to the definition and use of iterative search procedures for the set of active slip systems.

3.1.3. Update of the other variables

After convergence of the Newton–Raphson procedure, the slip increments $\Delta\gamma_\beta$ as well as the set of active slip systems are known and all other variables can be updated using the constitutive equations:

- The plastic and elastic parts of the deformation gradient are computed as follows:

$$\mathbf{f}_p(t_0+\Delta t)=(\mathbf{1}-\Delta\gamma_\beta\,\mathbf{M}_\beta)^{-1}.\mathbf{f}_p(t_0)\quad;\quad \mathbf{f}_e(t_0+\Delta t)=\mathbf{f}(t_0+\Delta t).\mathbf{f}_p^{-1}(t_0+\Delta t)\quad;\quad \beta\in\mathcal{P}. \tag{52}$$

- Then, the second Piola–Kirchhoff and Cauchy stress tensors are determined by

$$\begin{aligned}\mathbf{T}(t_0+\Delta t)&=\mathbf{T}^{tr}(t\ +\Delta t)-\sum_{\beta\in\mathcal{P}}\Delta\gamma_\beta\,\boldsymbol{\mathcal{L}}_e:\mathrm{sym}\left(\mathbf{c}_e^{tr}(t_0+\Delta t).\mathbf{M}_\beta^0\right)\\ \boldsymbol{\sigma}(t_0+\Delta t)&=\left[\det(\mathbf{f}_e(t_0+\Delta t))\right]^{-1}\mathbf{f}_e(t_0+\Delta t).\,\mathbf{T}(t_0+\Delta t).\,\mathbf{f}_e^{T}(t_0+\Delta t)\end{aligned}. \tag{53}$$

- The resolved shear stresses are computed by

$$\forall\,\alpha=1,...,2N_s:\quad \tau_\alpha(t_0+\Delta t)=\mathbf{T}(t_0+\Delta t):\mathbf{M}_\alpha^0. \tag{54}$$

- The critical shear stresses are then updated

$$\forall\,\alpha=1,...,2N_s:\quad \tau_\alpha(t_0+\Delta t)=\tau_\alpha^c(t_0)+h_{\alpha\beta}(t_0+\Delta t)\,\Delta\gamma_\beta\quad;\quad \beta\in\mathcal{P}. \tag{55}$$

The Schmid law is checked for the different slip systems $\alpha=1,...,2N_s$, i.e.

$$\tau_\alpha^c(t_0+\Delta t)=\tau_\alpha(t_0+\Delta t), \tag{56}$$

if the slip system is active, or

$$\tau_\alpha(t_0+\Delta t)<\tau_\alpha^c(t_0+\Delta t), \tag{57}$$

if it is inactive.

In fact, Eqs. (56) and (57) are obviously satisfied for the potentially active slip systems. But occasionally, one or several non-potential slip systems may violate inequality (57). In such a case, these slip systems must be added to the set of potentially active slip systems, and the procedure detailed in Sections 3.1.2 and 3.1.3 is repeated until complete satisfaction of Eqs. (56) and (57).

Finally, the lattice rotation $\mathbf{r}$ is derived by polar decomposition of the elastic part of the deformation gradient

$$\mathbf{r}(t_0+\Delta t)=\mathbf{v}_e^{-1}(t_0+\Delta t).\,\mathbf{f}_e(t_0+\Delta t)\ . \tag{58}$$

3.1.4. Some remarks on the return-mapping algorithm

- Although the stretching tensor $\mathbf{v}_e$ is assumed to be very small, Eq. $(53)_{(2)}$ is used to update the Cauchy stress tensor $\boldsymbol{\sigma}$ without any approximation (i.e., $\mathbf{f}_e(t_0+\Delta t)$ is not replaced by $\mathbf{r}(t_0+\Delta t)$ in Eq. $(53)_{(2)}$).

- The return-mapping algorithm detailed above is not able to accurately satisfy the incremental incompressibility. Indeed, an isochoric loading (det(**f**)=1) does not necessarily lead to a deviatoric form for the stress tensors **T** and $\boldsymbol{\sigma}$. This point will be further discussed in Section 4.3.
- The return-mapping algorithm developed in the previous section is based on a Lagrangian hyperelastic formulation. Accordingly, one of its advantages is that one does not have to check its incremental objectivity. Indeed, the time integration scheme naturally satisfies this important requirement.

3.2. Ultimate algorithm

For convenience, the Eulerian formulation is used in the development of the following version of the so-called ultimate algorithm. In this case, the time increment I^{Δ} is divided into several discrete sub-increments I^{δ}, over which the Schmid criterion is fulfilled at each time t. The end of each sub-increment I^{δ} corresponds to a change in the set of potentially active slip systems (through addition of new systems or suppression of existing systems).
The ultimate algorithm is defined by the following main steps:

3.2.1. Determination of the set of potentially active slip systems

The set of potentially active slip systems $\mathcal{P}$ is identified at t_0 as

$$\mathcal{P}=\{\alpha=1,...,2N\ ;\ \boldsymbol{\sigma}(t_0):\mathbf{M}_\alpha=\underline{\boldsymbol{\sigma}}(t_0):\underline{\mathbf{M}}_\alpha=\underline{\boldsymbol{\sigma}}(t_0):\mathbf{M}^0_\alpha=\tau_\alpha(t_0)\}, \tag{59}$$

where $\underline{\boldsymbol{\sigma}}(t_0)$ is equal to $\mathbf{r}^T(t_0).\boldsymbol{\sigma}(t_0).\mathbf{r}(t_0)$.

At this stage, two possibilities may occur:

- If (a) $\mathcal{P}=\varnothing$ or (b) $\mathcal{P}\neq\varnothing$ and $\mathbf{M}^0_\alpha:\boldsymbol{\mathcal{L}}_e:_(t_0)\leq 0$ for all $\alpha\in\mathcal{P}$, then the mechanical behavior is purely elastic over I^{δ} (with $\underline{\mathbf{d}}(t_0)$ being equal to $\mathbf{r}^T(t_0).\mathbf{d}.\mathbf{r}(t\ )$). In this case, the algorithm detailed in Section 3.2.2 is followed.
- If $\mathcal{P}\neq\varnothing$ and $^{0}_{\alpha}:\boldsymbol{\mathcal{L}}_e:\underline{\mathbf{d}}(t_0)>0$ for at least one system $\alpha\in\mathcal{P}$, then the mechanical behavior is elastic–plastic over I^{δ}. In this case, the algorithm detailed in Section 3.2.3 is followed.

In spite of multiple similarities between the algorithmic treatments of the two possibilities, the two cases are studied separately (Sections 3.2.2 and 3.2.3) for the sake of clarity and consistency of the presentation.

3.2.2. Purely elastic phase

The first aim of this elastic algorithm is to compute the length δt of the sub-increment I^{δ} over which the behavior remains purely elastic. The second aim is to update the different mechanical variables at $t_0 + \delta t$.

δt corresponds to the time required to reach the first facet of the yield surface. It must be comprised between 0 and Δt ($0 \le \delta t \le \Delta t$). δt can then be computed as follows:

$$\delta t = \min_{\alpha=1,\dots,2N_s} \left\{ \Delta t,\ \frac{\tau_\alpha^c(t_0) - \underline{\boldsymbol{\sigma}}(t_0) : \mathbf{M}_\alpha^0}{\dot{\tau}_\alpha(t_0)} \quad \frac{\tau_\alpha^c(t_0) - \underline{\boldsymbol{\sigma}}(t_0) : \mathbf{M}_\alpha^0}{\mathbf{M}_\alpha^0 : \mathcal{L}_e \ _(t_0)} \right\}. \tag{60}$$

Generally, the length δt of the elastic phase is very small. Therefore, the different mechanical variables can be updated by a forward scheme

$$\begin{aligned} &\underline{\boldsymbol{\sigma}}(t_0 + \delta t) = \underline{\boldsymbol{\sigma}}(t_0) + \delta t\, \dot{\underline{\boldsymbol{\sigma}}}(t_0) = \underline{\boldsymbol{\sigma}}(t_0) + \delta t\, \mathcal{L}_e : \underline{\mathbf{d}}(t_0) \\ &\mathbf{r}(t_0 + \delta t) = \mathbf{r}(t_0).\mathbf{e}^{\delta t\, \underline{\mathbf{w}}(t_0)} \quad ; \quad \boldsymbol{\sigma}(t_0 + \delta t) = \mathbf{r}(t_0 + \delta t).\, \underline{\boldsymbol{\sigma}}(t_0 + \delta t).\, \mathbf{r}^T(t_0 + \delta t) \end{aligned}. \tag{61}$$

As this phase is purely elastic, the accumulated slips γ_α and the critical shear stresses τ_α^c of the different slip systems remain constant over I^{δ} and equal to their values at t_0.

After application of this update stage, the computation must be restarted from Section 3.2.1, with a new sub-increment I^{δ}.

3.2.3. Elastic–plastic phase

3.2.3.1. Determination of the slip rates of the potentially active slip systems

For each slip system belonging to $\mathcal{P}$, the Schmid criterion should be verified at t_0 and at each instant t over I^{δ}.

$$\forall \alpha \in \mathcal{P}, \forall t \in I^{\delta} : \quad \dot{\gamma}_\alpha(t) \ge 0 \quad ; \quad (\tau_\alpha(t) - \tau_\alpha^c(t)) \le 0 \quad ; \quad \dot{\gamma}_\alpha(t)(\tau_\alpha(t) - \tau_\alpha^c(t)) = 0 . \tag{62}$$

The resolved shear stress $\tau_\alpha(t)$ and the critical shear stress $\tau_\alpha^c(t)$ can be approximated over I^{δ} by the following expression (which can be obtained, e.g., by backward integration):

$$\forall \alpha = 1,\dots,2N_s, \forall t \in I^{\delta} : \begin{cases} \tau_\alpha(t) = \tau_\alpha(t_0) + (t - t_0)\dot{\tau}_\alpha(t) \\ \tau_\alpha^c(t) = \tau_\alpha^c(t_0) + (t - t_0)\dot{\tau}_\alpha^c(t) \end{cases}. \tag{63}$$

As all the slip systems $\alpha \in \mathcal{P}$ are potentially active, $\tau_\alpha(t_0)$ is equal to $\tau^c(t_0)$. Using this, in conjunction with Eq. (63), the Schmid criterion (62) becomes equivalent to the following consistency condition:

$$\forall \alpha \in \mathcal{P}, \forall t \in I^{\delta} : \quad \dot{\gamma}_\alpha(t) \ge 0 \quad ; \quad (\dot{\tau}_\alpha(t) - \dot{\tau}_\alpha^c(t)) \le 0 \quad ; \quad \dot{\gamma}_\alpha(t)(\dot{\tau}_\alpha(t) - \dot{\tau}_\alpha^c(t)) = 0 . \tag{64}$$

Knowing that only the slip systems belonging to $\mathcal{P}$ may be active, $\dot{\tau}_\alpha(t)$ and $\dot{\tau}^c_\alpha(t)$ can be recast in the following form:

$$\forall \alpha = 1,...,2N_s, \forall t \in I^\delta : \quad \begin{cases} \dot{\tau}^c_\alpha(t) = h_{\alpha\beta}(t)\,\dot{\gamma}_\beta(t) \\ \dot{\tau}_\alpha(t) = \dot{\underline{\boldsymbol{\sigma}}} : \mathbf{M}^0_\alpha = \mathbf{M}^0_\alpha : \boldsymbol{\mathcal{L}}_e : \underline{\mathbf{d}}(t) - \dot{\gamma}_\beta(t)(\mathbf{M}^0_\alpha : \boldsymbol{\mathcal{L}}_e : \mathbf{M}^0_\beta) \end{cases} \quad ; \quad \beta \in \mathcal{P}. \quad (65)$$

After straightforward algebraic manipulations, the insertion of (65) into (64) yields

$$\forall \alpha \in \mathcal{P}, \forall t \in I^\delta : \quad \dot{\gamma}_\alpha(t) \geq 0 \quad ; \quad \dot{f}_\alpha(t) = \tilde{A}_{\alpha\beta}(t)\dot{\gamma}_\beta(t) - \tilde{b}_\alpha(t) \leq 0 \quad ; \quad \dot{\gamma}_\alpha(t)\,\dot{f}_\alpha(t) = 0 \quad ; \quad \beta \in \mathcal{P}. \quad (66)$$

The components of $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{b}}$ are given by

$$\forall \alpha,\beta \in \mathcal{P}, \forall t \in I^\delta : \quad \tilde{A}_{\alpha\beta}(t) = -\mathbf{M}^0_\alpha : \boldsymbol{\mathcal{L}}_e : \mathbf{M}^0_\beta - h_{\alpha\beta}(t) \quad ; \quad \tilde{b}_\alpha(t) = -\mathbf{M}^0_\alpha : \boldsymbol{\mathcal{L}}_e : \underline{\mathbf{d}}(t). \quad (67)$$

The non-smooth formulation of the consistency condition given by Eq. (66) is a Non-Linear Complementarity Problem (NLCP). Unlike the return-mapping algorithm, this NLCP cannot be solved by an implicit scheme over I^δ, because at the initial time t_0 of I^δ, its length δt is not a priori known. To compute the slip rates of the potentially active slip systems, a new algorithm is developed. This algorithm requires one or two phases, depending on the situation: an explicit phase (automatically achieved), followed or not by an implicit correction (performed if δt is equal to Δt). This algorithm is described by the following three steps:

- *Step 1:* In this step, the NLCP given by Eq. (66) is evaluated at t_0 as

  $$\forall \alpha \in \mathcal{P}: \quad \dot{\gamma}_\alpha(t_0) \geq 0 \quad ; \quad \dot{f}_\alpha(t_0) = \tilde{A}_{\alpha\beta}(t_0)\dot{\gamma}_\beta(t_0) - \tilde{b}_\alpha(t_0) \leq 0 \quad ; \quad \dot{\gamma}_\alpha(t_0)\,\dot{f}_\alpha(t_0) = 0 \quad ; \quad \beta \in \mathcal{P}. \quad (68)$$

  With this choice, the NLCP is transformed into a Linear Complementarity Problem (LCP), since $\tilde{\mathbf{b}}(t_0)$ and $\tilde{\mathbf{A}}(t_0)$ are known and do not depend on the value of $\dot{\boldsymbol{\gamma}}(t_0)$. This LCP can be solved by using a semi-smooth formulation similar to that given by Eq. (51). However, it turns out that this semi-smooth form is not the most optimal approach to solve Eq. (68). Consequently, as alternative, a robust and efficient iterative search strategy is proposed to detect the set of active slip systems and thus to solve this LCP. This strategy, developed in the spirit of the algorithm proposed in [11], is based on the following three sub-steps:

  - ✓ Sub-step 1.1: We start this sub-step by assuming that the active set $\mathcal{A}$ at t_0 coincides with the set of potentially active slip systems $\mathcal{P}$. The slip rates of the slip systems belonging to $\mathcal{A}$ are determined by solving the following linear algebraic equation system:

    $$\tilde{\mathbf{A}}(t_0).\,\dot{\boldsymbol{\gamma}}(t_0) - \tilde{\mathbf{b}}(t_0) = \mathbf{0} \Leftrightarrow \tilde{\mathbf{A}}(t_0).\,\dot{\boldsymbol{\gamma}}(t_0) = \tilde{\mathbf{b}}(t_0). \quad (69)$$

If for all slip systems α belonging to the set $\mathcal{A}$ (=$\mathcal{P}$) the corresponding components $\dot{\gamma}_\alpha(t_0)$, solution of (69), are strictly positive, then $\dot{\boldsymbol{\gamma}}(t_0)$ is a solution of the LCP (68). In this case, we go to Step 2; otherwise, we go to Sub-step 1.2. If the matrix $\tilde{\mathbf{A}}(t_0)$ is singular, the pseudo-inversion method is used to compute its inverse and then to solve Eq. (69). Further details about the pseudo-inversion method are given in Appendix C.

- ✓ Sub-step 1.2: If some components $\dot{\gamma}_\alpha(t_0)$, solution of (69), are negative or equal to zero, then the corresponding slip systems are inactive and, accordingly, we remove them from the set of active slip systems $\mathcal{A}$ and we resolve the algebraic system (69) for this new set $\mathcal{A}$. Sub-step 1.2 is repeated until a solution for (69) is found, with all slip rates $\dot{\gamma}_\alpha(t_0)$, $\alpha \in \mathcal{A}$, being strictly positive. The slip rates of the slip systems belonging to $\mathcal{P} \setminus \mathcal{A}$ are set to 0.
- ✓ Sub-step 1.3: In this sub-step, the consistency condition (68) is checked for all slip systems belonging to $\mathcal{P}$. The first inequality is naturally fulfilled, as a result of Sub-step 1.2. The second inequality of (68) is obviously satisfied for the slip systems belonging to the set $\mathcal{A}$. If some slip systems belonging to $\mathcal{P} \setminus \mathcal{A}$ violate this inequality, then these slip systems are added to the set of active slip systems $\mathcal{A}$ and one returns to Sub-step 1.2. Otherwise, the vector $\dot{\boldsymbol{\gamma}}(t_0)$ computed through Sub-step 1.2 is a solution to the LCP (68). The combinatorial search procedure, defined by Sub-steps 1.2 and 1.3, always allows identifying the set of active slip systems after a few iterations, as will be demonstrated in Figure 18. Moreover, it is known that the set of active slip systems $\mathcal{A}$ is included in the set of potentially active slip systems $\mathcal{P}$. Hence, if the set $\mathcal{P}$ contains *n* slip systems, the set $\mathcal{A}$ is at least one of the 2^n sub-sets of $\mathcal{P}$. Consequently, the application of a trivial combinatorial search procedure necessarily allows finding at least one set of active slip systems $\mathcal{A}$, as a solution of the consistency condition (68), among the 2^n sub-sets of $\mathcal{P}$. However, this trivial search procedure seems to be computationally expensive (for instance, for a set of 8 potentially active slip systems, 256 sub-sets must be checked). In order to quickly identify the set of active slip systems, the search procedure consisting of Sub-steps 1.2 and 1.3 is followed. The same procedure has been successfully used in several previous works in the literature (see, e.g., [11] and [13]). In the numerical code developed on the basis of this algorithm, the trivial search procedure has been incorporated in case the combinatorial algorithm presented in Sub-steps 1.2 and 1.3 fails to converge after 10 iterations. From personal experience, and on the basis of the various simulations carried out at the single crystal and polycrystal scales, it is found that the set of active slip

systems is generally determined after 3 or 4 iterations. Therefore, from a practical point of view, the recourse to a trivial search procedure has never been required so far, although the latter has been implemented in the code.

- *Step 2:* The aim of this step is to compute the length δt of the time sub-increment $I^{\delta}=[t_0,\ t_0+\delta t]$ over which the Schmid criterion is satisfied. In view of the previous steps, it can be shown that this criterion is fulfilled for the potentially active slip systems. For the other systems ($\notin \mathcal{P}$), the following condition must be checked:

$$\forall \alpha \notin \mathcal{P}: \quad \tau_\alpha(t_0+\delta t) \leq \tau_\alpha^c(t_0+\delta t)\,. \tag{70}$$

By using a Taylor expansion combined with Eq. (65), one obtains

$$\forall \alpha \notin \mathcal{P}: \quad \begin{cases} \tau_\alpha(t_0+\delta t) = \mathbf{M}_\alpha^0 : \underline{\boldsymbol{\sigma}}(t_0) + \delta t\, \mathbf{M}_\alpha : \dot{\underline{\boldsymbol{\sigma}}}(t_0) \\ \tau_\alpha^c(t_0+\delta t) = \tau_\alpha(t_0) + \delta t\, h_{\alpha\beta}(t_0)\, \dot{\gamma}_\beta(t_0) \end{cases} \quad ; \quad \beta \in \mathcal{P}\,, \tag{71}$$

where the stress rate $\dot{\underline{\boldsymbol{\sigma}}}(t_0)$ is computed by the following relation:

$$\dot{\underline{\boldsymbol{\sigma}}}(t_0) = \mathcal{L}_e : \left(\underline{\mathbf{d}}(t_0) - \dot{\gamma}_\beta(t_0)\, \mathbf{M}_\beta^0\right) \quad ; \quad \beta \in \mathcal{P}\,. \tag{72}$$

The combination of Eqs. (70) and (71) gives the following condition on δt, which must, of course, be greater than 0 and inferior or equal to Δt:

$$\delta t = \min_{\alpha \notin \mathcal{P}} \left\{ \Delta t,\ \frac{\tau_\alpha^c(t_0) - \underline{\boldsymbol{\sigma}}(t_0)}{\mathbf{M}_\alpha^0 : \mathcal{L}_e : \left(\underline{\mathbf{d}}(t_0) - \dot{\gamma}_\beta(t_0)\mathbf{M}_\beta\right) - h_{\alpha\beta}(t_0)\, \dot{\gamma}_\beta(t_0)} \quad ; \quad \beta \in \mathcal{P} \right\}. \tag{73}$$

If $\delta t < \Delta t$, then an implicit evaluation of the slip rates turns out to be very complicated. Indeed, in this case, an iterative loop is required to update the length of the time sub-increment δt, in addition to the iterative loop used to implicitly evaluate the slip rates for a given value of δt [24]. Furthermore, the value of δt is generally small (most often below Δt). For these reasons, the explicit evaluation $\dot{\boldsymbol{\gamma}}(t_0)$ of the slip rates obtained through Step 1 is considered here to be reasonable, and the implicit correction is not required. In case $\delta t = \Delta t$, the iterative loop to update δt is not required. Therefore, a straightforward implicit correction can be achieved to accurately compute the slip rates at $t_0+\delta t$ $(= t_0 + \Delta t)$. This implicit correction is detailed in the next step.

- *Step 3:* In this step, the slip rates of the potentially active slip systems are implicitly corrected in order to ensure the overall accuracy and stability of the solution. This implicit correction allows the use of large time increments. The set of active slip systems at $t_0+\delta t$ is assumed to be the same as that explicitly obtained in Step 1 at t_0. In order to compute the slip rates of these active slip systems, the linear system (69) is replaced by the following system of non-linear equations:

$$\forall \alpha \in \mathcal{A}: \quad \tilde{A}_{\alpha\beta}(t_0+\delta t)\dot{\gamma}_\beta(t_0+\delta t) = \tilde{b}_\alpha(t_0+\delta t) \quad ; \quad \beta \in \mathcal{A} \tag{74}$$

where the components of $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{b}}$ are given by

$$\forall \alpha,\beta \in \mathcal{A}: \quad \tilde{A}_{\alpha\beta}(t_0+\delta t) = \mathbf{M}_\alpha^0 : \boldsymbol{\mathcal{L}}_e : \mathbf{M}_\beta^0 + h_{\alpha\beta}(t_0+\delta t) \quad ; \quad \tilde{b}_\alpha(t_0+\delta t) = \mathbf{M}_\alpha^0 : \boldsymbol{\mathcal{L}}_e : \underline{\mathbf{d}}(t_0+\delta t). \tag{75}$$

The set of non-linear equations (74) can be solved by at least two iterative techniques: the Newton–Raphson method and the fixed point method. On the basis of various numerical tests and simulations, the latter method has been preferred. Indeed, this method does not require the analytical or numerical computation of some Jacobian matrix, which is needed in the application of the Newton–Raphson method. The detailed procedure of the fixed point method is given in Appendix D.

If some slip rates $\dot{\gamma}_\beta(t_0+\delta t)$, computed by the fixed point procedure, violate at least one constraint of the NLCP (66), then the set of active slip systems at $t_0+\delta t$ is different from that determined at t_0. In this particular case, the basic iterative search strategy developed in [24], [33] can be used to choose another set of active slip systems, and Step 3 is repeated until fulfillment of the different constraints of (66). Note that, fortunately, thanks to the slow evolution of matrix $\tilde{\mathbf{A}}$ and vector $\tilde{\mathbf{b}}$ over I^δ, this particular case is seldom encountered.

3.2.3.2. Update of the other variables

- If the slip rates are computed explicitly (i.e., without the use of the implicit correction), the different mechanical variables are updated as follows:

$$\begin{aligned}
&\underline{\boldsymbol{\sigma}}(t_0+\delta t) = \underline{\boldsymbol{\sigma}}(t_0) + \delta t\, \underline{\dot{\boldsymbol{\sigma}}}(t_0) = \underline{\boldsymbol{\sigma}}(t_0) + \delta t\, \boldsymbol{\mathcal{L}}_e : (\underline{\mathbf{d}}(t_0) - \dot{\gamma}_\beta(t_0)\,\mathbf{M}_\beta^0) \quad ; \quad \beta \in \mathcal{A} \\
&\mathbf{r}(t_0+\delta t) = \mathbf{r}(t_0).\mathbf{e}^{\delta t\, \underline{\mathbf{w}}\,(t_0+\delta t)} \quad ; \quad \boldsymbol{\sigma}(t_0+\delta t) = \mathbf{r}(t_0+\delta t).\,\underline{\boldsymbol{\sigma}}(t_0+\delta t).\,\mathbf{r}^T(t_0+\delta t) \\
&\forall \alpha \in \mathcal{A}: \quad \gamma_\alpha(t_0+\delta t) = \gamma_\alpha(t_0) + \delta t\, \dot{\gamma}_\alpha(t_0) \\
&\forall \alpha = 1,...,2N_s: \quad \tau_\alpha(t_0+\delta t) = \tau_\alpha^c(t_0) + \delta t\, h_{\alpha\beta}(t_0)\,\dot{\gamma}_\beta(t_0) \quad ; \quad \beta \in \mathcal{A}
\end{aligned} \tag{76}$$

The time increment Δt and the initial time t_0 are also updated

$$\Delta t = \Delta t - \delta t \quad ; \quad t_0 = t + \delta t. \tag{77}$$

After this update stage, the computation must be restarted from Section 3.2.1, with a new sub-increment I^δ.

- If the implicit correction is used, then the different mechanical variables are updated as follows:

$$\begin{aligned}
&\underline{\boldsymbol{\sigma}}(t_0+\delta t) = \underline{\boldsymbol{\sigma}}(t_0) + \delta t\, \underline{\dot{\boldsymbol{\sigma}}}(t_0+\delta t) = \underline{\boldsymbol{\sigma}}(t_0) + \delta t\, \boldsymbol{\mathcal{L}}_e : (\underline{\mathbf{d}}(t_0+\delta t) - \dot{\gamma}_\beta(t_0+\delta t)\,\mathbf{M}_\beta^0) \quad ; \quad \beta \in \mathcal{A} \\
&\boldsymbol{\sigma}(t_0+\delta t) = \mathbf{r}(t_0+\delta t).\,\underline{\boldsymbol{\sigma}}(t_0+\delta t).\,\mathbf{r}^T(t_0+\delta t) \\
&\forall \alpha \in \mathcal{A}: \quad \gamma_\alpha(t_0+\delta t) = \gamma_\alpha(t_0) + \delta t\, \dot{\gamma}_\alpha(t_0+\delta t) \\
&\forall \alpha = 1,...,2N_s: \quad \tau_\alpha^c(t_0+\delta t) = \tau_\alpha(t_0) + \delta t\, h_{\alpha\beta}(t_0+\delta t)\,\dot{\gamma}_\beta(t_0+\delta t) \quad ; \quad \beta \in \mathcal{A}
\end{aligned} \tag{78}$$

where $\mathbf{r}(t_0+\delta t)$ and $h_{\alpha\beta}(t_0+\delta t)$ are taken equal to their respective last converged values, computed in the application of the fixed point procedure (see Eqs. (D.1) and (D.5), respectively, in Appendix D).

After application of this update stage, the computation must be restarted from Section 3.2.1, with a new sub-increment I^{δ}.

