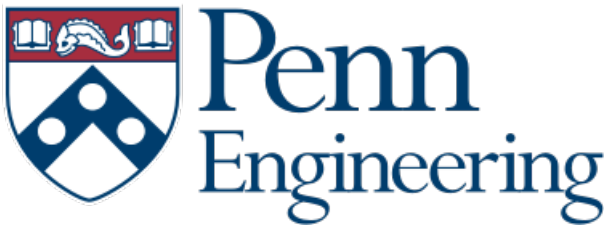
Final work : Homogenization of single crystal nickel based super-alloys. An overview of the creep behavior

Auteur : Carrillo Segura, Simon
Promoteur(s) : Noels, Ludovic
Faculté : Faculté des Sciences appliquées
Diplôme : Master en ingénieur civil en aérospatiale, à finalité spécialisée en "turbomachinery aeromechanics (THRUST)"
Année académique : 2019-2020
URI/URL : http://hdl.handle.net/2268.2/8789



Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania

Faculty of Applied Sciences, University of Liège

Homogenization of single crystal nickel based super-alloys. An overview of the creep behavior

---

Graduation Studies conducted for obtaining the ***Master's degree in Aerospace Engineering in Aeromechanical*** & ***Turbomachinery***
by Simon Carrillo Segura

Supervised by: *Pedro Ponte Castañeda* & *Ludovic Noels*

Author: *Simon Carrillo Segura*

Academic Year: 2019/2020

Acknowledgments

First of all, I would like to express my most honest gratitude to my advisors, Prof. Ludovic Noels and Prof. Pedro Ponte-Castaneda for making possible this collaborative project between University of Liège and University of Pennsylvania. I am deeply in debt with you for such incredible opportunity.

Thanks to Shuvrangsu Das, Christoph Kammer and Jose Cotelo for the innumerable discussions leading to a successful Finite Element Model.

This work sets the final dot of the THRUST Master, where unforgettable professionals deserve to be mention: Nenad Glodic, Bob Kielb and Paul Petrie-Repar. Thank you.

This journey has been easier thanks to my awesome classmates. From the unforgettable talks with Harish Rajan, to manufacturing works in the KTH Formula Student garage with Luca Tadei and Davide Di Nubila. Late night talks with Davide Di Nubila made my year in KTH Royal Institute of Technology the best in my lifetime. Thank you.

However, I don't want to forget those who have always been there, supporting me in every decision, opening my eyes in academic and non-academic topics. Thank you for being there Samuel Ruiz Capel, Roberto Barbero and Ignacio Pons Gutierrez.

I don't want to forget about Madhu Sukumar and Ted Wilson, my quarantine roommates during my research stay in Philadelphia. Thanks for all the funny moments in 615 apartment.

A mi madre, padre y hermanos, por apoyarme todos estos años en mi larga formación. En especial a mi madre, Luisa, la persona más luchadora que conozco. Nada de esto hubiera sido posible sin ti.

"Either write something worth reading or do something worth writing"

Benjamin Franklin

Abstract

An overview of the creep behaviour of an aeronautical-type single crystal turbine blade is analyzed by means of two methodologies: a fully-optimized second order homogenization method and a Finite Element numerical approach. Micro-structure, i.e porosity level, is shown to have an important role in creep behaviour of porous FCC single crystals.

The homogenization-based constitutive model developed by Ponte-Castaneda makes use of the fully optimized second order variational approach of (Ponte Castañeda, 2015), along with the iterated homogenization method of (Agoras and Ponte Castañeda, 2013) to define a constitutive model for the finite-strain macroscopic response of porous single crystal in the sense of visco-plasticity. For the computations (Song and Ponte Castañeda, 2017a), Song et al. implemented a numerical implementation in Fortran language.

The numerical finite element calculations are carried out using a three dimensional Finite element code of a Unit Cell.The single crystal matrix is defined by a simple power law viscous crystal plasticity constitutive relation. The Unit Cell is initially cubic with a sphere or ellipsoid located in the center, constituting the inclusion phase. Fully periodic boundary conditions are imposed in the Unit Cell Finite element model by means of the MPC capability of ABAQUS and the "dummy node" technique.

The effect of crystal orientation and loading conditions on the micro-structure evolution in a face center cubic (FCC) single crystal is analyzed. Two different initial crystal orientations are considered. The calculations are carried out for six different values of stress triaxiality and for three different Lode parameter. Additionally, the effect of an initial ellipsoidal void shape and the effect of the initial porosity level is addressed.

Micro-structure evolution in an FCC single crystal may produce a softening or hardening effect related to the void growth or collapse, setting the base for further research in terms of enhancement of creep properties of FCC single crystals. Strain rates along deformation were analysed allowing to understand the physics behind micro-structure evolution and its consequence in creep properties. Moreover, stress concentration around the inclusion phase depends highly on the crystal orientation and loading conditions.

Contents

List of Figures

List of Tables

1 Introduction

High pressure turbine blades entitles one of the most loading-demanding harsh environments any material may face. Finding a material able to withstand highly unsteady unsymmetrical loading, at more than $1600^{o}C$ within a turbulent flow coming at high velocities, containing the corrosive agents, resulting of the combustion between kerosene and oxygen, is not an easy task. All the mentioned above set the stage for creep to occur. Creep is the progressive time-dependent non-elastic deformation under constant mechanical load and high temperatures. Additionally, creep has been proven to be accompanied by micro-structural evolution such as dislocation motion or grain boundary diffusion. In fact, creep failure is one of the most critical failure modes in turbine blades.

In the early 40's, the development of nickel super-alloys such Inconel (Akca and Gürsel, 2015) was mainly conceived towards the mentioned high performance that engines in aviation started to demand. The nickel-based super-alloys are known to be corrosion and oxidation-resistant materials, having nickel as predominant component, followed by chromium. The election of Nickel as the main alloy element is not an arbitrary decision. Nickel performs a face-centre cubic (FCC) crystalline structure. FCC structures (Jin et al., 2019) are known to provide a high toughness and ductility properties. Additionally, stability at high temperatures is ensured for this super-alloys, as well as promising properties in the creep behaviour. The initial approach for engine turbine blade made use of the aforementioned nickel-based super-alloys with a poly-crystalline micro-structure. However, the presence of many grain boundaries played a detrimental role in creep and corrosion mechanisms (Joshi and Stein, 1972).

Later, in the 1970's, directional solidification manufacturing techniques have been used for turbine blades (Galantucci and Tricarico, 1998). In turn, the grain boundaries are aligned in one direction, normally the most stressed direction, i.e. along the blade. Creep resistance and stiffness were proven to increase substantially with respect to its predecessors, the classical poly-crystals nickel super-alloys (Kermanpur et al., 2000). But it was not until 1980 when the idea of manufacturing a turbine blade as a single crystal aroused. Early researches confirmed the benefits of single crystal in engines: the fatigue life and creep behaviour was increased more than an order of magnitude with respect to the classical poly-crystalline super-alloys.

A lot of effort was made in recent years to understand the suitability of single crystal for aviation engines. The complexity behind a single crystal resides in its inner deformation behaviour. It is known that the main deformation mechanism of a single crystal involves slip phenomena (Staroselsky and Cassenti, 2011)(Gell and Duhl, 1986). Additionally, some studies have been carried out recently regarding the porosity distribution and its effects in the creep life of the single crystal (Bokstein et al., 2007)(Anton and Giamei, 1985)(Caron and Lavigne, 2011). A big concern raised in this matter in the aviation industry and further techniques are in need of development to characterize the mechanical behaviour of single crystal avoiding expensive complex experimental tests. In fact, it has been proven that the presence of microscopic inclusions has a big impact on the macroscopic constitute response of solids in general. Failure in ductile solids takes place by nucleation, growth and coalescence of microscopic voids (Tvergaard, 1989). As a consequence, one of the main lines of work in solid mechanics has been the development of constitutive model that accounts for the aforementioned micro-structure evolution.

The main goal of this work is to provide an overview of the creep response of a classical aviation porous single crystal turbine blade nickel-based super-alloy, the ***PWA1484 Ni-based single crystal super-alloy*** (Srivastava and Needleman, 2012) under different loading states. Two approaches will be addressed for this purpose: an analytical homogenization method for porous visco-plastic single crystal and a numerical approach involving Finite Element computations of a Unit Cell using a commercial software.

While the analytical homogenization method provides a reduced-cost solution, its development process is complex and sometimes involves a loose of the physics behind it. Its accuracy is not always optimal and needs to be checked against experimental or numerical finite element computations. In the other hand, the computational homogenization approaches can accommodate a wider range of cases and physical behaviours, although it is more expensive to be conducted.

It is important to recall that metals, where our line of work focuses, consist of aggregates of porous single-crystals, forming a poly-crystalline structure. Initial homogenization approaches assumed isotropicity of the material as in (Gurson, 1977). More sophisticated models started to use non-linear homogenization techniques. A Variational Homogenization (VH) approach, that made use of a Linear Comparison Composite (LCC), was employed by (P. Castañeda, 1991) to pioneer in the definition of the bounds of the Hashin-Shritkman type for porous isotropic visco-plastic materials. Similar approaches, which made use of this innovative Variational method, led to other similar constitutive models (Willis, 1991) (P. P. Castañeda and Suquet, 1997) (P. Castañeda and Zaidman, 1994a). All the above mentioned accounted for the simplest case of micro-structure evolution, i.e. no void rotation, and isotropic visco-plastic matrix phases. The complex phenomena associated with the rotation of the inclusion phase was taken into account in latter works (Kailasam et al., 1997) (Aravas and Castañeda, 2004).

However, the innovative Variational Homogenization (VH) bounds were not able to predict the micro-structure evolution of low porosity visco-plastic materials for purely deviatoric loadings, leading to a over-stiff response. As a consequence, the homogenization approach needed to evolve. Authors started to include the so called second-order approaches (SO) in their constitutive models.

One of the first authors to propose an improved finite-strain constitutive model for porous single-crystal materials was Ponte-Castaneda in (Danas and Castañeda], 2009a) (Danas and Castañeda], 2009b). In spite of providing an accurate result for a wide range of loading configurations and porosity levels while comparing with available numerical results, it required some fitting parameters, not being entirely predictive. In the search for the fully predictive homogenization model, (Agoras and Ponte Castañeda, 2013) proposed an iterated variational homogenization approach as accurate as its previous version (Danas and Castañeda], 2009a)(Danas and Castañeda], 2009b), but not involving any ad-hoc modifications. Further improvements of this iterative approach were developed in (Song et al., 2015), where void rotation was taken into account for general shearing loads.

The upcoming improvements in homogenization techniques considered heterogeneous materials with crystalline visco-plastic phases. (DeBotton and Ponte Castaneda, 1995) and (Idiart and Ponte Castañeda, 2007) proposed an extension of the classical Variatinal Homogenization (VH) bounds to account for general anisotropicity of the phases. Additionally, in (Mbiakop et al., 2015) an ad-hoc modification of the VH bounds of (DeBotton and Ponte Castaneda, 1995) to obtain macroscopic accurate flow potentials for porous single crystal with general ellipsoidal inclusion phases under general loading was implemented. This method is the so called Modified Variational (MVAR).

Last achievements in homogenization methodologies were achieved in (Song and Ponte Castañeda, 2017b). A more general implementation was detailed in (Song and Ponte Castañeda, 2017b), where the VH bounds described in (DeBotton and Ponte Castaneda, 1995) were generalised for porous visco-plastic single crystal under large deformations. Moreover, this homogenization methodology was further improved by making use of the iterative second order approach (ISO), pioneered in (Agoras and Ponte Castañeda, 2013); allowing for improved accuracy for low porosities conditions and purely hydrostatic loadings. In fact, this homogenization approach will be used in this work.

Nevertheless, the accuracy of new second order variational homogenization approaches needs to be assessed. Computational homogenization techniques allow to simulate in a general manner the micro-structure evolution and the macroscopic response for the desired conditions. Pioneers in Finite Element Unit Cell computations succeeded in obtaining results for highly symmetric single crystals, such as FCC crystal with 100 initial crystal orientation (Srivastava and Needleman, 2012) (Srivastava and Needleman, 2015) (Gan et al., 2006). On the other hand, for highly anisotropic phases such the case of HCP single crystals, no available numerical results were found.

Recent achievements in Crystal Plasticity Finite Element Methods (CPFEM). CPFEM are suitable to represent the isothermal, anisotropic, large deformation of polycyrstalline metals for FCC, BCC and HCP structures (Marin, 2006).

However, the available results in literature are limited to a few values of loading conditions; and analysis of such important quantities in solid mechanics as the Strain

Rate, $\boldsymbol{D}$ is not considered. As a consequence, this work has its aim on filling this gap in porous visco-plastic FCC single crystal with an application in aerospace turbine blades. These FEM results will compared with the innovative homogenization approach proposed in (Song and Ponte Castañeda, 2017b).

In summary, this work will be divided in five sections. Firstly, the problem to address will be fully described by setting up the single crystal plasticity constitutive model as well as the details of the turbine blade specimen to model. Then, the Fully-Optimized Second Order (FOSO) together with the Iterative Second Order (ISO) approach of Ponte-Casteñeda will be detailed, paying special attention to the finite strain response and the description of the micro-structure evolution. Once the homogenization approach was presented, a Finite Element model of a Unit Cell with Periodic Boundary Conditions (PBC) will be implemented by using ABAQUS commercial software. This FEM model will be compared with the presented FOSO homogenization approach (Ponte Castañeda, 2015). The comparison of both approaches will be held for a range of loading configurations. For brevity, further analysis in initial crystal orientations, porosity levels, initial inclusion shape and creep properties will be exposed solely for the Unit Cell Finite Element approach. At the end of this chapter, the reader will have a complete overview of the creep behaviour in term of single crystal visco-plasticity for an FCC crystal. Future work and some conclusions in terms of numerical simulations and homogenization techniques will be drawn in a last instance.

