Proper Generalized Decomposition (PGD) for the numerical simulation of polycrystalline aggregates under cyclic loading

Mohamed Aziz Nasri, Camille Robert, Amine Ammar, Saber El Arem, Franck Morel

Proper Generalized Decomposition (PGD) for the numerical simulation of polycrystalline aggregates under cyclic loading

Mohamed Aziz Nasri *, Camille Robert, Amine Ammar, Saber El Arem, Franck Morel

Arts et Métiers ParisTech, Campus d'Angers, LAMPA, 2, bd du Ronceray, 49035 Angers cedex 1, France

Keywords:
Finite Element method
Numerical modelling
Fatigue
Model reduction
PGD
Elasto-viscoplasticity

A B S T R A C T

The numerical modelling of the behaviour of materials at the microstructural scale has been greatly developed over the last two decades. Unfortunately, conventional resolution methods cannot simulate polycrystalline aggregates beyond tens of loading cycles, and they do not remain quantitative due to the plasticity behaviour. This work presents the development of a numerical solver for the resolution of the Finite Element modelling of polycrystalline aggregates subjected to cyclic mechanical loading. The method is based on two concepts. The first one consists in maintaining a constant stiffness matrix. The second uses a time/space model reduction method. In order to analyse the applicability and the performance of the use of a space–time separated representation, the simulations are carried out on a three-dimensional polycrystalline aggregate under cyclic loading. Different numbers of elements per grain and two time increments per cycle are investigated. The results show a significant CPU time saving while maintaining good precision. Moreover, increasing the number of elements and the number of time increments per cycle, the model reduction method is faster than the standard solver.

1. Introduction

High Cycle Fatigue (HCF) of metallic materials is an important issue for many components involved in many areas of mechanics. It is characterized by a low level of loading and leads to a number of cycles required to cause failure generally between 10^5 and 10^7. In this regime, the modelling of crack initiation observed at the scale where the mechanisms take place (scale of the microstructure) is not straightforward. Indeed, one of the most challenging tasks in the mechanics of materials is to get access to the mechanical response at the grain scale of a polycrystalline aggregate, especially when it is subjected to cyclic loading [1]. Local plasticity, microstructural heterogeneities, and crystal orientations play an essential role in the initiation and growth of cracks. Regarding this matter, it is often necessary to reach a stabilized state to get the various mechanical states of stress and strain in the critical zones.

From a numerical point of view, performing computation under monotonic loading is not sufficient. A simulation of the entire cyclic load history is needed. The numerical modelling of materials at the microstructural scale has been greatly developed over the last two decades. Different numerical methods have been used to explicitly model a polycrystal: the

* Corresponding author.
E-mail address: Mohamed-Aziz.Nasri@ensam.com (M.A. Nasri).

https://doi.org/10.1016/j.crme.2017.11.009

Notation

A	scalar	$\underline{\underline{A}}$	second-order tensor
$\{A\}$	vector	$\mathbb{A}$	fourth-order tensor
$[A]$	matrix		
$\underline{A}$	first-order tensor (vector)	$\{0\}$	zero vector

Finite Elements Method (FEM) [2–4] and the spectral method [5–7]. The spectral method, with the use of the Fast Fourier Transform (FFT) algorithm to solve numerically the integral equation associated with the heterogeneous problem, allows us to reach better performances than the FEM [8]. However, the FFT method is limited to the case of periodic problems. The FEM remains the most commonly used approach for numerical simulation of polycrystalline aggregates. The problem is solved step by step through an incremental scheme. For each increment loading, a Newton–Raphson-type resolution scheme is carried out. This method is taken as a reference and is commonly used in commercial and academic computation codes. Unfortunately, for problems with a large number of degrees of freedom (dof), the resolution of the linear system is expensive in terms of CPU time [9]. In fact, the tangent stiffness matrix is computed at each iteration and creates a significant computational cost for a high number of dofs. Moreover, the computation time is proportional to the number of increments and consequently to the number of simulated cycles.

To reduce significantly the computation time of the FEM, many authors seek to develop accelerated numerical methods that can be classified as follows:

(i) methods that deal with the local variable evolution at every point of the structure as the LATIN method [10,11], the direct cyclic method [12–14], the cycles skip method [15,16], and the Zarka's method [17,18];
(ii) methods that approximate the local variables at the critical point, called the Neuber methods [19];
(iii) methods based on model reduction and called Reduced Order Models (ROMs). They can greatly reduce the resolution cost of various mechanical problems appearing in many industrial applications. Two families are usually distinguished: the "*a posteriori*" methods, which consist of a construction of the reduced basis with an *a priori* knowledge of the solution, such as the Proper Orthogonal Decomposition (POD) [20,21], the Singular Value Decomposition (SVD) [22], and the Centroidal Voronoi Tessellations (CVT) [23]. A family of "*a priori*" model reduction techniques where the construction of the reduced-basis is done without an *a priori* knowledge of the solution, such as the A Priori Hyper Reduction method (APHR) [24,25] and the Proper Generalized Decomposition (PGD) [26–28].

According to the literature, accelerated numerical methods are generally used for the elastoplastic behaviour with von Mises plasticity criteria and linear kinematic hardening or nonlinear combined hardening. Some of these methods do not give a good satisfaction for any of the constitutive equations used for the materials (macroscopic or microscopic scale). For example, Zarka's method is limited to the case of linear kinematic hardening; Neuber's method is restricted to the confined plasticity case.

This paper presents the development of a numerical method based on a model reduction approach (technique) for the resolution with the FEM of nonlinear elasto-viscoplastic problems. The main purpose is to reduce the CPU time while keeping a good precision of the mechanical fields. Two approaches are used. The first one consists in keeping a constant stiffness matrix [29] all along the simulation in order to perform a factorization. The second is based on the use of a model reduction method to solve the space/time problem by decoupling the spatial and time coordinates. This is the idea of the Proper Generalized Decomposition (PGD) method [30]. The general principle consists in computing the solution defined on the entire space/time domain, in the form of a sum of product functions of each variable. This method has shown its efficiency in solving problems with a large number of dofs [31]. The space–time separated representation was initially introduced by Ladevèze et al. [10,32] ("*radial approximation method*") with the LATIN approach [33,34], a powerful nonlinear solver. A review of the state of the art and more recent ex-tensions of the LATIN-PGD method in the case of nonlinear solid mechanics problems could be found in [35,36]. Then, a new family of PGD methods has been proposed by Ammar et al. [26,37] that circumvent the difficulty related to multidimensional problems. This new family has been very successful for many applications: the quantum chemistry field [38], the chemical master equation [39], the resolution of the multidimensional, and parametric problems [40,41] and nonlinear problem in rheology [42]. Recently model reduction methods opened new perspectives for an accurate and fast simulation of mechanical systems [43–45]; however, nonlinear history-dependent behaviours remain still today challenging scenarios for the application of these techniques. For instance, Bergheau et al. [45] used the PGD method as a space–time integrator of the elastoplastic problem under cyclic loadings to determine the elastoplastic states with reducing the computation times. Similarly to Bergheau's work, we show in this paper a new methodology to develop the PGD method coupled to the FEM for problems involving high nonlinearity as well as a large number of dofs (scale of the microstructure). This type of application where plasticity dominates the material behaviour allows us to highlight the efficiency of the PGD solver in terms of precision and computation time when the number of space–time functions increases.

In this paper we focus on the application of the PGD in the field of polycrystal plasticity. Many computations have been carried out on polycrystalline aggregates with different mesh densities and time discretizations. Comparisons with the incremental method results, conventionally used in commercial computation codes, at the macroscopic (aggregate averages) and mesoscopic (grains averages) scales are presented and commented. In addition, the performance of the method in terms of CPU time is discussed.

2. Numerical methods

A structure defined in a spatial domain Ω is considered and submitted to a mechanical loading that evolves in the time interval $[0, T]$. The applied loading can be:

- a body force f_d in Ω;
- a surface force on the boundary part $\partial\Omega_2$;
- a fixed displacement field on another part $\partial\Omega_1$.

The objective is to determine the mechanical responses at each point of the structure and at any time instant. This is performed by solving the equilibrium, compatibility and behaviour equations. These equations are discretized in time and space. The spatial discretization is performed by the FEM and the time problem is traditionally solved by the incremental method. In the following, the dependence of space and time will be noted x and t, respectively.

2.1. Finite Element Method

The FEM is the most used method for solving mechanical problems [46]. The static equilibrium condition writes (by neglecting the body force f_d):

$$\underline{\nabla}_x \cdot \underline{\underline{\sigma}}(x,t) = 0 \qquad \forall\, (x \in \Omega),\ (t \in [0,T]) \tag{1}$$

where σ is the stress field. Using the Principle of Virtual Power, the weak formulation obtained is:

$$\int_\Omega (\underline{\nabla}_x \cdot \underline{\underline{\sigma}}) \cdot \underline{u}^* \, \mathrm{d}V = 0 \tag{2}$$

where u^* defines a test field. Applying the divergence theorem, the weak formulation becomes:

$$\int_\Omega \underline{\underline{\sigma}} : \underline{\underline{\varepsilon}}^* \, \mathrm{d}V = \int_{\partial\Omega} (\underline{\underline{\sigma}} \cdot \underline{n}) \cdot \underline{u}^* \, \mathrm{d}S \tag{3}$$

where $\underline{\underline{\varepsilon}}^*$ is the strain field deriving from the virtual displacement $\underline{u}^*$ and $\underline{n}$ is the normal vector to $\partial\Omega$. Taking into account the boundary conditions, Eq. (3) can be written as follows:

$$\int_\Omega \underline{\underline{\sigma}} : \underline{\underline{\varepsilon}}^* \, \mathrm{d}V = \int_{\partial\Omega_1} (\underline{\underline{\sigma}} \cdot \underline{n}) \cdot \underline{u}^* \, \mathrm{d}S + \int_{\partial\Omega_2} \underline{F_d} \cdot \underline{u}^* \, \mathrm{d}S \tag{4}$$

where F_d represents surface force on $\partial\Omega_2$. As the test field u^* vanishes on $\partial\Omega_1$, the above equation can be written as:

$$\int_\Omega \underline{\underline{\sigma}} : \underline{\underline{\varepsilon}}^* \, \mathrm{d}V = \int_{\partial\Omega_2} \underline{F_d} \cdot \underline{u}^* \, \mathrm{d}S \tag{5}$$

The FEM is based on the construction of an approximation of the field variables u via a subdomain. This approximation is constructed by the approximate values of the field at the nodes of the element. In other words, instead of calculating the solution to the problem over an infinite set of possibilities, we limit the search space to the function U and use basic functions giving the displacement field u and u^* at any point in Ω. Thus, a discrete form of the displacement field is given by:

$$\{u\} = [N]\{U\} \quad \text{and} \quad \{u^*\} = [N]\{U^*\} \tag{6}$$

where $\{U\}$ and $\{U^*\}$ are the nodal values and $[N]$ are the commonly used shape functions. The strain fields ε and ε^* in each point of the structure can be written as functions of the displacement field using the above equation, with:

$$\underline{\underline{\varepsilon}} = \frac{1}{2}(\nabla \underline{u} + \nabla^{\mathsf{T}} \underline{u}) \quad \text{and} \quad \underline{\underline{\varepsilon}}^* = \frac{1}{2}(\nabla \underline{u}^* + \nabla^{\mathsf{T}} \underline{u}^*) \tag{7}$$

$$\{\varepsilon\} = \begin{Bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ 2\varepsilon_{12} \end{Bmatrix} = [B]\{U\} \quad \text{and} \quad \{\varepsilon^*\} = [B]\{U^*\} \tag{8}$$

where ∇ represents the gradient operator and $[B]$ is the matrix of spatial derivatives of the shape functions. According to the above expressions of the strain tensors, using Eqs. (3) and (5), we obtain:

$$\int_{\Omega} \langle \sigma \rangle \{\varepsilon^*\} \, \mathrm{d}V = \langle u^* \rangle \int_{\Omega} [B]\{\sigma\} \, \mathrm{d}V = \langle u^* \rangle \{F^{\mathrm{int}}\} \tag{9}$$

$$\int_{\partial\Omega_2} \langle F_d \rangle \{u^*\} \, \mathrm{d}S = \langle u^* \rangle \{F^{\mathrm{ext}}\} \tag{10}$$

where $\{F^{\mathrm{int}}\}$ and $\{F^{\mathrm{ext}}\}$ are internal and external forces, respectively. The resolution of a mechanical problem using the FEM consists in searching the displacement field at each time (through an incremental scheme) by verifying the following weak formulation:

$$\{F^{\mathrm{ext}}\} - \{F^{\mathrm{int}}\} = \int_{\partial\Omega_2} \langle F_d \rangle \{u^*\} \, \mathrm{d}S - \langle u^* \rangle \int_{\Omega} [B]\{\sigma\} \, \mathrm{d}V = \{0\} \tag{11}$$

where $\{\sigma\}$ depends on the strain evolution.The equilibrium of Eq. (5) can be written as:

$$\langle u^* \rangle \int_{\Omega} [B]\{\sigma\} \, \mathrm{d}V = \langle u^* \rangle \left(\int_{\partial\Omega_2} [N]\{F_{\mathrm{d}}\} \, \mathrm{d}S \right) \tag{12}$$

After simplification, the overall equilibrium can be written as:

$$[K]\{U\} = \{F^{\mathrm{ext}}\} \tag{13}$$

where $[K]$ is the stiffness matrix. The resolution of the time problem in Eq. (13) is usually performed by the incremental method. However, the overall evolution is not resolved globally in one step, but the study interval is discretized in time increments and, consequently, Eq. (13) is solved in terms of displacement increment.