3.2.4. Incremental objectivity of the ultimate algorithm

Several works have been developed in the literature to demonstrate the incremental objectivity of numerical integration schemes (see, e.g., [38–40]). To demonstrate that the developed ultimate algorithm is incrementally objective, let us apply a pure rotation $\mathcal{R}$, as an increment of the deformation gradient $\mathbf{f}$ over a typical time increment $[t_0, t_0+\Delta t]$

$$\mathbf{f}(t_0+\Delta t) = \mathcal{R}.\mathbf{f}(t_0)\,. \tag{79}$$

The algorithm is incrementally objective if the stress tensor $\boldsymbol{\sigma}(t_0+\Delta t)$ computed by the ultimate integration algorithm can be deduced from $\boldsymbol{\sigma}(t_0)$ and $\mathcal{R}$ by the following relation ([38]):

$$\boldsymbol{\sigma}(t_0+\Delta t) = \mathcal{R}.\boldsymbol{\sigma}(t_0).\mathcal{R}^{T}\,. \tag{80}$$

In order to check whether relation (79) implies (80), let us consider unit base vectors $\vec{\mathbf{e}}_i$ (i=1,2,3) related to the deformed configuration of the single crystal at t_0. Without loss of generality, let us choose $\vec{\mathbf{e}}_1$ and $\vec{\mathbf{e}}_2$ equal to $\vec{\mathbf{m}}_1/\|\vec{\mathbf{m}}\|$ and $\vec{\mathbf{n}}_1/\|\vec{\mathbf{n}}\|$, respectively. Here, $\vec{\mathbf{m}}_1$ and $\vec{\mathbf{n}}_1$ are the slip direction and the normal to the slip plane of the slip system number 1 in the deformed configuration at t_0. $\vec{\mathbf{e}}_3$ can be determined automatically from $\vec{\mathbf{e}}_1$ and $\vec{\mathbf{e}}_2$ by the following cross product:

$$\vec{\mathbf{e}}_3 = \vec{\mathbf{e}}_1 \wedge \vec{\mathbf{e}}_2\,. \tag{81}$$

Due to rotation $\mathcal{R}$, the unit base vectors $\vec{\mathbf{e}}_i$ (i=1,2,3) are transformed into unit base vectors $\vec{\mathbf{g}}_i$ (i=1,2,3) at $t_0+\Delta t$

$$\forall\, i=1,2,3:\quad \vec{\mathbf{g}}_i = \mathcal{R}.\vec{\mathbf{e}}_i\,. \tag{82}$$

The rotation tensors between the deformed configuration and the intermediate configuration of the single crystal (see Figure 1) at t_0 and $t_0+\Delta t$ are denoted, respectively, $\mathbf{r}(t_0)$ and $\mathbf{r}(t_0+\Delta t)$. Using Eqs. (78)$_{(2)}$ and (78)$_{(1)}$ (for $\delta t=\Delta t$), the following relation is obtained:

$$\begin{aligned}
\boldsymbol{\sigma}(t_0+\Delta t) &= \mathbf{r}(t_0+\Delta t).\underline{\boldsymbol{\sigma}}(t_0+\Delta t).\mathbf{r}^{T}(t_0+\Delta t)\\
&= \mathbf{r}(t_0+\Delta t).(\underline{\boldsymbol{\sigma}}(t_0)\quad \Delta t\,\dot{\underline{\boldsymbol{\sigma}}}(t_0+\Delta t)).\mathbf{r}^{T}(t_0+\Delta t)\\
&= \mathbf{r}(t_0+\Delta t).(\ ^{T}(t_0).\boldsymbol{\sigma}(t_0).\mathbf{r}(t_0)\quad \Delta t\,\dot{\underline{\boldsymbol{\sigma}}}(t_0+\Delta t)).\mathbf{r}^{T}(t_0+\Delta t)\\
&= [\mathbf{r}(t_0+\Delta t).\ ^{T}(t_0)].\boldsymbol{\sigma}(t_0).[\mathbf{r}(t_0+\Delta t).\mathbf{r}^{T}(t_0)]^{T} + \Delta t\,\mathbf{r}(t_0+\Delta t).\dot{\underline{\boldsymbol{\sigma}}}(t_0+\Delta t)).\mathbf{r}^{T}(t_0+\Delta t)
\end{aligned} \tag{83}$$

Let us also introduce the unit base vectors $\vec{\mathbf{t}}_i$ (i=1,2,3) related to the intermediate configuration and defined as follows:

$$\vec{\mathbf{t}}_1 = \vec{\mathbf{m}}_1^0 \quad ; \quad \vec{\mathbf{t}}_2 = \vec{\mathbf{n}}_1 \quad ; \quad \vec{\mathbf{t}}_3 = \vec{\mathbf{t}}_1 \wedge \vec{\mathbf{t}}_2 . \tag{84}$$

Then, $\vec{\mathbf{g}}_i$ can be related to $\vec{\mathbf{t}}_i$ by the following relation (as long as the elastic deformation is assumed to be very small):

$$\forall\, i = 1,2,3: \quad \vec{\mathbf{g}}_i = \mathbf{r}(t_0 + \Delta t).\vec{\mathbf{t}}_i . \tag{85}$$

On the other hand, $\vec{\mathbf{t}}_i$ and $\vec{\mathbf{e}}_i$ are related by

$$\forall\, i = 1,2,3: \quad \vec{\mathbf{t}}_i = \mathbf{r}^T(t_0).\vec{\mathbf{e}}_i . \tag{86}$$

Replacing $\vec{\mathbf{t}}_i$ in Eq. (85) by its expression from Eq. (86), one obtains

$$\forall\, i = 1,2,3: \quad \vec{\mathbf{g}}_i = \mathbf{r}(t_0 + \Delta t).\mathbf{r}^T(t_0).\vec{\mathbf{e}}_i . \tag{87}$$

By comparing (82) and (87), one can easily deduce that

$$\boldsymbol{\mathcal{R}} = \mathbf{r}(t_0 + \Delta t).\mathbf{r}^T(t\ ) . \tag{88}$$

The deformation between t_0 and $t_0+\Delta t$ is a pure rotation as defined in (79) and, accordingly, the strain rate $\mathbf{d}$ is equal to $\mathbf{0}$. Hence, $\underline{\mathbf{d}}$ is also equal to $\mathbf{0}$ and the slip rates of all slip systems are equal to zero. This conclusion is easy to understand by analyzing the relation between the slip rates and the strain rate $\underline{\mathbf{d}}$ (see for example Eqs. (67) and (69)). Then, the plastic strain $\underline{\mathbf{d}}_p$, defined by Eq. $(16)_{(3)}$, is equal to $\mathbf{0}$ and, using Eq. $(16)_{(1)}$, we can deduce easily that $\underline{\mathbf{d}}_e$ is also equal to $\mathbf{0}$. Consequently, the stress rate $\dot{\underline{\boldsymbol{\sigma}}}(t_0 + \Delta t)$ is equal to $\mathbf{0}$ and, hence, Eq. (83) reduces to

$$\boldsymbol{\sigma}(t_0 + \Delta t) = [\mathbf{r}(t_0 + \Delta t).\mathbf{r}^T(t_0)].\boldsymbol{\sigma}(t_0).[\mathbf{r}(t_0 + \Delta t).\mathbf{r}^T(t_0)]^T . \tag{89}$$

By replacing $\mathbf{r}(t_0 + \Delta t).\mathbf{r}^T(t\ )$ by $\boldsymbol{\mathcal{R}}$, as stated in (88), Eq. (89) is transformed to

$$\boldsymbol{\sigma}(t_0 + \Delta t) = \boldsymbol{\mathcal{R}}.\boldsymbol{\sigma}(t\ ).\boldsymbol{\mathcal{R}}^T , \tag{90}$$

which is exactly the same relation as (80). This shows that the ultimate algorithm is incrementally objective.

3.2.5. Some remarks on the ultimate algorithm

The remarks below rely on the fictitious yield surface introduced in Figure 2.

- **Remark 1**: The first sub-increment of computation is elastic, as the single crystal is generally assumed to be initially stress-free (i.e. $\boldsymbol{\sigma}(t = 0) = \mathbf{0}$), and this sub-increment permits to catch the first facet of the yield surface, which corresponds to the activation of system 1'. As the crystal is loaded further, the stress state rides along the system 1', until it activates system 2'. In this

manner, other facets are progressively reached in the following sub-increments until arriving at a vertex of the yield surface (the intersection of at least 5 facets). Note that, during this process, some systems may be activated simultaneously. Generally, at this stage, only one sub-increment (for typical values of strain increments, which are about 0.1–1% in FE simulations) is required in order to reach the next facet. Therefore, for these first sub-increments (when the stress state is not yet located at a vertex of the yield surface), the explicit algorithm is used without implicit correction, as δt is generally smaller than Δt.

- **Remark 2**: When the stress state at t_0 is laid on a vertex, and unless changes in the loading path or unloading occur, $\mathcal{L}_e : \underline{\mathbf{d}}(t)$ is an outward vector with respect to the yield surface. As the yield surface is convex, the stress remains constrained to this vertex, and it is impossible to reach a novel facet of the yield surface during the remainder of the time increment I^{Δ}. Accordingly, there is no change in the set of potentially active slip systems in this case. This implies that δt is equal to Δt (see Eq. (73)), which means that the implicit correction is possible in this case.

In order to geometrically illustrate the previous two remarks, a simplified 2D schematic representation of the single crystal yield surface is introduced in Figure 2. The idea behind the progression shown in Figure 2 can be easily extended to the general case of the single crystal yield surface (which is represented in the linear five-dimensional deviatoric space).

3.3. Derivation of an analytical expression for the consistent tangent modulus

For the sake of brevity, only the tangent modulus, consistent with the ultimate integration scheme, is derived in this section. For the return-mapping integration scheme, several versions of the consistent tangent modulus have been given (in a Lagrangian formulation) in [41] and [42].

In order to simplify the following developments, the time increment $[t_0, t_0+\Delta t]$ is assumed to consist of a single sub-increment. Accordingly, I^{Δ} (resp. Δt) is equal to I^{δ} (resp. δt). From a practical point of view, this assumption is generally valid after few time increments (usually starting from the third or the fourth time increment). When the time increment is composed of more than one sub-increment, the following development remains valid, provided it is applied to the last sub-increment of $[t_0, t_0+\Delta t]$. In order to simplify notations, the argument $t_0+\Delta t$ is dropped in the following developments, with the implied understanding that the corresponding variable is evaluated at $t_0+\Delta t$.

In order to take the finite strain aspects into account (finite deformation as well as finite rotation), the consistent tangent modulus is computed in the co-rotational frame, where it is defined by the following expression (see, e.g., [43–44]):

$$\hat{\mathcal{L}}_{ep} = \frac{\partial \Delta\hat{\boldsymbol{\sigma}}}{\partial \Delta\hat{\boldsymbol{\varepsilon}}}, \tag{91}$$

in which $\Delta\hat{\boldsymbol{\sigma}}$ and $\Delta\hat{\boldsymbol{\varepsilon}}$ represent, respectively, the Cauchy stress increment and the corresponding strain increment (equal to $\Delta t\,\hat{\mathbf{d}}$, as $\hat{\mathbf{d}}$ is assumed to be constant over the time increment), both being expressed in the co-rotational frame. This co-rotational frame is determined by the anti-symmetric part $\mathbf{w}$ of the velocity gradient $\mathbf{g}$. Then, the rotation tensor $\hat{\mathbf{r}}$ is defined as the orientation of the co-rotational frame relative to the fixed frame. This rotation has the following evolution law:

$$\hat{\mathbf{r}} = e^{\Delta t\,\mathbf{w}}.\hat{\mathbf{r}}(t_0). \tag{92}$$

Any tensor $\mathbf{X}$ used in the current section will be denoted $\hat{\mathbf{X}}$ when expressed in the co-rotational frame. To compute $\hat{\mathcal{L}}_{ep}$, the so-called secant modulus $\hat{\mathcal{L}}_{s}$, which relates the increment of the Cauchy stress tensor $\Delta\hat{\boldsymbol{\sigma}}$ to the increment of the strain tensor $\Delta\hat{\boldsymbol{\varepsilon}}$, is first determined

$$\Delta\hat{\boldsymbol{\sigma}} = \hat{\mathcal{L}}_{s} : \Delta\hat{\boldsymbol{\varepsilon}}. \tag{93}$$

Indeed, the stress increment $\Delta\hat{\boldsymbol{\sigma}}$ can be expressed as follows:

$$\Delta\hat{\boldsymbol{\sigma}} = \hat{\boldsymbol{\sigma}} - \hat{\boldsymbol{\sigma}}(t_0) = \underset{\sim}{\mathbf{r}}.\underline{\boldsymbol{\sigma}}.\underset{\sim}{\mathbf{r}}^{T} - \hat{\boldsymbol{\sigma}}(t\ ), \tag{94}$$

where $\underset{\sim}{\mathbf{r}}$ is the rotation of the intermediate configuration relative to the co-rotational frame. Its evolution is defined by the following equation:

$$\underset{\sim}{\mathbf{r}} = e^{-\Delta t\,\hat{\mathbf{w}}_p}.\underset{\sim}{\mathbf{r}}(t_0) = e^{-\Delta t\,\dot{\gamma}_\alpha \hat{\mathbf{S}}_\alpha}.\underset{\sim}{\mathbf{r}}(t\ ) \quad ; \quad \alpha \in \mathcal{A}. \tag{95}$$

Using the update equation $(78)_{(1)}$, $\Delta\hat{\boldsymbol{\sigma}}$ can be rewritten as

$$\begin{aligned} \Delta\hat{\boldsymbol{\sigma}} &= \hat{\boldsymbol{\sigma}} - \hat{\boldsymbol{\sigma}}(t_0) = \underset{\sim}{\mathbf{r}}.(\underline{\boldsymbol{\sigma}}(t_0) + \Delta t\,\dot{\underline{\boldsymbol{\sigma}}}).\underset{\sim}{\mathbf{r}}^{T} - \hat{\boldsymbol{\sigma}}(t_0) \\ &= [\underset{\sim}{\mathbf{r}}.\underset{\sim}{\mathbf{r}}^{T}(t_0)].\boldsymbol{\sigma}(t_0).[\underset{\sim}{\mathbf{r}}.\underset{\sim}{\mathbf{r}}^{T}(t_0)]^{T} + \Delta t\,\underset{\sim}{\mathbf{r}}.\dot{\underline{\boldsymbol{\sigma}}}.\underset{\sim}{\mathbf{r}}^{T} - \hat{\boldsymbol{\sigma}}(t_0). \end{aligned} \tag{96}$$

By using Eq. (95), $\underset{\sim}{\mathbf{r}}.\underset{\sim}{\mathbf{r}}^{T}(t_0)$ and $[\underset{\sim}{\mathbf{r}}.\underset{\sim}{\mathbf{r}}^{T}(t_0)]^{T}$ can be expressed as

$$\underset{\sim}{\mathbf{r}}.\underset{\sim}{\mathbf{r}}^{T}(t_0) = e^{-\Delta t\,\dot{\gamma}_\alpha \hat{\mathbf{S}}_\alpha} \quad , \quad [\underset{\sim}{\mathbf{r}}.\underset{\sim}{\mathbf{r}}^{T}(t\ )]^{T} = e^{\Delta t\,\dot{\gamma}_\alpha\ \alpha} \quad ; \quad \alpha \in \mathcal{A}. \tag{97}$$

For small time increments, tensor $e^{-\Delta t\,\dot{\gamma}_\alpha \hat{\mathbf{S}}_\alpha}$ (resp. $e^{\Delta t\,\dot{\gamma}_\alpha \hat{\mathbf{S}}_\alpha}$) can be approximated by $\mathbf{1} - \Delta t\,\dot{\gamma}_\alpha \hat{\mathbf{S}}_\alpha$ (resp. $\mathbf{1} + \Delta t\,\dot{\gamma}_\alpha \hat{\mathbf{S}}_\alpha$), since the higher-order terms in Δt are negligible.

Hence, Eq. (96) can be transformed into

$$\begin{aligned} \Delta\hat{\boldsymbol{\sigma}} &= \hat{\boldsymbol{\sigma}} - \hat{\boldsymbol{\sigma}}(t_0) = \underset{\sim}{\mathbf{r}}.(\underline{\boldsymbol{\sigma}}(t_0) + \Delta t\,\dot{\underline{\boldsymbol{\sigma}}}).\underset{\sim}{\mathbf{r}}^{T} - \hat{\boldsymbol{\sigma}}(t_0) \\ &= [\mathbf{1} - \Delta t\,\dot{\gamma}_\alpha\ \hat{}_\alpha].\boldsymbol{\sigma}(t_0).[\mathbf{1} + \Delta t\,\dot{\gamma}_\alpha \mathbf{S}_\alpha] + \Delta t\,\underset{\sim}{\mathbf{r}}.\dot{\underline{\boldsymbol{\sigma}}}.\underset{\sim}{\mathbf{r}}^{T} - \hat{\boldsymbol{\sigma}}(t_0) \end{aligned} \quad ; \quad \alpha \in \mathcal{A}. \tag{98}$$

By neglecting the second-order terms in Δt, the expression of $\Delta\hat{\boldsymbol{\sigma}}$ can be reduced to

$$\begin{aligned}\Delta\hat{\boldsymbol{\sigma}} &= \hat{\boldsymbol{\sigma}}(t_0) - \Delta t\,\dot{\gamma}_\alpha\,\hat{\mathbf{S}}_\alpha .\hat{\boldsymbol{\sigma}}(t_0) + \Delta t\,\dot{\gamma}_\alpha\,\hat{\boldsymbol{\sigma}}(t_0).\mathbf{S}_\alpha + \Delta t\,\underset{\sim}{\mathbf{r}}.\underline{\dot{\boldsymbol{\sigma}}}.\underset{\sim}{\mathbf{r}}^{T} - \hat{\boldsymbol{\sigma}}(t_0) \\ &= -\Delta t\,\dot{\gamma}_\alpha\,\mathbf{S}_\alpha .\boldsymbol{\sigma}(t_0) + \Delta t\,\dot{\gamma}_\alpha\,\hat{\boldsymbol{\sigma}}(t_0).\hat{\mathbf{S}}_\alpha + \Delta t\,\underset{\sim}{\mathbf{r}}.\underline{\dot{\boldsymbol{\sigma}}}.\underset{\sim}{\mathbf{r}}^{T}.\end{aligned} \quad ; \quad \alpha \in \mathcal{A}\,. \tag{99}$$

By using Eq. $(78)_{(1)}$, $\underset{\sim}{\mathbf{r}}.\underline{\dot{\boldsymbol{\sigma}}}.\underset{\sim}{\mathbf{r}}^{T}$ can be determined in the following form:

$$\underset{\sim}{\mathbf{r}}.\underline{\dot{\boldsymbol{\sigma}}}.\underset{\sim}{\mathbf{r}}^{T} = \boldsymbol{\mathcal{L}}_e : (\hat{\mathbf{d}} - \dot{\gamma}_\alpha\,\hat{\mathbf{M}}_\alpha) = \boldsymbol{\mathcal{L}}_e : (\hat{\mathbf{d}} - \dot{\gamma}_\alpha\,\hat{\mathbf{R}}_\alpha) \quad ; \quad \alpha \in \mathcal{A}\,. \tag{100}$$

The combination of Eqs. (65)–(67) gives the following expression for the slip rates of the active slip systems:

$$\forall\,\alpha \in \mathcal{A}: \quad \dot{\gamma}_\alpha = \Lambda_{\alpha\beta} \quad {}_\beta : \boldsymbol{\mathcal{L}}_e : \hat{\mathbf{d}} \quad ; \quad \beta \in \mathcal{A}\,, \tag{101}$$

where the square matrix $\boldsymbol{\Lambda}$ is the inverse of $-\tilde{\mathbf{A}}$ defined in Eq. (67).

Taking into account expression (101) for $\dot{\gamma}_\alpha$, Eq. (100) can be rewritten as

$$\underset{\sim}{\mathbf{r}}.\underline{\dot{\boldsymbol{\sigma}}}.\underset{\sim}{\mathbf{r}}^{T} = [\boldsymbol{\mathcal{L}}_e - \Lambda_{\alpha\beta}\,(\boldsymbol{\mathcal{L}}_e : \hat{\mathbf{R}}_\alpha) \otimes (\hat{\mathbf{R}}_\beta : \boldsymbol{\mathcal{L}}_e)] : \hat{\mathbf{d}} \quad ; \quad \alpha,\beta \in \mathcal{A}\,. \tag{102}$$

By inserting Eq. (102) into Eq. (99), and using expression (101), one obtains

$$\begin{aligned}\Delta\hat{\boldsymbol{\sigma}} &= \Delta t\,[\boldsymbol{\mathcal{L}}_e - \Lambda_{\alpha\beta}\,(\boldsymbol{\mathcal{L}}_e : \hat{\mathbf{R}}_\alpha + \hat{\mathbf{S}}_\alpha .\hat{\boldsymbol{\sigma}}(t_0) - \hat{\boldsymbol{\sigma}}(t_0).\mathbf{S}_\alpha) \otimes (\hat{\mathbf{R}}_\beta : \boldsymbol{\mathcal{L}}_e)] : \hat{\mathbf{d}} \\ &= [\boldsymbol{\mathcal{L}}_e - \Lambda_{\alpha\beta}\,(\boldsymbol{\mathcal{L}}_e : \hat{\mathbf{R}}_\alpha + \hat{}_\alpha .\hat{\boldsymbol{\sigma}}(t_0) - \hat{\boldsymbol{\sigma}}(t_0).\mathbf{S}_\alpha) \otimes (\mathbf{R}_\beta : \boldsymbol{\mathcal{L}}_e)] : \Delta\hat{\boldsymbol{\varepsilon}}\end{aligned} \quad ; \quad \alpha,\beta \in \mathcal{A}\,. \tag{103}$$

The secant modulus $\hat{\boldsymbol{\mathcal{L}}}_s$ can then be identified by comparing Eqs. (93) and (103)

$$\hat{\boldsymbol{\mathcal{L}}}_s = [\boldsymbol{\mathcal{L}} \; - \Lambda_{\alpha\beta}\,(\boldsymbol{\mathcal{L}}_e : \hat{\mathbf{R}}_\alpha + \hat{\mathbf{S}}_\alpha .\hat{\boldsymbol{\sigma}}(t_0) - \hat{\boldsymbol{\sigma}}(t_0).\hat{\mathbf{S}}_\alpha) \otimes (\hat{\mathbf{R}}_\beta : \boldsymbol{\mathcal{L}}_e)] \quad ; \quad \alpha,\beta \in \mathcal{A}\,. \tag{104}$$

Finally, Eq. (93) is differentiated in order to derive the expression of the consistent tangent modulus $\hat{\boldsymbol{\mathcal{L}}}_{ep}$

$$\partial\Delta\hat{\boldsymbol{\sigma}} = \partial(\hat{\boldsymbol{\mathcal{L}}}_s : \Delta\hat{\boldsymbol{\varepsilon}}) = \hat{\boldsymbol{\mathcal{L}}}_s : \partial\Delta\hat{\boldsymbol{\varepsilon}} + \partial\hat{\boldsymbol{\mathcal{L}}}_s : \Delta\hat{\boldsymbol{\varepsilon}} = (\hat{\boldsymbol{\mathcal{L}}}_s + \hat{\boldsymbol{\mathcal{L}}}_s^{*}) : \partial\Delta\hat{\boldsymbol{\varepsilon}}\,, \tag{105}$$

where $\hat{\boldsymbol{\mathcal{L}}}_s^{*}$ is a fourth-order tensor, which is determined by the following relation:

$$\hat{\boldsymbol{\mathcal{L}}}_s^{*} : \partial\Delta\hat{\boldsymbol{\varepsilon}} = \partial\hat{\boldsymbol{\mathcal{L}}} : \Delta\hat{\boldsymbol{\varepsilon}}\,. \tag{106}$$

The analytical expression of $\hat{\boldsymbol{\mathcal{L}}}_s^{*}$ is derived in Appendix E.

Once $\hat{\boldsymbol{\mathcal{L}}}_s^{*}$ has been determined, the consistent tangent modulus $\hat{\boldsymbol{\mathcal{L}}}_{ep}$ is obtained as follows:

$$\hat{\boldsymbol{\mathcal{L}}}_{ep} = \hat{\boldsymbol{\mathcal{L}}}_s + \hat{\boldsymbol{\mathcal{L}}}_s^{*}\,. \tag{107}$$

Thereafter, $\boldsymbol{\mathcal{L}}_{ep}$ can be determined by rotating $\hat{\boldsymbol{\mathcal{L}}}_{ep}$ from the co-rotational frame to the fixed frame.

It is worth noting that, in addition to the analytical determination of the consistent tangent operator, the latter can also be determined numerically by combining the perturbation technique with the finite difference method [45]. For this numerical evaluation of the consistent tangent modulus, the integration scheme of Section 3.2 has to be applied seven times: six times for building the consistent tangent

modulus, column by column, by perturbing the six components of $\Delta\hat{\boldsymbol{\varepsilon}}$ one at a time, and once for updating the different mechanical variables. Hence, the use of this numerical version of the consistent tangent modulus appears to be very expensive in terms of computational time. In Section 6, a comparison between these two techniques will be presented, both in terms of numerical values for the consistent tangent matrix components and also in terms of the quadratic convergence of the Newton–Raphson method associated with an initial boundary value problem.

4. Single crystal scale: Numerical results and discussions

The objective of this section is to compare the accuracy, efficiency and robustness of the two developed integration algorithms at the single crystal scale. To this end, both algorithms, namely the return-mapping and the ultimate algorithm, with their different versions and formulations, have been implemented using the multi-paradigm numerical computing environment *Matlab (R2014)*. This choice is motivated by the fact that this software offers efficient and powerful tools and functionalities in order to optimize the numerical implementation. The simulations in this paper are all performed on a personal computer with 2.00 GHz of CPU frequency and 6.00 Go of RAM memory. To extensively compare the respective performances of these algorithms, the simulations are carried out in the current section on single crystals subjected to both constant and non-constant loadings with various initial orientations.

4.1. Initial state, hardening law and material parameters

The initial state for each single crystal, in terms of stress and plastic slip, is characterized by

$$\boldsymbol{\sigma}(t=0)=\mathbf{0} \quad ; \quad (\alpha=1,...,2N_s): \quad \gamma_\alpha(t=0)=0. \tag{108}$$

The hardening law considered here was initially introduced by Chang and Asaro [46], and recently employed by Miehe and Schröder [12] for FCC single crystals. This hardening law is adapted here to the case of BCC single crystals. With this law, the components of the hardening modulus $\mathbf{h}$ are defined by the following expressions:

$$\forall\, \alpha,\beta=1,...,24: \quad h_{\alpha\beta}=\hat{h}(A)\left[q+(1-q)\delta_{\alpha\beta}\right. \quad ; \quad h(A)=h_0\,\mathrm{sech}^2\left(\frac{h_0 A}{\tau_{sat}-\tau_0}\right), \tag{109}$$

where $\delta_{\alpha\beta}$ is the Krönecker symbol and A is the sum of the accumulated plastic slip on all slip systems ($=\sum_{\alpha=1}^{48}\gamma_\alpha$). As to τ_{sat},τ_0,q and h_0, they represent material parameters. The values of both elasticity and hardening parameters are reported in Table 1, see also [12].

The numerical results of Section 4 are obtained by considering four different initial orientations of single crystals (as reported in Table 2).

4.2. Constant loading 1: Combined shearing–stretching test

This section presents the simulations of the mechanical behavior of a single crystal subjected to constant loading. This loading corresponds to a combined shearing–stretching test characterized by the following Eulerian velocity gradient:

$$\mathbf{g} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & -0.5 \end{bmatrix} \ (\mathrm{s}^{-1}) \,. \tag{110}$$

4.2.1. Limitations of the non-smooth formulation coupled with the return-mapping algorithm

As shown in Eq. (26), the return-mapping algorithm is classically used with the non-smooth formulation of the Schmid law. With such a formulation, and for an implicit integration scheme, two nested loops are required: an external loop to determine the set of active slip systems among the potentially active ones, and an internal loop to iteratively compute (with the Newton–Raphson method) the slip increments of the active slip systems. This formulation has been adopted in the majority of previous works devoted to the numerical integration of the constitutive equations of FCC single crystals, see e.g., [11], [12], [14]. Unfortunately, this return-mapping algorithm associated with the non-smooth formulation fails sometimes, especially for relatively large time increments. To illustrate this failure, three simulations are carried out to predict the mechanical response of the single crystal with the initial orientation # 2 given in Table 2. From one simulation to another, only the size of the time increment Δt is changed (3×10^{-4}, 4×10^{-4} and 5×10^{-4} s). The simulations are planned to be performed until the tenth time increment is reached (independently of its size). The evolution of both sets of slip systems, potentially active and active, during the simulations is detailed in Table 3. As revealed by this table, the set of potentially active slip systems $\mathcal{P}$ seems to be dependent on the value of $\Delta t \mathbf{g}$. When the product $\Delta t \mathbf{g}$ increases, the cardinal of $\mathcal{P}$ also increases. This cardinal may, in some cases, reach N_s (but it is never greater than N_s). This latter point will be further discussed in Section 4.2.3.

When the time increment Δt is equal to 3×10^{-4} s, the computation is successfully carried out up to the last time increment. At the first four increments, the predicted behavior is purely elastic. For this reason, the set $\mathcal{P}$ (and hence the set $\mathcal{A}$) are empty. At the fifth increment, four slip systems seem to be potentially active. Using the iterative search strategy for the determination of active slip systems (the same as that used by Anand and Kothari [11]), we find that only two slip systems belonging to $\mathcal{P}$ are active. The same iterative search strategy is applied successfully to the remainder of the time increments.

When the time increment Δt is equal to 4 x 10^{-4} s, the mechanical response at the first three increments is found to be purely elastic, and becomes elastic–plastic at the fourth increment. At this fourth increment, the iterative search procedure is applied with success to integrate the constitutive equations. At the fifth increment, the set $\mathcal{P}$ consists of 6 slip systems. Following Anand and Kothari [11], the set of active slip systems is assumed to be identical to $\mathcal{P}$ at the first iteration of the iterative search strategy. To determine the slip increments for this combination, we seek to iteratively solve the following set of non-linear equations:

$$\forall\, \alpha \in \mathcal{A} = \mathcal{P}: \quad \tau_\alpha(t_0 + \Delta t) = \tau^c(t\ + \Delta t) \Rightarrow A_{\alpha\beta}\,\Delta\gamma_\beta = b_\alpha \quad ; \quad \beta \in \mathcal{A} = \mathcal{P}, \tag{111}$$

where $\mathbf{A}$ and $\mathbf{b}$ are, respectively, the matrix and the vector introduced in Eq. (50).

After several attempts to iteratively solve system (111), it seems that it has no solution. This is due to the fact that the facets corresponding to the six slip systems of $\mathcal{A}(=\mathcal{P})$ cannot be intersected to define a vertex of the yield surface. Therefore, the slip systems belonging to $\mathcal{A}$ cannot be activated all together. This implies that there is at least one slip system α that is inactive and, for this slip system, the critical shear stress $\tau_\alpha^c(t_0 + \Delta t)$ is strictly superior to the resolved shear stress $\tau_\alpha(t_0 + \Delta t)$. More specifically, the number of inactive slip systems is found to be equal to 4 (systems 4, 22, 24, 30). For this reason, the convergence of Eq. (111) cannot be reached. Accordingly, the computation is stopped at this level, because the iterative search strategy applied to this non-smooth form of the return-mapping algorithm cannot be continued any further. Indeed, to proceed further, the slip increments of the slip systems belonging to $\mathcal{A}$ need to be determined because they would be used to update the set of active slip systems (see the iterative search procedure introduced in [11]).

The problem above may be geometrically illustrated, by considering a different fictitious situation of three potentially active slip systems denoted by 1', 2', 3' (see Figure 3). In this figure, we have

$$\begin{gathered}(\tau_{1'}^{Tr}(t_0 + \Delta t) = T^{Tr}(t_0 + \Delta t) : \mathbf{R}_{1'}^0) > \tau_{1'}^c(t_0) \quad ; \quad (\tau_{2'}^{Tr}(t_0 + \Delta t) = T^{Tr}(t_0 + \Delta t) : \mathbf{R}_{2'}^0) > \tau_{2'}^c(t_0) \\ (\tau_{3'}^{Tr}(t_0 + \Delta t) = T^{Tr}(t_0 + \Delta t) : \mathbf{R}_{3'}^0) > \tau_{3'}^c(t_0)\end{gathered} . \tag{112}$$

Therefore, the set $\mathcal{P}$ is equal to {1', 2', 3'}. At the first iteration of the iterative search procedure, the set of active slip systems $\mathcal{A}$ is assumed to be equal to $\mathcal{P}$. However, the facets corresponding to the slip systems 1', 2', 3' are unable to form together a vertex of the yield surface. Indeed, only couples {1', 2'} and {2', 3'} are vertices of the yield surface. Accordingly, the following mathematical system has no solution:

$$\tau_{1'}(t_0 + \Delta t) = \tau_{1'}^c(t_0 + \Delta t) \quad ; \quad \tau_{2'}(t_0 + \Delta t) = \tau_{2'}^c(t_0 + \Delta t) \quad ; \quad \tau_{3'}(t_0 + \Delta t) = \tau_{3'}^c(t_0 + \Delta t). \tag{113}$$

When the time increment Δt is equal to 5×10^{-4} s, the computations are no longer consistent at the fourth time increment for the same reasons explained in the previous case.

The simulations with the other initial crystallographic orientations exhibit similar behavior, and the same problems encountered for orientation # 2 are observed once again. Therefore, they are not presented for conciseness.