In this work, in terms of notation, scalars quantities will be denoted by italic Roman, a , or Greek letters, β; vectorial values by italic boldface Roman letters, $\boldsymbol{l}$; second-order tensors by boldface Roman letters or Greek letters, $\mathbf{Z}$; and fourth order tensors by barred letters, $\mathbb{M}$.

2 Problem Description

2.1 Single Crystal Plasticity

Large deformation kinematics theory is considered in this work. Large finite-deformations consider the deformation of a initial or reference configuration, into a deformed or current configuration. Illustration 2.1 shows the framework of finite-deformation kinematics.

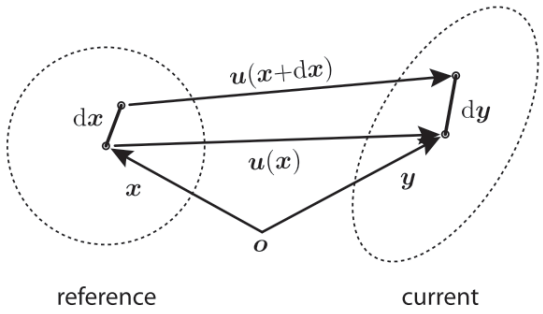


Figure 2.1: Initial and current configurations framework in large finite-deformation kinematics (Roters et al., 2010).

In that sense, the deformation gradient can be defined as:

$$\boxed{\mathbf{F} = \frac{\partial \mathbf{y}}{\partial \mathbf{x}} = \nabla \mathbf{y} \Leftrightarrow F_{ij} = \frac{\partial y_i}{\partial x_j} = y_{i,j}} \tag{2.1}$$

Flow of dislocations along crystallographic slip systems is the main mechanism of plastic deformation in metal crystal plasticity. In fact, dislocation motion is volume preserving, as it involves neither expansion nor compression of the crystallographic lattice of the single crystal. As a consequence, the crystal lattice remains undistorted once the dislocations have gotten past.

Additionally, a slip system is defined as a close packed atomic direction on a closed packed atomic plane, so that, the smallest Burgers Vector magnitude is achieved. As a consequence, the most favourable direction for dislocation motion through slip is found. In the case of concern, FCC single crystal, the slips systems are characterized

by [111][110], with 12 activated slip systems through deformation.

To understand deformation mechanisms in single crystals, it is important to interpret the concept of crystallographic lattice rotation. An example, where only one slip system is activated, is illustrated in Figure 2.2.

An out-of-balance moment is generated in the specimen as a consequence of the finite deformation nature of slip. In order to achieve equilibrium, the crystal lattice must rotate. It is important to note that this crystal lattice rotation is generally not uniform through the single crystal, having local behaviour.

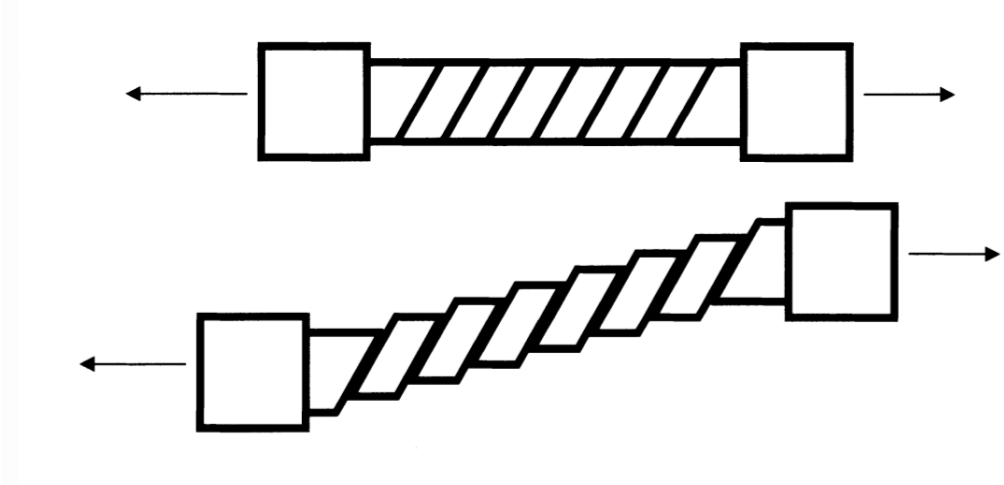


Figure 2.2: Tensile test in a single crystal specimen (Bohm, 2004)

Thus, the goal is to relate dislocation motion due to slip phenomena and plastic deformation in a continuum mechanics framework. For that purpose, a plastic shear strain, denoted as γ, is defined to represent dislocation flow in the single crystal. A sketch of the defined plastic shear strain is represented in Figure 2.3.

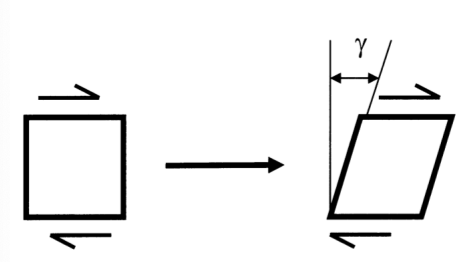


Figure 2.3: Plastic shear strain, γ for a given slip system (Bohm, 2004).

The deformation gradient in the framework of single crystal plasticity theory, $\mathbf{F}$ can be then written as:

$$\boxed{\mathbf{F} = \mathbf{F}^* \cdot \mathbf{F}^P} \tag{2.2}$$

$\mathbf{F}^P$ accounts for the plastic crystallographic slip and $\mathbf{F}^*$ for the stretching and rotation of the lattice of the single crystal. An illustrative representation can be found in Figure 2.4.

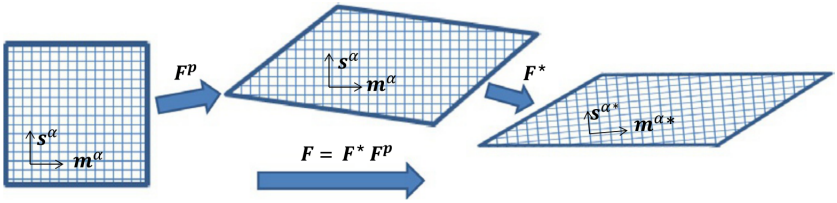


Figure 2.4: Initial and current configurations framework in large finite-deformation kinematics (Segurado et al., 2018).

In the reference configuration, the slip plane normal and slip direction unit vectors along a specific α slip system of the single crystal are denoted by $\boldsymbol{m}^{(\alpha)}$ and $\boldsymbol{s}^{(\alpha)}$.

In the current configuration, these quantities can be described as:

$$\boldsymbol{s}^{(\alpha)*} = \mathbf{F}^* \cdot \boldsymbol{s}^{(\alpha)} \quad \boldsymbol{m}^{(\alpha)*} = \boldsymbol{m}^{(\alpha)} \cdot \mathbf{F}^{*-1} \tag{2.3}$$

Taking the time derivative of Eq. (2.2):

$$\boxed{\dot{\mathbf{F}} \cdot \mathbf{F}^{-1} = \mathbf{D} + \boldsymbol{\Omega} = (\mathbf{D}^* + \boldsymbol{\Omega}^*) + (\mathbf{D}^p + \boldsymbol{\Omega}^p)} \tag{2.4}$$

where $\mathbf{D}^*$ denotes the elastic part of the stretching and $\boldsymbol{\Omega}^*$ is the so called "elastic" spin tensor. Additionally, the plastic part of the deformation is described by $\boldsymbol{\Omega}^p$ and $\mathbf{D}^p$ as follows:

$$\boxed{\mathbf{D}^p = \sum_\alpha \dot{\gamma}^{(\alpha)} \mathbf{P}^{(\alpha)}; \quad \boldsymbol{\Omega}^p = \sum_\alpha \dot{\gamma}^{(\alpha)} \mathbf{W}^{(\alpha)}} \tag{2.5}$$

In Eq. (2.5), $\dot{\gamma}^{(\alpha)}$ denotes the shear rate on a given slip system (α). A comment in this magnitude will be detailed in Eq. (2.8).

Moreover, the quantities $\mathbf{P}^{(\alpha)}$ and $\mathbf{W}^{(\alpha)}$ are given by:

$$\boxed{\mathbf{P}^{(\alpha)} = \frac{1}{2}\left(\mathbf{s}^{(\alpha)*} \otimes \mathbf{m}^{(\alpha)*} + \mathbf{m}^{(\alpha)*} \otimes \mathbf{s}^{(\alpha)*}\right)} \tag{2.6}$$

$$\boxed{\mathbf{W}^{(\alpha)} = \frac{1}{2}\left(\mathbf{s}^{(\alpha)*} \otimes \mathbf{m}^{(\alpha)*} - \mathbf{m}^{(\alpha)*} \otimes \mathbf{s}^{(\alpha)*}\right)} \tag{2.7}$$

For the case of concern, all twelve primary octahedral slip systems characterizing the FCC single crystal undergo slip phenomena that obey Schimd's law. As a consequence, the slip rate, $\dot{\gamma}^{(\alpha)}$ only depends on the resolved shear stress or Schmid stress, $\tau^{(\alpha)}$. Furthermore, the creep relation for the porous FCC single crystal is assumed to follow the power law secondary creep form, as follows:

$$\boxed{\dot{\gamma}^{(\alpha)} = \left\{\dot{\gamma}_0 \left|\frac{\tau^{(\alpha)}}{\tau_0}\right|^n\right\} \operatorname{sgn}\left(\tau^{(\alpha)}\right)} \tag{2.8}$$

where $\dot{\gamma}_0$ denotes the initial shear strain rate, τ_0 corresponds to the reference shear stress, and n to the "empirical" creep exponent.

Furthermore, it is of interest to have constitutive equations which are frame indifferent, i.e. objective. The Jaumann rate of Cauchy stress is one of most used objective stress rates due to its implementation simplicity.

Elastic strains are assumed to be small compared with plastic deformation. Besides, the Jaumann derivate of Cauchy stress with respect to the lattice, $\hat{\boldsymbol{\sigma}}^*$ (Srivastava and Needleman, 2012), can be written as:

$$\boxed{\hat{\sigma}^* = \dot{\sigma} + \sigma \cdot \boldsymbol{\Omega}^* - \boldsymbol{\Omega}^* \cdot \sigma = \mathbf{L} : \mathbf{D}^* - \sigma\left(\mathbf{I} : \mathbf{D}^*\right)} \tag{2.9}$$

where $\mathbf{L}$ denotes the elastic moduli tensor. Moreover, the Jaumann derivative with respect to the material may be given by:

$$\boxed{\hat{\sigma} = \dot{\sigma} - \boldsymbol{\Omega} \cdot \sigma + \sigma \cdot \boldsymbol{\Omega}} \tag{2.10}$$

If now, the difference between both corotationals is computed:

$$\hat{\sigma}^* - \hat{\sigma} = \sum_{\alpha} \dot{\gamma}^{(\alpha)} \mathbf{W}^{(\alpha)} \cdot \sigma - \sum_{\alpha} \dot{\gamma}^{(\alpha)} \sigma \cdot \mathbf{W}^{(\alpha)} \tag{2.11}$$

Then, by introducing:

$$\psi^{(\alpha)} = \mathbf{W}^{(\alpha)} \cdot \sigma - \sigma \cdot \mathbf{W}^{(\alpha)} \tag{2.12}$$

and making use of Eq. (2.4), (2.5), (2.9) and (2.10), the Jaumann derivative with respect to the material:

$$\boxed{\hat{\sigma} = (\mathbf{L} - \sigma \otimes \mathbf{I}) : \mathbf{D} - \sum_{\alpha} \dot{\gamma}^{(\alpha)} \mathbf{R}^{(\alpha)}} \tag{2.13}$$

having $\mathbf{R}^{(\alpha)} = \mathbf{L} : \mathbf{P}^{(\alpha)} + \psi^{(\alpha)}$.

Finally, the Schmid or shear resolved stress is defined by:

$$\boxed{\tau^{(\alpha)} = \boldsymbol{m}^{(\alpha)*} \cdot \sigma \cdot \boldsymbol{s}^{(\alpha)*} = \sigma : \mathbf{P}^{(\alpha)}} \tag{2.14}$$

The kinematics and the constitutive laws for the single crystal plasticity problem have been described.

The specimen of concern is the ***PWA1484 Ni-based single crystal super-alloy*** which has been shown to contain coherent FCC-based ordered $L1_2$ precipitates in a FCC matrix (Anton and Giamei, 1985). In this work, the nickel-based super-alloy will be modelled as an porous FCC single crystal. The FCC crystal will be modelled with the twelve primary octahedral slip systems (111) (110) active during deformation.

The ***PWA1484 Ni-based single crystal super-alloy*** is characterised by the following elastic constants: $C_{11} = 283.3GPa$, $C_{12} = 197.5GPa$ and $C_{44} = 112GPa$ (Srivastava and Needleman, 2012). The creep parameters of the specimen are: $n = 5$, $\tau_0 = 245MPa$ and $\dot{\gamma}_0^{(\alpha)} = 1.53 \cdot 10^{-9} s^{-1}$. The complete description of the mentioned variables will be addressed in detail in the next chapters.