2.2. Nonlinear resolution strategy

2.2.1. Implicit approach (nonlinearity at the left-hand side)

The main goal is to solve the nonlinear Finite Element equations that satisfy equilibrium (13). In other words, the resulting force of the internal stresses must balance the externally applied loads. This leads to a vector representation given by the values of the displacement field $\{U\}$ in order to minimize the residual $\{R\}$:

$$\{R(U)\} = \{[K(U)] \cdot \{U\}\} - \{F^{\mathrm{ext}}\} = \{0\} \tag{14}$$

The resolution of a nonlinear system of equations such as (14) relies, often, on iterative algorithms such as in the Newton–Raphson method. When dealing with linear elastic materials, the relation (14) is linear. In this case, no Newton–Raphson linearization or temporal discretization is required. In general, the most frequently used methods for the resolution of nonlinear Finite Element equations (e.g., equation (14)) are the Newton–Raphson ones. Alternatively, a class of methods known as quasi-Newton methods have been developed for solving nonlinear systems of equations (see [47,48]) such as BFGS (Broyden–Fletcher–Goldfarb–Shanno). This last method is the most efficient one in the family of quasi-Newton solvers (see [49,50]).

The Newton method, just like the resolution of scalar equations, is based, for each iteration, on the Taylor expansion of the residual of the previous iteration:

$$\{R(U)\}^{j+1} \approx \{R(U)\}^{j} + \frac{\partial R}{\partial U}(\{U\}^{j})\{\delta U\}^{j} \tag{15}$$

where j is the iteration of the Newton–Raphson method, $\{R(U)\}^{j}$ is the residual given at the iteration j. The term $\{\delta U\}^{j}$ is the displacement increment made on the solution at the current iteration and defined by:

$$\{U\}^{j+1} = \{U\}^{j} + \{\delta U\}^{j} \tag{16}$$

The correction $\{\delta U\}^{j}$ is therefore repeated in order to find the quantity of $\{U\}^{j+1}$ close to $\{U\}^{j}$ in order to minimize the residual of Eq. (15). It is therefore needed to solve the following linear equation system:

$$\{R(U)\}^{j} + [K_{\mathrm{T}}]\{\delta U\}^{j} = 0 \tag{17}$$

The correction at iteration j is given by:

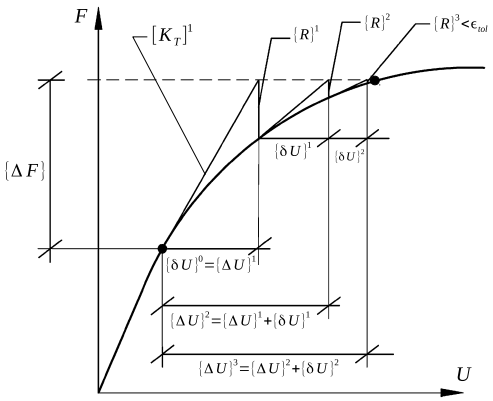


Fig. 1. Strategy for equilibrium resolution with elastoplastic loading.

$$\{\delta U\}^j = -[K_{\mathrm{T}}^{-1}]^j \{R\}^j \tag{18}$$

with $[K_{\mathrm{T}}]$ is the tangent stiffness matrix [51,52] expressed at each iteration j by:

$$[K_{\mathrm{T}}]^j = \nabla_U \cdot [R(U^j)] = \frac{\partial R}{\partial U}(\{U\}^j) = \int_\Omega \frac{\partial \varepsilon}{\partial U} : \left[C^{\mathrm{nl}}\right]^j : \varepsilon^* \,\mathrm{d}V = \int_\Omega \frac{\partial \varepsilon}{\partial U} : \left(\frac{\partial \sigma}{\partial \varepsilon}\right)^j : \varepsilon^* \,\mathrm{d}V = \int_\Omega [B]^{\mathrm{T}} \left[\frac{\partial \sigma}{\partial \varepsilon}\right]^j [B] \cdot \{\varepsilon^*\} \,\mathrm{d}V \tag{19}$$

where $[C^{\mathrm{nl}}]$ is the nonlinear operator, the term $[\frac{\Delta\sigma}{\Delta\varepsilon}]$ reflecting nonlinearity. Let $\{\Delta U\}$ be an increment of the displacement field. Taking into account the previous relations, the incremental relationship of the overall equilibrium is given by:

$$[K_{\mathrm{T}}]^j \{\delta U\}_i^j = \{\Delta F^{\mathrm{ext}}(\Delta U^{j-1})\}^j \tag{20}$$

where the subindex 'i' represents the current increment. The problem is solved step by step through an incremental scheme. The computations consist in applying an increment $\{\Delta F^{\mathrm{ext}}\}$ and computing the corresponding displacement increment from Eq. (20) at iteration j. If convergence is obtained at this iteration, we proceed further with the next increment $i+1$. Otherwise, a new increment of displacement is calculated at the following iteration $j+1$ (Eq. (16)) from a new residual $\{R(U)\}^{j+1}$ (Eq. (15)). This iterative scheme is repeated until the convergence of the solution is obtained for each increment. This strategy is illustrated in Fig. 1. That is, the norm of R in Eq. (14), which is required to be lower than a threshold value ϵ_{tol}, can be written as:

$$\|Res\| = \frac{\|R\|}{\|\Delta F^{\mathrm{ext}}\|} < \epsilon_{\mathrm{tol}} \tag{21}$$

If the norm of the residual $\|Res\|$ verifies $\|Res\| < \epsilon_{\mathrm{tol}}$, we can continue with the next increment i. On the contrary, we recompute additional iteration until convergence. This method, called the incremental method, is taken as a reference and is very used in commercial and academic computation codes. The major inconvenience is that the tangent stiffness matrix is recalculated at every iteration. Indeed, when the Newton–Raphson method is applied to solve the problem at iteration j subjected to finite inelastic deformations, the linearization of the weak formulation is needed. With this linearization, it is necessary to compute the tangent matrix [53], which leads to a significant computational cost, especially with a high number of dofs. In the following, the second version will be described to represent the nonlinear term.

2.2.2. Semi-explicit approach (nonlinearity on the right-hand side)

In this case, it is assumed that nonlinearity is only due to material behaviour. The elastic stiffness tensor $[C^{\mathrm{el}}]$ defines the relationship between the stress increments and the strain increments following the nonlinear equation:

$$\{\Delta\sigma\} = [C^{\mathrm{el}}](\{\Delta\varepsilon - \Delta\varepsilon^{\mathrm{p}}\}) \tag{22}$$

were $\{\Delta\varepsilon^{\mathrm{p}}\}$ is the increment of the plastic strain. By using relation (22), Eq. (9) can be written as:

$$\int_\Omega \sigma : \varepsilon^* \,\mathrm{d}V = \langle u^* \rangle \int_\Omega [B]^\intercal \{\Delta\sigma\} \,\mathrm{d}V = \langle u^* \rangle \int_\Omega [B]^\intercal [C^{\mathrm{el}}](\{\Delta\varepsilon - \Delta\varepsilon^{\mathrm{p}}\}) \,\mathrm{d}V$$
$$= \langle u^* \rangle \left(\int_\Omega [B]^\intercal [C^{\mathrm{el}}] \{\Delta\varepsilon\} \,\mathrm{d}V - \int_\Omega [B]^\intercal [C] \{\Delta\varepsilon^{\mathrm{p}}\} \,\mathrm{d}V \right) \tag{23}$$

Using the relations of the previous section, Eq. (23) can be expressed as:

$$\int_\Omega [B]^\intercal \{\Delta\sigma\} \,\mathrm{d}V = [K]\{\Delta U\} - \int_\Omega [B]^\intercal [C^{\mathrm{el}}] \{\Delta\varepsilon^{\mathrm{p}}\} \,\mathrm{d}V \tag{24}$$

That is, the equation for the overall equilibrium of Eq. (13) for the previous method, will be replaced by:

$$[K]\{\Delta U\}_i^j = \{\Delta F^{\mathrm{ext}}\}_i^j + \{\Delta Q^{\mathrm{p}}\}_i^j \tag{25}$$

with:

$$[K] = \int_\Omega [B]^\intercal [C^{\mathrm{el}}][B] \,\mathrm{d}V \tag{26}$$

$$\{\Delta F^{\mathrm{ext}}\} = \int_\Omega [B]^\intercal \{\Delta\sigma^{\mathrm{el}}\} \,\mathrm{d}V = \int_\Omega [B]^\intercal \{\Delta\sigma^{\mathrm{SA}}\} \,\mathrm{d}V = \int_{\partial\Omega_2} [N]\{F_{\mathrm{d}}\} \,\mathrm{d}S \tag{27}$$

$$\{\Delta Q^{\mathrm{p}}\} = \int_\Omega [B]^\intercal [C^{\mathrm{el}}] \{\Delta\varepsilon^{\mathrm{p}}\} \,\mathrm{d}V \tag{28}$$

where $\{\Delta Q^{\mathrm{p}}\}$ is the vector of nodal forces resulting from the increment of the plastic strain field $\{\Delta\varepsilon^{\mathrm{p}}\}$. $\{\sigma^{\mathrm{SA}}\}$ is the field of statically admissible stresses calculated from Eq. (22). $\{\Delta Q^{\mathrm{p}}\}$ represents the nonlinear term of the equilibrium of Eq. (25). The idea to put the nonlinearity in the right-hand side of Eq. (25) has also been used by Spiliopoulos et al. [54,55] in the Direct Method to predict cyclic creep response [54,55]. Spiliopoulos et al. [29] also estimate the behaviour of an elastic-perfectly plastic structure subjected to cyclic loading. Maouche and Pommier [12,13] determine directly the asymptotic solution to a problem by using this approach. Maouche [12] uses the incremental method for the resolution of the global step (i.e. the equilibrium equation) and Pommier [13] uses the decomposition into Fourier series to reduce the computing time of the global step.

Within the framework of the small perturbation theory, the matrix of spatial derivatives of the shape functions $[B]$ is constant. Assuming that the elastic material properties are time independent, the matrix $[K]$ is constant. Thus, maintaining a constant stiffness matrix allows us to do just one single factorization in the first iteration of the simulation. This allows high CPU time saving compared to the incremental method of the previous section.

The solution to the above spatial problem is performed by the FEM. The resolution of the temporal problem is performed by the incremental method. Knowing the increment of the plastic strain field $\{\Delta\varepsilon^{\mathrm{p}}\}$, the first step consists in solving the equation of equilibrium with a global system over the whole structure (Eq. (25)). From a practical point of view, the initialization of the increment of the nodal forces resulting from the increment of the plastic strain field is done with:

$$\{\Delta Q^{\mathrm{p}}\}^{\mathrm{I}} = \int_\Omega [B]^\intercal [C^{\mathrm{el}}] \{\Delta\varepsilon^{\mathrm{p}}\}^{\mathrm{I}} \,\mathrm{d}V = \{0\} \tag{29}$$

where $\{\Delta\varepsilon^{\mathrm{p}}\}^{\mathrm{I}}$ is the increment of the initial plastic strain field which is equal to zero for the first iteration. Finally, the convergence criterion is based on the evaluation of the nodal force, with:

$$\frac{\|F(\Delta\sigma^{\mathrm{SA}}) - F(\Delta\sigma^{\mathrm{PA}})\|}{\|F(\Delta\sigma^{\mathrm{SA}})\|} = \frac{\| \int_\Omega ([B]^\intercal \{\sigma^{\mathrm{SA}}\}_i^j) \,\mathrm{d}V - \int_\Omega ([B]^\intercal \{\sigma^{\mathrm{PA}}\}_i^j) \,\mathrm{d}V \|}{\| \int_\Omega ([B]^\intercal \{\sigma^{\mathrm{SA}}\}_i^j) \,\mathrm{d}V \|} < \epsilon_{\mathrm{tol}} \tag{30}$$

where $\{\Delta\sigma^{\mathrm{PA}}\}$ is the field of plastically admissible stress calculated from constitutive equations, $F(\Delta\sigma^{\mathrm{SA}})$ and $F(\Delta\sigma^{\mathrm{PA}})$ are the nodal forces resulting from the increment of the field of statically and plastically admissible stresses respectively. The stress field $\{\sigma^{\mathrm{PA}}\}$ is called plastically admissible if it satisfies the constitutive equations. In the following, this method will be called incremental right method.