Unfortunately, the critical size of the time increment, which should not be exceeded in order to successfully apply the iterative search procedure, is not a priori known. Therefore, to overcome the previous problem, and thus ensure the numerical convergence of the return-mapping algorithm, three solutions may be followed:

- The use of very small time increments. This choice permits to reduce the cardinal of the set of potentially active slip systems, and thus to minimize the risk of non-convergence of Eq. (111). However, it generally leads to a significant increase in the CPU time, which represents major restrictions and shortcomings of the non-smooth formulation of the return-mapping algorithm (especially for the computation of polycrystalline structures).
- The use of an adaptive time step strategy. This strategy consists in dividing the time step by two, in case of non-convergence of the computation after 50 combinations of the set of potentially active slip systems have been tested. In such a case, the computation is restarted from the latest converged configuration.
- The use of another iterative search procedure, as alternative to that developed in [11] and followed here. Such an alternative procedure would be effective if the assumed set of active slip systems can form a valid intersection of the facets of the yield surface. This alternative seems to be also time consuming, considering the large number of combinations that would be involved.
- The application of a different approach, alternative to the non-smooth formulation, where this kind of problems is inherently avoided. This is possible by taking advantage of the semi-smooth formulation.

4.2.2. Comparison between the semi-smooth and non-smooth formulations of the return-mapping algorithm

In this section, the semi-smooth formulation of the return-mapping algorithm is adopted to integrate the constitutive equations. With this formulation, the iterative search procedure intended to the determination of the set of active slip systems is not required. Thanks to the mathematical form of this semi-smooth formulation, the problem encountered in Section 4.2.1 is inherently avoided here. In fact, with this semi-smooth formulation, the stress state at the end of the time increment necessarily lies on a vertex of the yield surface. Therefore, unlike the non-smooth formulation, the computation always converges regardless of the size of the time increment. In practice, the integration of the semi-smooth formulation is based on the global Newton–Raphson method, which always converges in this case (see

Table 4). These results emphasize the efficiency of the semi-smooth formulation as compared to its non-smooth counterpart in the context of the numerical integration of crystal plasticity constitutive equations. To confirm these conclusions, the simulations are performed once again with the other initial orientations of Table 2 (orientations # 1, 3 and 4) showing that the computations always converge.

The results predicted by the use of both formulations (the non-smooth and the semi-smooth) are further compared in Figure 4. To this end, and in order to ensure the convergence of the non-smooth formulation of the return-mapping algorithm, the time increment Δt is taken to be equal to 2×10^{-4} s. The computations are stopped at t=0.05 s. The solid lines represent the predictions of the non-smooth formulation, while the dotted graphs correspond to the results obtained with the semi-smooth formulation. In Figure 4a, the evolution of the accumulated slip γ of the active slip systems is plotted versus time t. The number of activated slip systems is shown in this figure. It appears that the two distinct formulations predict the same set of active slip systems and the same accumulated slips. This equivalence in terms of numerical predictions is also observed for the evolution of the components σ_{11} and σ_{12} of the stress tensor, as demonstrated in Figure 4b. This perfect agreement between the results predicted by the semi-smooth formulation and those yielded by the non-smooth formulation is confirmed by additional simulations corresponding to the other initial orientations of Table 2.

From various numerical simulations, it is concluded that, in contrast to the non-smooth formulation, the algorithm based on the semi-smooth formulation always converges (independently of the value of the time increment). However, the convergence rate of this algorithm depends on several numerical parameters and choices, such as the parameters used in the line search strategy (see Appendix B), the size of the time increment and especially the initial guess for the slip increments, which is required at each increment in the global Newton–Raphson procedure. In fact, the choice of the initial guess for the slip increments plays a crucial role in the convergence rate of the global Newton–Raphson procedure. An illustration of its impact on the CPU time is provided in Figure 5. In Figure 5a, the simulations are performed over a single time increment with a size Δt of 5×10^{-4} s. The results are presented for the different single crystal initial orientations of Table 2. The initial guess for the slip increments is assumed to be the same for the different potentially active slip systems. For each initial orientation, three simulations are performed: the first with an initial guess $\Delta\gamma^{(0)}=10^{-1}$, the second with an initial guess $\Delta\gamma^{(0)}=10^{-2}$ and the third with an initial guess $\Delta\gamma^{(0)}=10^{-3}$. It is clear that the CPU time depends on the value of the initial guess. However, at this stage, it is not obvious to know which initial guess should be selected to minimize the CPU time and therefore to optimize the numerical integration. To the best of our knowledge, there is no universal method for choosing an initial guess that minimizes the CPU time for this problem. As the evolution of the slip increments from one increment to another is very slow, a more optimal choice for the initial guess of the slip increments would be the converged values of the slip increments at the previous time step (denoted in the following by ‘LCS’ for ‘last

converged solution'). However, this choice is possible only if the set of potentially active slip systems remains the same from one time step to another. This is not always possible, especially in case of complex loading, when the loading direction evolves continuously during the loading history (see Section 5). On the other hand, there is no guarantee that a potentially active slip system will continue to be potential, even for a constant loading. The impact of the choice of the initial guess on a multi-step computation is illustrated in Figure 5b. In this case, the computations are carried out over 10 time increments and the size of each increment is equal to 5×10^{-4} s. For each initial orientation, three types of initial guess for the slip increments are compared: two predefined guesses ($\Delta\gamma^{(0)}=10^{-1}$ and $\Delta\gamma^{(0)}=10^{-2}$ for the potentially active slip systems), as well as the more optimal choice detailed previously. For the different orientations, the more optimal choice seems to be a "good" initial guess because it allows minimizing the CPU time.

Analyzing the results of Figure 5a, one can observe that the CPU time required for the execution of one time increment for one single crystal always exceeds 0.05 s, regardless of the initial orientation or the initial guess. This trend shows that the CPU time will be very high if a multiscale simulation is performed over a large number of time steps. This drawback is likely to considerably limit the performance of the return-mapping algorithm based on the semi-smooth formulation. As will be demonstrated later, this limitation can be easily overcome by using the ultimate algorithm.

4.2.3. Comparison between the ultimate and return-mapping algorithms

Before comparing the respective efficiency of the ultimate and return-mapping algorithms, a prerequisite would be to verify that the two algorithms allow predicting the same mechanical response at the single crystal scale. In Figure 6, the accumulated slips of the active slip systems predicted by the ultimate algorithm (explicit/implicit algorithm) are compared with their counterparts predicted by the return-mapping algorithm (semi-smooth formulation). For all the simulations presented in this section, the time increment Δt is fixed to 5×10^{-4} s. For both algorithms used, the computations are stopped at t=0.2 s. The solid lines represent the predictions of the ultimate algorithm, while the dotted graphs correspond to the results obtained with the return-mapping algorithm. One can observe that the simulations by the two algorithms result in identical sets of active slip systems and exactly the same accumulated slips for these systems for all of the initial crystallographic orientations considered. This obviously leads to the same evolution for the stress components as well as for the crystallographic texture.

A comparative study of the efficiency, in terms of CPU time, for the two algorithms is given in Figure 7. In this figure, "R-M" (resp. "UL") stands for "return-mapping" (resp. "ultimate") algorithm. In these simulations, two different sizes for the time increment are used: $\Delta t=5 \times 10^{-4}$ s, in Figure 7a, and $\Delta t= 10^{-3}$ s, in Figure 7b. For both cases, the computations are stopped at t=0.2 s. Two types of initial guess

are used when the return-mapping algorithm is applied: a predefined constant guess ($\Delta\gamma^{(0)}=10^{-2}$), and the "LCS" guess strategy (i.e., $\Delta\gamma^{(0)}=10^{-2}$, whenever a different set of potentially active slip systems is involved, and the converged value at the previous time increment, when the set of potentially active slip systems is the same as that at the previous time increment). The dependence of the CPU time on the choice of the initial guess for the slip increments, when the return-mapping algorithm is applied, is confirmed here once again. In this case, the application of the "LCS" initial guess allows considerably reducing the CPU time. This beneficial feature is especially true in this case, because the loading applied to the single crystal is constant throughout the deformation. Indeed, the set of potentially active slip systems changes much less frequently during such types of loading.

A noteworthy observation is that, although the number of increments required for the simulations of Figure 7a is twice larger than that needed for the simulations of Figure 7b, the required CPU time is not necessarily proportional to this factor; it is sometimes comparable or even less for some simulations of Figure 7a. This implies that the CPU time required to solve the problem for each increment tends to increase with the size of the time increment. This fact is obvious considering the following two reasons:

- The number of potentially active slip systems, and consequently the size of the mathematical system to be solved at each increment, increases with the size of the time increment. This trend can be clearly observed through the results of Figure 8 (which are also conformal with the results of Table 4). It is also observed in Figure 8 that the number of potentially active slip systems may be 1.5 to 3 times larger than the number of active slip systems.
- The number of iterations also generally increases with the size of the time increment as demonstrated in Figure 9. In this figure, the total number of Newton–Raphson iterations is averaged by the total number of time increments (and will be denoted by ‘AIN’, for ‘average iteration number’ in Figure 9).

To avoid the two problems mentioned above, an adaptive time step strategy can be used when the number of iterations required to solve the Fischer–Burmeister complementarity functions of Eq. (51) reaches a user-defined threshold value (here this value is fixed to 50). This strategy obviously increases the number of time increments for the whole simulation; nevertheless, due to its efficiency, the total CPU time would decrease. The return-mapping algorithm, enhanced with this adaptive time step strategy, will be systematically used in Section 5, corresponding to the polycrystalline simulations.

On the other hand, the ultimate explicit/implicit algorithm seems to be largely more efficient than the return-mapping algorithm. This is due to the following facts:

- The search strategy detailed in Step 1 of Section 3.2.3, for the determination of active slip systems and their rates in the explicit phase, turns out to be a safe scheme to handle the complex structural changes in the slip activity, with sufficient accuracy and reasonable CPU time, as compared to alternative search strategies (such as that adopted in [12]). Indeed, this search procedure, applied in this constant loading case, converges very often at the first or the second

search iteration. This result is clearly demonstrated in Figure 8. One can see from this figure that, for the ultimate algorithm, the number of potentially active slip systems is always equal to the number of active slip systems. Therefore, the iterative search procedure converges at the first iteration. By following this search procedure, the set of active slip systems is most often the same as the set of potentially active slip systems (contrary to the return-mapping algorithm).

- The solution $\dot{\boldsymbol{\gamma}}(t_0)$ of the LCP (68) is a good initial guess for the non-linear system (74), and may obviously be considered as a reasonable initial guess for the implicit phase (whenever this phase is considered). Indeed, the slip rates evolve rather slowly over I^{δ} and, accordingly, the fixed point method most often requires less than four iterations, regardless of the length of I^{δ}. This result is confirmed by Figure 9, in which the total number of iterations for the fixed point method is averaged by the total number of time increments (when the implicit phase is used). For this reason, the CPU time needed for each increment is almost independent of the increment size. This results in a linear proportionality between the CPU time and the number of increments, as clearly observed in Figure 7.

Unlike the return-mapping algorithm (with both formulations), the ultimate algorithm is relevant without any restriction on the length of the time increment. Therefore, the adaptive time step strategy is not required in this latter case. This point is an additional advantage of the ultimate algorithm, as compared to the return-mapping one.

It should be noted that, despite its high cost in general, the convergence rate of the global Newton–Raphson procedure applied to the return-mapping algorithm remains relatively high, thanks to the analytical computation of the Jacobian matrix. In some cases, such as [11], [15], it is impossible to analytically derive this matrix, and the finite difference method may be used to numerically compute the Jacobian matrix. Such situations obviously lead to an increase in the CPU time. It is worth noting that the derivation of some analytical or numerical Jacobian matrix is not required when the fixed point method is used in the ultimate algorithm.

For all the above-mentioned reasons, the application of the ultimate algorithm, instead of the return-mapping one (with the "LCS" initial guess), allows an interesting reduction in the CPU time. For instance, this CPU time is reduced by a factor ranging between 1.3 and 2.6, when the time increment is equal to 5×10^{-4} s, and a factor ranging between 2.2 and 4.7, when the time increment is equal to 10^{-3} s. These results clearly show that the ultimate algorithm is much more efficient than the return-mapping one.

The evolution of the independent components of the plastic spin w_{p12}, w_{p23} and w_{p13}, corresponding to the loading path described by Eq. (110), is displayed in Figure 10. This evolution can be related to the evolution of the accumulated slip of the active slip systems, reported in Figure 6, through Eq. $(13)_2$. The different curves show that the ultimate algorithm predicts exactly the same evolution of the plastic

spin as the return-mapping algorithm. This result is obvious considering the perfect similarity between the predictions in Figure 6. Due to the crystallographic symmetries, inherent to BCC single crystals, the components $w_{p\,23}$ and $w_{p\,13}$ are always equal to zero for the first three initial crystallographic orientations (orientations # 1, 2 and 3). The abrupt change in the evolution of the components of the plastic spin tensor, observed especially in Figure 10a and Figure 10d, corresponds to the activation of new crystallographic slip systems (systems 10, 16 for the case of orientation #1, and system 2 for orientation #4). These results are confirmed when compared to the evolution of accumulated slip given in Figure 6.

4.2.4. Effect of variation of time increment size on the numerical results

The different numerical simulations in the previous sections have been carried out with constant time increments. These simulations aimed to compare the different algorithms and formulations in terms of accuracy and efficiency. In the current section, the impact of the variation of the size of the time increment on the numerical results is investigated. To this end, both the ultimate and return-mapping algorithms are applied with both constant and variable time increments. Only the semi-smooth formulation is used when the return-mapping algorithm is applied, with a predefined constant guess set to $\Delta\gamma^{(0)}=10^{-2}$. In the various simulations that have been carried out, the velocity gradient given by Eq. (110) is used as loading path, and the final time is fixed to 0.2 s. The value of Δt is fixed to 5×10^{-4} s for the simulations with constant size for the time step. Thus, the number of time steps is equal to 400. The number of time increments for the simulations with variable time steps is also fixed to 400. In order to obtain a random variation for the size Δt in these latter simulations, the following procedure is applied:

- Select randomly 400 values in the interval [0,1].
- Compute the sum of these 400 values. This sum is denoted "sum".
- Multiply each of the 400 values by 0.2/sum. This operation gives, in a straightforward way, the size of each time increment Δt.

This procedure allows generating 400 random values for Δt. Each of them corresponds to a given time increment. In the end, the final time will be equal to 0.2 s, the same as for the simulations with a constant size for the time increment. The evolution of Δt as a function of time t is given in Figure 11.

Figure 12 analyzes the effect of variation of the size of the time increment on the evolution of the diagonal components of the stress tensor $\boldsymbol{\sigma}$. In this figure, only the results corresponding to the single crystal with initial orientation #2 (see Table 2) are presented. The other single crystals, associated with different initial orientations, follow the same trend. Figure 12a shows the evolution of the diagonal components of the Cauchy stress predicted on the basis of the ultimate algorithm. In Figure 12b, the results predicted by the return-mapping algorithm are plotted. By comparing Figure 12a and Figure

12b, we can easily confirm the perfect similarity between the predictions based on the ultimate algorithm and those based on the return-mapping algorithm for both evolutions of the size of the time increment. Figure 12 also shows that the evolution of □11, □22 and □33 is not much influenced by the variation of the size of the time increment, especially in the time interval [0, 0.15]. However, in the second part of time interval, namely [0.15,0.2], the components σ_{22} and σ_{33} are more noticeably affected by variation of Δt. Considering the extreme variation of the value of Δt over the loading history (see Figure 11), this result may be considered as quite acceptable, and does not question the robustness of the two numerical algorithms. Indeed, the variation shown in Figure 11 should be viewed as an extreme case, while in usual applications (based on the CPFEM, for example), Δt may evolve from one increment to another, but much more smoothly than the evolution displayed in Figure 11. As to the CPU time required in these simulations, it is almost independent of the variation of Δt: it is equal to 13 s and 60 s for the simulations with the ultimate algorithm, respectively, the return-mapping algorithm.

4.3. Constant loading 2: Simple shear test

To further assess the robustness of the developed numerical algorithms, a simple shear test is simulated. This simple shear test is defined by the following Eulerian velocity gradient, which is applied to the four single crystals listed in Table 2:

$$\mathbf{g} = \begin{vmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & & \end{vmatrix} (s^{-1}) . \qquad (114)$$

The different simulations of the current section are carried out up to t=0.2 s. The time step Δt is fixed to 5×10^{-4} s. The explicit/implicit version of the ultimate algorithm is used. For the return-mapping algorithm, the semi-smooth formulation with a predefined constant guess option (with $\Delta\gamma^{(0)}=10^{-2}$) is employed. The evolution of the components σ_{11}, σ_{22} and σ_{12} of the stress tensor as a function of time is given in Figure 13 (the other components of $\boldsymbol{\sigma}$ are small as compared to σ_{11}, σ_{22} and σ_{12}). On the whole, the predictions of the two algorithms are very close. However, a very slight difference is observed at the end of the deformation (in the large strain range) for single crystals of orientation #1 and orientation #2. In what follows, we will try to understand the origin of this difference. In this aim, the evolution of the mean stress σ_m (equal to $(1/3)\mathrm{tr}(\boldsymbol{\sigma})$) as function of time is plotted in Figure 14. To evaluate objectively this evolution, the mean stress is normalized (divided) by the von Mises equivalent stress σ_{eq} in Figure 14. The ratio σ_m/σ_{eq} is always equal to 0 for the simulations carried out with the ultimate algorithm. This result is obvious considering the fact that the loading is isochoric, on the one hand, and that the constitutive equations, on which the ultimate algorithm is based, satisfy the incremental incompressibility, on the other hand. However, this ratio is different from zero for the simulations with the return-mapping algorithm, although it remains very small during the deformation (it does not exceed 0.005 for all simulations). This result reveals that the incremental incompressibility is not fully

satisfied by the return-mapping algorithm. This violation is attributable to the formulation of the constitutive equations (32), (33), (45) and (53). Indeed, the trace of the trial elastic Green–Lagrange strain $\mathbf{e}_e^{tr}$ in Eq. (32) is in general different from zero, even when the deformation is isochoric ($\det(\mathbf{f})=1$). As a result, the trace of $\mathbf{T}^{tr}$ in Eq. (33) is also different from zero. Consequently, the trace of $\mathbf{T}$ and $\boldsymbol{\sigma}$ are also different from zero (see Eqs. (45) and (53)). This fact shows that the incremental incompressibility is violated by the return-mapping algorithm. Nevertheless, since the elastic strain is assumed to be sufficiently small for metallic materials as compared to the plastic strain, this violation remains very small and does not affect the robustness and the overall accuracy of this algorithm. This difference in the formulation of the constitutive equations, between the ultimate and return-mapping algorithms, may justify the slight differences observed in Figure 13a and Figure 13b.

On the other hand, the different curves in Figure 13 are smooth showing that the numerical oscillations, observed with other integration schemes, do not appear here, which confirms once again the robustness of both algorithms.

4.4. Non-constant loading

To further assess the efficiency of the two integration algorithms, the single crystal with the initial orientations of Table 2 is subjected to a non-constant loading, characterized by the following Eulerian velocity gradient:

$$\mathbf{g} = \begin{bmatrix} t & (0.33-t) & 0 \\ 0 & t/2 & 0 \\ 0 & 0 & -t/2 \end{bmatrix} (s^{-1}) . \qquad (115)$$

This loading is defined, with an increasing value of time, up to t=0.666 s. The evolution of the strain rate components d_{ij} over the time interval $[0, 0.666]$ is plotted in Figure 15. The size of the time increment Δt is taken equal to 0.00333 s in the following simulations so that the total number of increments is equal to 200.

Figure 16 depicts the evolution of the accumulated slips of the active slip systems for the four different initial crystallographic orientations. The solid lines represent the predictions of the ultimate algorithm, while the dotted graphs correspond to the results obtained with the return-mapping algorithm. It can be observed that both numerical algorithms predict the same activated systems as well as the same evolution of their accumulated slip. Contrary to the case of constant loading, the change of the set of active slip systems is more frequent when the single crystal is subjected to non-constant loading. For example, systems 4, 10, 16, 22 are earlier activated for the orientation # 1, and are then rapidly deactivated for t>0.15 s. Conversely, systems 18 and 36 are inactive at the beginning of the simulation (until t=0.35 s), and become active after this instant.

The change in slip activity of the slip systems is obviously the result of the regular change of the set of potentially active slip systems, which is a direct consequence of the continuous evolution of the velocity gradient. Thus, compared to the case of constant loading, the beneficial impact of the "LCS" strategy is reduced in the present case. Because in practical applications, such as the FE simulation of forming processes or the use of some elaborate micro–macro transition schemes, the loading is generally complex and non-constant, the "LCS" strategy for the initial guess is likely to lose some of its interest, and thus becomes less relevant. As expected, Figure 17a demonstrates that the ultimate algorithm exhibits higher performance when compared to the return-mapping algorithm. More specifically, the CPU time decreases by a factor comprised between 2 and 3 when the ultimate algorithm is applied instead of the return-mapping algorithm combined with the "LCS" initial guess.

Also, for the above two algorithms, the evolution of card($\mathcal{P}$) and card($\mathcal{A}$) during the loading are depicted in Figure 17b, 17c and 17d. These figures clearly reveal the continuous evolution of these sets (in contrast to the results of Figure 8), which may be correlated with the results of Figure 16.

As expected, the number of active slip systems is not always equal to the number of potentially active slip systems when the ultimate algorithm is used (as demonstrated in Figure 17b, 17c and 17d). This implies that several iterations (at least two) for the iterative search procedure are sometimes required. Figure 18 depicts the evolution of the number of iterations during the loading for the different initial orientations. It comes that, over the 200 time increments performed, the iterative search procedure is required for 4% of these time increments. Thus, for most of the time increments, the set of active slip systems is equal to the set of potentially active slip systems and only a single iteration is required. Also, in all cases, the number of iterations per time increment does not exceed 3 iterations, as demonstrated in Figure 18. Note however that, more generally, the number of iterations per time increment may be higher, when the loading path is more abrupt than that applied in Eq. (115), which evolves rather slowly.

5. Polycrystalline aggregate scale: Numerical results and discussions

To further compare the ultimate and return-mapping algorithms in terms of accuracy and efficiency, three representative tests will be simulated in this section for a polycrystalline aggregate. For these different tests, the Taylor model is used to derive the constitutive equations of the polycrystalline aggregate from the constitutive equations of its microscopic constituents (the single crystals). As for the single crystal simulations, the numerical tools developed to simulate the three tests are implemented in the numerical computing environment *Matlab*. However, the novelty here compared to the single crystal simulations, is the use of the *Parallel Computing Toolbox*^{TM} of *Matlab* in order to decrease the CPU time. For each simulation performed at the polycrystalline scale, 4 workers (cores) are simultaneously used. The time increment is fixed to 5×10^{-4} s. In all of the figures that are included in

this section, the solid lines represent the predictions corresponding to the use of the ultimate algorithm, while the dotted graphs correspond to the results obtained with the return-mapping algorithm.

The three different representative tests are sorted in an increasing order of complexity:

- For the first one, the macroscopic velocity gradient is assumed to be known and constant during the deformation.
- In the second test, the polycrystalline aggregate is deformed under plane-stress condition [47]. This condition implies that the component 33 of the macroscopic velocity gradient becomes variable throughout the deformation, whereas the other components are held constant.
- The third test deals with material instability prediction under in-plane biaxial stretching. More specifically, the prediction of localized necking in the polycrystalline aggregate, subjected to some particular proportional strain paths, is carried out using the plastic flow localization criterion based on the initial imperfection approach. The aim of this third test is to illustrate a typical case where more than one component of the macroscopic velocity gradient is variable all along the deformation.

For the simulation of the above-mentioned tests, the following numerical choices are made when the return-mapping algorithm is used:

- ✓ Only the semi-smooth formulation, based on the Fischer–Burmeister complementarity functions, is used.
- ✓ The adaptive time increment explained in Section 4.2.3 is employed.
- ✓ At the beginning of the application of the Newton–Raphson method, corresponding to the resolution of the equations based on Fischer–Burmeister complementarity functions, the trial value of the slip increment is fixed to 10^{-3} for all of the potentially active slip systems.

5.1. Initial texture of the polycrystalline aggregate

The polycrystalline aggregate, which has been introduced above for the simulation of the three representative tests, is made of 100 single crystals. Its initial crystallographic texture is assumed to be orthotropic and is represented by the pole figures of Figure 19. For each single crystal contained in the polycrystalline aggregate, the material parameters as well as the initial values for the Cauchy stress and plastic slip are the same as those used in Section 4.1.

5.2. First test: Constant macroscopic loading

In this first test, the macroscopic velocity gradient is held constant and is defined by the following expression:

$$\mathbf{G} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & -1/2 & 0 \\ 0 & 0 & -1/2 \end{vmatrix} \quad (\mathrm{s}^{-1}). \tag{116}$$

To assess the efficiency of each numerical algorithm at the polycrystalline scale, two simulations are carried out up to a value of 0.2 for the component e_{11} of the logarithmic strain. The first simulation is based on the ultimate algorithm, while the second is based on the return-mapping algorithm. Figure 20a depicts the evolution of the diagonal components of the Cauchy stress tensor (σ_{11}, σ_{22} and σ_{33}) for one of the single crystals (single crystal #1), comprised in the set of single crystals that make up the polycrystalline aggregate, as function of e_{11}. The other stress components (σ_{12}, σ_{23} and σ_{13}) are negligible as compared to the diagonal terms. This first result shows that the application of both numerical algorithms at the single crystal scale leads to the same results. Note that the same observation is made for all the other single crystals composing the aggregate; for conciseness, the associated numerical results are not reported here. Consequently, the components of the macroscopic stress tensor predicted by the two numerical integration algorithms are exactly the same as shown in Figure 20b. We recall that, as the Taylor scheme is used in these simulations, the microscopic component e_{11} is equal to its macroscopic counterpart $E_{11,}$ which in turns is equal to t G_{11}. Figure 20c and Figure 20d depict the {111} pole figures determined at the end of the computations and which are associated with the ultimate algorithm and the return-mapping algorithm, respectively. These pole figures seem to be perfectly similar. This confirms the idea that the two integration algorithms predict exactly the same mechanical responses. These observations are consistent with the results reported in Section 4.2.3. On the other hand, although the predictions are very similar, the CPU time required for running the return-mapping algorithm turns out to be significantly larger than that needed for the application of the ultimate algorithm: 672 s versus 149 s. This result confirms the higher performance of the latter algorithm compared to the former.

5.3. Second test: Macroscopic loading under plane-stress state

In this second test, the polycrystalline aggregate is deformed under plane-stress condition in direction 3 (Σ_{i3}=0, i=1,2,3). This loading is defined by the following mixed boundary conditions:

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1/2 & 0 \\ 0 & 0 & ? \end{bmatrix} (\mathrm{s}^{-1}) \quad ; \quad \mathbf{\Sigma} = \begin{bmatrix} ? & ? & 0 \\ ? & ? & 0 \\ 0 & 0 & 0 \end{bmatrix} (\mathrm{MPa}), \tag{117}$$

where symbol '?' designates the unknown components, which need to be determined at each time step of the simulation. As the crystallographic texture is initially orthotropic (and remains orthotropic during the deformation), the plane-stress conditions imply that the components G_{13}, G_{31}, G_{23} and G_{32} are also

equal to zero. This well-known property (see, e.g., [13], [47], [48]) has been successfully checked from our numerical results. In this test, the component G_{33} is neither constant nor a priori known during the deformation, thus illustrating the complexity of the loading. Indeed, this component is determined iteratively (at each time increment) by imposing the plane-stress condition ($\Sigma_{33}=0$). The numerical details related to the determination of G_{33} can be found in [47].

The evolution of the component G_{33}, as a function of E_{11}, predicted by both integration algorithms is plotted in Figure 21a. It is worth noting that the two algorithms predict the same evolution of G_{33}. In the range of small strains ($E_{11}<0.05$), the component G_{33} varies between −0.21 (which correspond to the value of G_{33} for a purely elastic loading) and −0.5. In the range of large strains ($E_{11}>0.05$), the elastic strain becomes very small compared to the plastic one and, accordingly, G_{33} is almost equal to −0.5. This result is obvious considering the fact that the elastic deformation is very small and that the plastic deformation is incompressible. It must be noted that this plane-stress condition leads to a non-isochoric loading, especially in the range of small strains ($\mathrm{tr}(\mathbf{G})\neq 0$). To further evaluate the difference between the two predictions, the evolution of the ratio G_{33} (UL)/ G_{33} (R-M) in terms of E_{11} is plotted in Figure 21b. Here, G_{33} (UL) (resp. G_{33} (R-M)) refers to the component G_{33} predicted by the ultimate algorithm (resp. the return-mapping algorithm). This figure confirms the result of Figure 21a: the two algorithms predict the same evolution of G_{33} (in spite of very slight oscillations at the beginning of the deformation, which are related to the activation of slip systems). The evolution of the diagonal components of the microscopic stress tensor (σ_{11}, σ_{22} and σ_{33}) for single crystal #1 is plotted in Figure 21c. It can be seen that, contrary to the component 33 of the macroscopic Cauchy stress, the microscopic counterpart does not necessarily satisfy the zero-stress condition. On the whole, several conclusions can be drawn from this figure:

- The two algorithms predict the same stress tensor $\boldsymbol{\sigma}$. This result is consistent with all the above results.
- Although the macroscopic stress component Σ_{33} is equal to zero (as a result of the plane-stress condition), the microscopic stress component σ_{33} is different from zero. In other words, the plane-stress condition is satisfied at the macroscopic scale (the scale of the polycrystalline aggregate) and not at the microscopic scale.
- The evolution of the macroscopic components Σ_{11} and Σ_{22} (the component Σ_{33} being equal to zero) as function of E_{11} is plotted in Figure 21d. The results yielded by the two numerical algorithms are almost indistinguishable.
- Unlike the previous tests, the loading is non-isochoric at the scale of both the single crystal and the polycrystalline aggregate. Indeed, the trace of the macroscopic and microscopic stress tensors is different from zero, as clearly shown in Figure 21c and Figure 21d. Hence, the volume variation due to elasticity is apparent.

- Figure 21c and 21d also demonstrate that both numerical schemes (namely, the ultimate and the return-mapping) do not exhibit pressure oscillations, which are observed in similar situations when other integration schemes are applied. This result illustrates the robustness of both numerical algorithms.

Similar to the case of constant loading, investigated in the first test of Section 5.2, the ultimate algorithm appears to be significantly more efficient, in terms of CPU time, than the return-mapping algorithm, in the context of macroscopic loading. Indeed, 318 s are required to run the simulation of the macroscopic loading under plane-stress state with the ultimate algorithm as compared to 2610 s needed with the return-mapping algorithm.