The single crystal plasticity constitutive model is based on (R. Asaro and Needleman, 1985) and implemented in a User-defined Material Subroutine by (Huang, 1991).

3 Analytical Model

Structural mechanics finite element simulations need to take into account the microstructure evolution as its contribution has been proven to be crucial in the material macroscopic response of highly heterogeneous materials. However, the computational expense related to this approach would be over-demanding, new analytical "multi-scale" approaches are of interest. This chapter will focus in the line of homogenization methods.

In the line of homogenization methods, the macroscopic structure, Ω defines a Boundary-value problem (BVP) which is solved by assuming homogenized material properties in a specific material point at a macroscopic scale. These homogenized material properties come from the resolution of a meso-scale Boundary-value (BVP), ω. The aforementioned meso-scale structure will be denoted from this point as Representative Volume Element (RVE). An illustrative representation can be found in Figure 3.1.

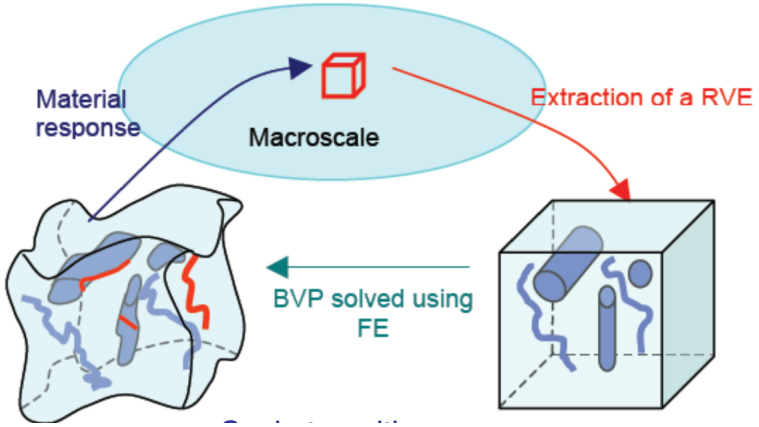


Figure 3.1: Schema of Homogenization approached in multi-scale problems (Wu et al., 2017).

By solving the RVE problem, an estimate of the average macroscopic strains and stress may be obtained. As depicted in the introductory chapter, homogenization-based approaches have been extensively developed in recent years. The interest in this work is for FCC porous single crystal, which presents non-linear material behaviour. As a

consequence, and in order to capture non-linear properties, it is necessary to use the so called Linear Comparison Composite (LCC).

The Linear Comparison Composite (LCC) is considered as a virtual composite material, containing the same micro-structural properties as the non-linear porous material, which behaves in a similar fashion as the linearized properties of the real phases for a certain strain/stress state. In fact, the LCC is the key component on obtaining accurate estimates of the macroscopic material properties in many homogenization-based approaches.

Additionally, important concepts in homogenization-based approaches are the first and second statistical moments of the stress and strains field.

The first and second statistical moments of the stress and strains fields are defined, respectively as follow:

$$\overline{\sigma}^{(r)} = \langle \sigma \rangle^{(r)}; \quad \overline{\epsilon}^{(r)} = \langle \epsilon \rangle^{(r)} \tag{3.1}$$

$$\langle \sigma \otimes \sigma \rangle^{(r)}; \quad \langle \epsilon \otimes \epsilon \rangle^{(r)} \tag{3.2}$$

From this point the triangular brackets denote volume averages over a Representative Volume Element (RVE) of porous material.

The majority of the homogenization methods briefly discussed in the introductory chapter of this work accounted only for the so called first statistical moment values for the phases when considering the visco-plastic behaviour. Nevertheless, in order to capture non-linear material properties accurately, the variations of the first moment estimates, such as $\boldsymbol{\Delta}\sigma$ and $\boldsymbol{\Delta}\epsilon$ need to be accounted for. This is achieved thanks to the second order statistical moments defined in Eq. (3.2).

From the use of these second order statistical moments of the stress field the Fully Optimized Second Order (FOSO) takes its last name.

As introduced in the previous chapters, a general finite-strain constitutive model for porous visco-plastic single crystals has been developed in (Song and Ponte Castañeda, 2017a). This homogenization approach provides, in a first instance, the instantaneous

response of the porous medium for a given initial micro-structure state; added to the capability of accounting for the micro-structure evolution, i.e. the evolution of the average void shape and size as well as the crystal anisotropicity caused by the lattice rotation during deformation. This method will be shown to be predictive (Agoras and Ponte Castañeda, 2013).

This method makes use of the fully optimised second-order (FOSO) variational approach in (Ponte Castañeda, 2015), together with the iterated variational homogenization procedure presented in (Agoras and Ponte Castañeda, 2013). A further development of this approach will be detailed later in the chapter, allowing to understand the need of using concepts such as Second order variational approach or iterated homogenization.

Furthermore, in (Song and Ponte Castañeda, 2017a) estimates for the average strain rate and spin fields in both phases of the porous single crystal were used to write evolution equations for the micro-structure change of the single crystal, i.e. porosity, void orientation and shape, and lattice rotation.

This homogenization approach will be compared with Finite Element Unit Cell computations and some conclusions will be drawn in the following chapters.

3.1 Theoretical Background

In (Song and Ponte Castañeda, 2017a) the porous visco-plastic single crystals are idealized as two-phase materials: void phase embedded in the single crystal matrix. The single crystal matrix (phase 1) is assumed to have a crystal lattice fully defined by three linearly independent crystallographic directions $\boldsymbol{l}_1$, $\boldsymbol{l}_2$ and $\boldsymbol{l}_3$. An illustration of the mentioned crystallographic axes can be found in Figure 3.2.

Classically, two main deformation mechanisms characterize the single crystal-matrix: (a) the elastic behaviour of the lattice and (b) the plastic deformation associated with the motion of dislocations. Many works have already shown that under large deformations, which is the concern of this work, the elastic component of the strains is of few orders of magnitudes smaller than the plastic strains. For this reason, the elastic

strains will be considered negligible in this work (Song and Ponte Castañeda, 2017a). From this point, to alleviate notation, the superscript p referring to plastic component used in previous sections will be omitted, unless specified.

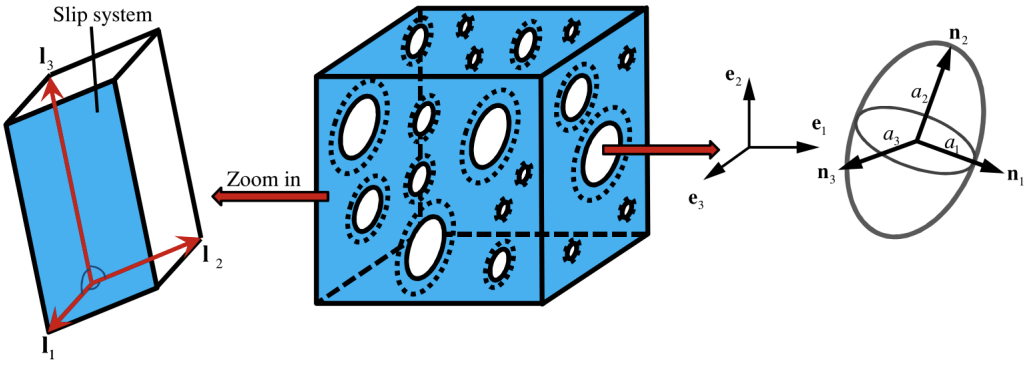


Figure 3.2: Representation of a porous single crystal consisting of aligned, ellipsoidal voids which are distributed with the same ellipsoidal symmetry in a single-crystal matrix (Song and Ponte Castañeda, 2017a).

As a conclusion, the single crystal-matrix will be assumed to deform solely by dislocation motion along α crystallographic slip systems, taking the local constitute behaviour of the matrix phase as visco-plastic.

The single crystal matrix behaviour can be characterized by:

$$\boxed{\mathbf{D} = \frac{\partial u^{(1)}(\sigma)}{\partial \sigma}, \quad u^{(1)}(\sigma) = \sum_{\alpha} \phi_{(\alpha)}\left(\tau_{(\alpha)}\right)} \tag{3.3}$$

In Eq. (3.3), $\mathbf{D}$ denotes the Eulerian strain rate; $u^{(1)}$ is the stress potential for the crystal matrix and σ the Cauchy stress. Additionally, for purposes of characterizing the response of the α slip systems, the convex functions $\phi_{(\alpha)}$ are defined, being dependent on the Schmid stresses, $\tau_{(\alpha)}$ as follows:

$$\tau_{(\alpha)} = \sigma : \mathbf{P}^{(\alpha)}; \quad \textit{where} \quad \mathbf{P}^{(\alpha)} = \frac{1}{2}\left(\boldsymbol{s}^{(\alpha)*} \otimes \boldsymbol{m}^{(\alpha)*} + \boldsymbol{m}^{(\alpha)*} \otimes \boldsymbol{s}^{(\alpha)*}\right) \tag{3.4}$$

As specified in Eq. (3.4), the $\mathbf{P}^{(\alpha)}$ is a second-order Schmid tensor obtained from the symmetric part of the dyadic product of the unit vectors characterizing the normal to the slip plane, $\mathbf{m}^{(\alpha)*}$, and along the slip direction, $\mathbf{s}^{(\alpha)*}$, of a specific crystallographic

slip system.

The previously defined slip potentials, $\phi_{(\alpha)}$, have been considered in (Song and Ponte Castañeda, 2017a) to be of the classic simple power-law type:

$$\phi_{(\alpha)}(\tau) = \frac{\dot{\gamma}_0 \left(\tau_0\right)_{(\alpha)}}{n+1} \left| \frac{\tau}{\left(\tau_0\right)_{(\alpha)}} \right|^{n+1} \tag{3.5}$$

In Eq. (3.5), $\dot{\gamma}_0$ represents the reference or initial strain rate; τ_0 the initial shear stress of a slip system and n is the so called creep exponent. The same power-law form will be used for the Finite Element computations. In the past years, it has been shown how this simple relationship is able to capture the non-linear behaviour for a range of materials (Lou and Schapery, 1971) (Chen and Argon, 1979).

The phase 2, the voids are taken to be of a general ellipsoidal shape, oriented in a certain direction and dispersed with a random distribution in the single crystal-matrix. This behaviour has been detailed in previous work (Agoras and Ponte Castañeda, 2013). As the concern of this work involves low porosities, it is assumed that the ellipsoidal shape and orientation of the distribution function are identical to the ellipsoidal shape and orientation of the voids (Agoras and Ponte Castañeda, 2014).

At this point, a set of micro-structural variables (Eq. (3.6)) may be defined to fully describe the micro-structure of a porous visco-plastic single-crystal.

$$\boxed{\mathbf{s} \equiv \{\boldsymbol{l}_1, \boldsymbol{l}_2, \boldsymbol{l}_3, f, w_1, w_2, \boldsymbol{n}_1, \boldsymbol{n}_2, \boldsymbol{n}_3\}} \tag{3.6}$$

Thus, 9 variables are needed to describe the micro-structure of a porous single-crystal:

- $\boldsymbol{l}_1, \boldsymbol{l}_2, \boldsymbol{l}_3$: The lattice vectors of the matrix phase.
- $f, w_1 = \frac{a_3}{a_1}, w_2 = \frac{a_3}{a_2}$: Volume fraction and aspect ratios characterizing the void size and shape.
- $\boldsymbol{n}_1, \boldsymbol{n}_2, \boldsymbol{n}_3$: Unit vectors aligned with the three principal directions of the ellipsoid.

In summary, from (Ponte Castañeda and Willis, 1999) the instantaneous macro-

scopic response of a porous single crystal of the form presented in this chapter can be completely determined by computing the effective stress potential $\widetilde{u}$ as follows:

$$\boxed{\overline{\mathbf{D}} = \frac{\partial \widetilde{u}(\overline{\sigma})}{\partial \overline{\sigma}}, \quad \widetilde{u}(\overline{\sigma}) = (1-f) \min_{\sigma \in \mathcal{S}(\bar{\sigma})} \langle u(\mathbf{x}, \sigma) \rangle^{(1)}} \tag{3.7}$$

Where $\widetilde{u}$ is the effective stress potential, $\mathcal{S}(\bar{\sigma})$ is the set of statistically admissible stress fields (Song and Ponte Castañeda, 2017a).

Due to the non-linear single crystal plasticity material behaviour, it would be practically impossible to obtain an exact solution to the resulting set of non-linear partial differential equations with random oscillating coefficients (Song and Ponte Castañeda, 2017a).

Here it rises the need for the fully optimized second order homogenization-based approach for computing approximate estimates for the material properties.

Thus, a novel iterative homogenization scheme was developed in (Agoras and Ponte Castañeda, 2013). This iterative approach will make use of the innovative fully optimized second-order (FOSO) variational homogenization method (Ponte Castañeda, 2015) to approximate estimates for the effective potential, $\widetilde{u}$.

3.2 FOSO variational estimates

In (Ponte Castañeda, 2015), it was developed a fully optimized second-order variational method (FOSO) which takes advantage of the effective behaviour of an appropriate selected porous "Linear Comparison Composite" (LCC). The Linear Comparison Composite has uniform matrix properties and a micro-structure which is identical to the porous non-liner single crystal of concern.