2.3. The PGD method

In the present work, we use a model reduction method in the framework of a space–time formulation. Thanks to the PGD method the vector of displacement increment is given by:

$$[\Delta U]_{(x,t)} \approx [\Delta U]_m = \sum_{i=1}^{m} \alpha^i \{\Delta U_x\}^i \otimes \{\Delta U_t\}^i \tag{31}$$

where $\{\Delta U_x\}^i$ is a vector of space function which size is equal to the number of dof, $\{\Delta U_t\}^i$ is the temporal function which size is equal to the discretization size of the temporal dimension and m is the reduced-basis order. In the following paragraph, the implementation of the PGD method in the case of the plasticity problem is presented.

2.3.1. Plasticity problem

There are different ways to implement the PGD method. Here, we focus on the version of the time–space decomposition with the tensor form. Supposing that the nonlinearity in the right-hand side is written as follows:

$$\{\Delta F^{\text{nl}}(x,t)\} = \{\Delta F^{\text{ext}}(x,t)\} + \{\Delta Q^{\text{p}}(x,t)\} \tag{32}$$

If the time steps N_t (again time in the sense of loading) is to be considered, the equilibrium equation with the non-incremental form can be written as:

$$\begin{cases} [K]\cdot\{\Delta U\}_1 &= \{\Delta F^{\text{nl}}\}_1 \\ [K]\cdot\{\Delta U\}_2 &= \{\Delta F^{\text{nl}}\}_2 \\ & \vdots \\ [K]\cdot\{\Delta U\}_{N_t} &= \{\Delta F^{\text{nl}}\}_{N_t} \end{cases} \tag{33}$$

The plastic history is given by the matrix $[F^{\text{nl}}]$:

$$[F^{\text{nl}}] = [\{\Delta F^{\text{nl}}\}_1, \{\Delta F^{\text{nl}}\}_2, \ldots, \{\Delta F^{\text{nl}}\}_{N_t}] \tag{34}$$

The PGD is an iterative method consisting in searching an approximation of the displacement field $\Delta U(x,t)$ under a form giving by Eq. (31). To compute the unknowns $\{\Delta U_x^i\}$, $\{\Delta U_t^i\}$ and α^i, the PGD consists in three steps: enrichment step, projection step, and convergence-checking step. Before detailing the three steps of resolution of PGD, a tensor decomposition of the operators is requested. In our case, we have:

$$K(x,t) = [K_x] \otimes [K_t] \tag{35}$$

$$\Delta F^{\text{nl}}(x,t) = \sum_{i=1}^{f_{\Delta F}} \{\Delta F_x\}^i \otimes \{\Delta F_t\}^i \tag{36}$$

$$\Delta U(x,t) = \sum_{i=1}^{f_{\Delta u}} \alpha^i \left(\{\Delta U_x\}^i \otimes \{\Delta U_t\}^i \right) \tag{37}$$

where $\{\Delta U_x\}^i$ is a vector that corresponds to the space dimension (i.e. the number of dofs) and $\{\Delta U_t\}^i$ corresponds to the temporal dimension (i.e. the number of time steps). The stiffness matrix $[K]$ is constant and is decomposed with one single product where K_x represents the $(N_x \times N_x)$ stiffness matrix and K_t the $(N_t \times N_t)$ identity matrix. The plastic history evolution $\Delta F^{\text{nl}}(x,t)$ has to be updated after each enrichment step and has to be written in the separated tensor form. This is done with an "*a posteriori*" method requesting the expansion of a large matrix before obtaining the space–time separated form. It is important to recall that the number of snapshots in (34) could be extremely large. To alleviate such a calculation, a Truncated SVD is used to decompose the second member. Also, a POD-based reduced basis for the plastic history representation can be used and proposed in [56].

The matrix $[F^{\text{nl}}]$ given by Eq. (34) contains the set of snapshots of the second member (the plastic history) with N_x rows and N_t columns. Applying a truncated SVD, the matrix $[F^{\text{nl}}]$ can be factorized into three matrices:

$$[F^{\text{nl}}] = [\Delta F_x][S][\Delta F_t] \tag{38}$$

The matrices $[\Delta F_x]$ and $[\Delta F_t]$ contain respectively the right and the left singular vectors of the matrix $[F^{\text{nl}}]$. Here, $[S]$ is a diagonal matrix that contains the $f_{\Delta F}$ highest singular values, which is the most representative information extracted from $[F^{\text{nl}}]$. This choice is made from a truncation that depends on an parameter ϵ_{SVD}, which is normally fixed to 10^{-6}. Thus, it is possible to obtain a space/time decomposition of the second member whose the matrices $[\Delta F_x]$ and $[\Delta F_t]$ are equivalent to a space and temporal basis of dimensions $(N_x \times f_{\Delta F})$ and $(N_t \times f_{\Delta F})$ respectively.

By using the separated representation form of the problem, the equilibrium equation can be written as:

$$([K_x]\otimes[K_t])\sum_{i=1}^{f_{\Delta u}}\alpha^i\left(\{\Delta U_x\}^i\otimes\{\Delta U_t\}^i\right)=\sum_{j=1}^{f_{\Delta F}}\{\Delta F_x\}^j\otimes\{\Delta F_t\}^j+\mathcal{R} \tag{39}$$

where $\mathcal{R}$ is a residual drawn from the approximation of the solution. The different steps to solve Eq. (39) by the PGD method are as follows.

Enrichment step Let us assume that we are at the order m of Eq. (31). The first $m-1$ functions $\{\Delta U_x\}^i$ and $\{\Delta U_t\}^i$ and the coefficients α^i are assumed to be known. Thus, at this stage, it can be written:

$$[\Delta U_m]=\sum_{i=1}^{m-1}\alpha_i\left(\{\Delta U_x\}^i\otimes\{\Delta U_t\}^i\right)+\{\Delta U_x\}^m\otimes\{\Delta U_t\}^m \tag{40}$$

where $\{\Delta U_x\}^m$ and $\{\Delta U_t\}^m$ are the new enriched vectors. We look here for the new functions $\{\Delta U_x\}^m$ and $\{\Delta U_x\}^m$. Introducing Eq. (40) in Eq. (39), we obtain:

$$[K_x]\{\Delta U_x\}^m\otimes[K_t]\{\Delta U_t\}^m=\sum_{j=1}^{f_{\Delta F}}\{\Delta F_x\}^j\otimes\{\Delta F_t\}^j-([K_x]\otimes[K_t])\sum_{i=1}^{m-1}\alpha^i(\{\Delta U_x\}^i\otimes\{\Delta U_t\}^i)+\mathcal{R}^{m-1} \tag{41}$$

To calculate the new enrichment vectors, Eq. (41) is projected onto each of the unknowns $\{\Delta U_x\}^m$ and $\{\Delta U_t\}^m$, with:

$$\begin{aligned}&\left\langle\left([K_x]\{\Delta U_x\}^m\otimes[K_t]\{\Delta U_t\}^m\right),\{U_t\}^m\right\rangle_{L^2([0,T])}\\&=\left\langle\sum_{j=1}^{f_{\Delta F}}\{\Delta F_x\}^j\otimes\{\Delta F_t\}^j-([K_x]\otimes[K_t])\sum_{i=1}^{m-1}\alpha^i(\{\Delta U_x\}^i\otimes\{\Delta U_t\}^i),\{U_t\}^m\right\rangle_{L^2([0,T])}+\left\langle\mathcal{R}^{m-1},\{U_t\}^m\right\rangle_{L^2([0,T])}\end{aligned} \tag{42}$$

$$\begin{aligned}&\left\langle\left([K_x]\{\Delta U_x\}^m\otimes[K_t]\{\Delta U_t\}^m\right),\{U_x\}^m\right\rangle_{L^2(\Omega)}\\&=\left\langle\sum_{j=1}^{f_{\Delta F}}\{\Delta F_x\}^j\otimes\{\Delta F_t\}^j-([K_x]\otimes[K_t])\sum_{i=1}^{m-1}\alpha^i(\{\Delta U_x\}^i\otimes\{\Delta U_t\}^i),\{U_x\}^m\right\rangle_{L^2(\Omega)}+\left\langle\mathcal{R}^{m-1},\{U_x\}^m\right\rangle_{L^2(\Omega)}\end{aligned} \tag{43}$$

where $\langle.,.\rangle_{L^2(\Omega)}$ and $\langle.,.\rangle_{L^2(]0,T[)}$ are the scalar products of L^2 defined on Ω and $[0,T]$. In addition, in order to improve optimality of the basis of the solution in the Galerkin sense, the residual must be orthogonal to the $\{\Delta U_x\}^m$, $\{\Delta U_t\}^m$ functions. Thus, we obtain:

$$\left(\langle\Delta U_t\rangle^m[K_t]\{\Delta U_t\}^m[K_x]\right)\cdot\{\Delta U_x\}^m=\sum_{j=1}^{f_{\Delta F}}\langle\Delta U_t\rangle^m\{\Delta F_t\}^j\{\Delta F_x\}^j-\sum_{i=1}^{m-1}\langle\Delta U_t\rangle^m[K_t]\{\Delta U_t\}^i\alpha^i[K_x]\{\Delta U_x\}^i \tag{44}$$

$$\left(\langle\Delta U_x\rangle^m[K_x]\{\Delta U_x\}^m[K_t]\right)\cdot\{\Delta U_t\}^m=\sum_{j=1}^{f_{\Delta F}}\langle\Delta U_x\rangle^m\{\Delta F_x\}^j\{\Delta F_t\}^j-\sum_{i=1}^{m-1}\langle\Delta U_x\rangle^m[K_x]\{\Delta U_x\}^i\alpha^i[K_t]\{\Delta U_t\}^i \tag{45}$$

Equations (44) and (45) are two equations with two unknowns $\{\Delta U_x\}^m$ and $\{\Delta U_t\}^m$. In this case, to compute $\{\Delta U_x\}^m$ and $\{\Delta U_t\}^m$, we use the fixed point algorithm. The general description is as follows: $\{\Delta U_t\}^m$ (respectively $\{\Delta U_x\}^m$) is fixed, and we solve Eq. (44) (respectively Eq. (45)) to obtain the new value of $\{\Delta U_x\}^m$ ($\{\Delta U_t\}^m$, respectively). This iterative scheme is repeated until convergence:

$$\|(\{\Delta U_x\}^m\{\Delta U_t\}^m)^q-(\{\Delta U_x\}^m\{\Delta U_t\}^m)^{q-1}\|<\epsilon_{fp} \tag{46}$$

where ϵ_{fp} is a tolerance set by the user. It is important to notice that the stiffness matrix $[K_x]$ is constant, except for the first iteration of the calculation (we need to factorize $[K_x]$ in the first iteration). In addition, the term $\left(\langle\Delta U_x\rangle^m[K_x]\{\Delta U_x\}^m[K_t]\right)$ is a scalar, and $[K_t]$ is an identity matrix. Thus, the resolution of linear systems Eqs. (44) and (45) at each iteration is not expensive in terms of CPU time.