5.4. Third test: Initial imperfection approach

In this section, four simulations are carried out for the prediction of localized necking in the polycrystalline aggregate subjected to some particular proportional strain paths. The parameters corresponding to these simulations are listed in Table 5.

For all the simulations performed in this section, the initial imperfection ratio ξ_I (see Eq. (F.1)) is fixed to 10^{-3} (which is a typical value for usual applications of the initial imperfection approach).
The main details on the governing equations relating to the initial imperfection approach as well as the corresponding terminology are given in Appendix F. Additional details regarding the numerical implementation of the initial imperfection approach can be found in the literature (see, e.g., [47–49]).
The non-zero components of the macroscopic velocity gradient applied to the safe zone are:

- G^S_{11}, which is constant and equal to 1.
- G^S_{22}, which is constant and equal to ρ.
- G^S_{33}, which evolves rapidly in the range of small strains and becomes almost constant and equal to $-(1+\rho)$ in the range of large strains, as illustrated in Section 5.3 (for $\rho=-1/2$).

Accordingly, except for the beginning of the deformation, the loading in the safe zone becomes almost monotonic and constant. Therefore, it is more interesting to analyze and to follow the evolution of the non-zero components of $\mathbf{G}^B$ during the deformation. This task is carried out, for the different simulations of Table 5, in Figure 22. From the curves of Figure 22, the following observations can be made:

- For the case of Simulation #1, the components G^B_{12} and G^B_{21} remain equal to 0 all along the deformation. This is a natural outcome of the value for the initial band orientation θ_I, taken equal to zero, and of the component $\dot{C}_2$ of the jump vector. Indeed, Eq. (F.5) reveals that the current orientation θ remains equal to 0 and, therefore, the band remains normal to the X_1 direction. Because the component $\dot{C}_2$ is equal to zero, the component G^B_{22} remains constant and equal to

G^S_{22} (which is equal to ρ). The other components of $\mathbf{G}^B$ (G^B_{11} and G^B_{33}) exhibit a non-linear evolution during the deformation. Localized necking is detected when the ratio G^B_{33}/G_{33} increases abruptly (which means that significant localized thinning takes place rapidly in the band as compared to the safe zone). In the present case, localized necking was predicted at a critical strain of $E^S_{11} = 0.282$.

- In contrast to Simulation #1, the components G^B_{12} and G^B_{21} computed in Simulation #2 are different from zero. In this latter case, the components $\dot{C}_1$ and $\dot{C}_2$ of the jump vector, which are initially equal to zero, increase in absolute value during the deformation. The orientation θ of the band evolves also, as its initial value is different from zero. These evolutions for the jump vector and localization band orientation explain the evolution of the components G^B_{12} and G^B_{21}. As to the components G^B_{11} and G^B_{33}, they evolve in a manner very similar to Simulation #1. Localized necking is observed when E^S_{11} reaches a value of 0.23.
- The evolution of the components of $\mathbf{G}^B$ in Simulation #3 follows the same trend as in Simulation #2, but with different magnitudes (the initial band orientation is equal here to 20° instead of 10°). In this case, localized necking is detected a little earlier, namely for a critical limit strain E^S_{11} equal to 0.142.
- In Simulation #4, the initial orientation of the band is equal to 0°, as in the case of Simulation #1. Therefore, the orientation of the band remains equal to zero (in virtue of Eq. (F.5)). On the other hand, the component $\dot{C}_2$ of the jump vector remains equal to zero during the deformation. As a consequence, both G^B_{12} and G^B_{21} are equal to zero, and G^B_{22} is equal to G^S_{22}, which is equal to ρ=0 (plane-strain tensile state). The critical limit strain E^S_{11} corresponding to the occurrence of localized necking is found equal to 0.052.

The results obtained from the above-mentioned four simulations reveal that the application of the initial imperfection approach for the prediction of material instability is a good numerical example for evaluating the efficiency and accuracy of the two numerical algorithms. Indeed, with the initial imperfection approach, the loading is complex and a priori unknown (especially in the localization band). Figure 22 confirms the main conclusions drawn from the numerical results of the tests conducted in Sections 5.2 and 5.3: the ultimate and return-mapping algorithms lead to the same simulation results. Indeed, the evolution of the unknown components of $\mathbf{G}^B$ predicted by the ultimate algorithm is very close to that predicted by the return-mapping algorithm. Consequently, both algorithms predict also the same value of the critical limit strain E^S_{11} corresponding to the occurrence of localized necking. It must be noted that the curves in Figure 22 are not very smooth, due to the relatively small number of single crystals taken to represent the polycrystalline aggregate (100 single crystals).

To assess the efficiency of each numerical algorithm, a comparative study in terms of the CPU time required for each simulation is reported in Table 6. In accordance with the previous tests, the ultimate algorithm appears to be significantly more efficient than the return-mapping algorithm.

6. Validation of the expression of the consistent tangent operator

In this section, the reliability and relevance of the analytical consistent tangent modulus (denoted $\mathcal{L}_{ep}^{A}$ hereafter, for brevity) derived in Section 3.3 are assessed. In this aim, the numerical results obtained by applying $\mathcal{L}_{ep}^{A}$ are compared with those obtained by using the consistent tangent modulus numerically approximated (denoted $\mathcal{L}_{ep}^{N}$ hereafter, for brevity). This $\mathcal{L}_{ep}^{N}$ is considered as reference modulus.

6.1. First test: Deviation between the analytical and the numerical tangent modulus

In this first test, two different velocity gradients are applied as boundary conditions on a BCC single crystal. The time interval on which the integration of the single crystal constitutive relations is carried out is [0, 0.2 s]:

$$\mathbf{g}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1/2 & 0 \\ 0 & 0 & -1/2 \end{bmatrix} (\mathrm{s}^{-1}) \quad ; \qquad = \begin{bmatrix} 1 & 1 & 0 \\ 0 & -1/2 & 0 \\ 0 & 0 & -1/2 \end{bmatrix} (\mathrm{s} \quad) . \tag{118}$$

In both cases, the time increment Δt is fixed to 10^{-3} s.

The material parameters of the studied single crystal are the same as those given in Table 1, while its initial crystallographic orientation (φ_1, φ_2, φ_3) is randomly chosen. The latter is equal to (103.56°, 87.03°, 1.04°). To evaluate the deviation between $\mathcal{L}_{ep}^{A}$ and $\mathcal{L}_{ep}^{N}$, we introduce the relative deviation ratio (RD) defined by the following expression:

$$\mathrm{RD} = \frac{\sqrt{\sum_{m,n,p,q=1}^{3} (\mathcal{L}_{ep\,mnpq}^{N} - \mathcal{L}_{ep\,mnpq}^{A})^2}}{\sqrt{\sum_{m,n,p,q=1}^{3} (\mathcal{L}_{ep\,mnpq})^2}} . \tag{119}$$

The evolution of RD as a function of time t is plotted in Figure 23. This figure demonstrates that RD never exceeds 6.10^{-4} for both simulations. This result gives a first validation of the reliability of the expression of the analytical consistent tangent modulus.

6.2. Second test: Simulation of a uniaxial tensile state

The aim of this second test is to assess whether the slight differences between $\mathcal{L}_{ep}^{A}$ and $\mathcal{L}_{ep}^{N}$, which have been observed in Figure 23, would have any effect on the convergence of a more complex simulation. In this aim, the polycrystalline aggregate defined in Section 5.1 is subjected to a uniaxial tensile test. From a practical point of view, applying this uniaxial tensile test is equivalent to solving the following equation:

$$\dot{\boldsymbol{\Sigma}} - \dot{\boldsymbol{\Sigma}}_0 = \mathbf{0}. \tag{120}$$

This uniaxial tensile test is defined by the following mixed boundary conditions:

$$\dot{\boldsymbol{\Sigma}}_0 = \begin{bmatrix} \alpha & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} (\text{MPa}) \quad ; \quad \mathbf{D} = \begin{bmatrix} 1 & D_{12} & D_{13} \\ D_{12} & D_{22} & D_{23} \\ D_{13} & D_{23} & D_{33} \end{bmatrix} (s^{-1}) \quad ; \quad \mathbf{W} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} (s \ ). \tag{121}$$

As $\mathbf{W} = \mathbf{0}$, the co-rotational frame coincides with the fixed frame throughout the loading.

The macroscopic behavior of the polycrystalline aggregate is determined from the behavior of its microscopic constituents (the single crystals) by using the Taylor model.

To solve Eq. (120), an incremental approach may be adopted. Accordingly, at each time increment $[t_0, t_0+\Delta t]$, this equation is equivalent to

$$\Delta\boldsymbol{\Sigma} - \Delta t \dot{\boldsymbol{\Sigma}}_0 = \mathbf{0}. \tag{122}$$

The unknowns of this incremental problem are: α, D_{22}, D_{33}, D_{12}, D_{13}, D_{23}, which can be stored in an unknown matrix $\mathbf{X}$

$$\mathbf{X} = \begin{vmatrix} \alpha & D_{12} & D_{13} \\ D_{12} & D_{22} & D_{23} \\ D_{13} & D_{23} & D_{33} \end{vmatrix}. \tag{123}$$

Therefore, Eq. (122) can be rewritten as

$$\mathbf{R}(\mathbf{X}) = \Delta\boldsymbol{\Sigma} - \Delta\boldsymbol{\Sigma}_0 = \mathbf{0}. \tag{124}$$

where $\mathbf{R}$ is the residual tensor. Eq. (124) may be iteratively solved by using the Newton–Raphson method. The standard update relation for the unknown matrix $\mathbf{X}$ at iteration (k) results in

$$\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)} + \delta\mathbf{X}^{(k+1)}. \tag{125}$$

where

$$\delta\mathbf{X}^{(k+1)} = -\mathbf{J}^{(k)^{-1}} : \mathbf{R}(\mathbf{X}^{(k)}). \tag{126}$$

The fourth-order Jacobian matrix $\mathbf{J}$ is equal to $\partial\mathbf{R}/\partial\mathbf{X}$ and is defined by the following components:

$$\begin{aligned} &J_{1111} = \frac{\partial R_{11}}{\partial X_{11}} = -\Delta t \quad ; \quad \forall (m,n) \neq (1,1) \quad : \quad J_{mn11} = \frac{\partial \Delta \Sigma_{mn}}{\partial \Delta E_{11}} = 0 \\ &\forall m,n \quad \text{and} \quad \forall (p,q) \neq (1,1) \quad : \quad J_{mnpq} = \frac{\partial R_{mn}}{\partial X_{pq}} = \frac{\partial \Delta \Sigma_{mn}}{\partial \Delta E_{pq}} = \Delta t \frac{\partial \Delta \Sigma_{mn}}{\partial \Delta E_{pq}} = \Delta t \, \mathcal{L}^{*}_{ep\,mnpq} \end{aligned} \quad . \tag{127}$$

where the macroscopic consistent tangent modulus $\mathcal{L}^{*}_{ep}$ is related to its microscopic counterpart $\mathcal{L}_{ep}$ by the following classic averaging relation:

$$\mathcal{L}^{*}_{ep} = \frac{1}{V} \int_{V} \mathcal{L}_{ep} \, dV \, , \tag{128}$$

where V is the volume of the polycrystalline aggregate.

Therefore, in order to solve with the Newton–Raphson method the non-linear system induced by the uniaxial tensile state, the microscopic consistent tangent modulus $\mathcal{L}_{ep}$ should be calculated. For comparison purposes, both the analytical and numerical versions of the consistent tangent modulus are used to perform the simulation of the uniaxial tensile test. The number of iterations of the Newton–Raphson method, averaged by the number of increments required in the simulation performed with the analytical (resp. numerical) consistent tangent modulus, is equal to 2.21 (resp. 2.135). It is noteworthy that, in terms of CPU time, the computations carried out by using the numerical consistent tangent modulus take 1870.75s, whereas those performed with the analytical consistent tangent modulus require 276.85s only. Considering that the constitutive equations for the single crystal are integrated seven times (resp. once) at each Newton–Raphson iteration when the numerical (resp. analytical) tangent modulus is used, this result is quite expectable. It confirms the efficiency of the analytical approach for the consistent tangent modulus.

The comparison between the simulation results obtained by using the analytical and numerical consistent tangent moduli is given in Figure 24. This figure shows that the use of one form or the other for the consistent tangent modulus leads to the same numerical predictions.

7. Conclusions

In this paper, two novel integration algorithms for predicting the mechanical response of BCC single crystals, within the framework of large strain rate-independent plasticity, have been developed and compared. The first algorithm is implicit and based on the well-known return-mapping scheme. In this algorithm, the non-smooth Kuhn–Tucker inequalities and constraints are replaced by a set of non-linear equations involving the so-called Fischer–Burmeister complementarity functions. This new semi-smooth formulation allows combining both tasks, namely the identification of the set of active slip systems and the determination of the slip increments of the resulting active slip systems. To solve the associated non-linear system, a global Newton–Raphson procedure based on the line search technique has been developed and used. The second approach is based on the ultimate algorithm. The latter is

defined by one or two stages (depending on the situation): an explicit phase (automatically achieved), followed by an implicit correction (performed under a condition on the size of the time increment). This algorithm can be easily transformed into an explicit scheme, by avoiding the application of the implicit correction. In this second algorithm, the determination of the set of active slip systems is based on an iterative search procedure, while the computation of the corresponding slip rates is based on the fixed point method.

The analysis of the simulation results for single crystals with various initial crystallographic orientations and subjected to monotonic and complex loadings allows the following conclusions to be drawn:

- The use of the semi-smooth formulation of the return-mapping algorithm, instead of the classical formulation, permits to avoid several problems, especially those related to the identification of the active slip systems. Indeed, no iterative search procedure, for the determination of the active slip systems, is required when the semi-smooth formulation is applied. Accordingly, all discussions on the choice and the development of such search procedures are obviously avoided here.
- The use of the semi-smooth formulation of the return-mapping algorithm involves too much iteration for reaching the solution; therefore, significant computational resources are required, especially when the number of potentially active slip systems is much larger than that of active slip systems. Additionally, the non-linear treatment of this algorithm is based on an elaborate line search procedure, which improves the global Newton–Raphson iterative method.
- In the case of the ultimate algorithm, the iterative search procedure, for the determination of the set of active slip systems, turns out to be a reliable and efficient procedure because it converges in little iteration. Moreover, the fixed point method used to compute the slip rates of the active slip systems appears to be very efficient.
- The various simulations for the single crystal response, performed by using the ultimate algorithm, clearly demonstrate the decrease in computational cost by a factor of 1.5–5 as compared to the integration scheme based on the return-mapping algorithm. This factor depends on the initial crystallographic orientation and, especially, on the size of the adopted time increment.
- The proposed ultimate algorithm provides a computationally efficient and robust semi-implicit integration scheme for the time integration of crystal plasticity constitutive models. This new algorithm shows more benefits for potential applications in materials science, and the associated large-scale simulations of forming processes, as compared to the return-mapping algorithm.

To further assess the respective efficiency of the ultimate and return-mapping algorithms, three typical mechanical tests have been simulated for a polycrystalline aggregate. These mechanical tests range from constant and a priori known loading, in the first test, to complex unknown loading in the third test.

For the different tests, the Taylor model is used to derive the mechanical behavior of the polycrystalline aggregate from the constitutive equations of its microscopic constituents. The above conclusions, drawn at the single crystal scale, are confirmed through the simulation results obtained at the polycrystalline scale. Indeed, the numerical results given by the polycrystalline simulations, and corresponding to the two algorithms used to integrate the constitutive equations at the single crystal scale, turn out to be indistinguishable. Despite the perfect similarity of the numerical results, the simulations based on the return-mapping algorithm are significantly more expensive than those based on the ultimate algorithm. Indeed, the ratio of the CPU time required for the simulations with the return-mapping algorithm to that associated with the ultimate algorithm ranges between 4.5 and 8.3 in the case of these polycrystalline simulations.

On the basis of these different observations and results, the ultimate algorithm may be considered as a relevant numerical scheme for the integration of the constitutive equations of rate-independent single crystal models, thanks to its performance in terms of robustness, accuracy and efficiency. It can then be successfully used in "large scale" simulation codes, based on the finite element method or other discretization methods (e.g., multiscale transition schemes), to simulate the behavior of metallic components and structures.

Appendix A. Slip systems for BCC single crystals

For BCC single crystals, the number of crystallographic slip systems N_s is equal to 24. Each slip system is characterized by two orthogonal vectors $(\vec{\mathbf{m}}_\alpha, \vec{\mathbf{n}}_\alpha)$ in the deformed configuration. $\vec{\mathbf{m}}_\alpha$ is the vector parallel to the slip line and $\vec{\mathbf{n}}_\alpha$ is the vector normal to the slip plane. In the intermediate configuration, both vectors are assumed to be constant and of unit length. Therefore, the following relations hold:

$$\forall \alpha = 1,...,N_s: \quad \left\|\vec{\mathbf{m}}_\alpha^0\right\| = \left\|\vec{\mathbf{n}}_\alpha^0\right\| = 1 \quad , \quad \vec{\mathbf{m}}_\alpha^0 . \vec{\mathbf{n}}_\alpha^0 = 0 . \tag{A.1}$$

Let us introduce the vectors $\vec{\mathbf{w}}_\alpha^0$ and $\vec{\mathbf{v}}_\alpha^0$ enumerated in Table A.1.

α	1	2	3	4	5	6	7	8
$\vec{\mathbf{v}}_\alpha^0$	<1 1 1>	<1 1 1>	<1 1 1>	<1 1 1>	<1 1 1>	<1 1 1>	<–1 1 1>	<–1 1 1>
$\vec{\mathbf{w}}_\alpha^0$	[1 –1 0]	[1 0 –1]	[0 1 –1]	[1 1 –2]	[1 –2 1]	[–2 1 1]	[1 1 0]	[1 0 1]

α	9	10	11	12	13	14	15	16
$\vec{\mathbf{v}}_\alpha^0$	<–1 1 1>	<–1 1 1>	<–1 1 1>	<–1 1 1>	<1 –1 1>	<1 –1 1>	<1 –1 1>	<1 –1 1>
$\vec{\mathbf{w}}_\alpha^0$	[0 1 –1]	[1 –1 2]	[1 2 –1]	[2 1 1]	[1 1 0]	[1 0 –1]	[0 1 1]	[–1 1 2]

α	17	18	19	20	21	22	23	24
$\vec{\mathbf{v}}_\alpha^0$	<1 –1 1>	<1 –1 1>	<1 1 –1>	<1 1 –1>	<1 1 –1>	<1 1 –1>	<1 1 –1>	<1 1 –1>
$\vec{\mathbf{w}}_\alpha^0$	[1 2 1]	[2 1 –1]	[1 –1 0]	[1 0 1]	[0 1 1]	[1 1 2]	[–1 2 1]	[2 –1 1]

Table A.1. The numbering of the slip systems of a BCC single crystal according to [50].

Then, $\vec{\mathbf{m}}_\alpha^0$ and $\vec{\mathbf{n}}_\alpha^0$ are defined in terms of $\vec{\mathbf{w}}_\alpha^0$ and $\vec{\mathbf{v}}_\alpha^0$ by the following relations:

$$\forall \alpha = 1,...,N_s: \quad \vec{\mathbf{m}}_\alpha^0 = \frac{\vec{\mathbf{w}}_\alpha^0}{\left\|\vec{\mathbf{w}}_\alpha^0\right\|} \quad ; \quad \vec{\mathbf{n}} = \frac{\vec{\mathbf{v}}}{\left\|\vec{\mathbf{v}}\right\|} . \tag{A.2}$$

Appendix B. The global Newton–Raphson method

The classical Newton–Raphson method is adopted to solve Eq. (51). The standard update relation of the slip increment at iteration (k) results in

$$\Delta\boldsymbol{\gamma}^{(k+1)} = \Delta\boldsymbol{\gamma}^{(k)} + \delta\Delta\boldsymbol{\gamma}^{(k)}, \tag{B.1}$$

where

$$\delta\Delta\boldsymbol{\gamma}^{(k)} = -\boldsymbol{\mathcal{D}}^{(k)^{-1}}\boldsymbol{\varphi}^{(k)}. \tag{B.2}$$

Here, $\boldsymbol{\mathcal{D}}^{(k)}$ is the kth iteration of the Jacobian matrix defined as $(\partial\boldsymbol{\varphi}/\partial\Delta\boldsymbol{\gamma})$. The $\alpha\beta^{th}$ component of $\boldsymbol{\mathcal{D}}$ is obtained as

$$\forall\alpha,\beta\in\mathcal{P}: \quad \mathcal{D}_{\alpha\beta} = \delta_{\alpha\beta}\left(\frac{\Delta\gamma_\alpha}{\sqrt{(A_{\alpha\delta}\Delta\gamma_\delta - b_\alpha)^2 + (\Delta\gamma_\alpha)^2}} - 1\right) + \frac{\partial f_\alpha}{\partial\Delta\gamma_\beta}\left(\frac{\alpha}{\sqrt{(A_{\alpha\delta}\Delta\gamma_\delta - b_\alpha)^2 + (\Delta\gamma_\alpha)^2}} + 1\right) \quad ; \quad \delta\in\mathcal{P} \tag{B.3}$$

in which $\partial f_\alpha / \partial\Delta\gamma_\beta$ is defined by the following relation:

$$\forall\alpha,\beta\in\mathcal{P}: \quad \frac{\partial f_\alpha}{\partial\Delta\gamma_\beta} = h_{\alpha\beta}(t_0+\Delta t) + \frac{h_{\alpha\delta}}{\Delta\gamma_\beta}(t_0+\Delta t)\Delta\gamma_\delta - \mathbf{M}^0_\alpha : \boldsymbol{\mathcal{L}}_e : \mathrm{sym}\left(\mathbf{c}_e^{tr}(t_0+\Delta t).\mathbf{M}^0_\beta\right) \quad ; \quad \delta\in\mathcal{P}. \tag{B.4}$$

The algorithm corresponding to the classical Newton–Raphson method is defined by the following three steps:

- ✓ Step 1:
  - Initialize the iteration index to 0: $k = 0$
  - Choose the initial guess $\Delta\boldsymbol{\gamma}^{(0)}$

For ($k \geq 1$)

- ✓ Step 2:
  - Compute $\boldsymbol{\varphi}^{(k)}$ and $\boldsymbol{\mathcal{D}}^{(k)}$
  - If $\left\|\boldsymbol{\varphi}^{(k)}\right\| = \sqrt{\sum_{\alpha\in\mathcal{P}}(\varphi_\alpha^{(k)})^2} \leq$ tol , then stop; else, go to Step 3
- ✓ Step 3:
  - Compute $\delta\Delta\boldsymbol{\gamma}^{(k)} = -\boldsymbol{\mathcal{D}}^{(k)^{-1}}\boldsymbol{\varphi}^{(k)}$
  - Compute $\Delta\boldsymbol{\gamma}^{(k+1)}$ $\Delta\boldsymbol{\gamma}^{(k)} + \delta\Delta\boldsymbol{\gamma}^{(k)}$, set $k \leftarrow k+1$ and go to Step 2

Note that the problem of non-uniqueness of the slip increments is still present with this semi-smooth formulation. Indeed, the Jacobian matrix $\boldsymbol{\mathcal{D}}$ of Eq. (B.3) may be singular, in which case it has to be

inverted using the pseudo-inversion technique as in [11], or the perturbation technique as in [12], for example. In the current work, the pseudo-inversion technique is used.

Unfortunately, the classical Newton–Raphson algorithm, defined by the previous three steps, is sometimes confronted to some numerical difficulties, such as the choice of the initial guess $\Delta\boldsymbol{\gamma}^{(0)}$ in order to guarantee a fast and correct convergence. Indeed, when the initial guess is relatively far from the solution, the method might not converge or may converge very slowly. To avoid this shortcoming, a line search strategy leading to a global Newton–Raphson method can be applied. This strategy is based on the limitation of the iteration length, by multiplying $\delta\Delta\boldsymbol{\gamma}^{(k)}$ in Eq. (B.1) by a scalar α comprised between 0 and 1. The aim of the line search strategy is to find α such that $\left\|\boldsymbol{\varphi}^{(k+1)}\right\| = \left\|\boldsymbol{\varphi}\ \Delta\boldsymbol{\gamma}^{(k+1)})\right\| = \left\|\boldsymbol{\varphi}(\Delta\boldsymbol{\gamma}^{(k)} + \alpha\delta\Delta\boldsymbol{\gamma}^{(k)})\right\|$ has sufficiently decreased. Several algorithms are available in the literature to compute α, such as the Armijo algorithm [51] or the algorithm developed in [52]. After several comparative tests, a relatively simple algorithm is followed. This algorithm is iteratively applied at each iteration of the global Newton–Raphson procedure. It is summarized by the following steps:

- ✓ Step 1:
  - Choose $\lambda \in [0, 0.5]$ and $\rho \in [0, 1]$
  - Set the iteration counter j to 0 and the first iteration $\alpha^{(0)}$ of α to 1
- ✓ Step 2:
  - Compute $\Delta\boldsymbol{\gamma}^{(k+1)} = \Delta\boldsymbol{\gamma}^{(k)} + \alpha^{(0)}\delta\Delta\boldsymbol{\gamma}^{(k)}$
- ✓ Step 3:
  - While $\left\|\boldsymbol{\varphi}^{(k+1)}\right\| > \left\|\boldsymbol{\varphi}(\Delta\boldsymbol{\gamma}^{(k)})\right\| \quad \lambda\ \alpha^{(j)}\left\|\boldsymbol{\varphi}(\Delta\ ^{(k)})\right\|^2$ do:
    - $\alpha^{(j+1)} = \rho\ \alpha^{(j)}$
    - $\Delta\boldsymbol{\gamma}^{(k+1)} = \Delta\boldsymbol{\gamma}^{(k)} + \alpha^{(j+1)}\delta\Delta\boldsymbol{\gamma}^{(k)}$
    - j=j+1

In the current paper, the numerical parameters λ and ρ are assumed to be equal to 10^{-4} and 0.5, respectively.

The algorithm for the determination of α is embedded in the Newton–Raphson algorithm explained in the beginning of this Appendix.

For the simulations performed in this paper, the partial derivatives $\partial h_{\alpha\delta} / \partial\Delta\gamma_\beta$ (where $h_{\alpha\delta}$ are the hardening components of the hardening law (109)) need to be computed in order to determine the components of the Jacobian matrix $\mathcal{D}$ (see Eq. (B.3))

$$\forall\ \alpha,\beta,\delta = 1,...,24: \quad \frac{\partial h_{\alpha\delta}}{\partial\Delta\gamma_\beta} = \frac{\partial\hat{h}(A)}{\Delta\gamma}\Big[q+(1-q)\,\delta_{\alpha\delta} \quad , \tag{B.5}$$

where $\partial\hat{h}/\partial\Delta\gamma_\beta$ is given by the following relation:

$$\forall \beta = 1,...,24: \quad \frac{\partial \hat{h}(A)}{\partial \Delta\gamma_\beta} = -\frac{2h_0 \hat{h}(A)}{\tau_{sat} - \tau_0} \tanh\left(\frac{h\ A}{\tau_{sat} - \tau_0} \right. \quad . \tag{B.6}$$

Appendix C. The pseudo-inversion method

A general square matrix $\mathbf{A}$ can be decomposed, by using the singular value decomposition [53], into the following form:

$$\mathbf{A} = \mathbf{U}^{\mathrm{T}}.\mathbf{S}.\mathbf{V}, \tag{C.1}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices. In the above decomposition, $\mathbf{S}$ is a diagonal matrix containing the singular values of $\mathbf{A}$. The pseudo-inverse of $\mathbf{A}$ is given by

$$\mathbf{A}^{+} = \mathbf{V}^{\mathrm{T}}.\mathbf{S}^{+}.\mathbf{U}, \tag{C.2}$$

in which $\mathbf{S}^{+}$ is also a diagonal matrix defined by

$$S_{ii}^{+} = \begin{cases} 1/S_{ii} \text{ if} & S_{ii} > 0 \\ 0 & \text{otherwise} \end{cases}. \tag{C.3}$$

When $\mathbf{A}$ is invertible, its pseudo-inverse simply reduces to its classical regular inverse.

Appendix D. The fixed point method

The system of Eq. (74) is strongly non-linear and its resolution is based on the iterative fixed point method. The application of this method involves the following sub-steps:

✓ Sub-step 3.1:

- Initialize the iteration index k to 0: k=0.

- Initialize the first iteration $\dot{\gamma}_\beta^{(0)}(t_0+\delta t)$ for the slip rates at $t_0+\delta t$ to the explicit solution computed by Step 1.

- Initialize the first iteration for the rotation $\mathbf{r}^{(0)}(t_0+\delta t)$ and for the hardening modulus $\mathbf{h}^{(0)}(t_0+\delta t)$ to their respective values at t_0.

- Set k=1.

For ($k \geq 1$), do the following sub-steps:

✓ Sub-step 3.2:

- Compute $\mathbf{r}^{(k)}(t_0+\delta t)$ as follows:

$$^{(k)}(t_0+\delta t) \quad \mathbf{r}(t\ ).e^{\delta t\, \underline{\mathbf{w}}_e^{(k-1)}}, \tag{D.1}$$

with

$$\underline{\mathbf{w}}_e^{(k-1)} = \underline{\mathbf{w}}^{(k-1)} - \underline{\mathbf{w}}_p^{(k-1)} = \mathbf{r}^{(k-1)T}(t_0+\delta t).\,\mathbf{w}.\,\mathbf{r}^{(k-1)}(t_0+\delta t) - \sum_{\beta\in\mathcal{A}} \dot{\gamma}_\beta^{(k-1)}(t_0+\delta t)\mathbf{S}_\beta^0 . \tag{D.2}$$

The above equations are the result of the implicit integration of Eq. (16) over I^δ.