At this point, it is necessary to describe the porous LCC that concerns the FOSO approach. The porous LCC will be characterized by a crystal matrix described by a quadratic stress potential as shown in Eq. (3.8); and a micro-structure identical to the porous single crystal of concern.

$$\boxed{u_L^{(1)}(\sigma) = \frac{1}{2}\sigma \cdot \mathbb{M}^{(1)}\sigma + \eta^{(1)} \cdot \sigma} \tag{3.8}$$

$\mathbb{M}^{(1)}$ represents the fourth order viscous compliance tensor and $\boldsymbol{\eta}^{(1)}$ is the second order eigenstrain-rate tensor, both characterizing crystal matrix properties. $\mathbb{M}^{(1)}$ and $\boldsymbol{\eta}^{(1)}$ have been defined in (Ponte Castañeda, 2015) as follows:

$$\mathbb{M}^{(1)} = \sum_{\alpha} \frac{1}{2\mu_{(\alpha)}} \mathbf{P}_{(\alpha)} \otimes \mathbf{P}_{(\alpha)}, \quad \text{and} \quad \boldsymbol{\eta}^{(1)} = \sum_{\alpha} \eta_{(\alpha)} \mathbf{P}_{(\alpha)} \tag{3.9}$$

Further details on how to compute the scalars $\mu_{(\alpha)}$ representing the slip viscosities and $\eta_{(\alpha)}$, slip eigenstrain rates, will be provided in next sections.

Regarding the constitutive relation for the inclusion phase, it will be also assumed of the form of Eq. (3.8).

At this point, it is important to introduce the Ponte-Castaneda Willis (PCW) estimates. The PCW estimates are new estimates of the Hashin-Shtrikman kind for composite materials, i.e. different phases, with micro-structure composed of randomly shaped particles which are distributed in a matrix phase with "ellipsoidal symmetry". This new PCW estimate, which were developed in (Castañeda and Willis, 1995), is of special interest in this work as it allows to account for non-linear composites with evolving micro-structures.

Thus, the effective potential $\widetilde{u}_L$ for a porous LCC will be estimated by means of the Ponte-Castaneda Willis (PCW) estimates, detailed in (Castañeda and Willis, 1995).

$$\widetilde{u}_L(\overline{\boldsymbol{\sigma}}) = \frac{1}{2}\overline{\boldsymbol{\sigma}} \cdot \tilde{\mathbb{M}}\overline{\boldsymbol{\sigma}} + \tilde{\boldsymbol{\eta}} \cdot \overline{\boldsymbol{\sigma}}; \quad \text{where:} \quad \widetilde{\mathbb{M}} = \mathbb{M}^{(1)} + \frac{f}{1-f}\mathbb{Q}^{-1}, \quad ; \quad \tilde{\boldsymbol{\eta}} = \boldsymbol{\eta}^{(1)} \tag{3.10}$$

In Eq. (3.10), $\mathbb{Q}$ represents the fourth-order micro-structural tensor being dependent on the matrix property and micro-structure properties, i.e. void shape, size, distribution and orientation; $\widetilde{\mathbb{M}}$ denotes the effective compliance tensor and $\tilde{\boldsymbol{\eta}}$ corresponds to the effective eigenstrains rate tensor.

The microstructural tensor, $\mathbb{Q}$ is defined as follows (Song and Ponte Castañeda, 2017a):

$$\mathbb{Q} = \frac{1}{4\pi w_1 w_2} \int_{|\boldsymbol{\zeta}|=1} \frac{\left(\mathbb{M}^{(1)}\right)^{-1} - \left(\mathbb{M}^{(1)}\right)^{-1} \mathbb{H}(\boldsymbol{\zeta}) \left(\mathbb{M}^{(1)}\right)^{-1}}{|\mathbf{Z}^{-1}\boldsymbol{\zeta}|^3} dS \tag{3.11}$$

In Eq. (3.11) the details in the computation of $\mathbb{H}(\boldsymbol{\zeta})$ can be found in (Song and Ponte Castañeda, 2017a); $\boldsymbol{\zeta}$ denotes a unit vector and $\mathbf{Z}$ represents a second-order tensor that includes the micro-structural information in terms of void size, shape and orientation as well as its distribution in the matrix phase.

The tensor $\mathbf{Z}$ may be defined as:

$$\mathbf{Z} = w_1 \boldsymbol{n}_1 \otimes \boldsymbol{n}_1 + w_2 \boldsymbol{n}_2 \otimes \boldsymbol{n}_2 + \boldsymbol{n}_3 \otimes \boldsymbol{n}_3 \tag{3.12}$$

A last step before showing the expression of the FOSO estimate for the effective stress potential, is to define the first and second moments of the stress field for a matrix of a porous LCC, applied to the FOSO approach for the non-linear porous material.

$$\overline{\sigma}^{(1)} = \frac{1}{1-f}\overline{\sigma}, \quad \text{and} \quad \langle \sigma \otimes \sigma \rangle^{(1)} = \frac{2}{1-f}\frac{\partial \widetilde{u}_L}{\partial \mathbb{M}^{(1)}} \tag{3.13}$$

Finally, the FOSO estimate for $\widetilde{u}$ is given by (Ponte Castañeda, 2015):

$$\boxed{\widetilde{u}^{\mathrm{SO}}(\overline{\sigma}) = (1-f) \sum_{\alpha} \left[\alpha \phi_{(\alpha)} \left(\check{\tau}_{(\alpha)} \right) + (1-\alpha) \phi_{(\alpha)} \left(\hat{\tau}_{(\alpha)} \right) \right]} \tag{3.14}$$

In the previous equation, it has been introduced a new parameter, α. α, not to be confused with (α) related to the activated slip systems, is a user-chosen weight factor ranging from 0 to 1. Further comments on this weight factor will be exposed in later sections of this work. Furthermore, $\check{\tau}_{(\alpha)}$ and $\hat{\tau}_{(\alpha)}$ are stress variables depending on the first and second moments (Eq. (3.13)).

The stress variables introduced in Eq. (3.14) satisfy the relations (Ponte Castañeda, 2015):

$$\bar{\tau}_{(\alpha)} = \alpha \tilde{\tau}_{(\alpha)} + (1-\alpha)\hat{\tau}_{(\alpha)} = \overline{\sigma}^{(1)} \cdot \mathbf{P}_{(\alpha)} \tag{3.15}$$

and

$$\overline{\overline{\tau}}_{(\alpha)} = \alpha \left(\check{\tau}_{(\alpha)}\right)^2 + (1-\alpha)\left(\hat{\tau}_{(\alpha)}\right)^2 = \mathbf{P}_{(\alpha)} \cdot \langle \boldsymbol{\sigma} \otimes \boldsymbol{\sigma} \rangle^{(1)} \mathbf{P}_{(\alpha)} \tag{3.16}$$

$\bar{\tau}_{(\alpha)}$ and $\overline{\overline{\tau}}_{(\alpha)}$ describe the first and second moments of the Schmid stresses over the α slip system of the porous LCC matrix.

Furthermore, in the FOSO approach, it is imposed that $\check{\tau}_{(\alpha)} \leq \hat{\tau}_{(\alpha)}$. By using Eq. (3.15) and (3.16), the user may compute in a simple way the quantities $\check{\tau}_{(\alpha)}$ and $\hat{\tau}_{(\alpha)}$, such that:

$$\check{\tau}_{(\alpha)} = \bar{\tau}_{(\alpha)} - \sqrt{\frac{1-\alpha}{\alpha}}\sqrt{\overline{\overline{\tau}}_{(\alpha)} - \bar{\tau}_{(\alpha)}^2} = \bar{\tau}_{(\alpha)} - \sqrt{\frac{1-\alpha}{\alpha}}\mathrm{SD}^{(1)}\left(\tau_{(\alpha)}\right) \tag{3.17}$$

and

$$\hat{\tau}_{(\alpha)} = \bar{\tau}_{(\alpha)} + \sqrt{\frac{\alpha}{1-\alpha}}\sqrt{\overline{\overline{\tau}}_{(\alpha)} - \bar{\tau}_{(\alpha)}^2} = \bar{\tau}_{(\alpha)} + \sqrt{\frac{\alpha}{1-\alpha}}\mathrm{SD}^{(1)}\left(\tau_{(\alpha)}\right) \tag{3.18}$$

with

$$\mathrm{SD}^{(1)}\left(\tau_{(\alpha)}\right) = \sqrt{\overline{\overline{\tau}}_{(\alpha)} - \bar{\tau}_{(\alpha)}^2} \tag{3.19}$$

where $\mathrm{SD}^{(1)}$ represents the standard deviation of the resolved shear stresses over a specific slip system in the Linear Comparison Composite matrix.

On the other hand, the material properties of the porous LCC of concern have to be specified such that $\mu_{(\alpha)}$ and $\eta_{(\alpha)}$ satisfy the linearization conditions defined by (Ponte Castañeda, 2015):

$$\phi'_{(\alpha)}\left(\hat{\tau}_{(\alpha)}\right) - \frac{1}{2\mu_{(\alpha)}}\hat{\tau}_{(\alpha)} = \eta_{(\alpha)} = \phi'_{(\alpha)}\left(\check{\tau}_{(\alpha)}\right) - \frac{1}{2\mu_{(\alpha)}}\check{\tau}_{(\alpha)} \tag{3.20}$$

The conditions commented previously in terms of $\check{\tau}_{(\alpha)} \leq \hat{\tau}_{(\alpha)}$ and the specified in Eq. (3.20) imply:

$$\boxed{\frac{1}{2\mu_{(\alpha)}} = \frac{\phi'_{(\alpha)}\left(\hat{\tau}_{(\alpha)}\right) - \phi'_{(\alpha)}\left(\check{\tau}_{(\alpha)}\right)}{\hat{\tau}_{(\alpha)} - \check{\tau}_{(\alpha)}}} \tag{3.21}$$

Eq. (3.21) is known as the "generalized secant" linearization condition. The reader

is referred to (Song and Ponte Castañeda, 2017a) and (Liu et al., 2003) for a complete understanding of the linearization approach employed.

Eq. 3.17-3.20 lead to a system of $4K$ non-linear algebraic equations which can be solved numerically to obtain the variables of concern, $\mu_{(\alpha)}$, $\eta_{(\alpha)}$, $\check{\tau}_{(\alpha)}$ and $\hat{\tau}_{(\alpha)}$.

Once these quantities are known, the FOSO estimate for the macroscopic strain rate $\overline{\mathbf{D}}$ of the porous single crystal under specific loading conditions $\overline{\boldsymbol{\sigma}}$, can be obtained by:

$$\boxed{\overline{\mathbf{D}} = \frac{\partial \tilde{u}^{S0}}{\partial \bar{\sigma}}(\overline{\boldsymbol{\sigma}}) = \frac{\partial \tilde{u}_L}{\partial \bar{\sigma}}(\overline{\boldsymbol{\sigma}}) = \widetilde{\mathbb{M}}\overline{\boldsymbol{\sigma}} + \tilde{\boldsymbol{\eta}}} \tag{3.22}$$

As a complement to the FOSO estimate approach, some additional results can be obtained in terms of average strain rate and spin fields for the inclusion phase. Those results will be of crucial usefulness while defining the evolution equations of the microstructure in a latter section. The reader is referred to (Song and Ponte Castañeda, 2017a) for further details regarding this matter.

In summary, the Fully Optimized Second order (FOSO) estimates are known to be exact to second order in the heterogeneity contrast as discussed in (Ponte Castañeda, 2015), when used in combination with estimates for the Linear Comparison Composite (LCC) that are exact to second order in the heterogeneity contrast. In fact, this entitles the main advantage with previous variational homogenization approaches which were only accurate to the first order.

However, the described Fully optimized Second Order (FOSO) estimates present some inaccuracies while dealing with low porosity levels and high non-linear behaviour for cases of high stress triaxiality, χ. The inaccuracies are related to the initial assumption of uniformity for the LCC matrix phase (Eq. (3.8)). For the reasons commented, an iterative homogenization-based approach is detailed in the next section, where non-uniform properties for the LCC matrix phase are considered.

A word must be said regarding the previously introduced weight parameter α. The results presented in this section are valid for any choice of α. At this moment, there are no further mathematical analysis available in literature in terms of choosing an optimal value of the weight parameter. For this work (Song and Ponte Castañeda, 2017a) it was selected the most symmetric case, $\alpha = \frac{1}{2}$.

3.3 Iterated Second Order (ISO) estimates

As developed in (Song and Ponte Castañeda, 2017a), the iterative homogenization approach (ISO) of (Agoras and Ponte Castañeda, 2013) together with the FOSO approach detailed in previous section provides a further improvement in the estimates for the effective stress potential of the porous single crystal of concern.

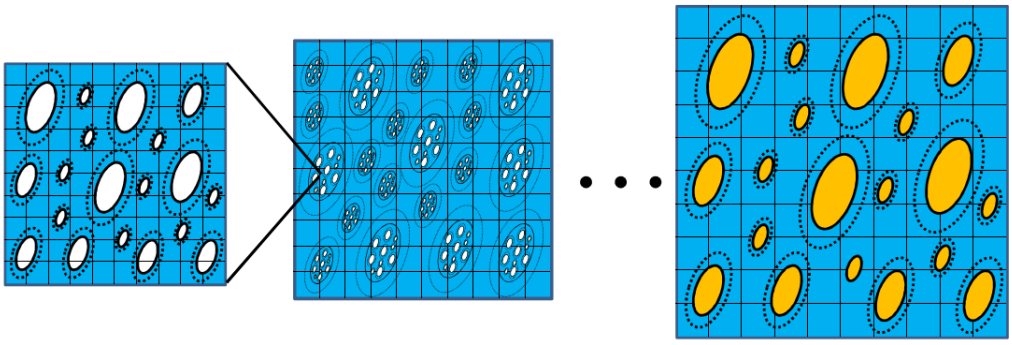


Figure 3.3: Representation of the iterated homogenization approach for a porous single crystal (Song and Ponte Castañeda, 2017a).