Once the calculation of $\{\Delta U_x\}^m$ and $\{\Delta U_t\}^m$ has been carried out successfully, and after normalization, the new solutions of $\{\Delta U_x\}^m$ and $\{\Delta U_t\}^m$ at the order m are given by:

$$\{\Delta U_x\}^m=\frac{\{\Delta U_x\}^m}{\|\{\Delta U_x\}^m\|}\qquad\text{and}\qquad\{\Delta U_t\}^m=\frac{\{\Delta U_t\}^m}{\|\{\Delta U_t\}^m\|} \tag{47}$$

Projection step The first m functions of $\{\Delta U_x\}^i$ and $\{\Delta U_t\}^i$ $(1 \leq i \leq m)$, solutions from the previous step, are known. At this stage, it is necessary to calculate the coefficients α^i $(1 \leq i \leq m)$. For this purpose, Eq. (39) is projected onto $\{\Delta U_x\}^j$ and $\{\Delta U_t\}^j$ $(1 \leq j \leq m)$, with:

$$\left\langle ([K_x] \otimes [K_t]) \sum_{i=1}^{f_{\Delta u}} \alpha^i \left(\{\Delta U_x\}^i \otimes \{\Delta U_t\}^i \right), \{\Delta U_x\}^j \otimes \{\Delta U_t\}^j \right\rangle_{L^2(\Omega \times [0,T])}$$
$$= \left\langle \sum_{j=1}^{f_{\Delta F}} \{\Delta F_x\}^j \otimes \{\Delta F_t\}^j, \{\Delta U_x\}^j \otimes \{\Delta U_t\}^j \right\rangle_{L^2(\Omega \times [0,T])} + \left\langle \mathcal{R}, \{\Delta U_x\}^j \otimes \{\Delta U_t\}^j \right\rangle_{L^2(\Omega \times [0,T])} \tag{48}$$

In addition, the residual is supposed to be orthogonal to $\{\Delta U_x\}^j$ and $\{\Delta U_t\}^j$, so it can be written as:

$$\sum_{i=1}^{m} \langle \Delta U_x \rangle^j [K_x] \{\Delta U_x\}^i \langle \Delta U_t \rangle^j [K_t] \{\Delta U_t\}^i \alpha^i = \sum_{k=1}^{f_{\Delta F}} \langle \Delta U_x \rangle^j \{\Delta F_x\}^k \langle \Delta U_t \rangle^j \{\Delta F_t\}^k \tag{49}$$

Corresponding to the above equation, to calculate the coefficients α^i, we are led to solve the following linear system:

$$[M]\{\alpha\} = \{D\} \tag{50}$$

With:

- $M_{ij} = \langle \Delta U_x \rangle^j [K_x] \{\Delta U_x\}^i \langle \Delta U_t \rangle^j [K_t] \{\Delta U_t\}^i$
- $D_j = \sum_{k=1}^{f_{\Delta F}} \langle \Delta U_x \rangle^j \{\Delta F_x\}^k \langle \Delta U_t \rangle^j \{\Delta F_t\}^k$
- $\langle \alpha \rangle = \{\alpha^1, \alpha^1, ..., \alpha^m\}$

$[M]$ is a matrix of dimension $(m \times m)$. Thus Eq. (50) is a linear system to solve with a low computational cost.

Checking convergence At this stage, an estimation of the convergence of the PGD algorithm at the order m is needed. For this, the residual of Eq. (39) is defined by:

$$\mathcal{R} = \sum_{i=1}^{m} \alpha^i \left([K_x] \{\Delta U_x\}^i \otimes [K_t] \{\Delta U_t\}^i \right) - \sum_{j=1}^{f_{\Delta F}} \{\Delta F_x\}^j \otimes \{\Delta F_t\}^j \tag{51}$$

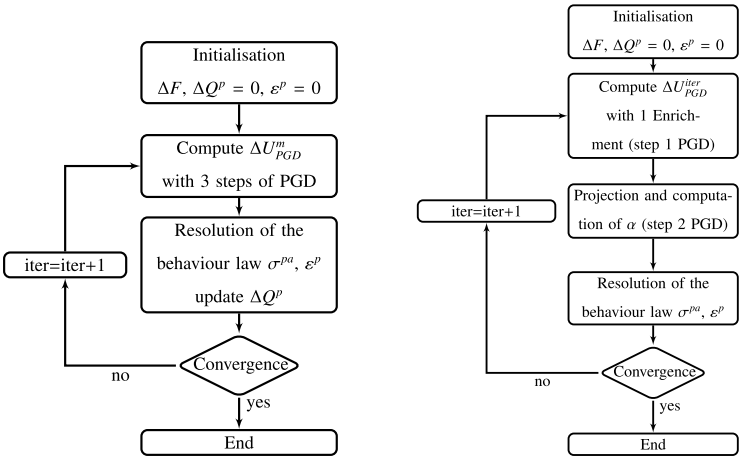
If the norm of the residual verifies $\|\mathcal{R}\| < \varepsilon_{\text{PGD}}$, where ε_{PGD} is a tolerance set by the user, the algorithm converges at this order m, and the solution to the problem is expressed with Eq. (31). Otherwise, a return to the enrichment step and an additional enrichment of the basis with different $\{U_x\}^{m+1}$ and $\{U_t\}^{m+1}$ have to be done.

2.3.2. Variants of the PGD method

The resolution with the PGD method is carried out successively until the convergence of the plastic strain field. However, the nonlinear term $\{\Delta Q^{\text{p}}\}$ of the equilibrium Eq. (25) is unknown and its computation is made iteratively. Two possibilities have been proposed to implement the PGD method. In the first version, we update the nonlinear term for a displacement field calculated for all the PGD functions. Here, the idea consists firstly in decoupling the equations of the problem (global and local step). Then, the resolution is carried out separately over the entire space/time domain using the PGD method as a solver of the global step (equilibrium equation). In this case, the nonlinear term $\{\Delta Q^{\text{p}}\}$ is updated from a converged displacement field $[\Delta U(x,t)]$ for a given second member. In the second version, we update the nonlinear term for a displacement field calculated for a single enrichment of the PGD. Thus, in this version, the nonlinear term is computed incrementally, and the numerical resolution of the problem is carried out iteratively. Each iteration leads to the following four steps: (i) enrichment, (ii) projection, (iii) local step, (iv) checking convergence. The algorithms associated with the two versions are illustrated in Fig. 2.

In the first version, the convergence criterion is based on the evaluation of the Eq. (30), whereas for the second version, the convergence is reached when:

$$\frac{\| [K] \sum_{i=1}^{m} \alpha^i \left(\{\Delta U_x\}^i \otimes \{\Delta U_t\}^i \right) - \sum_{j=1}^{f_{\Delta \mathcal{F}}} \{\Delta F_x\}^j \otimes \{\Delta F_t\}^j \|}{\| \sum_{j=1}^{f_{\Delta \mathcal{F}}} \{\Delta F_x\}^j \otimes \{\Delta F_t\}^j \|} < \epsilon_{\text{tol}} \tag{52}$$



(a) PGD version 1 (b) PGD version 2

Fig. 2. The two version of PGD resolution algorithm.

3. Constitutive equation of the viscoplastic crystalline behaviour

The viscoplastic crystalline behaviour law used in this paper was introduced by Meric and Cailletaud [57]. It is commonly used for the numerical simulation of polycrystalline aggregates [9,58,59]. The modelling is made in small strain assumption as usually done in high cycle fatigue modelling of materials. The local strain is decomposed into an elastic and an viscoplastic part:

$$\{\varepsilon\} = \{\varepsilon^{\mathrm{e}}\} + \{\varepsilon^{\mathrm{p}}\} \tag{53}$$

The Hooke law is used to describe the elastic behaviour. Cubic elasticity is considered and characterised by three coefficients defined in the crystal coordinate system: C_{11}, C_{22}, and C_{12}. The orientation tensor $\underline{\underline{m}}$ (Eq. (54)) is calculated on each slip system s by the tensor product of the normal to the slip plane $\underline{n}^s$ with the slip direction $\underline{l}^s$. This tensor allows one to compute the shear stress τ^s (Eq. (55)) and the plastic strain rate tensor $\dot{\varepsilon}^{\mathrm{p}}$ (Eq. (56)) on each slip system s.

$$\underline{\underline{m}}^s = \frac{\left(\underline{n}^s \otimes \underline{l}^s + \underline{l}^s \otimes \underline{n}^s\right)}{2} \tag{54}$$

$$\tau^s = \underline{\underline{m}}^s : \underline{\underline{\sigma}} \tag{55}$$

$$\underline{\underline{\dot{\varepsilon}}}^{\mathrm{p}} = \sum_s \dot{\gamma}^s \underline{\underline{m}}^s \tag{56}$$

where $\dot{\gamma}^s$ is the plastic slip rate (Eq. (57)) for the slip system s. τ^s is the resolved shear stress τ^s acting on s, x^s is the kinematic hardening (Eq. (58)) and r^s the isotropic hardening (Eq. (59)).

$$\dot{\gamma}^s = \left\langle \frac{|\tau^s - x^s| - r_0 - r^s}{K} \right\rangle^n \mathrm{sign}\left(\tau^s - x_s\right) = \dot{\nu}^s \,\mathrm{sign}\left(\tau^s - x^s\right) \tag{57}$$

$$x^s = c\alpha^s \text{ with } \dot{\alpha}^s = \dot{\gamma}^s - d\dot{\nu}^s\alpha^s \tag{58}$$

$$r^s = r_0 + Q\sum_r h^{\mathrm{rs}}\left(1 - \exp\left(-b\nu^r\right)\right) \tag{59}$$

where K and n are the viscosity parameters, ν^s is the cumulated viscoplastic slip on the slip system s and r_0 corresponds to the initial yield shear stress. Hardening material parameters are defined by c and d for kinematic hardening, and Q and b represent the isotropic hardening parameters. The interaction matrix h^{rs} is the one defined by Franciosi et al. [60]. It takes into account the influence of the accumulated plastic slip ν^r on the slip system s.

The material is a stainless steel 316L, and the parameters of the elastoviscoplastic behaviour law were identified by Guilhem et al. [61]. The Face Centered Cubic (FCC) structure (Fig. 3) shows the slip planes {111} and the slip directions ⟨110⟩. The interaction matrix is defined by six constants (h_0 to h_5). The material parameters values are summarised in Table 1.

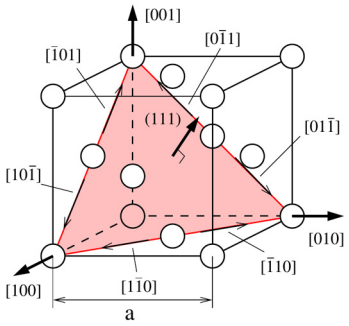
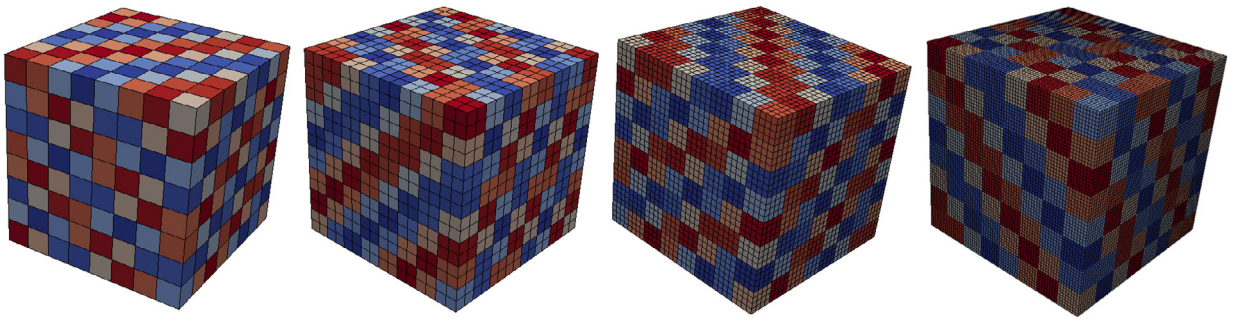


Fig. 3. The slip systems of the FCC crystal [6].

Table 1
Material parameters of the elasto-viscoplastic behaviour of the stainless steel 316L.

K [MPa.s$^{\frac{1}{n}}$]	n [–]	r_0 [MPa]	Q [MPa]	b [–]	c [MPa]	d [–]
12	11	40	10	3	40×10^3	1500

h_0 [–]	h_1 [–]	h_2 [–]	h_3 [–]	h_4 [–]	h_5 [–]	C_{11} [GPa]	C_{22} [GPa]	C_{12} [GPa]
1	1	0.6	12.3	1.6	1.8	197	125	122



(a) 1 element per grain (b) 8 element per grain (c) 64 element per grain (d) 512 element per grain

Fig. 4. Geometric model of polycrystalline aggregate consisting of 512 grains with several mesh densities.