- Compute the hardening modulus $h_{\alpha\beta}^{(k)}(t_0+\delta t)$ $(\alpha,\beta\in\mathcal{A})$ on the basis of $\gamma^{(k-1)}(t_0+\delta t)$, which is computed by the following relation:

$$\gamma_\beta^{(k\ 1)}(t_0+\delta t) = \gamma_\beta(t_0) + \delta t\, \dot{\gamma}_\beta^{(k-1)} . \tag{D.3}$$

- Compute $\underline{\mathbf{d}}^{(k)}$ by

$$\underline{\mathbf{d}}^{(k)} = \mathbf{r}^{(k)T}.\,\mathbf{d}.\,\mathbf{r}^{(k)} . \tag{D.4}$$

- Compute $\tilde{A}_{\alpha\beta}^{(k)}(t_0+\delta t)$ and $\tilde{b}^{(k)}(t_0+\delta t)$ by

$$\forall \alpha,\beta \quad \mathcal{A}: \quad \begin{cases} \tilde{A}_{\alpha\beta}^{(k)}(t_0+\delta t) = \mathbf{M}_\alpha^0 : \mathcal{L}_e : \ {}_\beta^0 + h_{\alpha\beta}^{(k)}(t_0+\delta t) \\ \tilde{b}_\alpha^{(k)}(t_0+\delta t) = \mathbf{M}_\alpha^0 : \mathcal{L}_e : _^{(k)}(t_0+\delta t) \end{cases} . \tag{D.5}$$

- Compute $\dot{\gamma}^{(k)}(t_0+\delta t)$ $(\beta\in\mathcal{A})$ by

$$\dot{\boldsymbol{\gamma}}^{(k)}(t_0+\delta t)=\tilde{\mathbf{A}}^{(k)-1}(t_0+\delta t).\,\tilde{\mathbf{b}}^{(k)}(t_0+\delta t)\,. \tag{D.6}$$

✓ Sub-step 3.3:

If $\left\|\dot{\boldsymbol{\gamma}}^{(k)}(t_0+\delta t)-\dot{\boldsymbol{\gamma}}^{(k-1)}(t\ +\delta t)\right\|<\varepsilon$, then $\dot{\boldsymbol{\gamma}}^{(k)}(t_0+\delta t)$ is the solution of (74); otherwise, set $k\leftarrow k+1$ and go to Sub-step 3.2. Here, ε is typically chosen to be equal to $10^{-5}\ s^{-1}$.

If the matrix $\tilde{\mathbf{A}}^{(k)}(t_0+\delta t)$ is singular, then the pseudo-inversion method is used to compute its inverse and then to solve Eq. (74).

Appendix E. Computation of matrix $\hat{\mathcal{L}}_s^*$

As discussed in Section 3.3, this part is devoted to the derivation of the fourth-order modulus $\hat{\mathcal{L}}_s^*$, which has been introduced in the above-mentioned section. It is noteworthy that all of the notations previously defined in that section are adopted here again. We recall that $\hat{\mathcal{L}}_s^*$ satisfies the following relation:

$$\hat{\mathcal{L}}_s^* : \partial\Delta\hat{\boldsymbol{\varepsilon}} = \partial\hat{\mathcal{L}} : \Delta\hat{\boldsymbol{\varepsilon}}\,, \tag{E.1}$$

where $\hat{\mathcal{L}}_s$ is given by the following relation:

$$\hat{\mathcal{L}}_s = [\mathcal{L} - \Lambda_{\alpha\beta}(\mathcal{L}_e : \hat{\mathbf{R}}_\alpha + \hat{\mathbf{S}}_\alpha.\hat{\boldsymbol{\sigma}}(t_0) - \hat{\boldsymbol{\sigma}}(t_0).\hat{\mathbf{S}}_\alpha) \otimes (\hat{\mathbf{R}}_\beta : \mathcal{L}_e)] \quad ; \quad \alpha,\beta \in \mathcal{A}\,. \tag{E.2}$$

By differentiating Eq. (E.2), one obtains

$$\begin{aligned} \partial\hat{\mathcal{L}}_s : \Delta\hat{\boldsymbol{\varepsilon}} = -\Delta t[&(\hat{\mathbf{R}}_\beta : \mathcal{L}_e : \hat{\mathbf{d}})(\mathcal{L}_e : \hat{\mathbf{R}}_\alpha + \hat{\mathbf{S}}_\alpha.\hat{\boldsymbol{\sigma}}(t_0) - \hat{\boldsymbol{\sigma}}(t_0).\hat{\mathbf{S}}_\alpha)\partial\Lambda_{\alpha\beta} \\ &+ \Lambda_{\alpha\beta}(\hat{\mathbf{R}}_\beta : \mathcal{L}_e : \hat{\mathbf{d}})(\mathcal{L}_e : \partial\hat{\mathbf{R}}_\alpha + \partial\hat{\mathbf{S}}_\alpha.\hat{\boldsymbol{\sigma}}(t_0) - \boldsymbol{\sigma}(t_0).\partial\mathbf{S}_\alpha) \quad ; \quad \alpha,\beta \in \mathcal{A}\,. \\ &+ \Lambda_{\alpha\beta}(\mathcal{L}_e : \hat{\mathbf{R}}_\alpha + \hat{}_\alpha.\hat{\boldsymbol{\sigma}}(t_0) - \boldsymbol{\sigma}(t_0).\mathbf{S}_\alpha)((\mathcal{L}_e : \hat{\mathbf{d}}) : \partial\hat{\mathbf{R}}_\beta)] \end{aligned} \tag{E.3}$$

In order to determine the analytical expression of $\hat{\mathcal{L}}_s^*$, the differential terms $\partial\Lambda_{\alpha\beta}$, $\partial\hat{\mathbf{R}}_\alpha$ and $\partial\hat{\mathbf{S}}_\alpha$ should be expressed as functions of the differential of the strain increment $\partial\Delta\hat{\boldsymbol{\varepsilon}}$. This is the purpose of the following developments.

In order to compute $\partial\Lambda$, the following relation is used:

$$\forall\alpha,\beta \in \mathcal{A}: \quad \Lambda_{\alpha\lambda}(\ {}_\lambda : \mathcal{L}_e : \hat{\mathbf{R}}_\beta + h_{\lambda\beta}) = \delta_{\alpha\beta} \quad ; \quad \lambda \in \mathcal{A}\,. \tag{E.4}$$

The differentiation of Eq. (E.4) leads to

$$\forall\alpha,\beta \in \mathcal{A}: \quad \partial\Lambda_{\alpha\lambda}(\ {}_\lambda : \mathcal{L}_e : \hat{\mathbf{R}}_\beta + h_{\lambda\beta}) + \Lambda_{\alpha\lambda}\partial h_{\lambda\beta} = 0 \quad ; \quad \lambda \in \mathcal{A}\,. \tag{E.5}$$

It is worth noting that the term $\hat{\mathbf{R}}_\lambda : \mathcal{L}_e\ {}_\beta$ is a scalar invariant, and hence Eq. (E.5) is equivalent to

$$\forall\alpha,\beta \in \mathcal{A}: \quad \partial\Lambda_{\alpha\beta} = -\Lambda_{\alpha\lambda}\partial h_{\lambda\zeta}\Lambda_{\zeta\beta} \quad ; \quad \lambda,\zeta \in \mathcal{A}\,. \tag{E.6}$$

Besides, the components of the hardening matrix are functions of the cumulated slip

$$\forall\,\lambda,\zeta \in \mathcal{A}: \quad \partial h_{\lambda\zeta} = \Delta t\frac{\partial h_{\lambda\zeta}}{\partial\gamma_\alpha}\partial\dot{\gamma}_\alpha \quad ; \quad \alpha \in \mathcal{A}, \tag{E.7}$$

where γ_α is the cumulated slip on slip system α.

Thereafter, the differentials of the symmetric parts of the Schmid orientation tensors are derived through the following relations:

$$\forall \alpha = 1,...,2N_s: \quad \partial \hat{\mathbf{R}}_\alpha = \partial(\underset{\sim}{\mathbf{r}}.\mathbf{R}^0.\underset{\sim}{\mathbf{r}}^T) = \partial\underset{\sim}{\mathbf{r}}.\mathbf{R}^0.\underset{\sim}{\mathbf{r}}^T + \underset{\sim}{\mathbf{r}}.\mathbf{R}^0_\alpha.(\partial\underset{\sim}{\mathbf{r}})^T \tag{E.8}$$

Let us introduce two fourth-order tensor operators denoted $\mathbf{A}^1$ and $\mathbf{A}^2$, respectively, such that for all second-order tensors $\mathbf{a}$ and $\mathbf{b}$,

$$\begin{cases} \mathbf{b}.\mathbf{a} = \mathbf{A}^1(\mathbf{a}) \ \mathbf{b} \\ \mathbf{a}.\mathbf{b} = \mathbf{A}^2(\mathbf{a}) \ \mathbf{b} \end{cases}. \tag{E.9}$$

The analytical expressions of $\mathbf{A}^1(\mathbf{a})$ and $\mathbf{A}^2(\mathbf{a})$ are given by the following relations:

$$\forall i, j, k, l \quad 1,2,3: \quad \begin{cases} A^1_{ijkl}(\mathbf{a}) = a_{lj}\delta_{ik} \\ A^2_{ijkl}(\mathbf{a}) = a_{il}\delta_{jk} \end{cases}. \tag{E.10}$$

Using these tensor operators, Eq. (E.8) can then be expressed in the following form:

$$\begin{aligned} \forall \alpha \in \mathcal{A}: \quad \partial \hat{\mathbf{R}}_\alpha &= \mathbf{A}^1(\mathbf{R}^0_\alpha.\underset{\sim}{\mathbf{r}}^T) : \partial\underset{\sim}{\mathbf{r}} + \mathbf{A}^2(\underset{\sim}{\mathbf{r}}.\mathbf{R}^0_\alpha) : \partial\underset{\sim}{\mathbf{r}} \\ &= \left(\mathbf{A}^1(\mathbf{R}^0.\underset{\sim}{\mathbf{r}}^T) + \mathbf{A}^2(\underset{\sim}{\mathbf{r}}.\mathbf{R}^0_\alpha)\right) : \partial\underset{\sim}{\mathbf{r}}, \\ &= \mathbf{B}(\mathbf{R}^0_\alpha \ : \partial\underset{\sim}{\mathbf{r}} \end{aligned} \tag{E.11}$$

where $\mathbf{B}$ is a fourth-order tensor operator, such that for any second-order tensor $\mathbf{a}$,

$$\mathbf{B}(\mathbf{a}) = \mathbf{A}^1(\mathbf{a}.\underset{\sim}{\mathbf{r}}^T) + \mathbf{A}^2(\underset{\sim}{\mathbf{r}}.\mathbf{a}). \tag{E.12}$$

Similarly, the differentials of the anti-symmetric parts of the Schmid orientation tensors read

$$\forall \alpha \in \mathcal{A}: \quad \partial \hat{\mathbf{S}} \ = \mathbf{B}(\mathbf{S}^0_\alpha) : \partial\underset{\sim}{\mathbf{r}}. \tag{E.13}$$

Next comes the determination of the differential of the rotation $\partial\underset{\sim}{\mathbf{r}}$. We recall the evolution law for the rotation $\underset{\sim}{\mathbf{r}}$

$$\underset{\sim}{\mathbf{r}} = e^{-\Delta t \hat{\mathbf{w}}_p}.\underset{\sim}{\mathbf{r}}(t_0) \approx [\mathbf{1} - \Delta t \dot{\gamma}_\alpha \hat{\mathbf{S}}_\alpha].\underset{\sim}{\mathbf{r}}(t_0) \quad ; \quad \alpha \in \mathcal{A}. \tag{E.14}$$

The differentiation of Eq. (E.14) leads to

$$\begin{aligned} \partial\underset{\sim}{\mathbf{r}} &= -\Delta t[\partial\dot{\gamma}_\alpha \mathbf{S}_\alpha + \dot{\gamma}_\alpha \partial\hat{\mathbf{S}}_\alpha].\underset{\sim}{\mathbf{r}}(t_0) \\ &= -\Delta t \mathbf{A}^1(\underset{\sim}{\mathbf{r}}(t_0)) : (\partial\dot{\gamma}_\alpha \hat{\mathbf{S}}_\alpha + \dot{\gamma}_\alpha \partial\mathbf{S}_\alpha) \end{aligned} \quad ; \quad \alpha \in \mathcal{A}. \tag{E.15}$$

By inserting Eq. (E.13) into Eq. (E.15), one obtains

$$\begin{aligned} \partial\underset{\sim}{\mathbf{r}} &= -\Delta t \partial\dot{\gamma}_\alpha \mathbf{A}^1(\underset{\sim}{\mathbf{r}}(t_0)) : \hat{\mathbf{S}}_\alpha - \Delta t \dot{\gamma}_\alpha \mathbf{A}^1(\underset{\sim}{\mathbf{r}}(t_0)) : \mathbf{B}(\mathbf{S}^0_\alpha) : \partial\underset{\sim}{\mathbf{r}} \\ &= -[\mathbf{I} + \Delta t \dot{\gamma}_\beta \mathbf{A}^1(\underset{\sim}{\mathbf{r}}(t_0)) : \mathbf{B}(\mathbf{S}^0_\beta)]^{-1} : [\mathbf{A}^1(\underset{\sim}{\mathbf{r}}(t_0)) : \hat{\mathbf{S}}_\alpha] \Delta t \partial\dot{\gamma}_\alpha \quad ; \quad \alpha,\beta \in \mathcal{A}, \\ &= -\mathbf{D}_\alpha \Delta t \partial\dot{\gamma} \end{aligned} \tag{E.16}$$

where $\mathbf{I}$ is the fourth-order unit tensor, and $\mathbf{D}_\alpha$ the second-order tensor associated with slip system α and defined by the following relation:

$$\forall \alpha \in \mathcal{A}: \quad \mathbf{D}_\alpha = [\mathbf{I} + \Delta t \dot{\gamma}_\beta \mathbf{A}^1(\underset{\sim}{\mathbf{r}}(t_0)) : \mathbf{B}(\mathbf{S}^0_\beta)]^{-1} : [\mathbf{A}^1(\underset{\sim}{\mathbf{r}}(t_0)) : \hat{\mathbf{S}}_\alpha] \quad ; \quad \beta \in \mathcal{A}. \tag{E.17}$$

Using Eqs. (E.7), (E.11) and (E.16), the differentials of the slip rates can be expressed in terms of the differential of the strain increment.

Indeed, Eq. (101) shows that the slip rates satisfy the following relation:

$$\forall \alpha \in \mathcal{A}: \quad (\hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \hat{\mathbf{R}}_\beta + h_{\alpha\beta})\dot{\gamma}_\beta = \hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \hat{\mathbf{d}} \quad ; \quad \beta \in \mathcal{A}. \tag{E.18}$$

The differentiation of Eq. (E.18) leads to

$$\forall \alpha \in \mathcal{A}: \quad \partial h_{\alpha\beta}\,\dot{\gamma}_\beta + (\hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \hat{\mathbf{R}}_\beta + h_{\alpha\beta})\partial\dot{\gamma}_\beta = \partial\hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \hat{\mathbf{d}} + \hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \partial\hat{\mathbf{d}} \quad ; \quad \beta \in \mathcal{A}. \tag{E.19}$$

By inserting Eqs. (E.7) and (E.11) into Eq. (E.19), one obtains

$$\forall \alpha \in \mathcal{A}: \quad (\hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \hat{\mathbf{R}}_\beta + h_{\alpha\beta} + \frac{\partial h_{\alpha\lambda}}{\partial\gamma_\beta}\Delta t\,\dot{\gamma}_\lambda)\partial\dot{\gamma}_\beta = (\boldsymbol{\mathcal{L}}_e : \hat{\mathbf{d}}) : \mathbf{B}(\mathbf{R}_\alpha^0) : \partial\underset{\sim}{\mathbf{r}} + \hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \partial\hat{\mathbf{d}} \quad ; \quad \beta,\lambda \in \mathcal{A}.$$

(E.20)

Using Eq. (E.16), Eq. (E.20) becomes

$$\forall \alpha \in \mathcal{A}: \quad [\hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \hat{\mathbf{R}}_\beta + h_{\alpha\beta} + \frac{\partial h_{\alpha\lambda}}{\partial\gamma_\beta}\Delta t\,\dot{\gamma}_\lambda + \Delta t(\boldsymbol{\mathcal{L}}_e : \hat{\mathbf{d}}) : \mathbf{B}(\mathbf{R}_\alpha^0) : \mathbf{D}_\beta]\partial\dot{\gamma}_\beta = \hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \partial\hat{\mathbf{d}} \quad ; \quad \beta,\lambda \in \mathcal{A}.$$

(E.21)

Let us introduce the tensor $\mathbf{M}$, which is the inverse of the second–order tensor $\mathbf{Z}$, whose components are given by the following expression:

$$\forall \alpha, \beta \in \mathcal{A}: \quad Z_{\alpha\beta} = \hat{\mathbf{R}}_\alpha : \boldsymbol{\mathcal{L}}_e : \hat{\mathbf{R}}_\beta + h_{\alpha\beta} + \frac{\partial h_{\alpha\lambda}}{\partial\gamma_\beta}\Delta t\,\dot{\gamma}_\lambda + \Delta t(\boldsymbol{\mathcal{L}}_e : \hat{\mathbf{d}}) : \mathbf{B}(\mathbf{R}_\alpha^0) : \mathbf{D}_\beta \quad ; \quad \lambda \in \mathcal{A}. \tag{E.22}$$

The differentials of the slip rates finally read

$$\forall \alpha \in \mathcal{A}: \quad \partial\dot{\gamma}_\alpha = M_{\alpha\beta\ \ \beta} : \boldsymbol{\mathcal{L}}_e : \partial\hat{\mathbf{d}} \quad ; \quad \beta \in \mathcal{A}. \tag{E.23}$$

By inserting Eq. (E.23) into (E.6), (E.11) and (E.13), we finally obtain the following relations:

$$\begin{aligned}
&\forall \alpha, \beta \in \mathcal{A}: \quad \partial\Lambda_{\alpha\beta} = -\Delta t[\Lambda_{\alpha\lambda}\frac{\partial h_{\lambda\zeta}}{\partial\gamma_\chi}\Lambda_{\zeta\beta}\,M_{\chi\rho}(\hat{}_\rho : \boldsymbol{\mathcal{L}}_e)] : \partial\hat{\mathbf{d}} = -\Delta t\,\boldsymbol{\Omega}_{\alpha\beta} : \partial\hat{} \quad ; \quad \lambda,\zeta,\rho,\chi \in \mathcal{A} \\
&\forall \alpha \in \mathcal{A}: \quad \partial\hat{\mathbf{R}}_\alpha = -\Delta t[M_{\beta\gamma}(\mathbf{B}(\mathbf{R}_\alpha^0) : \mathbf{D}_\beta) \otimes (\hat{\mathbf{R}}_\gamma : \boldsymbol{\mathcal{L}}_e)] : \partial\hat{\mathbf{d}} = -\Delta t\,\mathbf{P}_\alpha : \partial\hat{\mathbf{d}} \quad ; \quad \beta,\gamma \in \mathcal{A} \\
&\forall \alpha \in \mathcal{A}: \quad \partial\hat{\mathbf{S}}_\alpha = -\Delta t[M_{\beta\gamma}(\mathbf{B}(\mathbf{S}_\alpha^0) : \mathbf{D}_\beta) \otimes (\hat{\mathbf{R}}_\gamma : \boldsymbol{\mathcal{L}}_e)] : \partial\hat{\mathbf{d}} = -\Delta t\,\mathbf{Q}_\alpha : \partial\hat{\mathbf{d}} \quad ; \quad \beta,\gamma \in \mathcal{A}
\end{aligned} \tag{E.24}$$

In order to determine the expression of $\hat{\boldsymbol{\mathcal{L}}}_s^*$, let us introduce a fourth-order operator denoted $\mathbf{U}$, such that for all two second-order tensors $\mathbf{a}$ and $\mathbf{b}$,

$$\mathbf{b}.\mathbf{a} - \mathbf{a}.\mathbf{b} = \mathbf{U}(\mathbf{a}) : \mathbf{b}. \tag{E.25}$$

One can easily check that the analytical expression of $\mathbf{U}(\mathbf{a})$ reads

$$\forall i, j, k, l = 1,2,3: \quad U(\mathbf{a})_{ijkl} = a_{lj}\delta_{ik} - a_{ik}\delta_{lj}. \tag{E.26}$$

Using Eqs. (E.24) and (E.25), Eq. (E.3) becomes

$$
\begin{aligned}
\partial \hat{\boldsymbol{\mathcal{L}}}_{\mathrm{s}} : \Delta\hat{\boldsymbol{\varepsilon}} = [&(\mathbf{R}_{\beta} : \boldsymbol{\mathcal{L}}_{\mathrm{e}} : \Delta\hat{\boldsymbol{\varepsilon}})(\boldsymbol{\mathcal{L}}_{\mathrm{e}} : \hat{\mathbf{R}}_{\alpha} + \mathbf{U}(\hat{\boldsymbol{\sigma}}(\mathrm{t}_0)) : \hat{\mathbf{S}}_{\alpha}) \otimes \boldsymbol{\Omega}_{\alpha\beta} \\
&+ \Lambda_{\alpha\beta}(\hat{\mathbf{R}}_{\beta} : \boldsymbol{\mathcal{L}}_{\mathrm{e}} : \Delta\hat{\boldsymbol{\varepsilon}})(\boldsymbol{\mathcal{L}}_{\mathrm{e}} : \mathbf{P}_{\alpha} + \mathbf{U}(\boldsymbol{\sigma}(\mathrm{t}_0)) : \mathbf{Q}_{\alpha}) \qquad ; \quad \alpha, \beta \in \mathcal{A} \\
&+ \Lambda_{\alpha\beta}\left((\boldsymbol{\mathcal{L}}_{\mathrm{e}} : \hat{}_{\alpha} + \mathbf{U}(\hat{\boldsymbol{\sigma}}(\mathrm{t}_0)) : \hat{\mathbf{S}}_{\alpha}) \otimes ((\boldsymbol{\mathcal{L}}_{\mathrm{e}} : \Delta\hat{\boldsymbol{\varepsilon}}) : \mathbf{P}_{\beta})\right)] : \partial\Delta\hat{\boldsymbol{\varepsilon}} \\
= \hat{\boldsymbol{\mathcal{L}}}^{*}_{\mathrm{s}} &: \partial\Delta\hat{\boldsymbol{\varepsilon}}
\end{aligned}
\qquad . \qquad \text{(E.27)}
$$

Appendix F. The initial imperfection approach

The initial imperfection approach is widely used in the literature for the prediction of localized necking in sheet metals. This procedure was initially developed by Marciniak and Kuczynski [54], and subsequently extended by Hutchinson and Neale [55]. In this approach, an initial geometric imperfection, in the form of a band weaker than the rest of the sheet, is assumed to preexist in the studied sheet metal. Figure F.1 depicts the initial geometry of the sheet including the postulated groove. In the sequel, the following notations will be employed:

- H_I^B: the initial thickness of the band B.
- H_I^S: the initial thickness of the safe zone S (referred to also as the homogeneous zone).
- $\vec{\mathcal{N}}_I$: the initial unit normal to the band.
- θ_I: the initial orientation of the band.

According to the notations above, the initial imperfection factor ξ_I, defined as

$$\xi_I = 1 - \frac{H_I^B}{H_I^S}, \tag{F.1}$$

can be introduced.

The initial imperfection approach, formulated under the plane-stress assumption, is defined by the following relations:

- The kinematic compatibility condition between the band and the safe zone. This condition expresses the jump in the velocity gradient across the discontinuity surface

$$\forall \alpha,\beta = 1,2: \quad G_{\alpha\beta}^B = G_{\alpha\beta}^S + \dot{C}_\alpha \mathcal{N}_\beta, \tag{F.2}$$

  where $\dot{\vec{C}}$ and $\vec{\mathcal{N}}$ are the jump vector and the current normal to the band, respectively.

- The equilibrium condition, involving the normal and shear forces across the interface between the band and the safe zone, has also to be satisfied throughout the deformation. This relation can be expressed in terms of Cauchy stress tensor (in index form) as follows:

$$\forall \alpha = 1,2: \quad H^S \Sigma_{\alpha\beta}^S \mathcal{N}_\beta = H^B \Sigma_{\alpha\beta}^B \mathcal{N}_\beta \quad ; \quad \beta = 1,2. \tag{F.3}$$

  The rate form of (F.3) reads

$$\forall \alpha = 1,2: \quad (H^S \dot{\Sigma}_{\alpha\beta}^S \mathcal{N}_\beta) = (H^B \dot{\Sigma}_{\alpha\beta}^B \mathcal{N}_\beta) \quad ; \quad \beta = 1,2. \tag{F.4}$$

  The final expression (F.4) of the equilibrium condition is that used in the implementation of the initial imperfection approach.

- The evolution of the band orientation, determined by the current inclination θ of its normal unit vector with respect to the X_1 direction

$$\mathrm{Tan}(\theta) = e^{(E_{11}^S - E_{22})}\,\mathrm{Tan}(\theta_I)\,, \tag{F.5}$$

where

$$E_{11}^S = \int_0^t G_{11}^S\,dt \quad ; \quad E_{22}^S = \int G_{22}^S\,dt\,. \tag{F.6}$$

- The evolution equation for the current imperfection factor ξ, defined as

$$\xi = 1 - \frac{H^B}{H^S}\,. \tag{F.7}$$

Taking into account the following relations:

$$H^S = H^S\,e^{E_{33}^S} \quad ; \quad H^B \quad H_I^B\,e^{\ 33}\,, \tag{F.8}$$

the current imperfection factor ξ can be expressed as

$$\xi = 1 - (1 - \xi_I)e^{(E_{33}^B - E_{33}^S)}\,. \tag{F.9}$$

The strain components E_{33}^S and E_{33}^B in Eqs. (F.8) and (F.9) are determined by $\int_0^t G_{33}^S\,dt$ and $\int_0^t G_{33}^B\,dt$, respectively.

- The plane-stress condition in both the safe zone and the band, written as

$$\Sigma_{13}^S = \Sigma_{23}^S = \Sigma_{33}^S = 0 \quad ; \quad \Sigma_{13}^B = \Sigma_{23}^B = \Sigma_{33}^B = 0\,. \tag{F.10}$$

As the crystallographic texture of the polycrystalline aggregate studied in this paper is assumed to be orthotropic, (F.10) is equivalent to

$$G_{13}^S = G_{31}^S = G_{23}^S = G_{32}^S = 0 \quad ; \quad G_{13}^B = G_{31}^B = G_{23}^B = G_{32}^B = 0\,. \tag{F.11}$$

- The in-plane biaxial loading applied to the safe zone

$$\frac{G_{22}^S}{G_{11}^S} = \rho \quad ; \quad G_{12}^S = G_{21} = 0\,, \tag{F.12}$$

where the strain-path ratio ρ is assumed to be constant during the deformation.

- The constitutive equations of the polycrystal, which are derived from the constitutive equations of the single crystals via the Taylor scale-transition scheme.

The component G_{11}^S is fixed to 1 and, accordingly, the component G_{22}^S is equal to ρ (see Eq. $(F.12)_{(1)}$). As to the component G_{33}^S, it is determined iteratively by imposing the plane-stress condition given by Eq. $(F.10)_{(1)}$. The other components of $\mathbf{G^S}$ are equal to zero, as detailed in Eqs. $(F.11)_{(1)}$ and $(F.12)_{(2)}$. The in-plane components $G_{\alpha\beta}^B$ (α,β=1,2) are determined from the components $G_{\alpha\beta}^S$ by using Eq. (F.2), while the component G_{33}^B is derived from the plane-stress condition in the same way as for G_{33}^S.

Let us now summarize the set of the known and unknown components of $\mathbf{G}^S$ and $\mathbf{G}^B$, corresponding to the initial imperfection approach:

Known components:

- $G^S_{11} = 1$ (in fact, the value of G^S_{11} can be chosen quite freely; for convenience, it has been fixed here to 1).
- $G^S_{22} = \rho G_{11} = \rho$ (this value is a priori known, because ρ is fixed to a constant value throughout the deformation).
- $G^S_{12} = G^S_{21} = 0$ (this is the result of the in-plane biaxial loading applied in the safe zone).
- $G^S_{13} = G^S_{31} = G^S_{23} = G^S_{32} = 0$ (this is the combined result of the plane-stress condition in the safe zone and the orthotropic crystallographic texture used).
- $G^B_{13} = G^B_{31} = G^B_{23} = G^B_{32} = 0$ (this is the combined result of the plane-stress condition in the band and the orthotropic crystallographic texture used).

Unknown components:

- G^S_{33}, which should be determined by the plane-stress condition in the safe zone.
- G^B_{33}, which should be determined by the plane-stress condition in the band.
- $G^B_{11}, G^B_{12}, G^B_{21}$ and G^B_{22}, which are functions of the jump vector $\dot{\vec{\mathbf{C}}}$ and the normal to the band $\vec{\mathcal{N}}$. It must be noted that the unit vector $\vec{\mathcal{N}}$ is entirely determined by θ ($\vec{\mathcal{N}} = (\cos\theta, \sin\theta)$), which in turn is a function of the known components E^S_{11} $(= t)$, E^S_{22} $(= t\rho)$ and the initial band orientation θ_I (see Eq. (F.5)). Therefore, this quantity is known and $G^B_{11}, G^B_{12}, G^B_{21}$ and G^B_{22} are functions only of the components $\dot{C}_1$ and $\dot{C}_2$ of the jump vector $\dot{\vec{\mathbf{C}}}$.

To solve the governing equations of the initial imperfection approach, and then determine the unknown components of $\mathbf{G}^S$ and $\mathbf{G}^B$ as well as other mechanical variables both in the safe zone and the band, the incremental algorithm developed in [49] is followed.

References

1. Taylor GI, Elam CF. The distortion of an aluminum crystal during a tensile test. *Proceedings of the Royal Society of London. Series A* 1923; **102**:643–667.

2. Taylor GI. Plastic strain in metals. *Journal Institute of Metals* 1938; **62**:307–324.

3. Mandel J. Généralisation de la théorie de la plasticité de W.T. Koiter. *International Journal of Solids and Structures* 1965; **1**:273–295.

4. Hill R. Generalized constitutive relations for incremental deformation of metal crystals by multislip. *Journal of the Mechanics and Physics of Solids* 1966; **14**:95–102.

5. Rice JR. Inelastic constitutive relations for solids: an internal-variable theory and its application to metal plasticity. *Journal of the Mechanics and Physics of Solids* 1971; **19**:433–455.

6. Hill R, Rice JR. Constitutive analysis of elastic–plastic crystals at arbitrary strain. *Journal of the Mechanics and Physics of Solids* 1971; **20**:401–413.

7. Asaro RJ. Crystal plasticity. *Journal of Applied Mechanics* 1983; **50**:921–934.

8. Havner KS. Finite Plastic Deformation of Crystalline Solids. Cambridge University Press, Cambridge, 1992.

9. Kocks UF, Tomé CN, Wenk HR. (Eds.), Texture and Anisotropy: preferred Orientations in Polycrystals and their effect on Materials Properties. Cambridge University Press, 1998.

10. Follansbee PS, Kocks UF. A constitutive description of the deformation of copper based on the use of the mechanical threshold stress as an internal state variable. *Acta Metallurgica* 1988; **36**(1):81–93.

11. Anand L, Kothari M. A computational procedure for rate-independent crystal plasticity. *Journal of the Mechanics and Physics of Solids* 1996; **44**:525–558.