The idea behind the ISO approach is to build the porous micro-structure iteratively.

In the first iteration, the inclusions are dispersed in the single crystal matrix with a volume fraction $c^{(2)}_{[1]} > f$ and given shapes and distribution. In the second iteration, voids of the porous material from the first iteration are distributed in the same matrix phase with a volume fraction $c^{(2)}_{|2]}$ and with the identical voids and distribution shapes as the ones in the first iteration. The procedure, exposed in Figure 3.3, is repeated N times in such a manner that the resulting composite material *(i=N)* results in a non-linear porous material, so that:

$$\boxed{f = \prod_{i=1}^{N} c^{(2)}_{[i]}} \tag{3.23}$$

with $c^{(2)}_{[i]} (i = 2, \ldots, N)$ as the volume fraction of the "composite inclusion" in the *level-i* composite.

Indeed, the ISO approach makes use of, at each iteration, a Linear Comparison Composite (LCC) with an identical micro-structure to the non-linear composite of concern, although with linear single-crystal and inclusion phases. Thus, the stress potential adopts the same form as defined previously in Eq. (3.8) but with a redefinition of $\mathbb{M}$ and $\boldsymbol{\eta}$ tensors as:

$$\boxed{\mathbb{M}^{(1)}_{[i]} = \sum_{\alpha} \frac{1}{2\mu^{[i]}_{(\alpha)}} \mathbf{P}_{(\alpha)} \otimes \mathbf{P}_{(\alpha)}}, \quad \text{and} \quad \boxed{\boldsymbol{\eta}^{(1)}_{[i]} = \sum_{\alpha} \eta^{[i]}_{(\alpha)} \mathbf{P}_{(\alpha)}} \tag{3.24}$$

From this point, the label $[i]$ denotes quantities of the Linear Comparison Composite (LCC) at level i. Moreover, $\overline{\boldsymbol{\sigma}}_{[i]} = \langle \boldsymbol{\sigma} \rangle_{[i]}$ denotes the average stress field at the level-i composite.

Then, a set of equations, as defined for the FOSO estimates in the previous section, can be rewritten for the lowest iteration to characterize the effective behaviour of the level-1 porous single crystal:

$$\widetilde{u}^{ISO}_{[1]}\left(\bar{\sigma}_{[1]}\right) = \left(1 - c^{(2)}_{[1]}\right) \sum_{\alpha} \left[\alpha \phi_{(\alpha)}\left(\check{\tau}^{[1]}_{(\alpha)}\right) + (1-\alpha)\phi_{(\alpha)}\left(\hat{\tau}^{[1]}_{(\alpha)}\right)\right] \tag{3.25}$$

$$\check{\tau}^{[i]}_{(\alpha)} = \bar{\tau}^{[i]}_{(\alpha)} - \sqrt{\frac{1-\alpha}{\alpha}} \sqrt{\bar{\bar{\tau}}^{[i]}_{(\alpha)} - \left(\bar{\tau}^{[i]}_{(\alpha)}\right)^2} = \bar{\tau}^{[i]}_{(\alpha)} - \sqrt{\frac{1-\alpha}{\alpha}} SD^{(1)}_{[i]}\left(\tau^{[i]}_{(\alpha)}\right) \tag{3.26}$$

and

$$\hat{\tau}^{[i]}_{(\alpha)} = \bar{\tau}^{[i]}_{(\alpha)} + \sqrt{\frac{\alpha}{1-\alpha}} \sqrt{\bar{\bar{\tau}}^{[i]}_{(\alpha)} - \left(\bar{\tau}^{[i]}_{(\alpha)}\right)^2} = \bar{\tau}^{[i]}_{(\alpha)} + \sqrt{\frac{\alpha}{1-\alpha}} SD^{(1)}_{[i]}\left(\tau^{[i]}_{(\alpha)}\right) \tag{3.27}$$

Additionally, the slip viscosities and eigenstrains rates of the LCC matrix at the lowest level may be obtained by setting *i=1* in Eq. (3.28):

$$\frac{1}{2\mu^{[i]}_{(\alpha)}} = \frac{\phi'_{(\alpha)}\left(\hat{\tau}^{[i]}_{(\alpha)}\right) - \phi'_{(\alpha)}\left(\check{\tau}^{[i]}_{(\alpha)}\right)}{\hat{\tau}^{[i]}_{(\alpha)} - \check{\tau}^{[i]}_{(\alpha)}}, \quad \text{and} \quad \eta^{[i]}_{(\alpha)} = \phi'_{(\alpha)}\left(\tilde{\tau}^{[i]}_{(\alpha)}\right) - \frac{1}{2\mu^{[i]}_{(\alpha)}} \check{\tau}^{[i]}_{(\alpha)} \tag{3.28}$$

In the same fashion, the level-2 composite estimates may be obtain. It is important to remember that the conditions set for the FOSO estimates in the previous section are kept for the ISO approach, i.e. $\alpha = \frac{1}{2}$ and the stress variables $\check{\tau}_{(\alpha)}^{[i]}$ and $\hat{\tau}_{(\alpha)}^{[i]}$ $\left(\check{\tau}_{(\alpha)}^{[i]} \leq \hat{\tau}_{(\alpha)}^{[i]}\right)$.

In summary, the Iterated Second-Order (ISO) estimates for the effective stress potential of a porous single crystal at a general iteration i can be written as follows:

$$\boxed{\widetilde{u}_{[i]}^{ISO}\left(\bar{\sigma}_{[i]}\right) = \left(1 - c_{[i]}^{(2)}\right) \sum_{\alpha} \left[\alpha\phi_{(\alpha)}\left(\check{\tau}_{(\alpha)}^{[i]}\right) + (1-\alpha)\phi_{(\alpha)}\left(\hat{\tau}_{(\alpha)}^{[i]}\right)\right] + c_{[i]}^{(2)} \tilde{u}_{[i-1]}^{ISO}\left(\bar{\sigma}_{[i]}^{(2)}\right)} \quad (3.29)$$

Leading to the ISO estimate for the final iteration (i=N) as (Agoras and Ponte Castañeda, 2013):

$$\boxed{\widetilde{u}^{ISO}(\bar{\sigma}) \equiv \widetilde{u}_{[N]}^{ISO}\left(\bar{\sigma}_{[N]}\right) = \sum_{i=1}^{N}\left(1 - c_{[i]}^{(2)}\right)\left(\prod_{j=i+1}^{N} c_{|j|}^{(2)}\right)\left[\sum_{\alpha}\left(\alpha\phi_{(\alpha)}\left(\check{\tau}_{(\alpha)}^{[i]}\right) + (1-\alpha)\phi_{(\alpha)}\left(\hat{\tau}_{(\alpha)}^{[i]}\right)\right)\right]} \quad (3.30)$$

Regarding the $\mathbb{M}$ and $\boldsymbol{\eta}$ tensors for the inclusion phase, the conditions to be specified to level-i of the LCC are, in a similar fashion as in the FOSO estimates:

$$\mathbb{M}_{[1]}^{(2)} \to \infty, \quad \boldsymbol{\eta}_{[1]}^{(2)} = \mathbf{0} \quad \text{and} \quad \mathbb{M}_{[i]}^{(2)} = \widetilde{\mathbb{M}}_{[i-1]}, \quad \boldsymbol{\eta}_{[i]}^{(2)} = \tilde{\boldsymbol{\eta}}_{[i-1]}, \quad i = 2, \ldots, N \quad (3.31)$$

All in all, the computation of the ISO estimates from Eq. (3.30) involves the solution of *4K* x *N* non-linear algebraic equations where the unknowns are the stress variables, slip viscosities and slip eigenstrain rates, as in the FOSO estimates, but at each i-level of the composite.

A last remark may be said regarding the ISO estimates. It is expected that the accuracy of the ISO estimates (Eq. (3.30)) increases progressively with the number of levels, *N*. For this work, *N = 10* is used. This value has shown sufficiently accurate results in previous works (Song and Ponte Castañeda, 2017a).

3.4 Equations governing micro-structure evolution

The micro-structural variables which characterize the porous single crystal evolve with the deformation when the porous single-crystal is subjected to finite-strain deformations. The mentioned micro-structure evolution will have an effect on the macroscopic response of the non-linear single-crystal. To account for the change in crystallographic and morphological levels of the porous single crystal, a set of evolution laws is developed in (Song and Ponte Castañeda, 2017a).

As it is expected, the presence of inclusions will induce highly heterogeneous deformation and stress fields in the single crystal matrix phase. Nevertheless, for the homogenization matter, it is enough to account, during deformation, for the average crystallographic orientation of the lattice. As a consequence, the lattice will be taken to rotate rigidly with the average elastic spin $\bar{\Omega}_e^{(1)}$ in the single crystal matrix. Thus, the evolution laws characterizing the crystallographic axes $\boldsymbol{l}_1, \boldsymbol{l}_2$ and $\boldsymbol{l}_3$ can be written as:

$$\boxed{\boldsymbol{i}_1 = \bar{\Omega}^{*(1)}\boldsymbol{l}_1, \quad \boldsymbol{i}_2 = \bar{\Omega}^{*(1)}\boldsymbol{l}_2, \quad \boldsymbol{i}_3 = \bar{\Omega}^{*(1)}\boldsymbol{l}_3} \tag{3.32}$$

In Eq. (3.32), the $\bar{\Omega}^{*(1)}$ is the average micro-structural spin defined as the difference between the average "continuum" spin $\overline{\boldsymbol{\Omega}}^{(1)}$ and the average "plastic" spin $\overline{\boldsymbol{\Omega}}^{p(1)}$ in the matrix phase. For further details in the computation of both spin quantities, the reader is referred to the appendix of (Song and Ponte Castañeda, 2017a).

At this point, it is important to remember that the spin quantities, $\overline{\boldsymbol{\Omega}}^{(1)}$ and $\overline{\boldsymbol{\Omega}}^{p(1)}$ have been computed using the second moments of the stress field while the evolution laws use an averaging approach.

$$\overline{\boldsymbol{\Omega}}^{(1)} = \frac{1}{1-f}\left(\overline{\boldsymbol{\Omega}} - f\overline{\boldsymbol{\Omega}}^{(2)}\right) \tag{3.33}$$

$$\overline{\boldsymbol{\Omega}}^{p(1)} = \frac{1}{2}\sum_{\alpha}\bar{\gamma}_{(\alpha)}\left(\boldsymbol{s}^{(\alpha)*}\otimes\boldsymbol{m}^{(\alpha)*} - \boldsymbol{m}^{(\alpha)*}\otimes\boldsymbol{s}^{(\alpha)*}\right) \tag{3.34}$$

where $\bar{\gamma}_{(\alpha)}$ denotes the average slip rates, detailed in (Song and Ponte Castañeda,

2017a).

Recalling the incompressible behaviour of the single crystal matrix, the volume change of the porous single crystal is equal to the volume change of the inclusion phase. As a consequence, the porosity evolution law can be defined by (Gurson, 1977):

$$\boxed{\dot{f} = (1-f)\bar{D}_{kk}} \tag{3.35}$$

It is important to mention that void nucleation has not been considered in this work.

Following the work developed during the years in (P. Castañeda and Zaidman, 1994b), (Kailasam and Castañeda, 1998) and (Aravas and Castañeda, 2004), we may assume that the evolution of average shape and orientation of the voids is fully determined by the average strain rate and the average spin in the inclusion phase. Thus, the average shape of the voids remains ellipsoidal (and their distribution), but there is a change in its orientation and aspect ratios during deformation. In turn, the following evolution laws for the aspect ratios of the voids can be written:

$$\boxed{\dot{w}_1 = w_1\left(\bar{D}_{33}^{(2)'} - \bar{D}_{11}^{(2)'}\right)}, \quad \boxed{\dot{w}_2 = w_2\left(\bar{D}_{33}^{(2)'} - \bar{D}_{22}^{(2)'}\right)} \tag{3.36}$$

It is important to recall the definition of the aspect ratios: $w_1 = \frac{a_3}{a_1}$ and $w_2 = \frac{a_2}{a_1}$; and $\bar{\mathbf{D}}^{(2)}$ denotes the average strain rate of the inclusion phase.

Moreover, the evolution law for the orientation vectors may be expressed as:

$$\boxed{\dot{\boldsymbol{n}}_1 = \overline{\boldsymbol{\Omega}}^{(2)}\boldsymbol{n}_1, \quad \dot{\mathbf{n}}_2 = \overline{\boldsymbol{\Omega}}^{(2)}\boldsymbol{n}_2, \quad \dot{\boldsymbol{n}}_3 = \overline{\boldsymbol{\Omega}}^{(2)}\boldsymbol{n}_3} \tag{3.37}$$

As a contrast, in Eq. (3.37), the spin $\bar{\boldsymbol{\Omega}}^{(2)}$ is the spin of the Eulerian axes of the average deformation gradient of the void phase as in (Aravas and Castañeda, 2004). Further details in the computation of the non-zero components of $\bar{\boldsymbol{\Omega}}^{(2)}$ can be found in (Ogden, 1984).