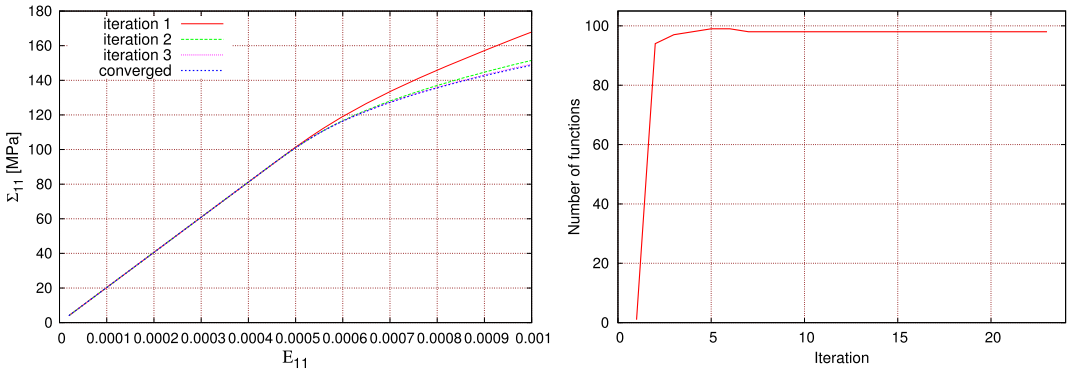
4. Numerical model

The simulations have been carried out on a 3D cubic polycrystalline aggregate. It is composed of 512 grains. In this configuration, the grains have fixed size and shape. The crystallographic orientation of the grain is chosen so as to get a global isotropic structure. The spatial discretization of the aggregate is done by a regular and periodic mesh with hexahedral Finite Elements with linear interpolation. In order to study the effect of the number of dofs, a study with different numbers of meshes was carried out. Four mesh densities are considered, from 1 element per grain to 512 elements per grain. The geometry and the Finite Element mesh for the microstructure of all cases treated is shown in (Fig. 4).

The polycrystalline aggregates considered in this work are subjected to a total strain controlled uniaxial loading. This loading is accomplished by imposing the volumetric average of strain on the polycrystalline aggregates. For all simulations investigated, the ratio used is $R_\varepsilon = \frac{\varepsilon_{\min}}{\varepsilon_{\max}} = -1$ and the value of the macroscopic strain imposed is 0.1%. 10 loading cycles are applied. Two load discretizations have been used for all simulations with 100 and 1000 increments per cycle.

5. Results

All methods, incremental, incremental right and the two versions of the PGD have been implemented in a computation code developed in LAMPA by Robert and Mareau [8]. The resolution of linear systems was performed using the MUMPS solver [62]. The calculations were performed on a Dell PowerEdge R710 server.



(a) Macroscopic mechanical response

(b) Number of functions and iterations

Fig. 5. Convergence of the PGD-V1 method.

5.1. Convergence of the PGD method

First, in order to clearly understand the results given by the PGD method, a study of the convergence of the first and the second versions of the PGD (labelled as PGD-V1 and PGD-V2, respectively) for a monotonic loading is conducted.

Fig. 5a shows the convergence of the PGD-V1, at the macroscopic level, for a monotonic loading. Fig. 5b illustrates the number of functions enriched and necessary for the convergence of space/time basis of the displacement field for the three steps of the PGD method according to the number of iterations necessary for the convergence of the mechanical response. In this case, the nonlinear term is updated with a separate representation of the displacement increment field calculated for the necessary PGD functions. The first iteration gives an almost linear response (red curve), which is logical as the vector $\{\Delta Q^{\mathrm{p}}\}$ is not known *a priori* (and therefore zero for the first iteration), and which corresponds to elastic behaviour. This first result is given by a single enrichment (Fig. 5b) sufficient for the convergence of the spatio-temporal base of the displacement field, since the response is quasi-linear. Then, the following iterations tends to be closer to the converged curve (blue curve) after 23 iterations with an enrichment average of 98 functions. The last time increments also are always the most distant for the solution due to the incremental computation of the constitutive equations.

Fig. 6 shows the convergence of the PGD-V2 for the first three enrichments with a tension–compression cyclic load. Figures from 6a to 6f stand for the PGD tensorial representation of the solution with the first three functions $\{\Delta U_x\}^i$ (Fig. 6a to Fig. 6c) and $\{\Delta U_t\}^i$ (Fig. 6d to Fig. 6f) ($i = 1, 2, 3$). In this case, the nonlinear term is updated with a displacement field calculated for a single enrichment of the PGD. Figs. 6g to 6i illustrate the results of the macroscopic stress–strain loop given by a separated representation of the displacement increment field calculated with the firsts three enrichments of the PGD. The first enrichment (Fig. 6g) is given by a nonlinear term equal to zero (which is not known *a priori*). The rest of enrichments are calculated by a nonlinear term updated from the previous enrichments (Fig. 6h and Fig. 6i). This resolution is made successively until convergence of the plastic strain field after 244 enrichments.

5.2. Mechanical response

Before analyzing the results in terms of CPU time, we compare the reliability of the different methods in terms of the results' accuracy. Two cases are analyzed: the mesoscopic and macroscopic ones.

5.2.1. Macroscopic results

We present here a comparison of the macroscopic response (aggregate average) of incremental, incremental right, and PGD simulations. Uniaxial strain cyclic tests with macroscopic strain amplitudes of 0.1% have been applied with two different time discretizations of 100 and 1000 increments for the four mesh densities. Fig. 7 illustrates the results for the 10th loading cycle with 32,768 elements and two different time discretizations. As can be seen, the results are very similar without any error. These results remain the same for the other mesh densities.

5.2.2. Mesoscopic results

The stress–strain responses at the scale grain for the different numerical methods have been studied now. The mean stress and strain values in each grain of the aggregate are given by:

$$\langle \sigma \rangle_G = \frac{1}{V_G} \int_{V_G} \sigma \, \mathrm{d}V \tag{60}$$

$$\langle \varepsilon \rangle_G = \frac{1}{V_G} \int_{V_G} \varepsilon \, \mathrm{d}V \tag{61}$$

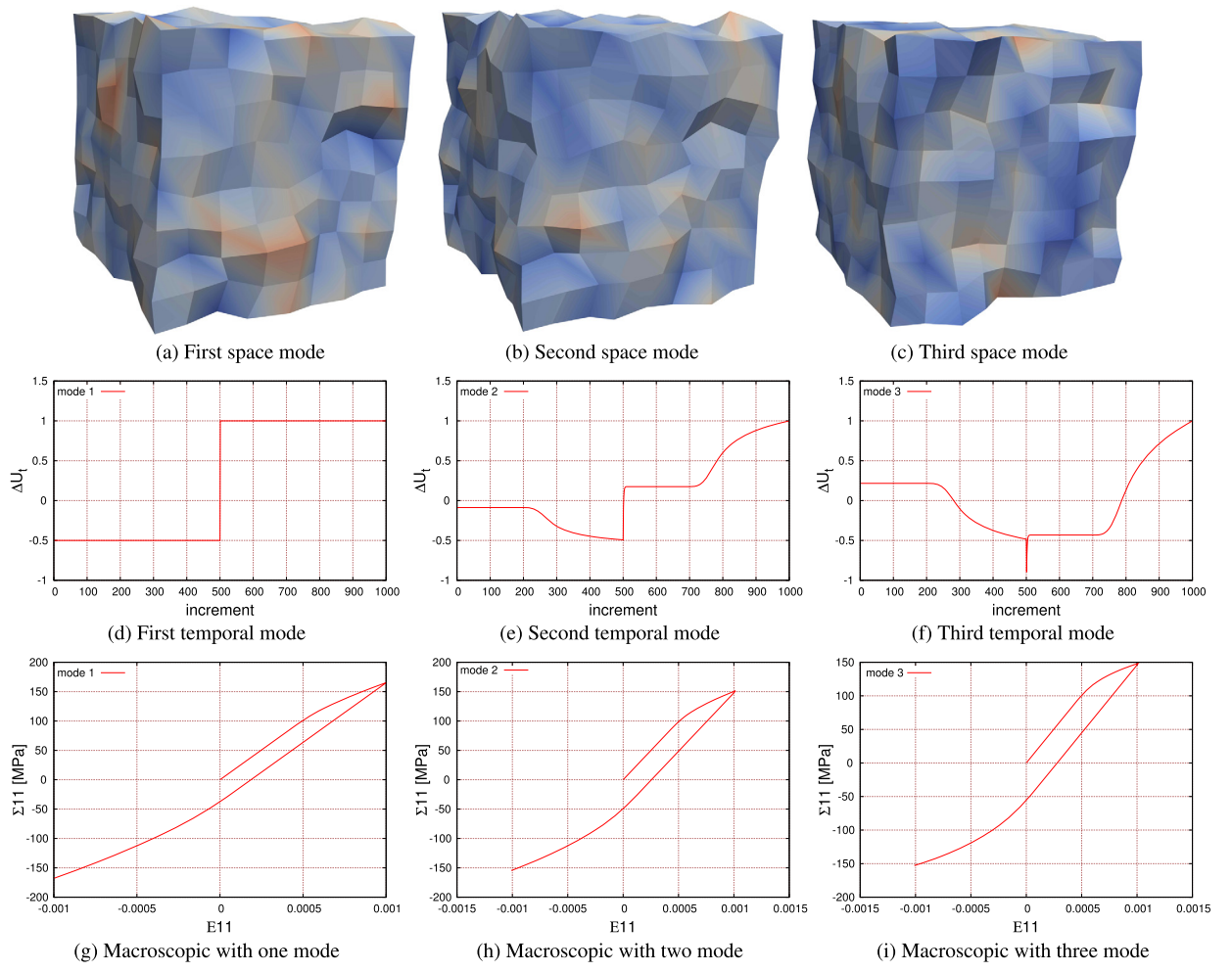


Fig. 6. Convergence of the PGD-V2 method with the first three modes for: (a, b, c) space basis, (d, e, f) temporal basis, and (g, h, i) evolution of macroscopic stress–strain.

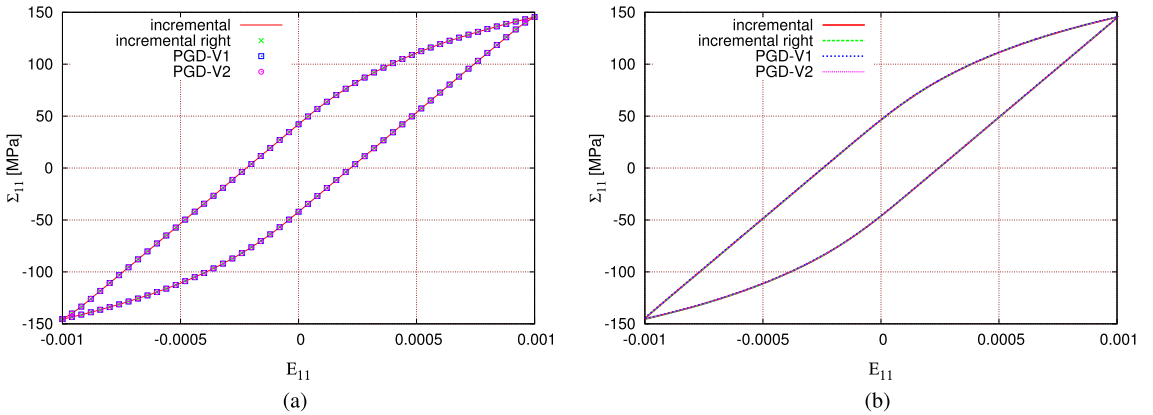


Fig. 7. Comparison between different methods at the macroscopic stress–strain $\Sigma_{11} - E_{11}$ responses of a polycrystalline aggregate consisting of 32,768 elements loaded with a uniaxial strain cyclic tests for two time discretizations: (a) with 100 increments (b) with 1000 increments.

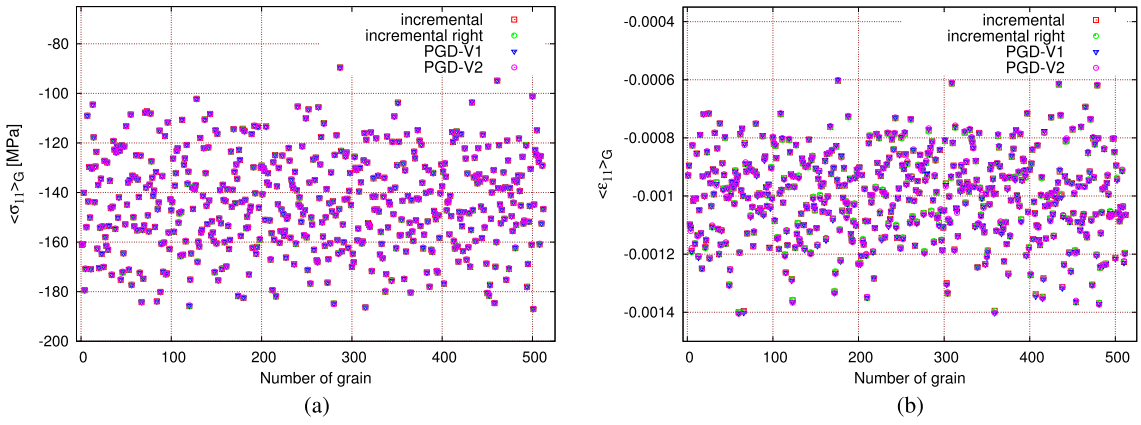


Fig. 8. Comparison between different methods for the mesoscopic responses ($\langle\sigma_{11}\rangle_G$ and $\langle\varepsilon_{11}\rangle_G$) of a polycrystalline aggregate consisting of 262,144 elements loaded in a uniaxial strain cyclic tests with 100 increments: (a) average stress in grains, (b) average strain in grains.