12. Miehe C, Schröder J. A comparative study of stress update algorithms for rate-independent and rate-dependent crystal plasticity. *International Journal for Numerical Methods in Engineering* 2001; **50**:273–298.

13. Knockaert R, Chastel Y, Massoni E. Rate-independent crystalline and polycrystalline plasticity, application to FCC materials. *International Journal of Plasticity* 2000; **16**:179–198.

14. Miehe C, Rosato D. Fast texture updates in fcc polycrystal plasticity based on a linear active-set-estimate of the lattice spin. *Journal of the Mechanics and Physics of Solids* 2007; **55**:2687–2716.

15. Franz G, Abed-Meraim F, Berveiller M. Strain localization analysis for single crystals and polycrystals: towards microstructure–ductility linkage. *International Journal of Plasticity* 2013; **48**:1–33.

16. Franz G, Abed-Meraim F, Lorrain J-P, Ben Zineb T, Lemoine X, Berveiller M. Ellipticity loss analysis for tangent moduli deduced from a large strain elastic–plastic self-consistent model. *International Journal of Plasticity* 2009; **25**:205–238.

17. Feyel F, Chaboche JL. FE^2 multiscale approach for modelling the elastoviscoplastic behaviour of long fibre SiC/Ti composite materials. *Computer Methods in Applied Mechanics and Engineering* 2000; **183**:309–330.

18. Feyel F. A multilevel finite element method (FE^2) to describe the response of highly non-linear structures using generalized continua. *Computer Methods in Applied Mechanics and Engineering* 2003; **192**:3233–3244.

19. Watanabe I, Terada K, de Souza Neto EA, Peric D. Characterization of macroscopic tensile strength of polycrystalline metals with two-scale finite element analysis. *Journal of the Mechanics and Physics of Solids* 2008; **56**:1105–1125.

20. Cuitiño AM, Ortiz M. Computational modelling of single crystals. *Modelling and Simulation in Materials Science and Engineering* 1992; **1**:225–263.

21. Schmidt-Baldassari M. Numerical concepts for rate-independent single crystal plasticity. *Computer Methods in Applied Mechanics and Engineering* 2003; **192**:1261–1280.

22. Han C-S, Wagoner RH, Barlat F. On precipitate induced hardening in crystal plasticity: algorithms and simulations. *International Journal of Plasticity* 2004; **20**:1441–1461.

23. Schmid E, Boas W. Kristallplastizität, Springer, 1935.

24. Ben Bettaieb M, Debordes O, Dogui A, Duchêne L, Keller C, On the numerical integration of rate independent single crystal behavior at large strain. *International Journal of Plasticity* 2012; **32-33**:184–217.

25. Fischer A. A special Newton-type optimization method. *Optimization* 1992; **24**:269–284.

26. Fischer A. Solution of monotone complementarity problems with locally Lipschitzian functions. *Mathematical Programming* 1997; **76**:513–532.

27. Rvachev VL. Methods of Logical Algebra in Mathematical Physics, *Naukova Dumka*, Kyiv, 1974 (in Russian).

28. Rvachev VL. Theory of R-functions and Some Applications, *Naukova Dumka*, Kyiv, 1982 (in Russian).

29. Mangasarian OL., Solodov MV. Nonlinear complementarity as unconstrained and constrained minimization. *Mathematical Programming* 1993; **62**:277–297.

30. Kindrachuk VM, Galanov BA. An efficient approach for numerical treatment of some inequalities in solid mechanics on examples of Kuhn–Tucker and Signorini–Fichera conditions. *Journal of the Mechanics and Physics of Solids* 2014; **63**:432–450.

31. Borja RI, Wren JR. Discrete micromechanics of elastoplastic crystals. *International Journal for Numerical Methods in Engineering* 1993; **36**:3815–3840.

32. McGinty RD, McDowell DL. A semi-implicit integration scheme for rate independent finite crystal plasticity. *International Journal of Plasticity* 2006; **22**:996–1025.

33. Ben Bettaieb M. Modélisation numérique du comportement de matériaux polycristallins par homogénéisation périodique. *Ph.D. Thesis*, Université de la Méditerranée, Marseille, France, 2006.

34. Amirkhizi AV, Nemat-Nasser S. A framework for numerical integration of crystal elasto-plastic constitutive equations compatible with explicit finite element codes. *International Journal of Plasticity* 2007; **23**:1918–1937.

35. Dogui A. Plasticité anisotrope en grandes déformations. *Thèse de doctorat d'état*, Université Claude Bernard, Lyon 1, France, 1989.

36. Bassani JL, Wu TY, Latent hardening in single crystals. II. Analytical characterization and predictions. *Proceedings of the Royal Society of London. Series A* 1991; **435**:21–41.

37. Simo JC, Kennedy JG, Govindjee S. Non-smooth multisurface plasticity and viscoplasticity. Loading/unloading conditions and numerical algorithms. *International Journal for Numerical Methods in Engineering* 1988; **26**:2161–2185.

38. Hughes TJR, Winget J. Finite rotation effects in numerical integration of rate constitutive equations arising in large-deformations analysis. *International Journal for Numerical Methods in Engineering* 1980; **15**:1862–1867.

39. Rubinstein R, Atluri SN. Objectivity of incremental constitutive relations over finite time steps in computational finite deformation analyses. *Computer Methods in Applied Mechanics and Engineering* 1983; **36**: 277–290.

40. Brunet M. Some computational aspects in three-dimensional and plane stress finite elastoplastic deformation problem. *Engineering Analysis with Boundary Elements* 1989; **6**:78–83.

41. Miehe C., Schröder J., Schotte J. Computational homogenization analysis in finite plasticity simulation of texture development in polycrystalline materials. *Computer Methods in Applied Mechanics and Engineering* 1999; **171**:387–418.

42. Terada K., Watanabe I. Computational aspects of tangent moduli tensors in rate-independent crystal elastoplasticity. *Computational Mechanics* 2007; **40**:497–511.

43. Hughes T.J.R., Winget J. Finite rotation effects in numerical integration of rate constitutive equations arising in large deformation analysis. *International Journal for Numerical Methods in Engineering* 1980; **15**:1862–1867.

44. Haddag B., Abed-Meraim F., Balan T. Strain localization analysis using a large deformation anisotropic elastic–plastic model coupled with damage. *International Journal of Plasticity* 2009; **25**:1970–1996.

45. Miehe C. Numerical computation of algorithmic (consistent) tangent moduli in large-strain computational inelasticity. *Computer Methods in Applied Mechanics and Engineering* 1996; **134**:223–240.

46. Chang YW, Asaro RJ. An experimental study of shear localization in aluminum-copper single crystals. *Acta Metallurgica* 1981; **29**:241–257.

47. Yoshida K, Kuroda M. Comparison of bifurcation and imperfection analyses of localized necking in rate-independent polycrystalline sheets. *International Journal of Solids and Structures* 2012; **49**:2073–2084.

48. Tadano Y, Yoshida K, Kuroda M. Plastic flow localization analysis of heterogeneous materials using homogenization-based finite element method. *International Journal of Mechanical Sciences* 2013; **72**:63–74.

49. Knockaert R, Chastel Y, Massoni E. Forming limit prediction using rate-independent polycrystalline plasticity. *International Journal of Plasticity* 2002; **18**:231–247.

50. Lorrain JP. Critère de ductilité basé sur la perte d'ellipticité du module tangent élastoplastique déduit d'un modèle autocohérent. *Ph.D. Thesis*, ENSAM, Metz, France, 2005.

51. Armijo L. Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific Journal* of *Mathematics* 1996; **16**:1–3.

52. Press WH, Teukolsky SA, Vetterling WT, Flannery BP, Numerical Recipes in Fortran 77: The Art of Scientific Computing. second edition, Cambridge Press, 1992.

53. Golub GH and Van Loan CF. Matrix Computations. *The Johns Hopkins University Press, Baltimore, MA* 1983

54. Marciniak Z, Kuczynski K. Limit strains in processes of stretch-forming sheet metal. *International Journal of Mechanical Sciences* 1967; **9**:609–620.

55. Hutchinson JW, Neale KW, Sheet necking-III. Strain-rate effects. *In: Koistinen, D.P., Wang, N.M. (Eds.), Mechanics of Sheet Metal Forming. Plenum* 1978; 269–285.

Table 1. Material parameters.

Elasticity		**Hardening**			
λ	μ	τ_{sat}	τ_0	q	h_0
42.03 GPa	28.3 GPa	0.108 GPa	0.06 GPa	1.4	0.534 GPa

Table 2. Initial crystal orientations in degrees.

Orientation	φ_1	φ_2	φ_3
1	0	0	0
2	15	0	0
3	30	0	0
4	15	30	0

Table 3. Convergence of the return-mapping algorithm, associated with the non-smooth formulation, when applied to the single crystal with initial orientation # 2.

Incr. Num.	**system set**	$\Delta t=3 \times 10^{-4}$ s	$\Delta t=4 \times 10^{-4}$ s	$\Delta t=5 \times 10^{-4}$ s
1	$\mathcal{P}$	{}	{}	{}
	$\mathcal{A}$	{}	{}	{}
2	$\mathcal{P}$	{}	{}	{}
	$\mathcal{A}$	{}	{}	{}
3	$\mathcal{P}$	{}	{}	{2, 4, 20, 22}
	$\mathcal{A}$	{}	{}	{2, 20}
4	$\mathcal{P}$	{}	{2, 4, 20, 22}	{2, 4, 20, 22, 24, 30}
	$\mathcal{A}$	{}	{2, 20}	No convergence
5	$\mathcal{P}$	{2, 4, 20, 22}	{2, 4, 20, 22, 24, 30}	
	$\mathcal{A}$	{2, 20}	No convergence	
6	$\mathcal{P}$	{2, 4, 20, 22}		
	$\mathcal{A}$	{2, 20}		
7	$\mathcal{P}$	{2, 4, 20, 22}		
	$\mathcal{A}$	{2, 20}		
8	$\mathcal{P}$	{2, 4, 20, 22}		
	$\mathcal{A}$	{2, 20}		
9	$\mathcal{P}$	{2, 4, 20, 22}		
	$\mathcal{A}$	{2, 20}		
10	$\mathcal{P}$	{2, 4, 20, 22}		
	$\mathcal{A}$	{2, 20}		

Table 4. Convergence of the semi-smooth formulation of the return-mapping algorithm when applied to the single crystal with initial orientation # 2.

Incr. Num.	**system set**	$\Delta t=3 \times 10^{-4}$ s	$\Delta t=4 \times 10^{-4}$ s	$\Delta t=5 \times 10^{-4}$ s
1	$\mathcal{P}$	{}	{}	{}
	$\mathcal{A}$	{}	{}	{}
2	$\mathcal{P}$	{}	{}	{}
	$\mathcal{A}$	{}	{}	{}
3	$\mathcal{P}$	{}	{}	{2, 4, 20, 22}
	$\mathcal{A}$	{}	{}	{2, 20}
4	$\mathcal{P}$	{}	{2, 4, 20, 22}	{2, 4, 20, 22, 24, 30}
	$\mathcal{A}$	{}	{2, 20}	{2, 20}
5	$\mathcal{P}$	{2, 4, 20, 22}	{2, 4, 20, 22, 24, 30}	{2, 4, 20, 22, 24, 30}
	$\mathcal{A}$	{2, 20}	{2, 20}	{2, 20}
6	$\mathcal{P}$	{2, 4, 20, 22}	{2, 4, 20, 22, 24, 30}	{2, 4, 20, 22, 24, 30}
	$\mathcal{A}$	{2, 20}	{2, 20}	{2, 20}
7	$\mathcal{P}$	{2, 4, 20, 22}	{2, 4, 20, 22, 24, 30}	{2, 4, 20, 22, 24, 30}
	$\mathcal{A}$	{2, 20}	{2, 20}	{2, 20}
8	$\mathcal{P}$	{2, 4, 20, 22}	{2, 4, 20, 22, 24, 30}	{2, 4,15, 20, 22, 24, 30, 33}
	$\mathcal{A}$	{2, 20}	{2, 20}	{2, 15, 20, 33}
9	$\mathcal{P}$	{2, 4, 20, 22}	{2, 4, 20, 22, 24, 30}	{2, 4,15, 20, 22, 24, 30, 33}
	$\mathcal{A}$	{2, 20}	{2, 20}	{2, 15, 20, 33}
10	$\mathcal{P}$	{2, 4, 20, 22}	{2, 4,15, 20, 22, 24, 30, 33}	{2, 4,15, 20, 22, 24, 30, 33}
	$\mathcal{A}$	{2, 20}	{2, 15, 20, 33}	{2, 4, 15, 20, 22, 33}

Table 5. Parameters and definition of the numerical simulations corresponding to the prediction of localized necking at the polycrystalline aggregate scale.

Simulation number	**Mechanical state**	**Strain path ratio ρ**	**Initial inclination θ_I**
Simulation #1	Uniaxial tensile state	–1/2	0°
Simulation #2	Uniaxial tensile state	–1/2	10°
Simulation #3	Uniaxial tensile state	–1/2	20°
Simulation #4	Plane-strain tensile state	0	0°

Table 6. CPU time corresponding to the different simulations for the third test relating to the prediction of localized necking using the imperfection approach.

Simulation number	**Ultimate algorithm**	**Return-mapping algorithm**
Simulation #1	1276 s	8548 s
Simulation #2	1208 s	9402 s
Simulation #3	768 s	5596 s
Simulation #4	374 s	3108 s

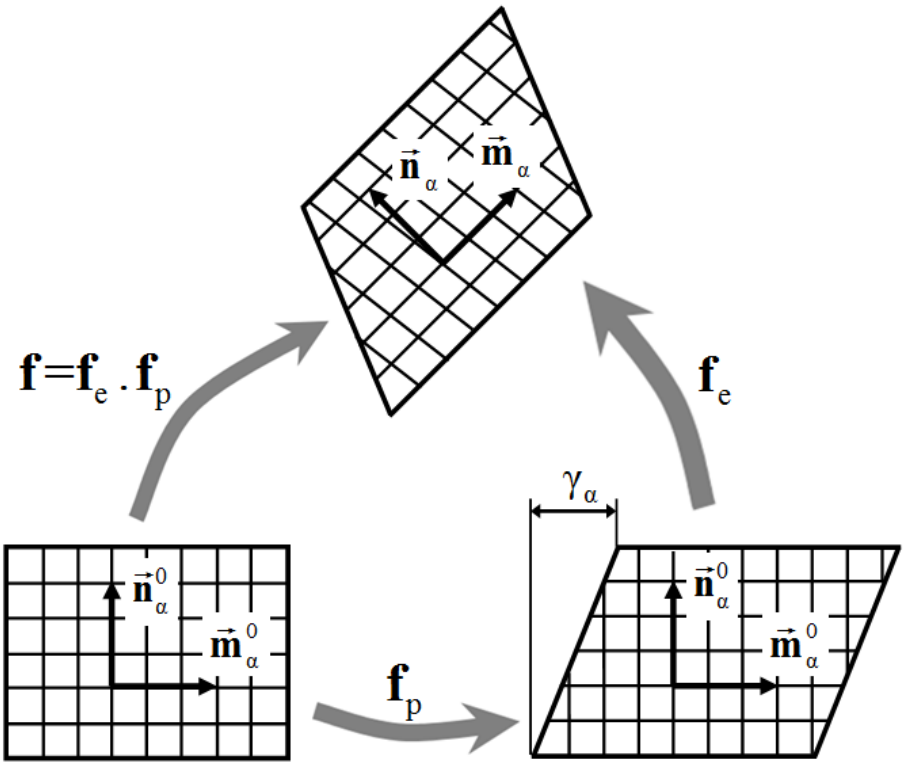


Figure 1. Schematic representation of the multiplicative decomposition of the deformation gradient: plastic slip and rigid rotation along with elastic distortion of the crystallographic lattice.

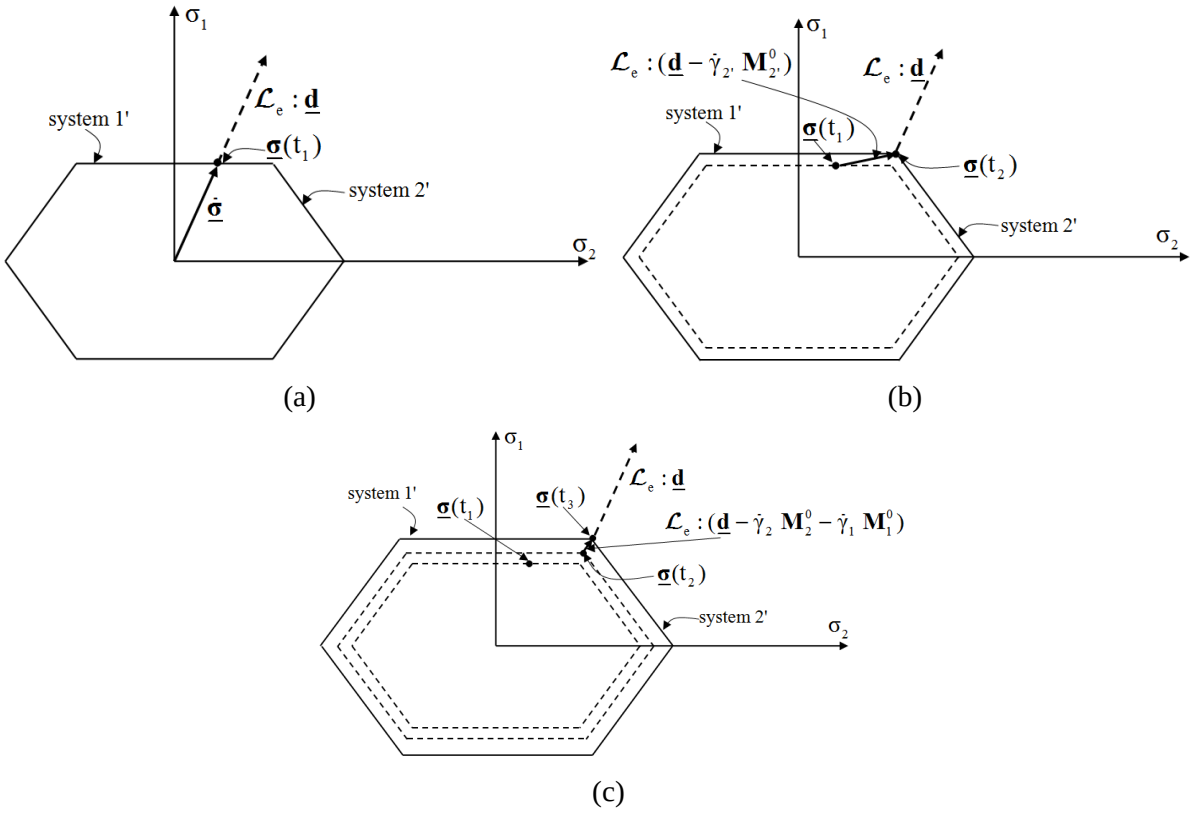


Figure 2. Evolution of the stress state on a 2D schematic representation of the single crystal yield surface: (a) First sub-increment [0, t_1], which corresponds to a purely elastic phase, (b) Second sub-increment [t_1, t_2], which corresponds to the phase where the first facet (system 1') is reached, (c) Third sub-increment [t_2, t_3], which corresponds to the phase where the second facet (system 2') is reached.

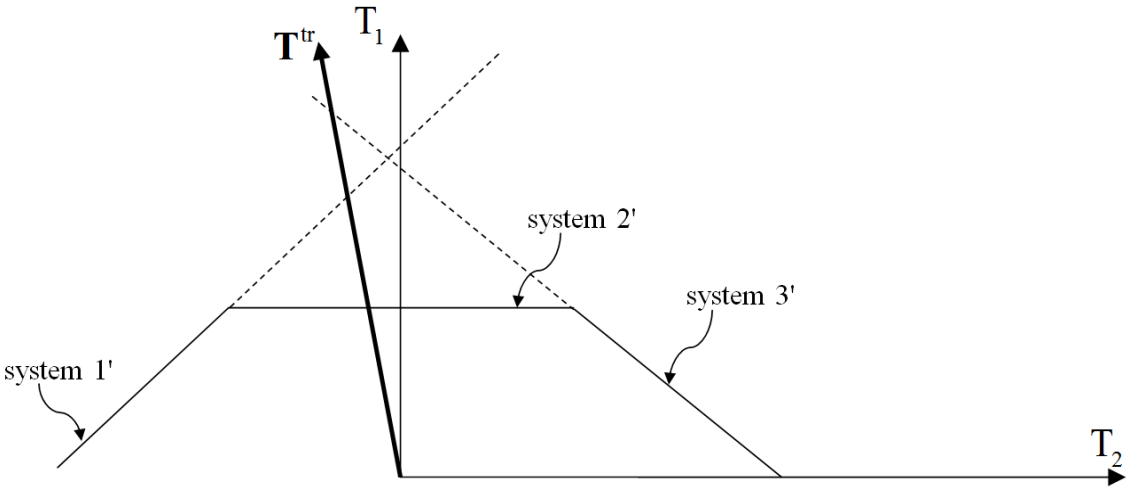
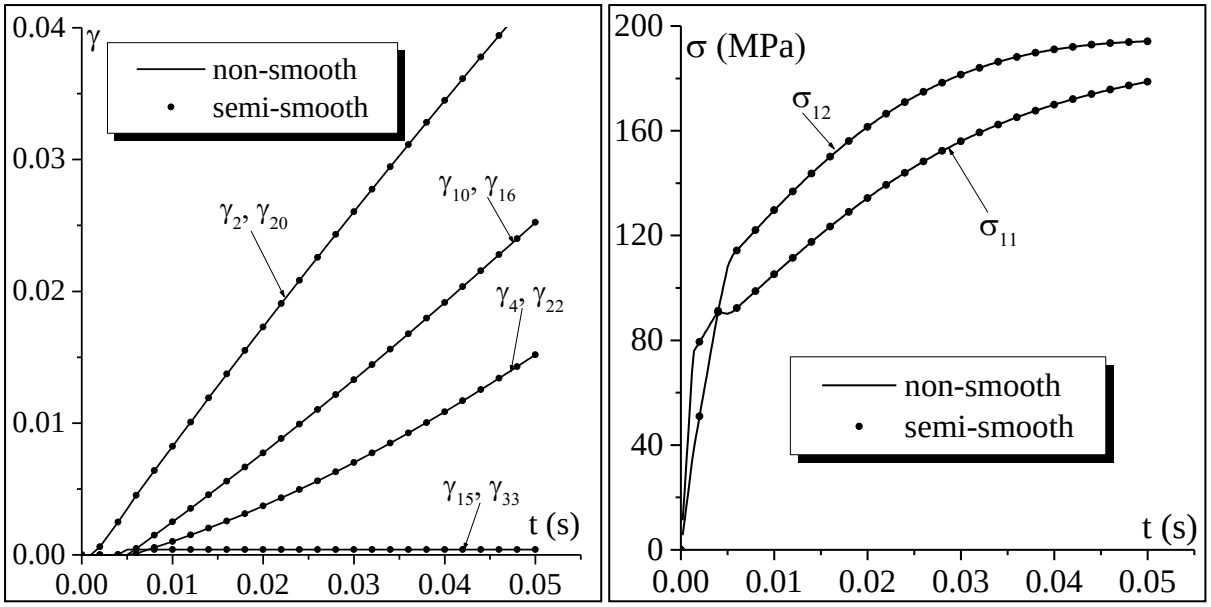
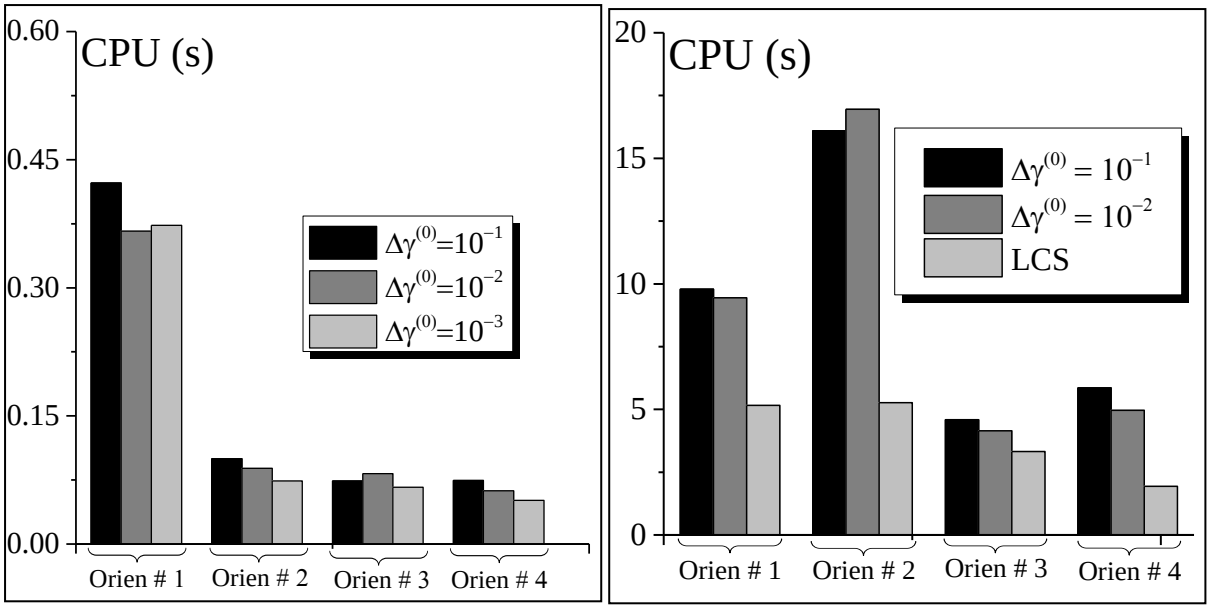


Figure 3. Illustration of the prediction of the set of potentially active slip systems by the return-mapping algorithm.



(a) (b)

Figure 4. Evolution of the accumulated slip γ of the active slip systems and the stress components σ_{11}, σ_{12} as functions of time t for the combined shearing–stretching test applied to single crystal with orientation #1; comparison between the results predicted by the non-smooth and semi-smooth formulations of the return-mapping algorithm: (a) the accumulated slip γ of the active slip systems; (b) the stress components σ_{11}, σ_{12}.



(a) (b)

Figure 5. Impact of the initial guess of the slip increments $\Delta\gamma^{(0)}$ on the CPU time required for the simulation of the combined shearing–stretching test: (a) The first time increment, (b) The first ten time increments.

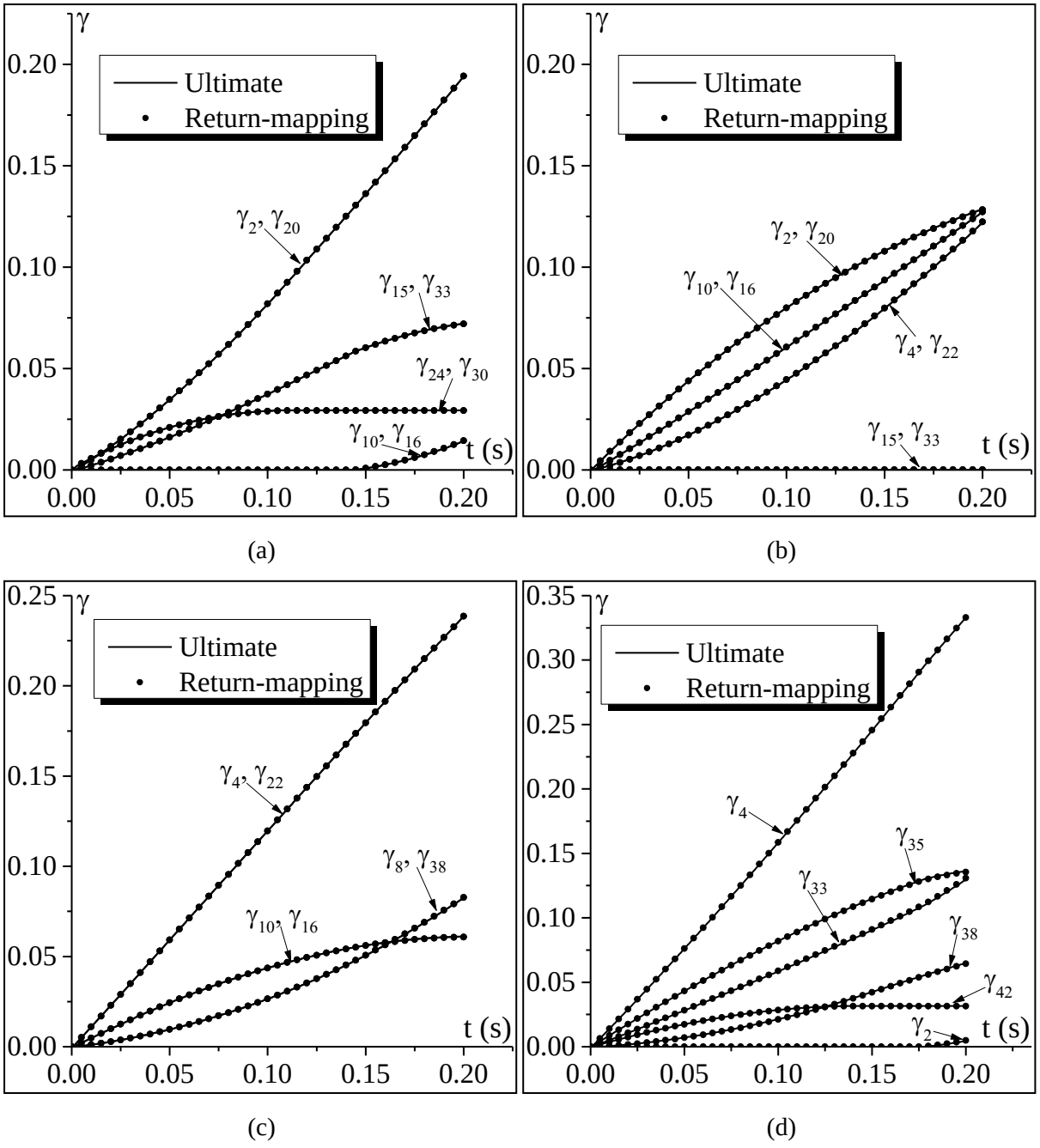


Figure 6. Evolution of the accumulated slip γ of the active slip systems as function of time t for the combined shearing–stretching test; comparison between the results predicted by the ultimate algorithm (explicit/implicit algorithm) and the return-mapping algorithm (semi-smooth formulation); (a) orientation # 1, (b) orientation # 2, (c) orientation # 3, (d) orientation # 4.