Another comment may be said regarding the evolution laws for shape and orientation of voids. While it could have been considered in a separate manner (Kailasam et al., 1997), the approach presented in this work considers them in a joint manner. This is justified since the non-linear single crystal of concern involves low to moderate porosity levels prior to void coalescence. In this case, the effect of void distribution on

the macroscopic behaviour is expected to be negligible.

Additionally, the strain hardening due to interaction between slip systems will not be considered in this homogenization approach, so that, the reference flow stresses or initial shear stress $(\tau_0)_{(\alpha)}$ characteristic of all the slip systems will remain fixed during deformation.

All in all, the macroscopic behaviour of the non-linear single crystal has been fully described by means of the constitutive relation detailed in Eq. (3.22) along with the evolution laws described in Eq. (3.32), (3.35)-(3.37). Additionally, the numerical integration scheme used for the iterative second order (ISO) constitutive approach may be implemented by making use of an explicit, forward-Euler integration scheme, as detailed in (Liu et al., 2003).

The model described in this chapter allows to account for the evolution of the micro-structure variables defined in Eq. (3.6), providing a visco-plastic model for the macroscopic response of a porous non-linear single crystal with general crystallographic anisotropy, general ellipsoidal voids, subjected to general, finite-strain loading conditions.

4 Computational Homogenization

The advantage of the variational homogenization-based approaches has been clearly stated in terms of saving costs; however the physics behind describing the material properties of a non-linear porous FCC single crystal needs to be tested. Computational homogenization methods rise to cover this need.

4.1 Theoretical Background

First of all, let's consider a macro-scale problem, Ω and its corresponding Representative Volume Element (RVE), ω in a specific material point.

Some basic assumptions need to be made in terms of macroscopic and microscopic material behaviours. The material of concern should be sufficiently homogeneous in a macroscopic scale but heterogeneous in the micro-scale, i.e. inclusions in a single crystal matrix. "Scale separation" must be ensured, such that the characteristic length at micro-scale must be much smaller than the characteristic length in the macroscopic level. Additionally, the characteristic size of the heterogeneities must be much smaller than the size of the Representative Volume Element (RVE).

The RVE must contain a complete information to characterize the micro-structure of the non-linear FCC single crystal in a reduced size, allowing for computational efficiency.

The equilibrium of the RVE problem will be consistent with the boundary conditions, directly related to the macroscopic fields.

In the context of computational homogenization, the homogenised macroscopic response is estimated from the resolution of a meso-scale Boundary Value Problem (BVP) by using a finite element method approach (Noels, n.d.).

Specifically this chapter tackles a finite-element approach of a **Unit Cell**. Assuming that the dynamical effects are negligible, the equilibrium equations in the micro-scale can be written as follows:

$$\begin{aligned} \mathbf{P}(\boldsymbol{x}) \cdot \boldsymbol{\nabla}_0 = 0 \quad & \forall \boldsymbol{x} \in \omega(\boldsymbol{X}) \\ \mathbf{P}(\boldsymbol{x}) \cdot \boldsymbol{n} = \boldsymbol{T}(\boldsymbol{x}) \quad & \forall \boldsymbol{x} \in \partial\omega(\boldsymbol{X}) \end{aligned} \tag{4.1}$$

In Eq. (4.1), $\mathbf{P}$ refers to the first Piola-Kirchhoff stress tensor, $\boldsymbol{\nabla}_0$ to the gradient operator in the reference configuration and $\boldsymbol{T}$ denotes the surface traction on the boundary, per unit reference surface.

At this point it is important to remember the relationship between the first Piola-Kirchoff stress tensor and the Cauchy stress tensor used for the simulation purposes, as follows:

$$\boxed{\sigma = J^{-1}\mathbf{P}\mathbf{F}^{\mathrm{T}}} \tag{4.2}$$

where J is the determinant of the deformation gradient, $\mathbf{F}$.

Moreover, the local constitute laws of the different phases at the current time t in the material point of interest need to be defined to complete the formulation of the micro-scale problem. This behaviour has been already introduced in ***Chapter 2 - Problem Statement***. under Single Crystal Plasticity.

Jumping to the macro-scale, the linear momentum equation may be written as follows, in absence of dynamical effects:

$$\bar{\mathbf{P}}(\boldsymbol{X}) \cdot \boldsymbol{\nabla}_0 + \bar{\boldsymbol{b}} = 0 \quad \forall \boldsymbol{X} \in \Omega \tag{4.3}$$

In Eq. (4.3) $\bar{\boldsymbol{b}}$ denotes the body forces per unit reference volume. Besides, the boundary conditions for the macro-scale problem are:

$$\begin{aligned} \bar{\boldsymbol{u}}(\boldsymbol{X}) &= \overline{\boldsymbol{u}_{\boldsymbol{BC}}} \quad \forall \boldsymbol{X} \in \partial_D \Omega \\ \bar{\mathbf{P}}(\boldsymbol{X}) \cdot \bar{\boldsymbol{n}} &= \overline{\boldsymbol{t}_{\boldsymbol{BC}}} \quad \forall \boldsymbol{X} \in \partial_N \Omega \end{aligned} \tag{4.4}$$

where $\overline{\boldsymbol{t}_{\boldsymbol{BC}}}$ and $\overline{\boldsymbol{u}_{\boldsymbol{BC}}}$ represent the conditions on the Neumann boundary, $\partial_N \Omega$ and Dirichlet boundary, $\partial_D \Omega$, respectively.

As specified for the local problem, a constitutive resolution needs to be defined to complete the macro-scale formulation, as exposed previously in ***Chapter 2 - Problem Statement***. under Single Crystal Plasticity.

Recovering the homogenization context of the problem, the macroscopic quantities

may be rewritten as the averages of the micro-scale quantities over the meso-scale volume element, as follows (Noels, n.d.):

$$\begin{aligned} \bar{\mathbf{F}}(\boldsymbol{X}) &= \frac{1}{V(\omega)} \int_{\omega(\boldsymbol{X})} \mathbf{F}(\boldsymbol{x}) d\boldsymbol{x}, \\ \bar{\mathbf{P}}(\boldsymbol{X}) &= \frac{1}{V(\omega)} \int_{\omega(\boldsymbol{X})} \mathbf{P}(\boldsymbol{x}) d\boldsymbol{x} \end{aligned} \tag{4.5}$$

Besides, the energy consistency between both scales, i.e. "scale transition", macro and micro, is stated by means of the Hill-Mandel condition (Noels, n.d.):

$$\bar{\mathbf{P}}(\boldsymbol{X}) : \delta\bar{\mathbf{F}}(\boldsymbol{X}) = \frac{1}{V(\omega)} \int_{\omega} \mathbf{P}(\boldsymbol{x}) : \delta\mathbf{F}(\boldsymbol{x}) d\boldsymbol{x} \tag{4.6}$$

A brief description of the theoretical background of the meso-scale BVP problem has been exposed. The practicalities of implementing a Unit Cell finite element for solving the aforementioned meso-scale BVP is tackled in the rest of the chapter.

4.2 A Crystal Plasticity Unit Cell model

This chapter details the construction of the Unit Cell Finite Element Model. For the purpose of FEM simulations, the commercial software *ABAQUS Standard 2018* has been used.

The efforts to complete the literature content in terms of Finite Element results for the Unit Cell problem have been increasing in the last years.

Since the pioneers in Finite Element computation for the Unit Cell problem simulated a simple model for single crystal plasticity (Srivastava and Needleman, 2012), the numerical techniques have evolved substantially.

In (Srivastava and Needleman, 2012) and (Srivastava and Needleman, 2015) a brief analysis was conducted regarding FCC single crystals for a variety of loading configurations, crystal orientations; and stress triaxiality and Lode parameter values. However, no calculations were made in terms of negative stress triaxiality values, creep exponent or initial void shape. Other results, included the analysis for higher porosity values with a random distribution of inclusions in a mono-disperse and poli-disperse fashion (Kouchmeshky and Zabaras, 2009) (Zhang et al., 2005) (Roters et al., 2010).

Previous work in the field of Finite Element computation for the creep response of a porous single crystal constructed a different Finite Element model for each case of initial crystallographic orientation, by taking advantage of the symmetry properties of its own lattice or the symmetry of its related slip deformation mechanism (Srivastava and Needleman, 2012) (Srivastava and Needleman, 2015).

From the author's point of view, while it is true that the computational time may be reduced for the simplest cases, i.e the most symmetric cases; this fact adds unnecessary complexity to the modelling task. For this reason, this work has the aim to build a unit cell model which can be used for any initial crystal orientation under any general loading conditions with fully periodic boundary conditions.

As stated in previous chapters, the aim of this work is to provide a complete overview of the creep properties of FCC single crystals, characteristic of nickel-based super-alloys used in aerospace applications, under a range of loading scenarios, in terms of numerical computations. For this purpose, all the range of stress triaxiality values, from negative to positive values, Lode parameter, other crystal orientations, different initial ellipsoidal void shapes and porosity levels will be reviewed.

The notation adopted for the crystal orientation definition is as follows:

- Global crystal orientation given by: [100][010][001].
- Relative initial crystal orientation defined by x_i axes. The x_i axes will define the main loading directions for simulation purposes.

One of the parameters to analyse while comparing with the homogenization approach of Ponte-Castaneda (Song and Ponte Castañeda, 2017b) is the initial crystal orientation. Two different cases will be computed. An info-graph gathering those cases can be found in Figure 4.1.

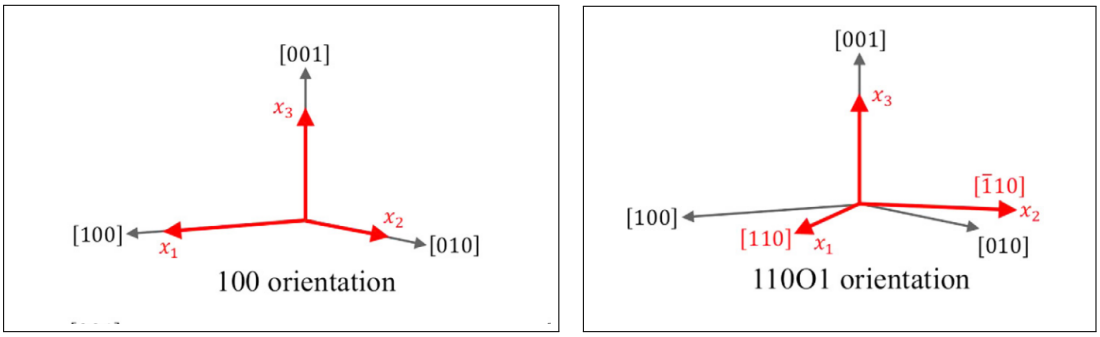


Figure 4.1: Initial crystal orientation. Relative orientation of the porous FCC single crystal (Srivastava and Needleman, 2015).

It is important to mention that the initial crystal orientation is given as an input to the User-defined Material subroutine (UMAT) as two pairs of vectors.

Furthermore, the periodic boundary conditions (PBC) need to be imposed. The periodic boundary conditions will be imposed by making use of the "dummy node" approach.

The last step to define completely the Unit Cell problem concerns the description of the loading parameters: stress triaxiality, χ and Lode parameter, L. The definition of the loading parameters will be addressed later in this chapter.

4.2.1 Geometry Description

The first step while conducting a Finite Element computation is to define the parameters that describe the geometry.

In this problem, the geometry is cubic of semi-length, a_0. The void is represented by a sphere or an ellipsoid depending on the case, located in the center of the Unit Cell.

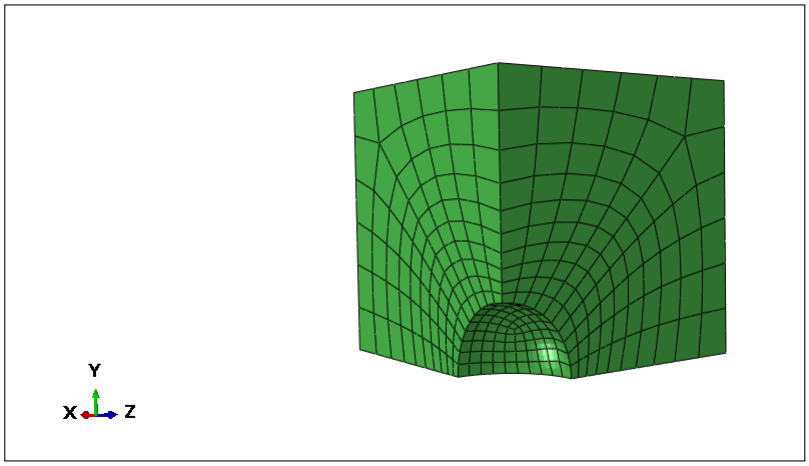


Figure 4.2: $\frac{1}{8}$ of the Unit Cell for a spherical inclusion.

The porosity level serves as an input and according to that, the radius of the sphere is computed as follows:

$$\boxed{r_0 = a_0 \sqrt[3]{\frac{6 \cdot f}{\pi}}} \tag{4.7}$$

where f represents the porosity level.