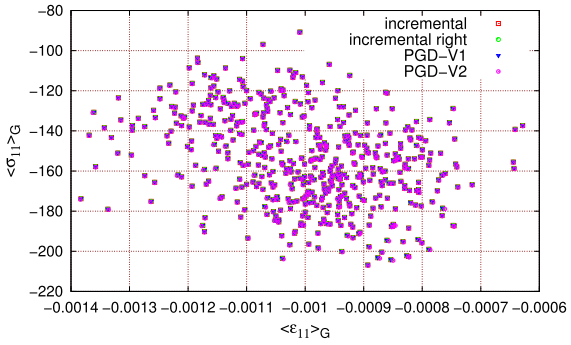


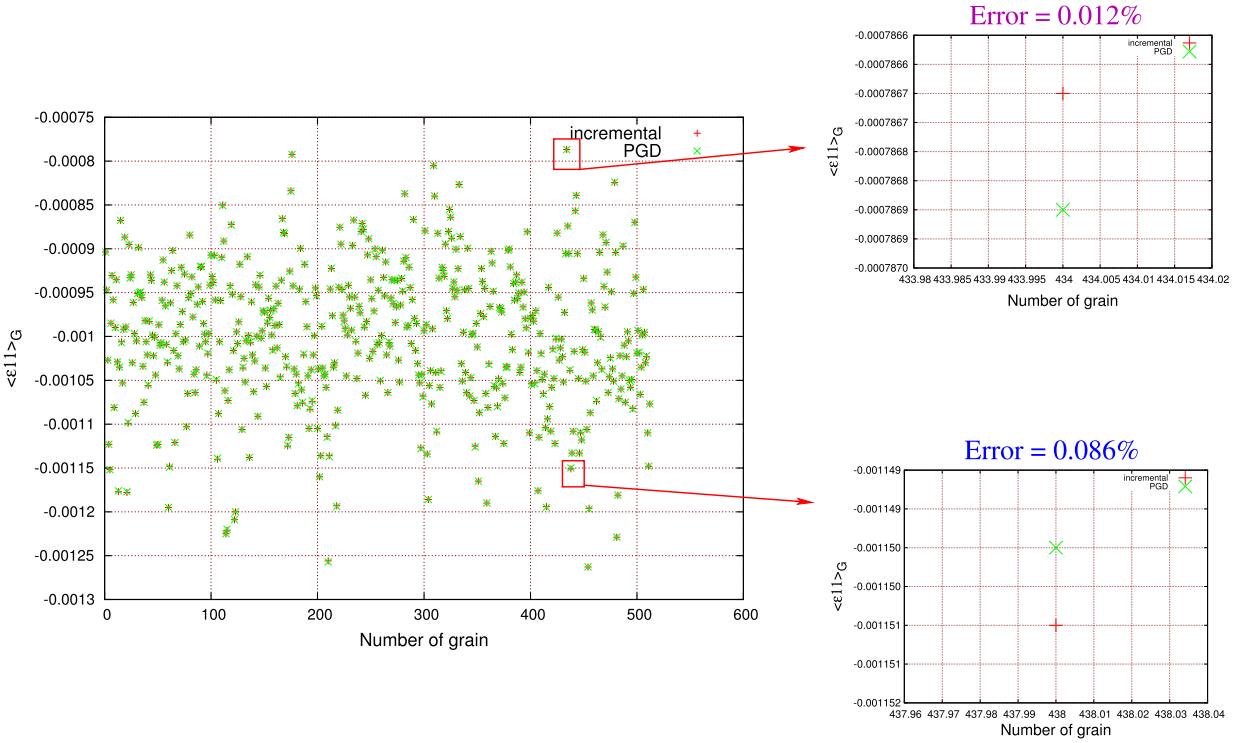
Fig. 9. Distribution of the average stress and strain in each grain obtained with the incremental method and the PGD methods for the polycrystalline aggregate consisting of 32,768 elements loaded with a uniaxial strain cyclic tests with 100 increments.

where V_G is the grain volume. Fig. 8 illustrates the results of the average axial stress $\langle\sigma_{11}\rangle_G$ and the average axial strain $\langle\varepsilon_{11}\rangle_G$ obtained for each grain G according to the loading path at the 10th loading cycle for the mesh with 262,144 elements. Each point in the figure corresponds to a grain. Fig. 8a shows that the developed PGD methods allow one to obtain results very close to those obtained with the incremental method for the average axial stress. For the average axial strain results, we observe very small differences between the developed methods and the incremental calculation (Fig. 8b), with a maximum relative difference less than 0.1%. Simulations were performed with 1000 cycles using a polycrystalline aggregate consisting of 512 and 4096 elements in order to see the evolution of the error with a larger number of cycles. Fig. 10 displays the results obtained for the 10th and 1000th loading cycles with incremental and PGD-V2 methods. These results show a small increase in the error after 1000 cycles compared to the results for two grains. For example, an evolution from 0.012% to 0.038% is observed. The results of the mesoscopic strain for the different methods developed seem to be quite acceptable. However, these differences can be reduced by decreasing the convergence tolerance but, in this case, the computation time increases significantly.

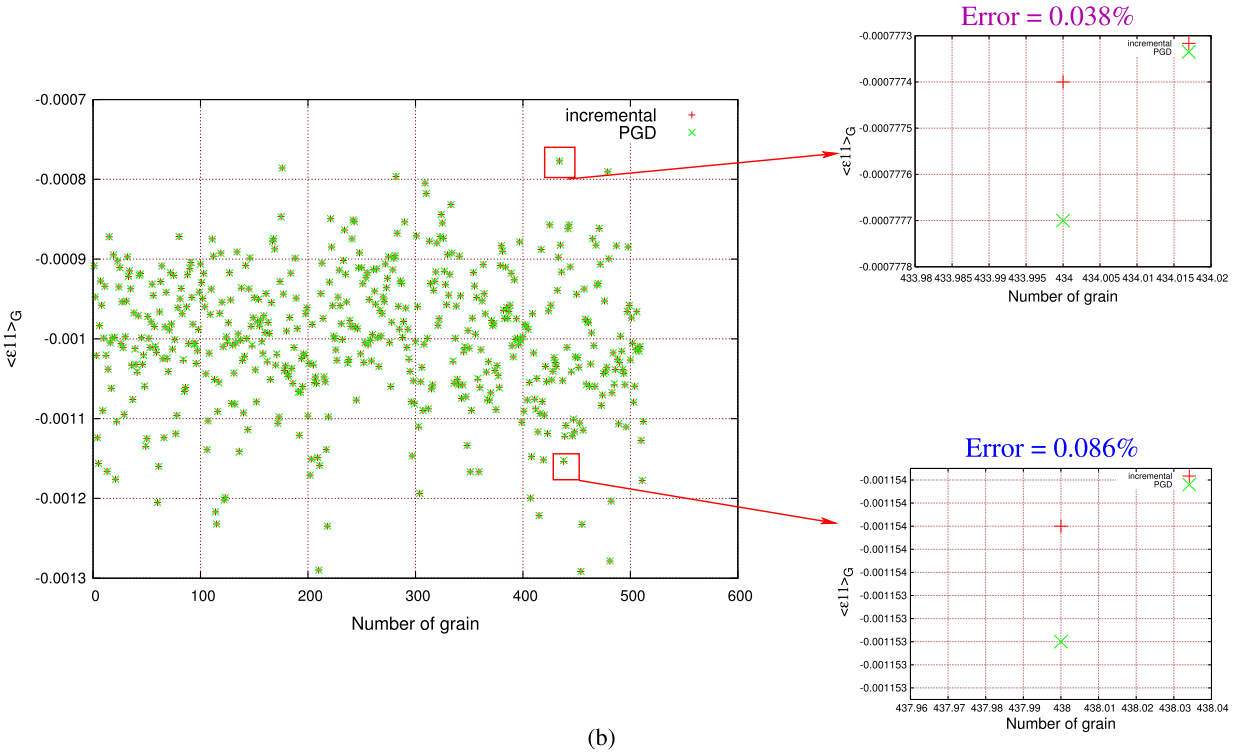
Fig. 9 shows the average stress and strain in each grain $\langle\sigma_{11}\rangle_G - \langle\varepsilon_{11}\rangle_G$ obtained at the 10th loading cycle for aggregate consisting of 32,768 elements and for a total macroscopic strain $E_{11} = 0.1\%$. These results confirm that the PGD method provides a good description of the mesoscopic results with respect to the two incremental methods.

5.3. Computation CPU time

Fig. 11 represents an example of the CPU time per cycle of the PGD-V1 and the PGD-V2 method, the number of functions required for the convergence and the iterations number of the PGD-V1. These results are obtained from a modelling of a polycrystalline aggregate consisting of 32,768 elements loaded with a uniaxial strain cyclic tests for 10 cycles. Note that the number of functions per cycle of the PGD-V1 method (continuum curve in Fig. 11d) corresponds to the number of total functions enriched for all the iterations required per cycle. These results show that the first cycles are the most expensive in terms of number of iterations (Fig. 11b) of the considerable behaviour evolution between cycles. After a certain number of cycles (in this case the fourth cycle), only four iterations are necessary to obtain convergence. These observations are also noticed with the number of functions enriched per cycle for both PGD methods (Fig. 11d). Indeed, the adaptation of the PGD basis is carried out mainly during the first cycles. Then, the longer the number of cycles, the faster the stabilization.



(a)



(b)

Fig. 10. Evolution of the error for the mesoscopic strain for two grains for incremental and PGD-V2 method for the (a) 10th loading cycle (b) 1000th loading cycle.

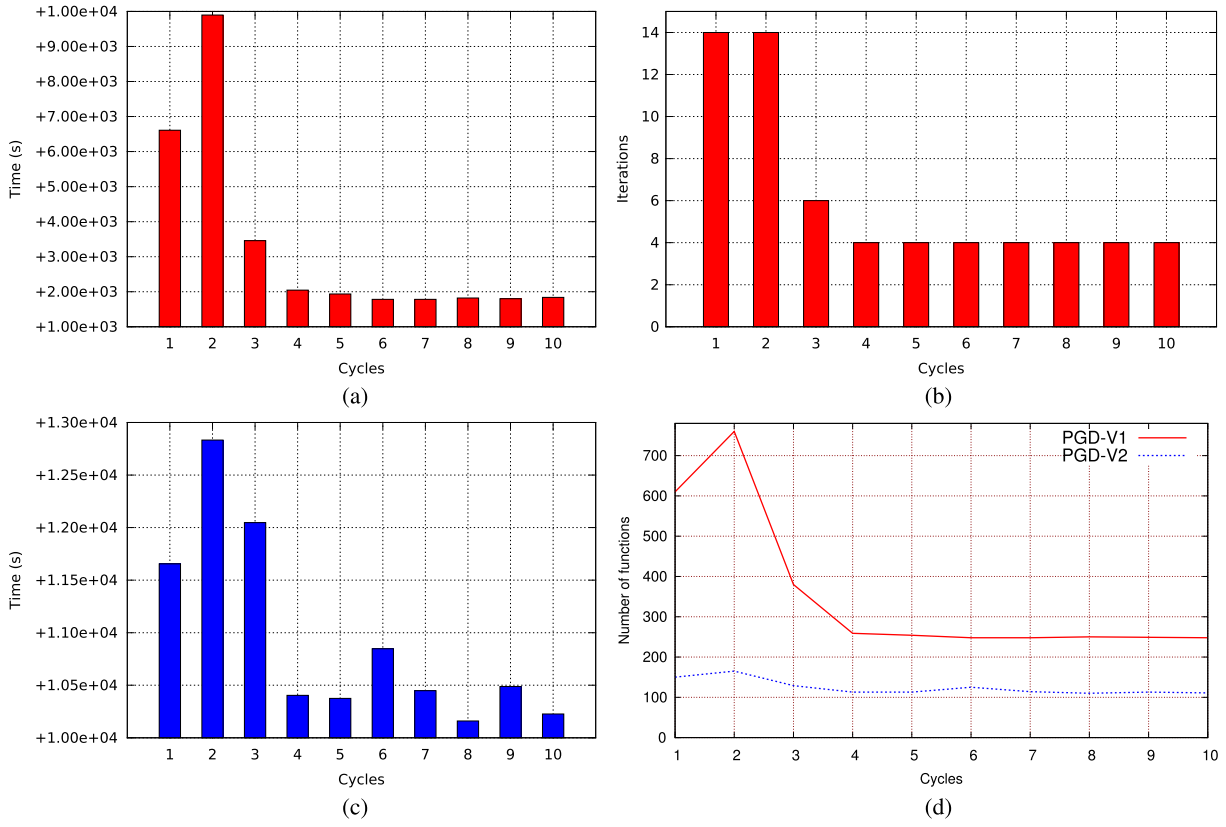


Fig. 11. CPU time, number of function and number of iterations per cycle for the PGD methods obtained with a polycrystalline aggregate consisting of 32,768 elements loaded with a uniaxial strain cyclic tests with 100 increments: (a) CPU time per cycle PGD-V1, (b) number of iterations PGD-V1, (c) CPU time per cycle PGD-V2, (d) number of functions.