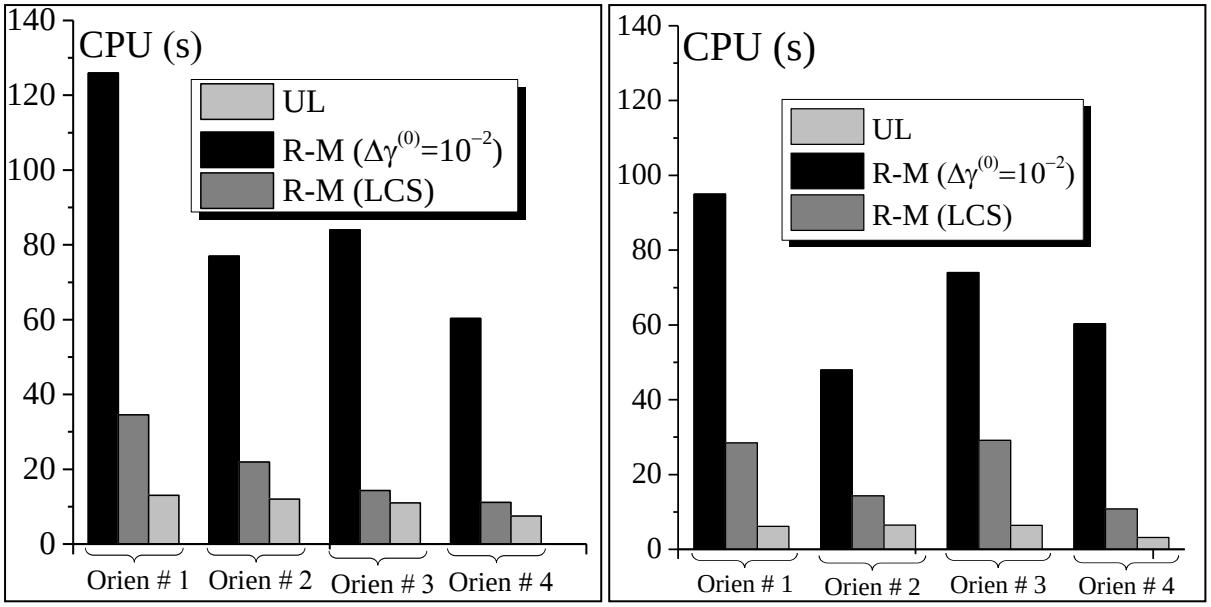


Figure 7. CPU time required in the simulation of the combined shearing–stretching test for different initial crystallographic orientations; comparison between the ultimate and return-mapping algorithms; (a) $\Delta t=5 \times 10^{-4}$ s, (b) $\Delta t= 10^{-3}$ s.

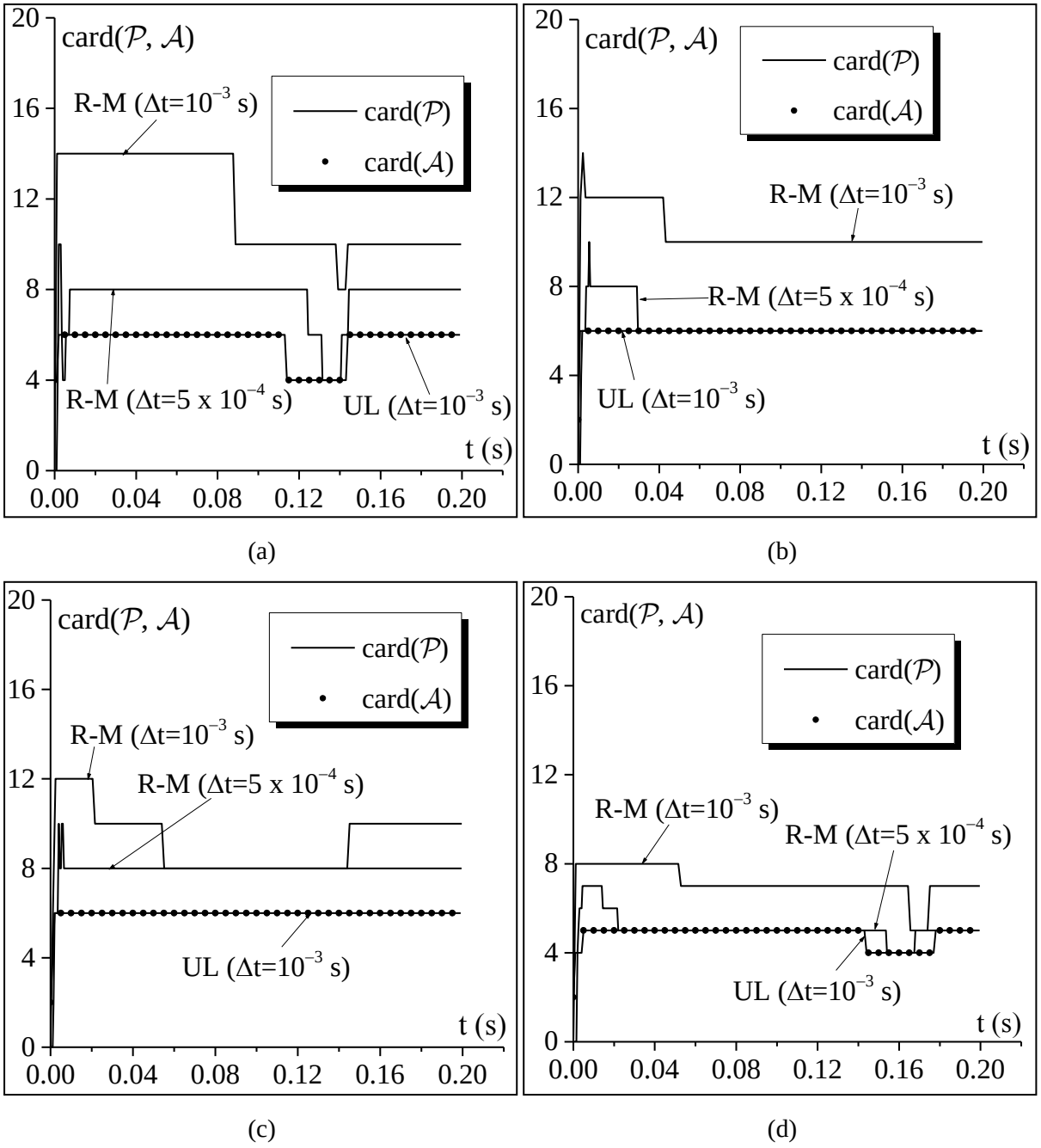


Figure 8. Evolution of the number of potentially active slip systems (denoted by card($\mathcal{P}$)) and the number of active slip systems (denoted by card($\mathcal{A}$)) as functions of time t for the combined shearing–stretching test; comparison between the ultimate and return-mapping algorithms; (a) orientation # 1, (b) orientation # 2, (c) orientation # 3, (d) orientation # 4.

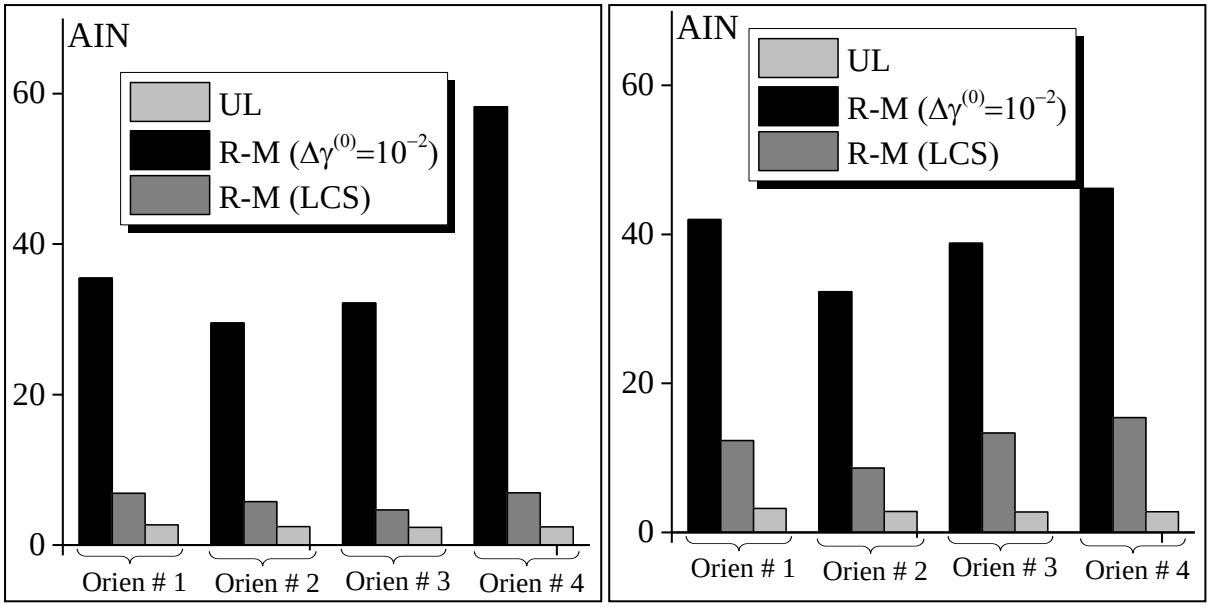


Figure 9. Average number of local iterations (AIN) in the simulation of the combined shearing–stretching test, for different initial crystallographic orientations; comparison between the ultimate and the return-mapping algorithms; (a) $\Delta t=5 \times 10^{-4}$ s, (b) $\Delta t= 10^{-3}$ s.

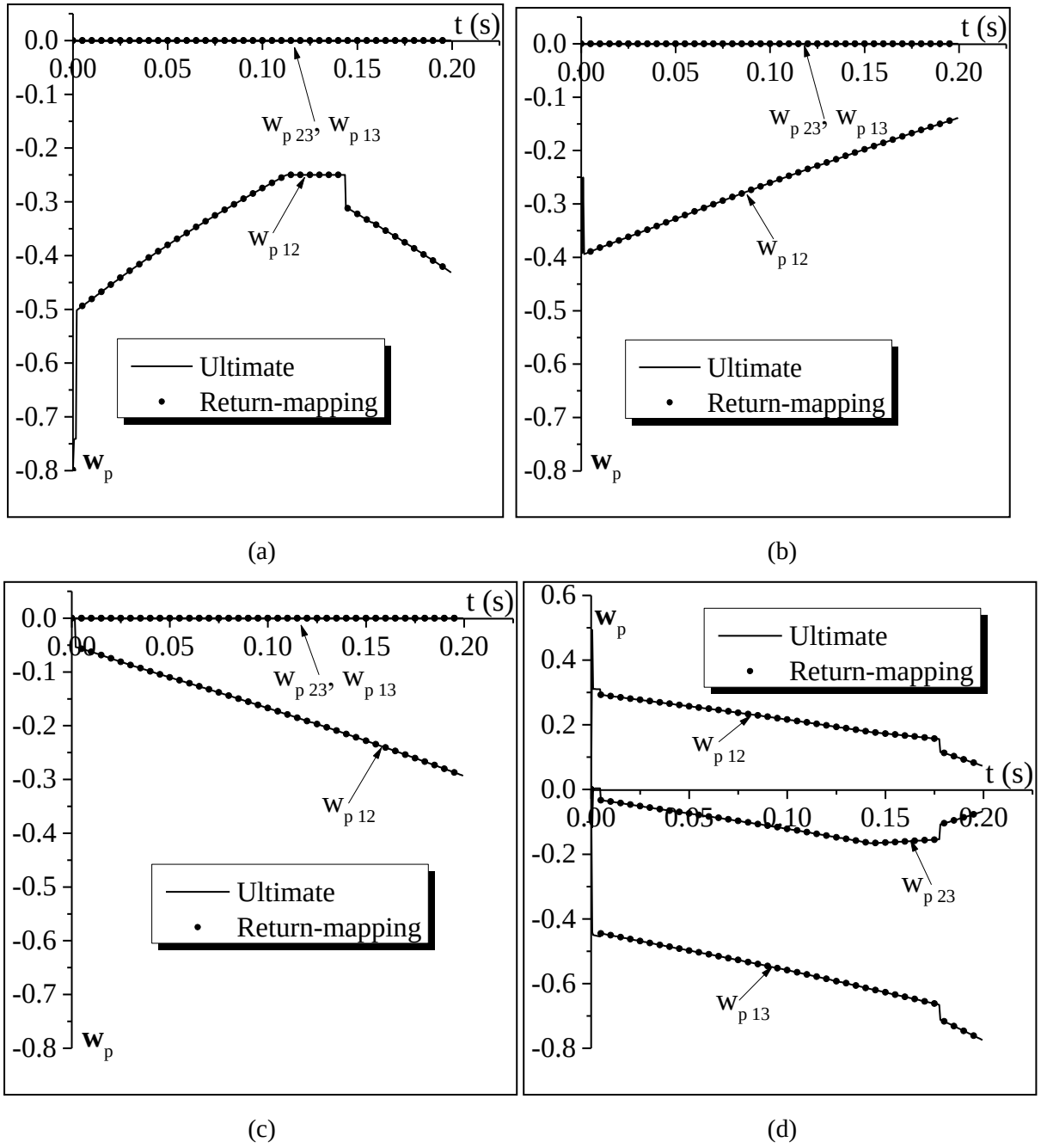


Figure 10. Evolution of the independent components of the plastic spin w_{p12}, w_{p23} and w_{p13} for the combined shearing–stretching test; comparison between the ultimate and return-mapping algorithms; (a) orientation # 1, (b) orientation # 2, (c) orientation # 3, (d) orientation # 4.

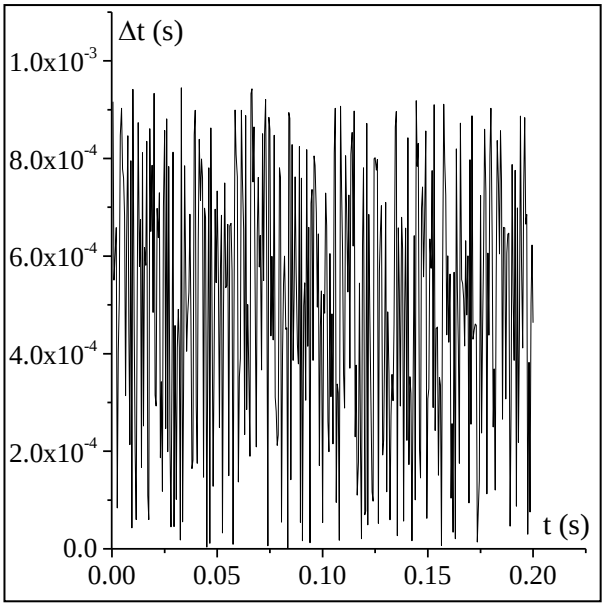
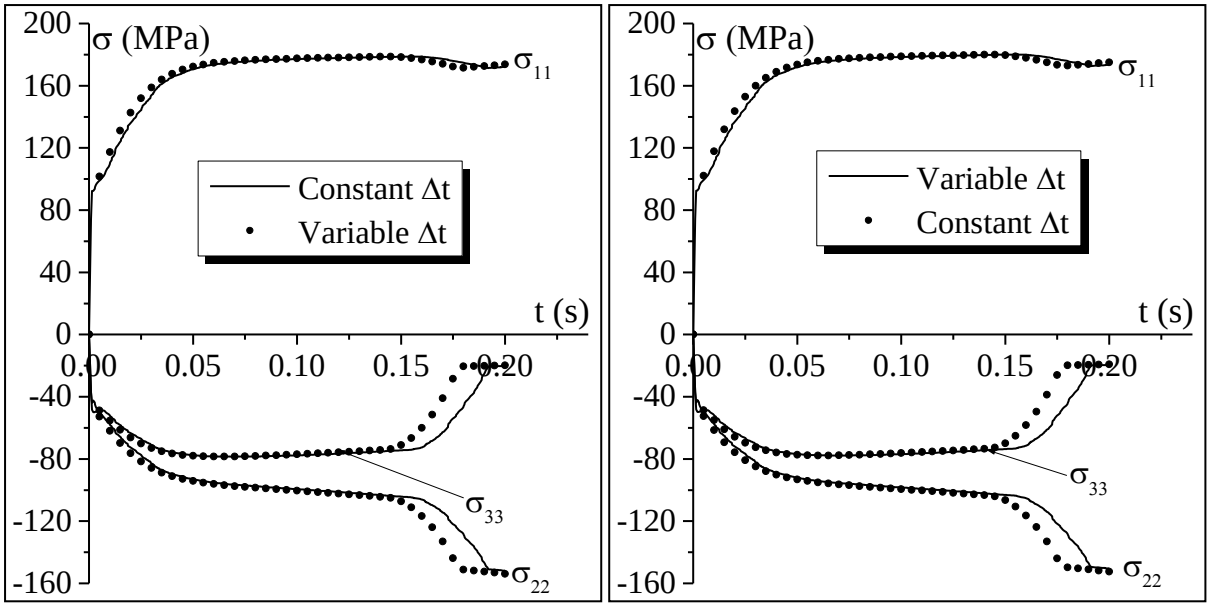


Figure 11. Evolution of the size Δt of the time increment as a function of time t for the simulations with variable size for the time increments.



(a) (b)

Figure 12. Evolution of the components σ_{11}, σ_{22} and σ_{33} as functions of time t for the combined shearing–stretching test applied to single crystal with orientation #2; comparison between the predictions based on constant and variable size for the time step; (a) Ultimate algorithm, (b) Return-mapping algorithm.

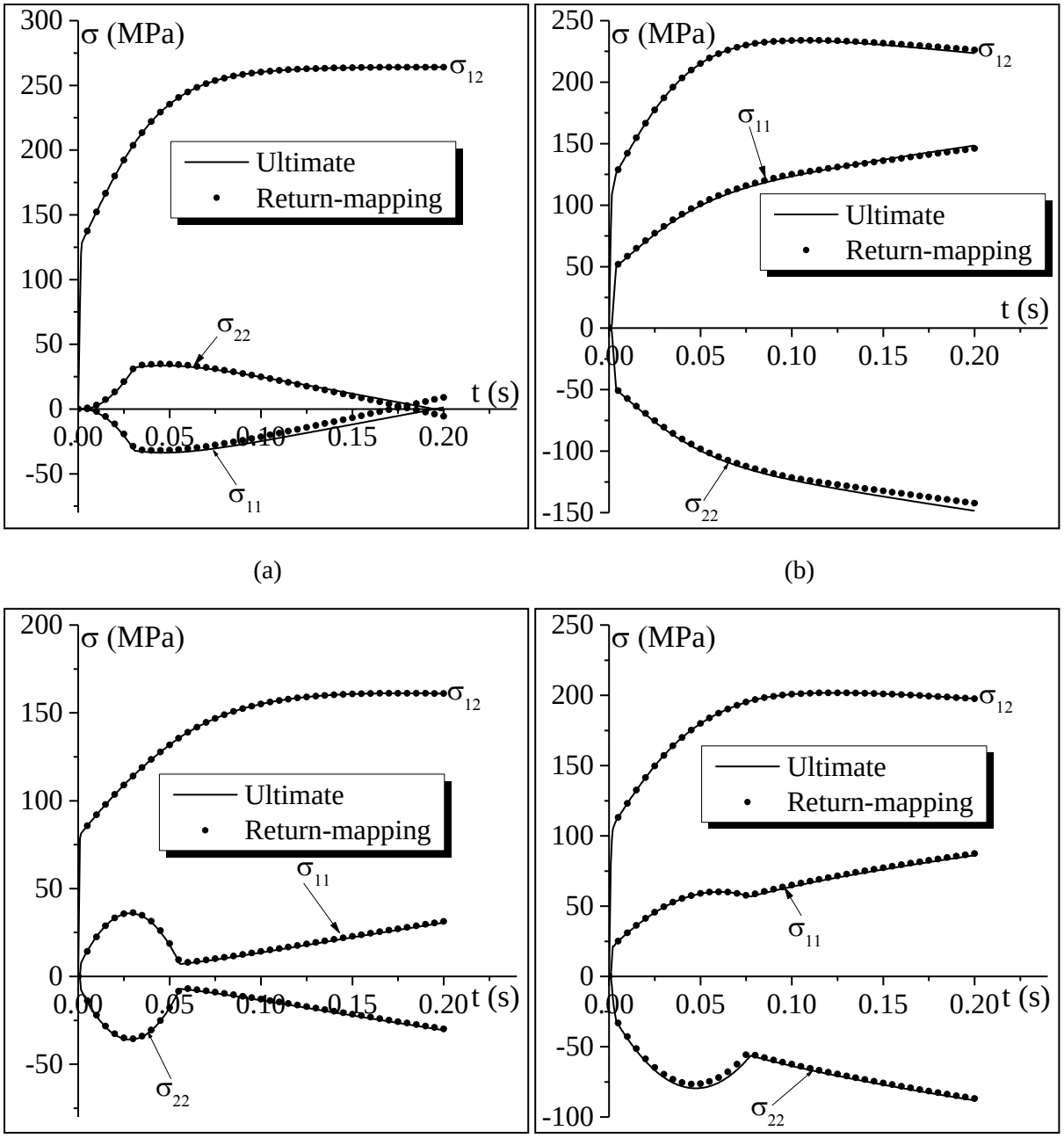


Figure 13. Evolution of the stress components σ_{11}, σ_{22} and σ_{12} as function of time t for the simple shear test; comparison between the results predicted by the ultimate and return-mapping algorithms; (a) orientation # 1, (b) orientation # 2, (c) orientation # 3, (d) orientation # 4.

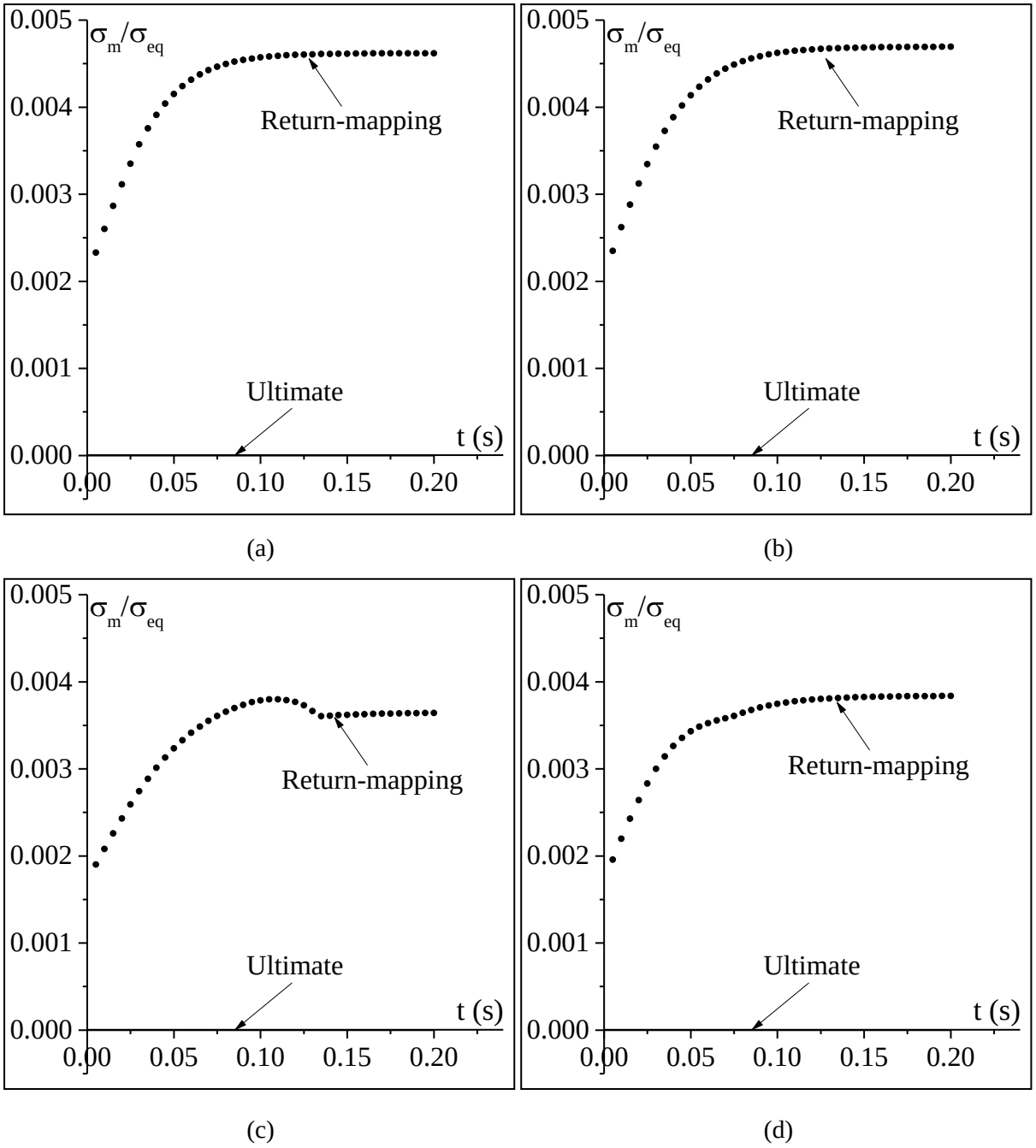


Figure 14. Evolution of the of mean stress normalized by the von Mises equivalent stress as function of time t for the simple shear test; comparison between the results predicted by the ultimate and return-mapping algorithms; (a) orientation # 1, (b) orientation # 2, (c) orientation # 3, (d) orientation # 4.

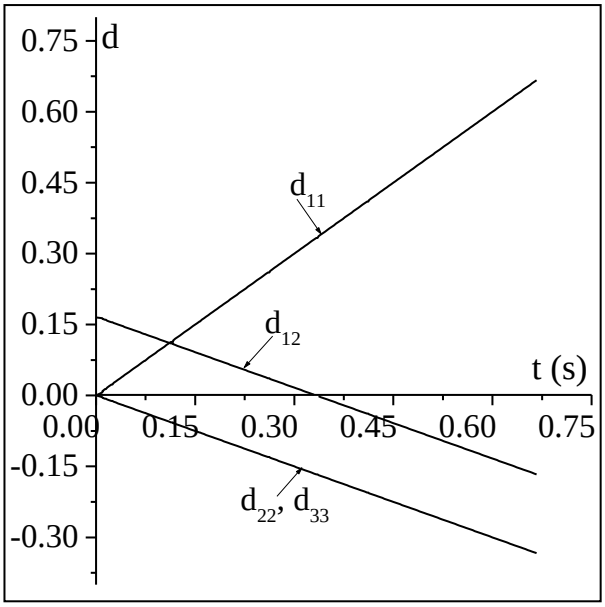


Figure 15. Evolution of the strain rate components for the non-constant loading.