Moreover, some cases will consider an initial ellipsoidal void shape. For this approach, the input parameters will be the initial porosity level and the asperity, w, defined as the ratio between the minor axis and the major axis, $w = \frac{r_2}{r_1}$. In this work

the third semi-axis will be identical as the major axis, meaning $\frac{r_3}{r_1} = 1$.

In that case the computation of the radius, r_2 and r_1 is given by:

$$\boxed{r_1 = a_0 \sqrt[3]{\frac{6f}{\pi w}}}; \quad \boxed{r_2 = r_1 \cdot w} \tag{4.8}$$

Practically, a cube of length a_0 is created. a_0 will be fixed to 0.5 mm for the whole analysis. Then a sphere of r_0 is superimposed to the previous cube. A new geometry is defined then by means of a Boolean cut between the cube and the sphere.

This procedure allows to obtain $\frac{1}{8}$ of the Unit Cell. The result of the described geometrical operations can be seen in Figure 4.2. The next step involves the "extension" in the appropriate directions until the full Unit Cell of length $2a_0$ with a full spherical void is achieved.

Details on the python scripting for the geometry construction can be found in Appendix A.

On the other hand, in any Finite Element Model, mesh convergence must be ensured in order to achieve meaningful results. All the details regarding the mesh parameters considered as well as a complete mesh convergence study is included at the beginning of next chapter.

4.2.2 Material Behaviour - UMAT Subroutine

The constitutive material law for single crystal plasticity has been implemented using the User-Material Subroutine (UMAT) developed by Huang (Huang, 1991).

This section will be used additionally as a review for the elasto-plastic constitutive formulation included in the mentioned UMAT Fortran subroutine.

ABAQUS software allows the possibility of a completely user-defined material. By using this capability, the stresses, the strains and other solution dependent quantities are solved in an incremental fashion. The UMAT subroutine is called at the start of

every time step and it is provided with the initial state of the increment, i.e. stress and solution dependent variables; and with the strain increments and the time increment. The UMAT subruoutine makes two computations:

- Update the stress and the solution dependent variables to their correspondent value at the end of the time increment.
- Provide the material Jacobian matrix. This is used by the constitutive laws as require for the implemented iterative Newton-Rhapson method.

The kinematics framework included in the mentioned subroutine is based on the work by (Rice, 1971) and (Hill and Rice, 1972).
In this work, plastic deformation is assumed to be due to crystallographic dislocation slip. Twinning, diffusion and grain boundary sliding is not considered in this work. Moreover, the Schmid stress is assumed to be the driving force for slip phenomena.

The kinematics and the constitutive behaviour included in the User-defined Material (UMAT) subroutine has been previously detailed in ***Chapter 2 - Problem Statement.***

Hardening law

The slip rate is considered to depend solely on the current stress state via the associated Schmid stress. Thus, we may write the slip rate, $\dot{\gamma}^{(\alpha)}$ as:

$$\boxed{\dot{\gamma}^{(\alpha)} = \left\{ \dot{\gamma}_0 \left| \frac{\tau^{(\alpha)}}{\tau_0} \right|^n \right\} \operatorname{sgn}\left(\tau^{(\alpha)}\right)} \tag{4.9}$$

From Eq. (4.9), it can be concluded that a viscous power law relationship has been used to characterize the mentioned slip rates in each α plane. It is also important to notice that only secondary creep behaviour will be considered in this work, as previously commented in ***Chapter 3 - Homogenization Model***. In Eq. (4.9), $\dot{\gamma}_0$, τ_0 and n have been already defined in previous chapters.

The capabilities of the mentioned UMAT subroutine allow the user to include two different hardening of rate-dependent laws: Pierce, Asaro and Needleman's self hardening moduli and Bassani and Wu's law. More details regarding the inclusion of self

and latent hardening behaviour between slip systems can be found in (Peirce et al., 1982),(R. J. Asaro, 1983) and (Bassani and Wu, 1991). This work will not consider any self nor latent hardening.

UMAT Forward Gradient Time Scheme

The last step while defining a User-Material subroutine is the time integration scheme. Huang's UMAT subroutine makes use of the tangent modulus method first described by Peirce in (Peirce et al., 1984).

In practice, the Finite Element code needs of 160 inputs parameters, as follows:

- Elastic moduli of the single cyrstal, C_{ij}.
- Number of slip systems to be activated.
- Normal to a slip plane and slip direction for the first set of slip systems.
- Initial orientation of the crystal, given by two pairs of vector defining local and global directions.
- Slipping rate dependence parameters, given by $\dot{\gamma_0}$ and n.
- Self and latent hardening parameter, which will depend on the chosen hardening law for interaction between slip systems.
- Parameters that define the integration scheme and the consideration of finite strain and finite rotation or not, given by θ and NLGEOM.
- Parameters to define the iterative procedure in terms of absolute error of shear strains in slip systems, γ_{err}; as well as the maximum number of iterations ITRMAX.

For the case of concern, a nickel-based super-alloy has been considered (Srivastava and Needleman, 2015). A summary of the general inputs for the UMAT single crystal plasticity model can be found in the Table 4.1.

Parameter	Value
C_{11}	283.3 GPa
C_{12}	197.5 GPa
C_{44}	112 GPa
NSET	12
Slip systems	[111][110]
Crystal orientation	[100][100];[010][010]
$\dot{\gamma_0}$	$1.53 \cdot 10^{-9} s^{-1}$
n	5
τ_0	245 MPa
θ	0.5
NLGEOM	1
ITRATN	1
ITRMAX	10
γ_{err}	10^{-5}

Table 4.1: UMAT input parameters for FCC single crystal.

All in all, it has been showed the easiness of changing the material parameters for a wide range of cases. The capability of implementing HCP crystal plasticity or more sophisticated hardening law for interaction between slip systems will be addressed at the end of this work.

4.2.3 Periodic Boundary Conditions (PBC)

As commented in the introduction of this chapter, periodic boundary conditions need to be applied in the Unit Cell boundaries.

Historically, the implementation of periodic boundary conditions have been a matter of concern due to its complexity. One of the most used approaches while implementing PBC in a Finite Element Unit Cell problem is the "dummy node" method (ABAQUS, 2014).

The "dummy node" technique simplifies the PBC problem to three reference points, *RFs*. These reference points will be used to link and track the displacement of paired opposite boundaries; one for each principal direction, RF_1, RF_2 and RF_3.

The conditions to be imposed to achieve a fully-periodic boundary Unit Cell problem

in the Finite Element model are given by:

$$
\begin{aligned}
u_1(a_0, x_2, x_3) &= u_1(-a_0, x_2, x_3) + U_1'(t) \\
u_2(a_0, x_2, x_3) &= u_2(-a_0, x_2, x_3) + U_2'(t) \\
u_3(a_0, x_2, x_3) &= u_3(-a_0, x_2, x_3) + U_3'(t) \\
\\
u_1(x_1, a_0, x_3) &= u_1(x_1, -a_0, x_3) + U_1''(t) \\
u_2(x_1, a_0, x_3) &= u_2(x_1, -a_0, x_3) + U_2''(t) \\
u_3(x_1, a_0, x_3) &= u_3(x_1, -a_0, x_3) + U_3''(t) \\
\\
u_1(x_1, x_2, a_0) &= u_1(x_1, x_2, -a_0) + U_1'''(t) \\
u_2(x_1, x_2, a_0) &= u_2(x_1, x_2, -a_0) + U_2'''(t) \\
u_3(x_1, x_2, a_0) &= u_3(x_1, x_2, -a_0) + U_3'''(t)
\end{aligned}
\tag{4.10}
$$

In Eq. (4.10), the quantities U_i', U_i'' and U_i''' represent the *ith* displacement degree of freedom of the dummy nodes. The time histories of these displacements are computed by the analysis to provide a specific loading condition as in (Srivastava and Needleman, 2015).

ABAQUS software allows to implement the periodic boundary conditions relations given by Eq. (4.10) by using the so called ***Multi-Point constraint capability (MPC)***(ABAQUS, 2014). In order to do so, a Python script has been developed. The python script to apply Periodic Boundary conditions to a cubic Unit Cell can be found under Appendix B.

The linear multi-point constraint capability is used to enforce Eq. (4.10) and determine the values of the unknowns U_i', U_i'' and U_i'''.

The mentioned script allows to identify pair nodes located in opposite faces. Once they are detected, it applies the relationships described in Eq. (4.10). In that sense, nodes corresponding to opposite face will be detected and associated through the described periodic relation to specific reference points, RF_s.

4.2.4 Loading Conditions in a Unit Cell PBC problem

A typical single crystal turbine blade is subjected to a variety of tests before entering service. One of the most critical test is the creep test. Typical values of the stresses

applied during a creep test that classical single crystal turbine blades face, are used in this work (Seetharaman and Pratt, 2003).

While conducting an experimental creep test it is know that the loading conditions are said to be Nominal Stress condition, i.e. Dead load experiment.

However, the objective of this work goes beyond that, it focuses on providing a parametric study in terms of the Lode Parameter, L and Stress triaxiality, χ. It has been detailed in (Srivastava and Needleman, 2012) how a True Stress Loading case, or constant Cauchy stress loading is the best option for the analysis of a phenomenological constitutive relation.

From this point, the true stresses or Cauchy Stresses, Σ_i , are described by the following relationships:

$$\boxed{\begin{aligned}\Sigma_1 &= \frac{1}{a_2 a_3}\int_0^{a_2}\int_0^{a_3}\sigma_{11}\left(a_1, x_2, x_3\right)\mathrm{d}x_2\mathrm{d}x_3\\ \Sigma_2 &= \frac{1}{a_1 a_3}\int_0^{a_1}\int_0^{a_3}\sigma_{22}\left(x_1, a_2, x_3\right)\mathrm{d}x_1\mathrm{d}x_3\\ \Sigma_3 &= \frac{1}{a_1 a_2}\int_0^{a_1}\int_0^{a_2}\sigma_{33}\left(x_1, x_2, a_3\right)\mathrm{d}x_1\mathrm{d}x_2\end{aligned}} \tag{4.11}$$

In Eq. (4.11), the quantities a_1, a_2 and a_3 are defined as:

$$\boxed{\begin{aligned}a_1 &= a_0 + U_1;\\ a_2 &= a_0 + U_2;\\ a_3 &= a_0 + U_3\end{aligned}} \tag{4.12}$$

Quantities defined in Eq. (4.12) represent the deformed length of the cube in the three principal directions.

a_1, a_2 and a_3 are computed thanks to the "sensor" capability implemented in ABAQUS software (ABAQUS, 2014). As it has been commented previously, the displacements, U_1, U_2 and U_3 are tracked during the entire simulation.

Having these quantities available, and taking into account that the faces of the cube undergo large deformations which lead to a change in the surface; the applied tractions in the reference points need to be "corrected". This correction is effective at the beginning at each time step and allows to efficiently implement a constant true, or Cauchy stress state.

The time history of the surface values in the three principal directions is computed as follows:

$$\boxed{\begin{aligned} S_1 &= (2a_0 + U_2) \cdot (2a_0 + U_3); \\ S_2 &= (2a_0 + U_1) \cdot (2a_0 + U_3); \\ S_3 &= (2a_0 + U_1) \cdot (2a_0 + U_2) \end{aligned}} \tag{4.13}$$

The previous relationship (Eq. (4.13)) is valid as far as the boundaries of the unit cell remain straight.

Once again, the update of applied tractions is possible thanks to a ***developed User-defined Amplitude subroutine (UAMP) capability*** of ABAQUS software (ABAQUS, 2014). The defined tractions, Σ_i are prescribed at the beginning of the simulations. These tractions are associated to a Amplitude Function controlled by the mentioned UAMP subroutine. This subroutine runs at the beginning at each time step, modifying the amplitude associated to each traction accordingly to the respective computed surface (current configuration).

Further details of the implementation of the User-defined Amplitude Subroutine can be found in Appendix E.

At this point, it is important to define the loading parameters, Lode parameter L and stress triaxiality, χ as:

$$\boxed{\chi = \frac{\Sigma_{\mathrm{h}}}{\Sigma_{\mathrm{e}}} = \frac{\sqrt{2}}{3} \frac{1 + \rho_2 + \rho_3}{\sqrt{(1 - \rho_2)^2 + (\rho_2 - \rho_3)^2 + (\rho_3 - 1)^2}}} \tag{4.14}$$

$$\boxed{L = \frac{2\Sigma_2 - \Sigma_1 - \Sigma_3}{\Sigma_1 - \Sigma_3} = \frac{2\rho_2 - 1 - \rho_3}{1 - \rho_3}} \tag{4.15}$$

where Σ_{e}, Σ_{h}, ρ_2 and ρ_3 are given by:

$$\rho_2 = \frac{\Sigma_2}{\Sigma_1}; \quad \rho_3 = \frac{\Sigma_3}{\Sigma_1} \tag{4.16}$$

$$\Sigma_{\mathrm{e}} = \frac{1}{\sqrt{2}}\sqrt{\left(\Sigma_1 - \Sigma_2\right)^2 + \left(\Sigma_2 - \Sigma_3\right)^2 + \left(\Sigma_3 - \Sigma_1\right)^2} \tag{4.17}$$

$$\Sigma_{\mathrm{h}} = \frac{1}{3}\left(\Sigma_1 + \Sigma_2 + \Sigma_3\right) \tag{4.18}$$

where Σ_e denotes the macroscopic effective stress, Σ_h represents the hydrostatic stress, and ρ_2 and ρ_3 are ratios that relate stress state in different loading directions.