This leads to a reduction in the number of functions required for convergence. Moreover, the computation time per cycle (Fig. 11a) is proportional to the number of functions and iterations. The first three cycles are the most costly in terms of computing time. The CPU time of the second cycle has increased compared to the first cycle, even though the number of iterations was the same. Indeed, in this case the first cycle requires 610 functions, whereas for the second cycle 760 functions are needed to achieve convergence.

Fig. 12 shows the evolution of the computation time obtained for ten cycles by the two incremental methods and the two versions of PGD according to the differences mesh densities.

The first observation is that the CPU time of the incremental method increases in a quadratic manner according to the number of dofs. It also occurs clearly that the incremental right method is faster than the incremental method and the PGD methods for all the cases studied. With 262,144 elements and 100 time increments per cycle (Fig. 12a), the calculation with the incremental right method was 91 times faster than the incremental method and 16 times faster than the PGD-V1 method. Similarly, with 1000 time increments and 32768 elements (Fig. 12b), the incremental right method was 56 times faster than the incremental method. Note that the CPU time of the incremental right method increases in a linear manner according to the number of dofs.

It is observed that the CPU time for the PGD-V1 is less important than in the incremental method. As observed, beyond 4096 elements with 100 time increments per cycle (Fig. 12a), the PGD-V1 CPU time appears to be shorter than the incremental solver CPU time, whereas for 1000 time increments per cycle, the PGD-V1 has proven to be always faster than the incremental method (Fig. 12b). In addition, an increase in the number of dofs allows greater time saving. For example, with 32768 elements, the PGD-V1 was twice as fast as the incremental method, whereas it was 6 times faster with 262144 elements. The time savings was more significant by increasing the number of time increments per cycle. Indeed, with a 100 time increment per cycle and 32,768 elements, the PGD-V1 was twice faster than the incremental method, while for 1000 time increments per cycle, the PGD-V1 becomes 10 times faster than the standard method. Unfortunately, compared to the incremental right method, the PGD method requires more computing time for ten cycles.

The results obtained with the PGD-V2 method clearly correspond to those obtained by the standard method in terms of mesoscopic and macroscopic responses. Regarding the CPU time, this method has not resulted in saving time compared to the incremental method. For 1000 time increments per cycle, the PGD-V2 was always slower than both the incremental and the PGD-V1 methods. Beyond 262,144 elements with 100 time increments per cycle, the PGD-V2 CPU time seems to

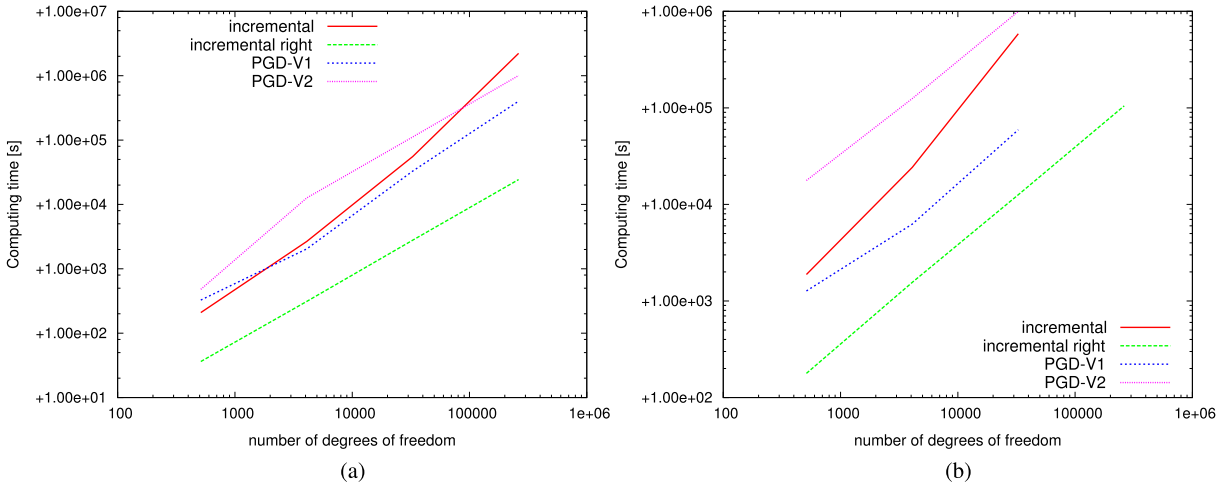


Fig. 12. CPU time obtained for the various methods depending on the number of elements: (a) 100 time increments per cycle, (b) 1000 time increments per cycle.

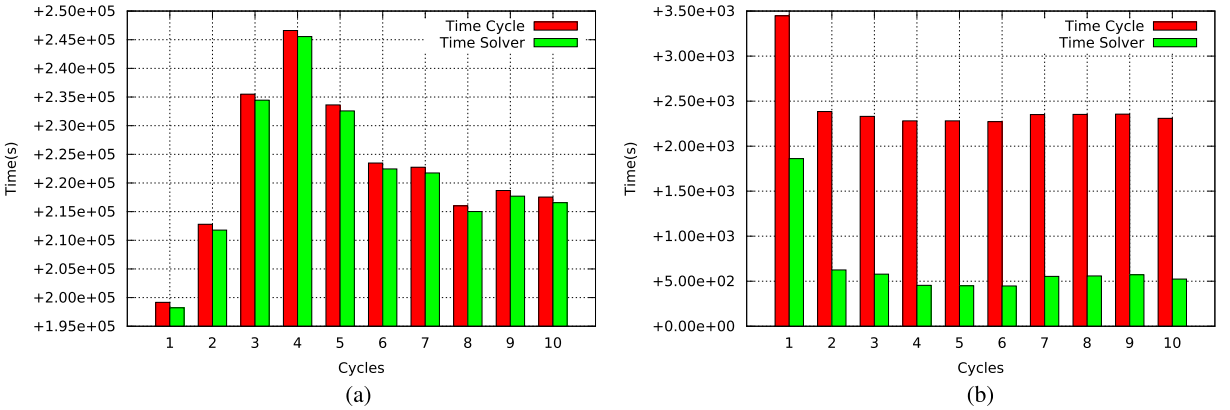


Fig. 13. CPU time per cycle and time solver obtained with a polycrystalline aggregate consisting of 262,144 elements loaded in a uniaxial strain cyclic test with 100 increments: (a) incremental method, (b) incremental-right method.

be shorter than in the incremental method, and it was twice faster but slower than in the other methods (Fig. 12b). The incremental method and PGD-V2 method are limited by the computation time, which is important for fine mesh resolutions and large time step increments cycle (case of 262,144 elements with 1000 time increments per cycle (Fig. 12b)). In addition, it is important to notice that both the PGD-V1 and PGD-V2 require more memory compared to the incremental method.

5.4. Discussion

The results presented previously show that the incremental-right method allows us to obtain a good mechanical response, with a computational time as fast as in the incremental method as well as with both PGD versions. This shows the interest of keeping a constant stiffness matrix for the resolution of the linear system (Eq. (25)) and to carry out one single factorization in the first iteration of the computation. This leads to very important time saving compared to the incremental method. Fig. 13 illustrates the comparison between the time per cycle and solver the time, which represent the time required to solve the global step. It appears that the solver time per cycle for the incremental-right method represents an average gain of 22% with regard to the computation time per cycle (Fig. 13b). For the first cycle, one factorization is required for the first iteration, which is not the case for the incremental method (Fig. 13b).

The PGD-V1 method provides results with an important reduced CPU time compared to the incremental method. Unfortunately, it is always slower than the incremental-right method, whereas the stiffness matrix is also constant in this case. Fig. 14a shows the time per cycle and the solver time for the PGD method. As noted, the solver time per cycle represents an average of 94% of computation time per cycle contrary to the incremental-right method. This explains that the PGD method requires a hundred functions to obtain convergence. Thus, to compute the nonlinear term of the second member, the resolution of the linear system must be done one hundred times for each iteration. Fig. 14b illustrates the number of functions required per cycle for the PGD-V1 method and the number of iterations per cycle for the incremental-right method. For

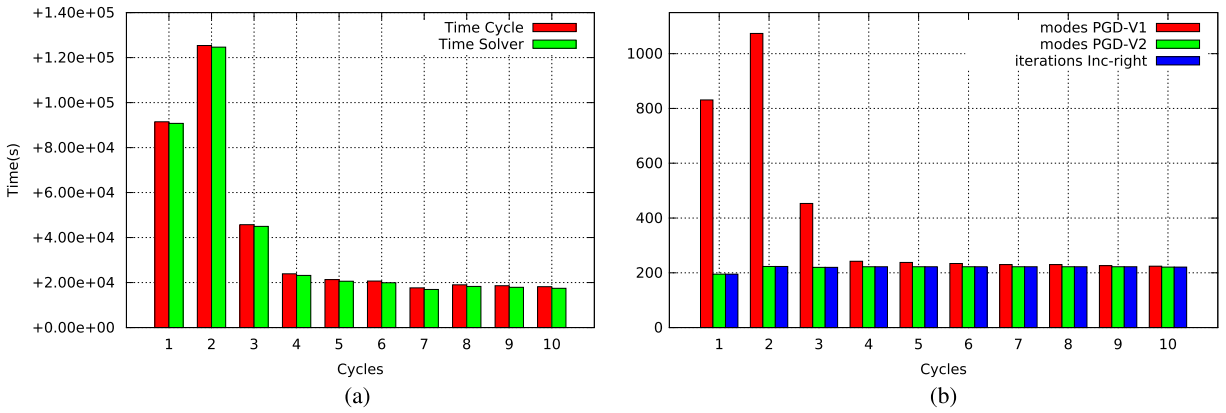


Fig. 14. Results obtained with a polycrystalline aggregate consisting of 262,144 elements loaded with a uniaxial strain cyclic tests with 100 increments: (a) CPU time per cycle and time solver for the PGD-V1 method, (b) number of modes and iterations required per cycle for PGD-V1, PGD-V2, and incremental-right method.

the third cycle, the PGD-V1 requires 453 functions for all iterations to reach convergence (with a tolerance of $\epsilon_{\text{PGD}} = 10^{-4}$). Taking into account the number of iterations of the fixed point algorithm (fixed at three iterations), the resolution of the linear system is executed (453×3) times, while the incremental-right method is performed 220 times. Furthermore, the nonlinear term of the second member is computed iteratively and, with the PGD method, the material behaviour equations are solved for each iteration and for all times steps. This takes more computation time, especially with material constitutive relations with high nonlinearity. This explains why the first version of the PGD is faster than the second one. With the PGD-V2 method, the nonlinear term is updated with a displacement field calculated for each enrichment of the PGD, unlike the first version, where the nonlinear term is updated with a separated representation of the displacement increment field calculated for all the necessary PGD functions. Accordingly, the resolution of the material behaviour equations are performed for each enrichment for the PGD-V2 method. As indicated by Fig. 14b and for the case of the third cycle, the number of enrichments is 121. Hence, the material constitutive relations are computed 121 times for all time steps, whereas for the PGD-V1 method, 6 computations are required (number of iterations in Fig. 11b).

6. Conclusion

This paper presents the development of an original numerical method to deal with nonlinear elasto-viscoplastic problems. FE simulations together with reduction methods are used. The aim was to simulate, within a reasonable CPU time, problems with a large number of dofs and on a large time scale for the prediction in the HCF regime. Three methods have been investigated: the incremental-right method and the PGD method with two versions. The firstone consists in keeping a constant stiffness to perform one single factorization, and the second uses a model reduction method (PGD) to decouple the dimensions of space and time in the global step. The simulations have been carried out on a 3D polycrystalline aggregate with an elastic–viscoplastic behaviour of stainless steel 316L. The responses have been computed with a cyclic tension–compression test and for various numbers of elements per grain and time increments per cycle. The results have been compared with the incremental method (conventionally used in commercial codes) in terms of mechanical response and CPU time.