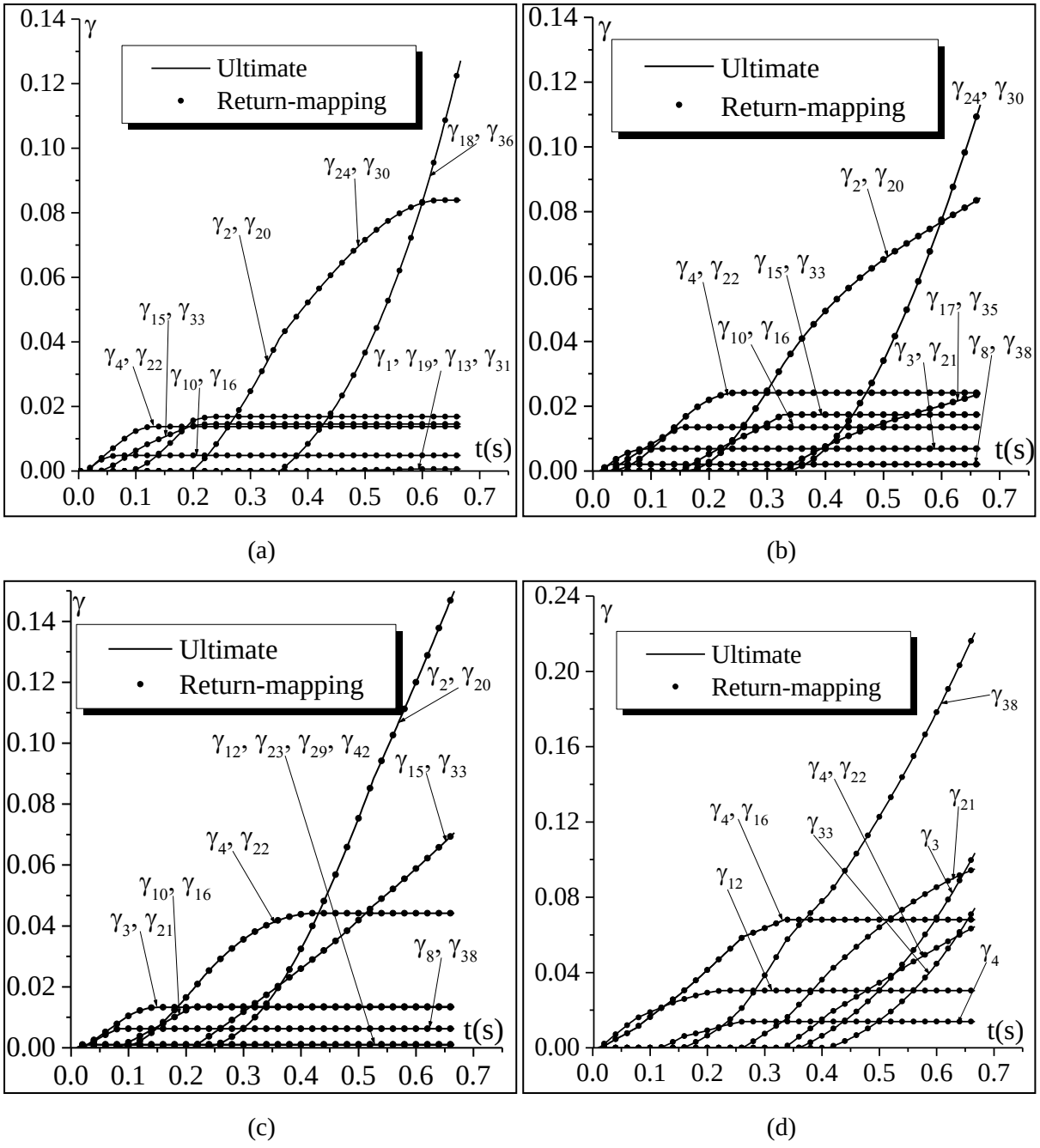
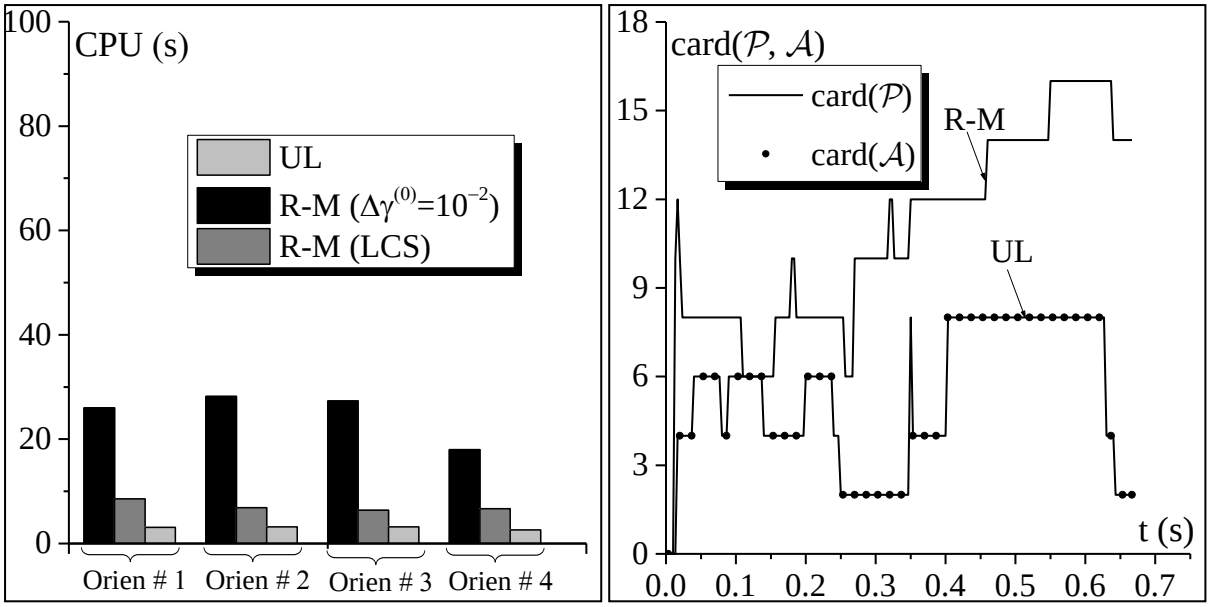
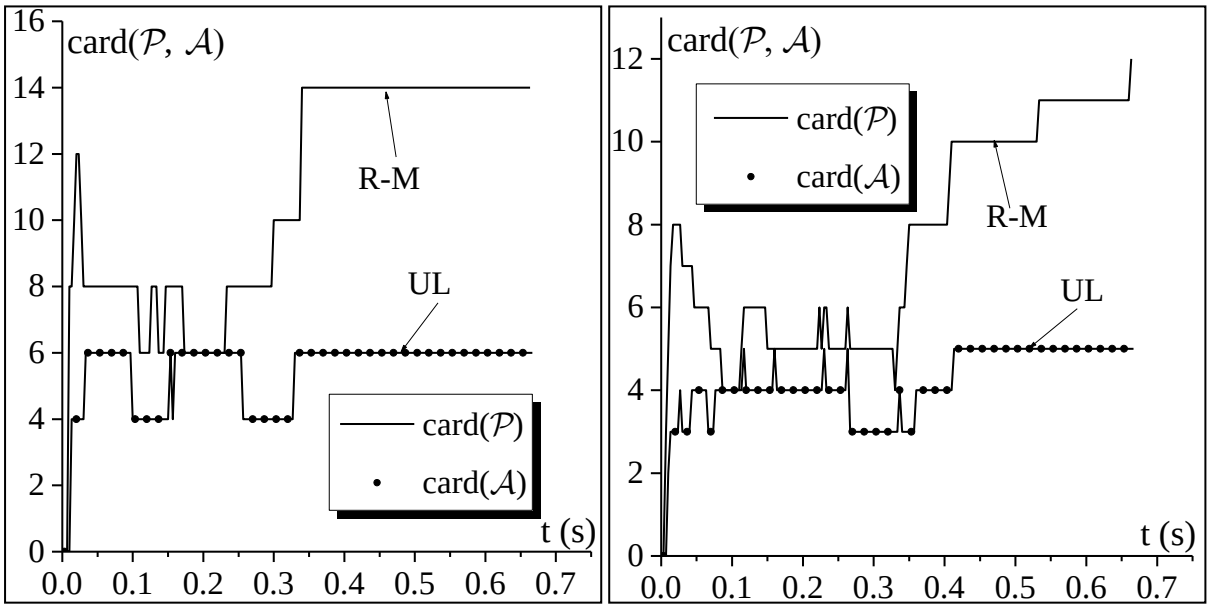


Figure 16. Evolution of the accumulated slip γ of the active slip systems as function of time t for the non-constant loading; comparison between the results predicted by the ultimate and return-mapping algorithms; (a) orientation # 1, (b) orientation # 2, (c) orientation # 3, (d) orientation # 4.



(a) (b)



(c) (d)

Figure 17. Evolution of the CPU time and of card($\mathcal{P}$, $\mathcal{A}$) as function of time for the non-constant loading; comparison between the ultimate and return-mapping algorithms; (a) evolution of the CPU time, (b) evolution of card($\mathcal{P}$, $\mathcal{A}$), orientation # 1, (c) evolution of card($\mathcal{P}$, $\mathcal{A}$), orientation # 2, (d) evolution of card($\mathcal{P}$, $\mathcal{A}$), orientation # 4.

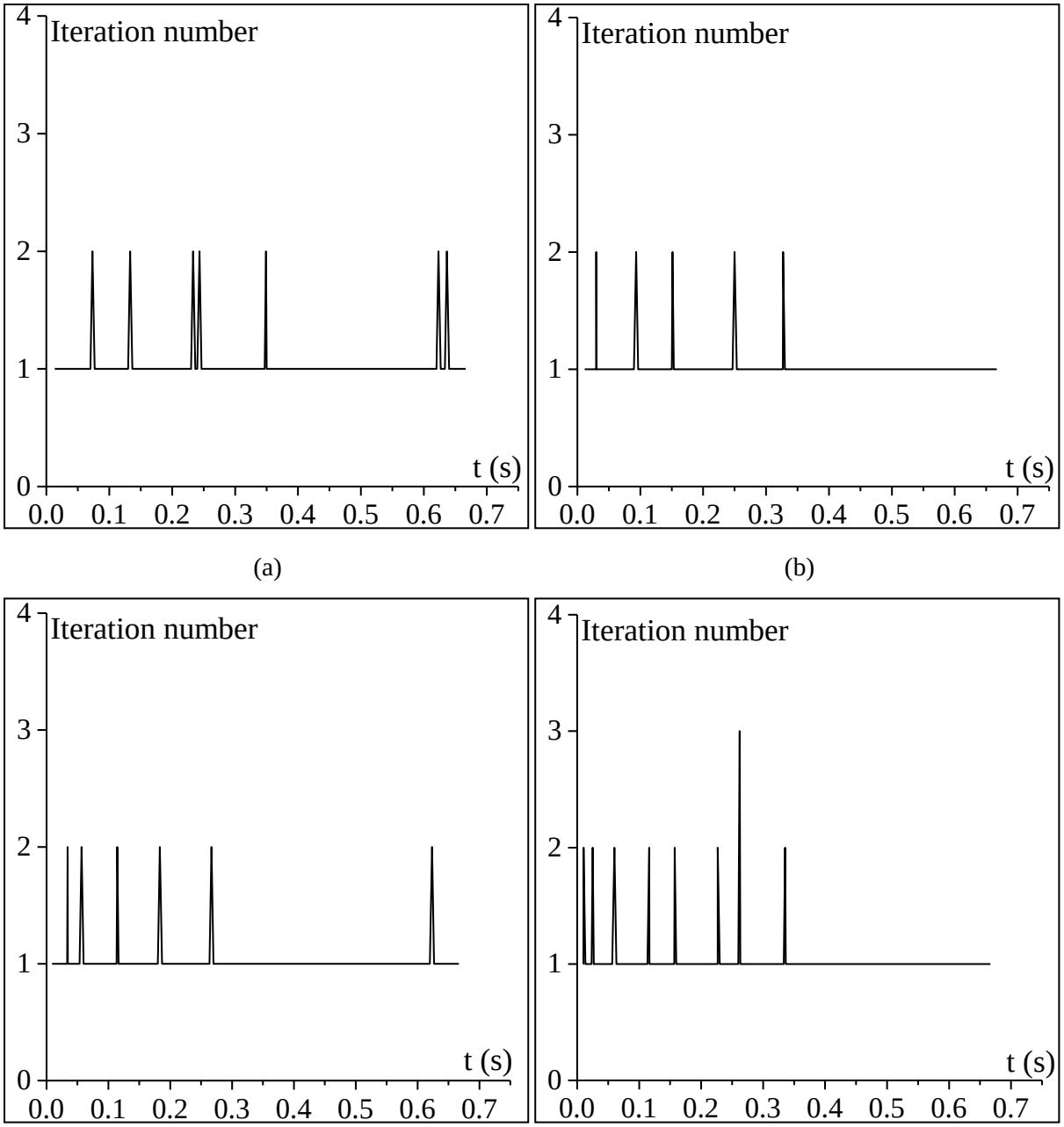
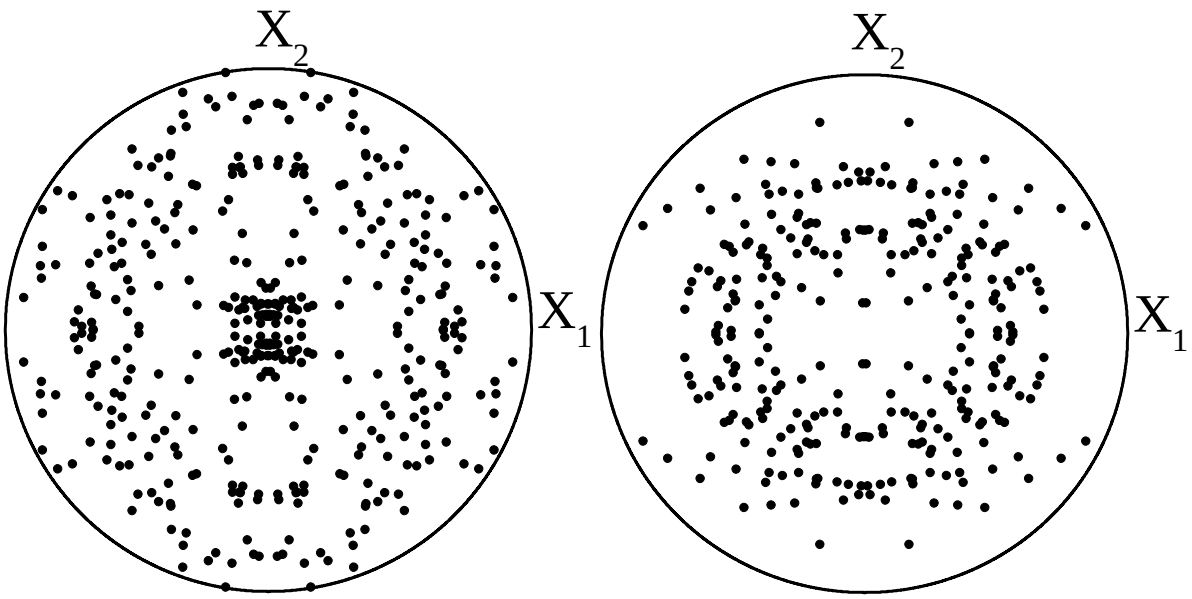


Figure 18. Evolution of the number of iterations for the iterative search procedure applied to the ultimate algorithm: (a) orientation # 1, (b) orientation # 2, (c) orientation # 3, (d) orientation # 4.



(a) (b)

Figure 19. Initial texture of the polycrystalline aggregate (100 single crystals): (a) {111} pole figure, (b) {100} pole figure.

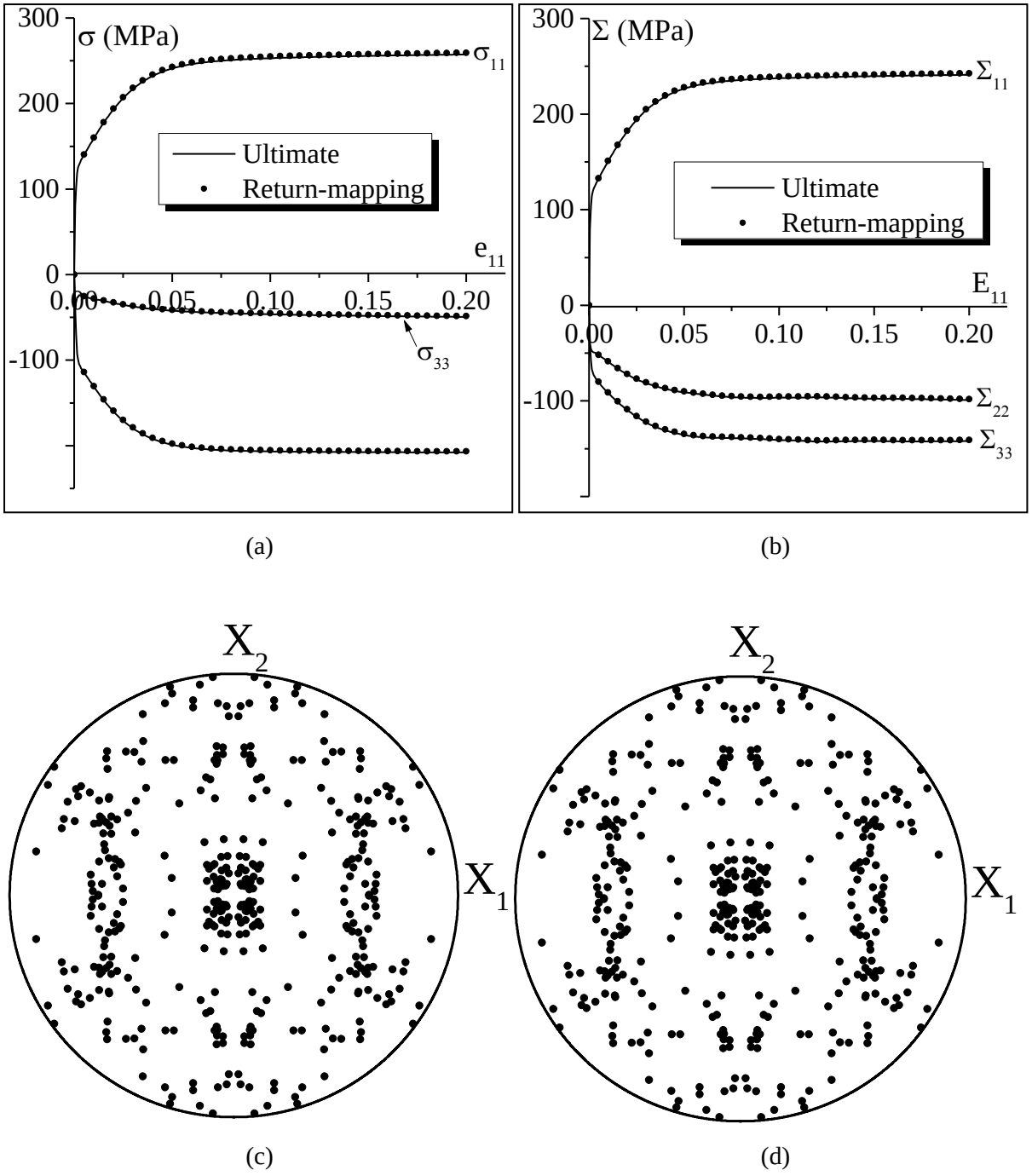


Figure 20. Comparison between the results obtained by the ultimate and return-mapping algorithms for the first test with constant macroscopic loading: (a) Evolution of the microscopic stress components σ_{11}, σ_{22} and σ_{33} for single crystal #1 as function of e_{11}, (b) Evolution of the macroscopic stress components Σ_{11}, Σ_{22} and Σ_{11} as function of E_{11} (= t G_{11}), (c) Final {111} pole figure obtained by the ultimate algorithm, (d) Final {111} pole figure obtained by the return-mapping algorithm.

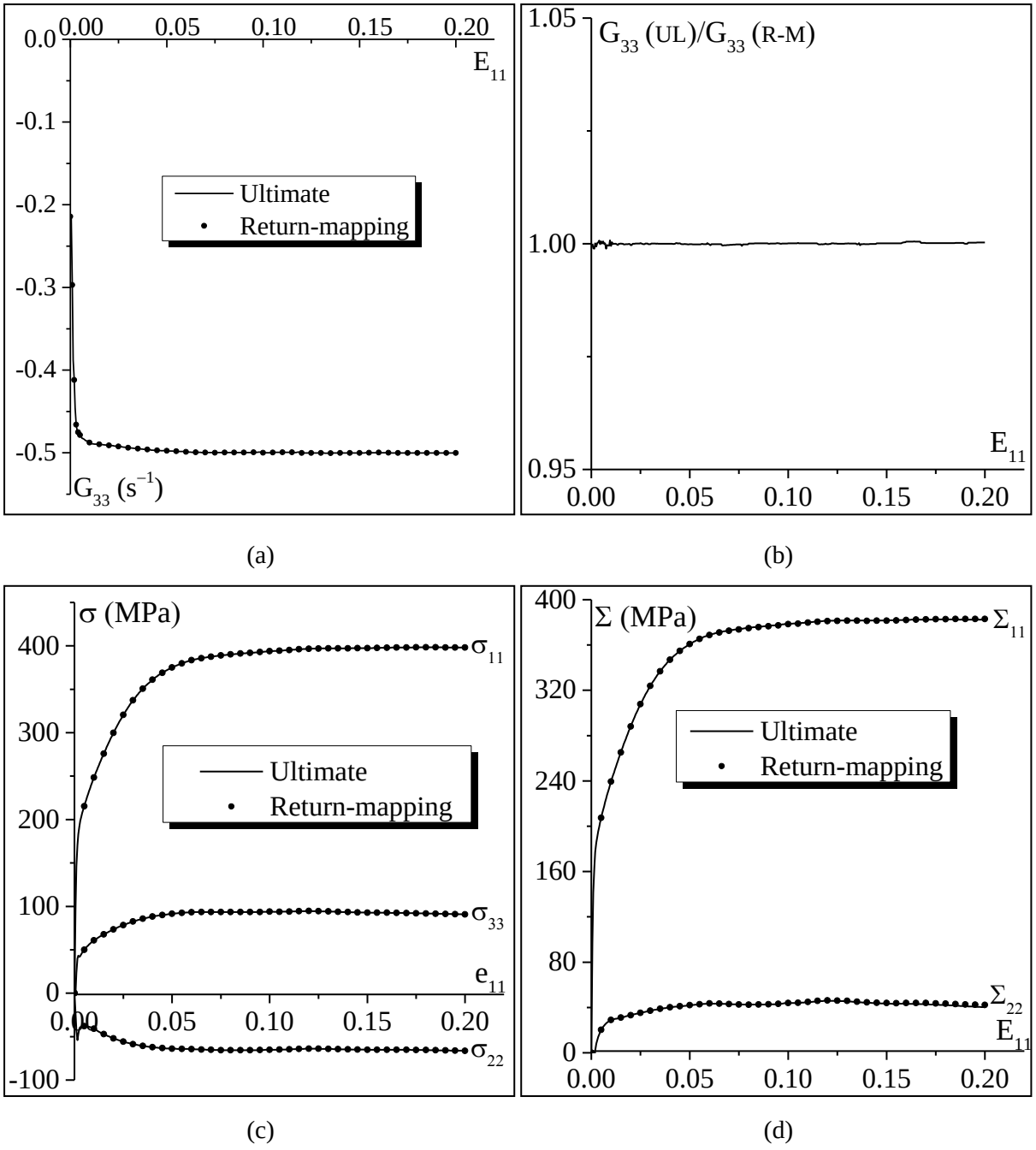


Figure 21. Comparison between the results obtained by the ultimate and return-mapping algorithms for the second test under plane-stress state: (a) Evolution of the macroscopic velocity gradient component G_{33} as a function of E_{11}, (b) Evolution of the ratio of G_{33}, determined by the ultimate algorithm, to that determined by the return-mapping algorithm, as a function of E_{11}, (c) Evolution of the microscopic stress components σ_{11}, σ_{22} and σ_{33} for single crystal #1 as function of e_{11}, (d) Evolution of the macroscopic stress components Σ_{11}, Σ_{22} as function of E_{11}.

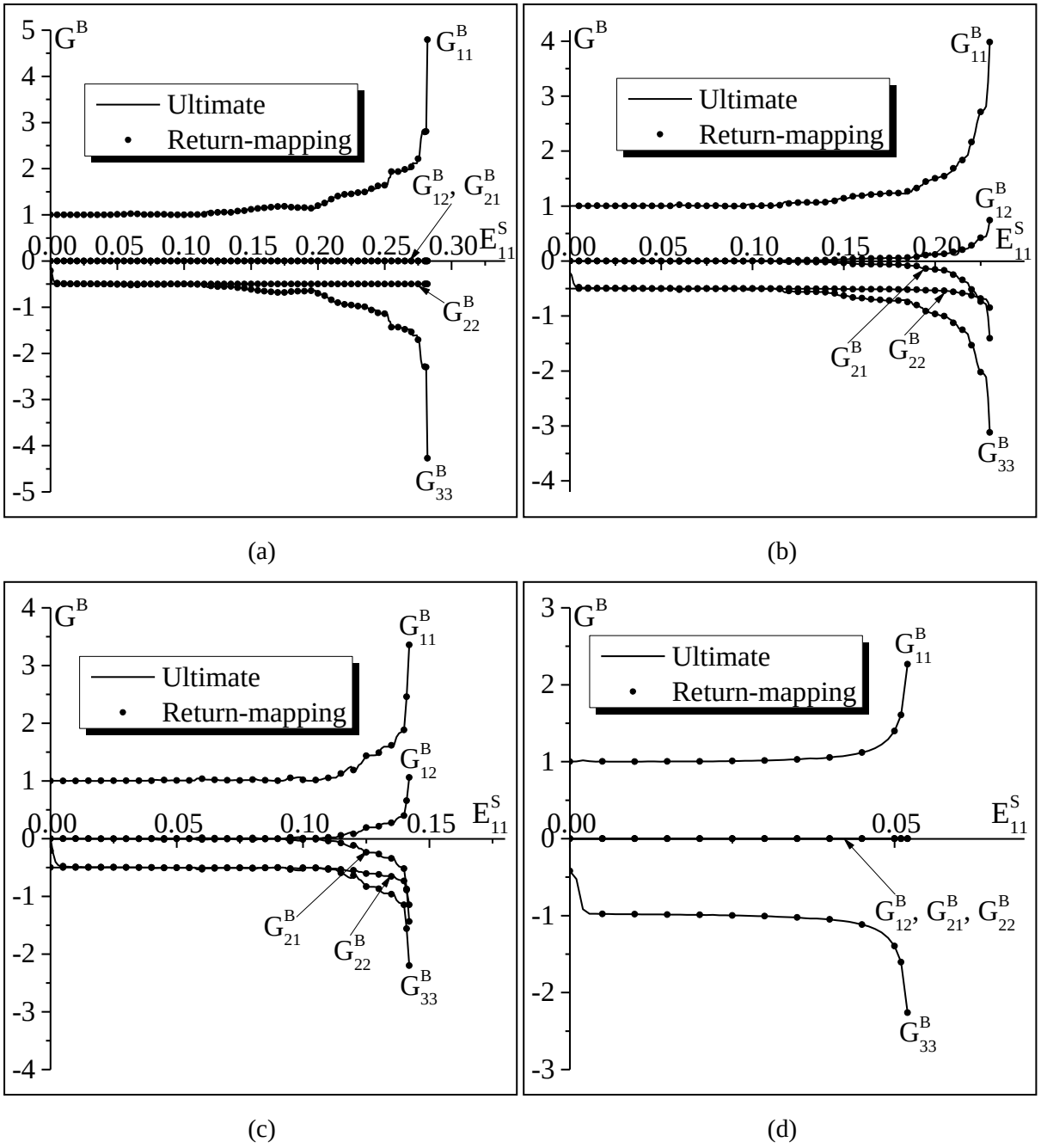


Figure 22. Evolution of the unknown components of the macroscopic velocity gradient tensor $\mathbf{G}^{B}$ in the band, as function of E_{11}^{S}, for the third test relating to the prediction of localized necking using the imperfection approach: (a) Simulation #1, (b) Simulation #2, (c) Simulation #3, (d) Simulation #4.

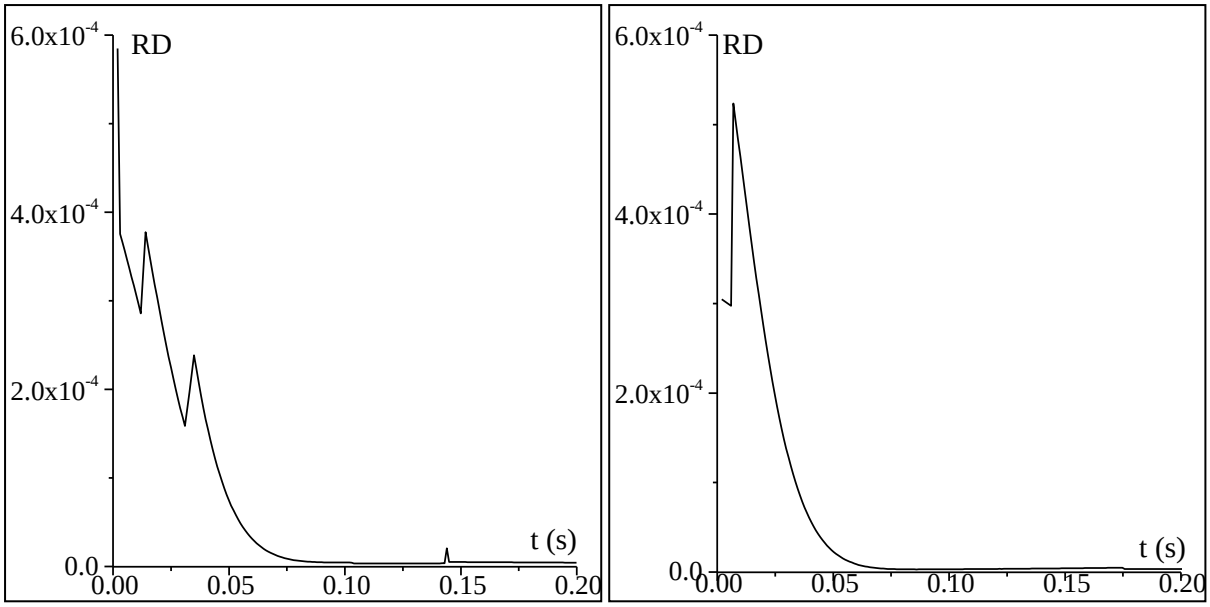
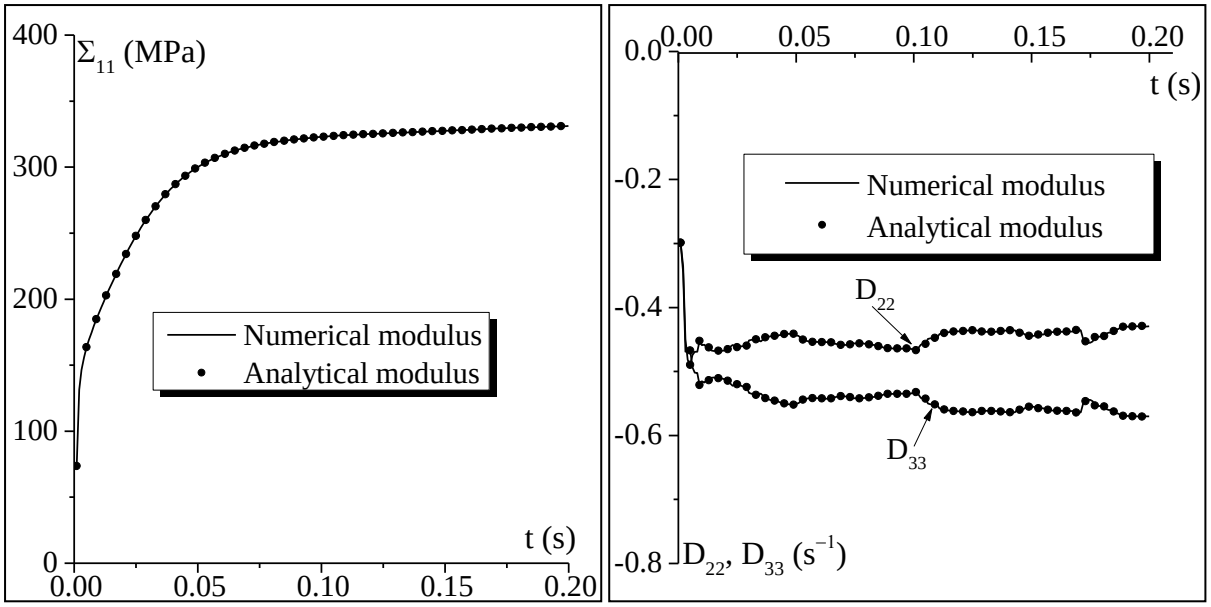


Figure 23. Evolution of RD as function of time t for the simulations with: (a) loading $\mathbf{g}_1$, (b) loading $\mathbf{g}_2$.



(a) (b)

Figure 24. Comparison between the simulation results obtained by using the analytical and numerical consistent tangent moduli: (a) Evolution of the stress component Σ_{11} as a function of time t, (b) Evolution of the strain rate components D_{22} and D_{33} as functions of time t.

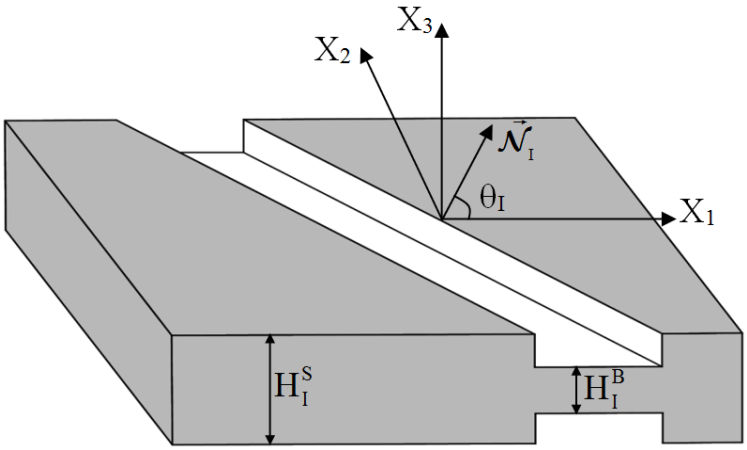


Figure F.1. Illustration of the initial imperfection approach.