For the macroscopic response, there is no difference between the different methods. Furthermore, the observation of the mesoscopic response shows that the results are very similar to those of the incremental method for the average stress. For the average axial strain results, it is possible to find discrepancies in some cases between the methods developed and the incremental method, but with a maximum relative difference less than 0.1% even for a large number of cycles.

For the computation time, we have showed that the PGD-V1 provides good results in terms of macroscopic and mesoscopic responses, with a computation time as fast as the one of the incremental method (reducing the CPU time by a factor of 10). In addition, we have demonstrated that increasing the number of elements and the number of time increments per cycle allows a greater time gain compared to the incremental method. Unfortunately, unlike in the previous case, the PGD-V2 has not resulted in saving time compared to the incremental method. Nevertheless, the idea of the nonlinearities in the second member to keep a constant stiffness matrix with the incremental method has enabled more CPU time saving compared to the PGD-V1 and PGD-V2 methods.

References

[1] R. Guerchais, Influence d'accidents géométriques et du mode de chargement sur le comportement en fatigue à grand nombre de cycles d'un acier inoxydable austénitique 316L, PhD thesis, École nationale supérieure des arts et métiers, Paris, 2014.
[2] M.A. Crisfield, Non-linear Finite Element Analysis of Solids and Structures, vol. 1, John Wiley and Sons Ltd., 2000.
[3] O.C. Zienkiewicz, Y.K. Cheung, Engineer, 1965, pp. 507–510.

[4] O.C. Zienkiewicz, The Finite Element Method in Structural and Continuum Mechanics, McGraw-Hill, New York, 1967.
[5] H. Moulinec, P. Suquet, A numerical method for computing the overall response of nonlinear composites with complex microstructure, Comput. Methods Appl. Mech. Eng. 157 (1998) 69–94.
[6] C. Mareau, D. Cuillerier, F. Morel, Experimental and numerical study of the evolution of stored and dissipated energies in a medium carbon steel under cyclic loading, Mech. Mater. 60 (2013) 93–106.
[7] R.A. Lebensohn, *N*-site modelling of a 3d viscoplastic polycrystal using fast Fourier transform, Acta Mater. 49 (2001) 2723–2737.
[8] C. Robert, C. Mareau, A comparison between different numerical methods for the modelling of polycrystalline materials with an elastic–viscoplastic behaviour, Comput. Mater. Sci. 103 (2015) 134–144.
[9] C. Robert, N. Saintier, T. Palin-Luc, F. Morel, Micro-mechanical modelling of high cycle fatigue behaviour of metals under multiaxial loads, Mech. Mater. 55 (2012) 112–129.
[10] P. Ladevèze, Sur une famille d'algorithmes en mécanique des structures, C. R. Acad. Sci. Paris, Ser. II 300 (1985) 41–44.
[11] P. Ladevèze, La méthode à grand incrément pour l'analyse de structures à comportement non linéaire décrit par variables internes, C. R. Acad. Sci. Paris, Ser. II 309 (11) (1989) 1095–1099.
[12] N. Maouche, Modélisation des phénomènes d'endommagement dus aux contacts à faible amplitude de débatement, PhD thesis, École nationale des ponts et chaussées, Paris, 1997.
[13] B. Pommier, Détermination de la réponse asymptotique d'une structure anélastique soumise à un chargement thermomécanique cyclique, PhD thesis, École polytechnique, 2003.
[14] H. Maitournam, B. Pommier, J.-J. Thomas, Determination of the asymptotic response of a structure under cyclic thermomechanical loading, C. R. Mecanique 330 (2002) 703–708.
[15] P.-M. Lesne, S. Savalle, An efficient cycles jump technique for viscoplastic structures calculations involving large number of cycles, in: 2nd International Conference on Computational Plasticity, Barcelona, Spain, 1989, pp. 591–602.
[16] K. Sai, Modèles à grand nombre de variables internes et méthodes numériques associées, PhD thesis, École nationale supérieure des mines de Paris, 1993.
[17] J. Zarka, J. Frelat, G. Inglebert, P. Kasmai-Navidi, A New Approach in Inelastic Analysis of Structures, CADLM, 1990.
[18] J. Zarka, P.K. Navidi, Simplified dynamical analysis of inelastic structures, Nucl. Eng. Des. 92 (1986) 89–103.
[19] H. Neuber, Theory of stress concentration for shear-strained prismatic bodies with arbitrary non-linear stress–strain law, J. Appl. Mech. 28 (1961) 544–551.
[20] E. Liberge, A. Hamdouni, Reduced order modelling method via proper orthogonal decomposition (POD) for flow around an oscillating cylinder, J. Fluids Struct. 26 (2) (2010) 292–311.
[21] L. Cordier, M. Bergmann, Réduction de dynamique par Décomposition Orthogonale aux Valeurs Propres (POD), in: Optimisation et Contrôle des Ecoulements et des Transferts, École de printemps OCET, 2006.
[22] G.H. Golub, C.F. Van Loan, Matrix Computations, Johns Hopkins University Press, Baltimore, MD, USA, 1990.
[23] J. Burkardt, M. Gunzburger, H.-C. Lee, POD and CVT-based reduced-order modelling of Navier–Stokes flows, Comput. Methods Appl. Mech. Eng. 196 (1–3) (2006) 337–355.
[24] D. Ryckelynck, A priori hyperreduction method: an adaptive approach, Comput. Methods Appl. Mech. Eng. 202 (2005) 346–366.
[25] A. Ammar, D. Ryckelynck, F. Chinesta, R. Keunings, On the reduction of kinetic theory models related to finitely extensible dumbbells, J. Non-Newton. Fluid Mech. 134 (2006) 136–147.
[26] A. Ammar, B. Mokdad, F. Chinesta, R. Keunings, A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modelling of complex fluids, J. Non-Newton. Fluid Mech. 139 (2006) 153–176.
[27] A. Ammar, F. Chinesta, P. Diez, A. Huerta, An error estimator for separated representations of highly multidimensional models, Comput. Methods Appl. Mech. Eng. 199 (2010) 1872–1880.
[28] A. Dumon, C. Allery, A. Ammar, Proper general decomposition (PGD) for the resolution of Navier–Stokes equations, J. Comput. Phys. 230 (2011) 1387–1407.
[29] K.V. Spiliopoulos, K.D. Panagiotou, Simplified methods for the steady state inelastic analysis of cyclically loaded structures, Comput. Methods Appl. Mech. Eng. 223–224 (2012) 186–198.
[30] A. Ammar, The proper generalized decomposition: a powerful tool for model reduction, Int. J. Mater. Form. 3 (2010) 89–102.
[31] F. Chinesta, A. Ammar, E. Cueto, Proper generalized decomposition of multiscale models, Int. J. Numer. Methods Eng. 89 (2010) 1114–1132.
[32] P. Ladevèze, Nonlinear Computational Structural Mechanics—New Approaches and Non-incremental Methods of Calculation, Springer Verlag, 1999.
[33] P. Ladevèze, A. Nouy, On a multiscale computational strategy with time and space homogenization for structural mechanics, Comput. Methods Appl. Mech. Eng. 192 (2003) 3061–3087.
[34] A. Nouy, P. Ladevèze, Multiscale computational strategy with time and space homogenization: a radial-type approximation technique for solving micro problems, Int. J. Multiscale Comput. Eng. (2004) 170.
[35] N. Relun, D. Néron, P.A. Boucard, A model reduction technique based on the PGD for elastic-viscoplastic computational analysis, Comput. Mech. 51 (2013) 83–92.
[36] P. Ladevèze, European journal of mechanics a/solids, Comput. Mech. 60 (2016) 227–237.
[37] A. Ammar, B. Mokdad, F. Chinesta, R. Keunings, A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modelling of complex fluids: part II: transient simulation using space–time separated representations, J. Non-Newton. Fluid Mech. 144 (2007) 98–121.
[38] F. Chinesta, A. Ammar, P. Joyot, The nanometric and micrometric scales of the structure and mechanics of materials revisited: an introduction to the challenges of fully deterministic numerical descriptions, Int. J. Multiscale Comput. Eng. 6 (3) (2008) 191–213.
[39] F. Chinesta, A. Ammar, E. Cueto, On the use of proper generalized decompositions for solving the multidimensional chemical master equation, Eur. J. Comput. Mech. 19 (1) (2010) 53–64.
[40] F. Chinesta, A. Ammar, E. Cueto, Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models, Arch. Comput. Methods Eng. 17 (2010) 327–350.
[41] E. Pruliere, F. Chinesta, A. Ammar, On the deterministic solution of multidimensional parametric models using the proper generalized decomposition, Math. Comput. Simul. 81 (2010) 791–810.
[42] A. Ammar, M. Normandin, F. Daim, D. Gonzalez, E. Cueto, F. Chinesta, Non incremental strategies based on separated representations: applications in computational rheology, Commun. Math. Sci. 8 (3) (2010) 671–695.
[43] D. Ryckelynck, K. Lampoh, S. Quilicy, Hyper-reduced predictions for lifetime assessment of elasto-plastic structures, Mecanica 51 (2016) 309–317.
[44] A. Radermacher, S. Reese, Model reduction in elastoplasticity: proper orthogonal decomposition combined with adaptive sub-structuring, Comput. Mech. 54 (2014) 677–687.
[45] J.M. Bergheau, S. Zuchiatti, J.C. Roux, E. Feulvarch, S. Tissot, G. Perrin, The proper generalized decomposition as a space–time integrator for elasto-plastic problems, C. R. Mecanique 344 (2016) 759–768.
[46] R.L. Taylor, O.C. Zienkiewicz, The Finite Element Method, vols. I–II, Butterworth-Heinemann, 2000.

[47] J.E. Dennis Jr., A brief survey of convergence results for quasi-Newton methods, in: SIAM-AMS Proceedings, vol. 9, 1976, pp. 185–199.
[48] K.-J. Bathe, Finite Element Procedures, Prentice-Hall, 1996.
[49] H. Matthies, G. Strang, The solution of nonlinear finite element equations, Int. J. Numer. Methods Eng. 14 (1979) 1613–1626.
[50] K.J. Bathe, A.P. Cimento, Some practical procedures for the solution of nonlinear finite element equations, Comput. Methods Appl. Mech. Eng. 22 (1980) 59–85.
[51] T.J.R. Hughes, J.C. Simo, Computational Inelasticity, Springer Verlag, 1997.
[52] J.C. Simo, R.L. Taylor, Consistent tangent operators for rate-independent plasticity, Comput. Methods Appl. Mech. Eng. 48 (1985) 101–118.
[53] P. Wriggers, Nonlinear Finite Element Methods, Springer, 2008.
[54] K.V. Spiliopoulos, A direct method to predict cyclic steady states of elastoplastic structures, in: D. Weichert, G. Maier (Eds.), Inelastic Analysis of Structures Under Variable Loads, Kluwer Academic Publishers, 2000, pp. 213–232.
[55] K.V. Spiliopoulos, A simplified method to predict the steady cyclic stress state of creeping structures, J. Appl. Mech. 69 (2002) 149–153.
[56] M.A. Nasri, J.V. Aguado, A. Ammar, E. Cueto, F. Chinesta, F. Morel, C. Robert, S. Elarem, Separated representation of incremental elastoplastic simulations, Key Eng. Mater. 651–653 (2015) 1285–1293.
[57] L. Méric, G. Cailletaud, Single crystal modelling for structural calculations. Part 2: finite element implementation, J. Eng. Mater. Technol. 113 (1991) 171–182.
[58] R. Guerchais, C. Robert, F. Morel, N. Saintier, Micromechanical study of the loading path effect in high cycle fatigue, Int. J. Fatigue 59 (2014) 64–75.
[59] F. Barbe, S. Forest, G. Cailletaud, Single crystal modelling for structural calculations. Part 2: finite element implementation, Int. J. Plast. 17 (2001) 537–563.
[60] P. Franciosi, Etude théorique et expérimentale du comportement élastoplastique des monocristaux métalliques se déformant par glissement: modélisation pour un chargement complexe quasi statique, PhD thesis, Université Paris-Nord (Paris-13), 1984.
[61] Y. Guilhem, S. Basseville, F. Curtit, J-M. Stéphan, G. Cailletaud, Numerical investigations of the free surface effect in three-dimensional polycrystalline aggregates, Comput. Mater. Sci. 70 (2013) 150–162.
[62] E. Agullo, P. Amestoy, A. Buttari, A. Guermouche, G. Joslin, J.-Y. L'Excellent, X.S. Li, A. Napov, F.-H. Rouet, M. Sid-Lakhdar, S. Wang, C. Weisbecker, I. Yamazaki, Recent advances in sparse direct solvers, in: 22nd Conference on Structural Mechanics in Reactor Technology, San Francisco, CA, USA, 2013.