Eq. (4.14)-(4.18) lead to a quadratic set of equations to be solved component-wise. The results are taken to be compatible with the following formulation:

$$\Sigma_1 \geq \Sigma_2 \geq \Sigma_3 \tag{4.19}$$

Until this point, all the ingredients needed for conducting a Unit Cell with Periodic Boundaries Finite Element computation have been detailed.

4.2.5 Extracting values from a Unit Cell problem

However one last step is needed in order to be able to extract meaningful results from the Finite element simulations: the outputs to be computed along the simulation, per time step.

The quantities to compute and compare with the homogenization approach are:

- Equivalent Creep Strain, E_e.
- Micro-structure evolution, $\frac{f}{f_0}$.
- Equivalent Strain rate, D_e.

In order to obtain them, and giving the fact that the unit cell is divided in many 3D mesh elements, the following quantities need to be extracted per time step:

- Logarithmic strain component in the centroid of the mesh element, ϵ_{ij}.
- Volume of each element.
- Strain rate component per mesh element, D_{ij}.
- Stress components per mesh element, σ_{ij}.

Added to this, and recalling how the true or Cauchy stress state is imposed, it is also required to track the displacements, U_i of the defined reference points, RF_i.

Once all these quantities are computed, they need to be extracted from the result file. This is achieved in a automatized way thanks to the scripting capability of ABAQUS. Further details of the output-request Python script can be found in Appendix C.

Finally, a post-processing of the extracted values is conducted using MATLAB software. Equivalent Creep strains, porosity evolution and Equivalent Strain rate are computed thanks to the Matlab script in Appendix D.

5 Results

In this chapter, a full overview of the creep response of a porous FCC single crystal is presented in terms of finite-strains and micro-structure evolution.

A total of 28 cases will be addressed, having five main topics to analyze: effect of loading configuration, effect of stress triaxiality, effect of initial porosity level and initial void shape and effect of the creep exponent.

The results will serve for a double-purpose: Finite Element Unit Cell computations will serve to assess the accuracy and efficiency of the proposed homogenization approach (Song and Ponte Castañeda, 2017b) for the initial analysis; and it will serve to completely understand the effect of crystal orientation and loading configuration in the creep life of single crystal turbine blades.

5.1 An overview of creep behaviour - Preliminary Considerations

Creep may be defined as a time-dependent material deformation under a constant stress state well below the material's yield strength.

As introduced in ***Chapter 4 - Numerical Approach***, this work considers a nonlinear porous face center cubic (FCC) single-crystal which contains a spherical or ellipsoidal void located in the center of the unit cell. The initial porosity level, f will be set to 1% unless it is specified the opposite. Additionally, self and latent hardening of the single crystal matrix is not considered in this work, so that $\tau_0 = 245$MPa along deformation. The reference shear strain, $\dot{\gamma_0}$ is fixed to $1.53 \cdot 10^{-9} s^{-1}$ and the creep exponent, n to 5, unless other value is specified.

Regarding loading conditions, a true stress or Cauchy stress state is imposed. This fact ensures a loading history with a constant stress triaxiality, χ (Srivastava and Needleman, 2012)(Song and Ponte Castañeda, 2017a). Specifically, Σ_e is set to $750MPa$. Thus, for a chosen Lode parameter value and Stress triaxiality, the tractions, Σ_i to be applied to the Unit Cell problem are calculated by solving a set of quadratic

equations already presented in previous chapter.

At this point it is important to recall the nature of the two methods to be compared. The Finite Element results accounts for Unit Cell computations of a porous FCC single crystal with a periodic distribution of voids, while the homogenization approach of Ponte pertains for non-linear porous FCC single crystal with random micro-structures. As a consequence, care has to be put in place while analyzing the results as it will be compared two different micro-structure distributions: periodic vs random.

Another fact to keep in mind, the Finite Element computations accounts for a elasto visco-plastic constitutive behaviour of the single crystal matrix while the ISO approach is solely capturing the visco-plastic response of a non-linear porous single crystal. Nevertheless, as commented previously, the effect of elasticity in the creep problem has been shown to be of few order of magnitude smaller than the visco-plastic response (Srivastava and Needleman, 2012), and considered negligible.

Taking into account the aforementioned notes, the finite-strain response of the porous FCC single crystal will be drawn by means of:

- Equivalent Creep Strain, E_e.
- Equivalent Strain Rate, D_e.
- Porosity evolution, $\frac{f}{f_0}$.
- Stress contour, $\frac{\sigma_{11}}{\Sigma_e}$ around the void.

The Equivalent Creep strain, E_e is defined as:

$$E_e = \frac{\sqrt{2}}{3}\sqrt{\left(\varepsilon_{11}-\varepsilon_{22}\right)^2+\left(\varepsilon_{22}-\varepsilon_{33}\right)^2+\left(\varepsilon_{11}-\varepsilon_{33}\right)^2+\frac{3}{2}\left(\varepsilon_{12}^2+\varepsilon_{23}^2+\varepsilon_{13}^2\right)} \tag{5.1}$$

In Eq. (5.1), ε_{ij}, are the volume average of the current values of the logarithmic strain components computed at the centroid of each mesh element. The volume average is computed as follows:

$$\boxed{\varepsilon_{ij}^t = \frac{\sum_i v^i \varepsilon_{ij}^i}{\sum_i v^i}} \tag{5.2}$$

where v^i represents the current volume of the mesh element. For comparison purposes, the Equivalent Creep strain, E_e is redefined to account only for the visco-plastic behaviour of the porous single crystal in the following form:

$$E_e = E_e - E_e^{elastic} \tag{5.3}$$

In Eq. (5.3), $E_e^{elastic}$ corresponds to the elastic part of the creep strain, considered to be achieved in the initial time steps, depending on the case of study.

In the same fashion, the Equivalent strain rate, D_e is computed as:

$$D_e = \frac{\sqrt{2}}{3}\sqrt{(D_{11}-D_{22})^2 + (D_{22}-D_{33})^2 + (D_{11}-D_{33})^2 + \frac{3}{2}\left(D_{12}^2 + D_{23}^2 + D_{13}^2\right)} \tag{5.4}$$

where D_{ij} are volume average of current values of the mechanical strain rate components computed at each mesh element.

Furthermore, the micro-structure evolution will be characterize in different ways. First, the porosity evolution, $\frac{f}{f_0}$ will be computed as follows:

$$\boxed{f = \frac{V_{cell} - V_{Matrix}}{V_{cell}}} \tag{5.5}$$

where V_{cell} is the current volume of the Unit Cell and V_{Matrix} is the current matrix volume, i.e. the sum of all the current elemental volumes.

In the homogenization (ISO) approach, the aspect ratio is defined as the minimum aspect ratio. This value will be used only in the first analysis.

$$\boxed{w = \frac{a_{min}}{a_{max}}} \tag{5.6}$$

In a last instance, the contour plots of the stress concentrations, $\frac{\sigma_{11}}{\Sigma_e}$ will be shown around the void for the different cases simulations.

Moreover, it is important to introduce here the concept of "onset on a rapid increase in creep strain" or "tertiary creep" as the phenomena where the equivalent creep strain rate, $\dot{E}_e$, i.e. slope of the creep strain curve versus time, increases rapidly with time. This fact will be observed in some of the loading cases simulated.

5.2 Mesh Convergence Study

A necessary step while constructing a finite element model is to asses mesh convergence. Having a converged mesh is of crucial importance to obtain meaningful and accurate numerical results.

In the case of the Unit Cell, as the main concern for this study entitles creep behaviour and micro-structure evolution, three output variables will be used to asses the convergence of the solution:

- Equivalent Creep Strain, E_e versus time, t.
- Porosity Evolution, $\frac{f}{f_0}$ versus Equivalent Creep Strain, E_e.
- Equivalent Strain rate, D_e versus Equivalent Creep Strain, E_e.

Additionally, due to the nature of the problem and the variety of loading cases to be simulated, two mesh convergence studies will be conducted: low stress triaxiliaty case ($\chi = \frac{1}{3}$) and high stress triaxiality case ($\chi = 3$).

Before starting with the analysis, some parameters need to be fixed along the simulation. Table 5.2 summarizes the physical parameters used for the aforementioned convergence studies.

Case	Crystal orientation	χ	L	f	n
A	100	$\frac{1}{3}$	-1	1%	5
B	110O1	3	-1	1%	5

Table 5.1: Physical parameters for mesh convergence studies.

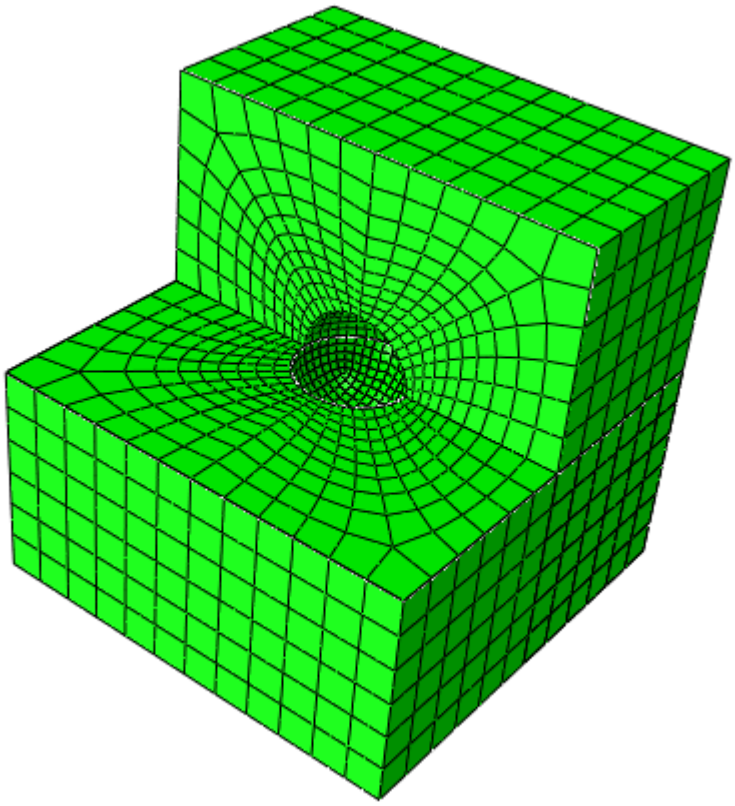


Figure 5.1: Cut view of the mesh for the Unit Cell problem ($f = 1\%$).

A finite element mesh composed by 20-node hybrid solid elements with quadratic displacement interpolation and linear pressure interpolation mesh elements, referenced as C3D20H elements in *ABAQUS* software will be used for this work.

The C3D20H elements have demonstrated in numerous studies the ability to capture accurately the physics behind the Unit Cell problem (Srivastava and Needleman, 2015)(Srivastava and Needleman, 2012). An example of a converged mesh can be observed in Figure 5.1.

As one of the objective of the Unit Cell problem is to capture the micro-structure evolution, i.e. the evolution of the void and its nearest region, it will be shown how an adequate mesh refinement has to be applied in the inclusion area. A detailed view of the void phase ares is shown in Figure 5.2. Indeed, the number of mesh elements in this region and the nearest adjacent area will be the variable to change during the mesh convergence study.

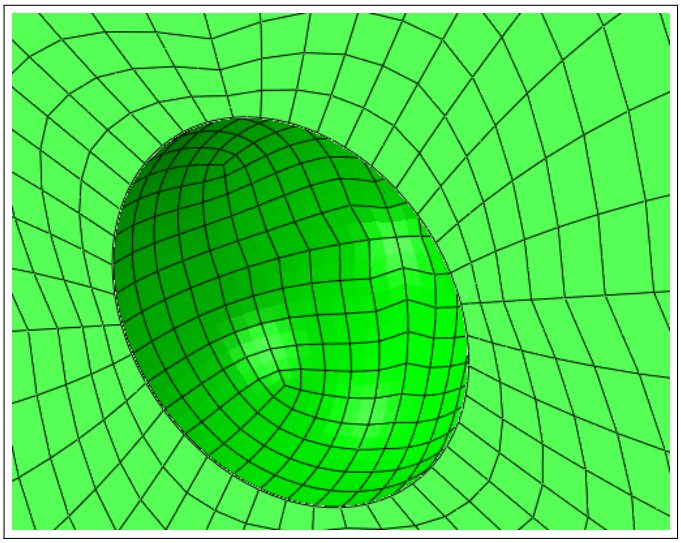


Figure 5.2: Zoom view of the mesh refinement near the void.

At the end of the analysis, conclusions in terms of number of elements around the void and in the ligament necessary for convergence will be drawn.

Table 5.2 gathers the 3 configurations computed to asses convergence.

Mesh	Aspect ratio	n_{void}	n_{lig}	N_{elem}
Coarse	0.2	20	5	1968
Intermediate	0.15	32	8	4208
Fine	0.15	40	10	6128

Table 5.2: Different mesh configurations.

In the previous table, n_{void} denotes the number of elements in the void circumference and n_{lig} represents the number of mesh elements in the ligament formed between the void and the cubic matrix.

Additionally, for validation purposes, the results in terms of creep strain and porosity evolution from (Srivastava and Needleman, 2015) will be included, denoted as red triangles. A perfect agreement will be clearly observed.

Stress Triaxiality, $\chi = \frac{1}{3}$

In Figures 5.3 and 5.4 it can be seen how for the low stress triaxiality case, there are no visible differences in terms of Equivalent creep strain, nor porosity evolution.

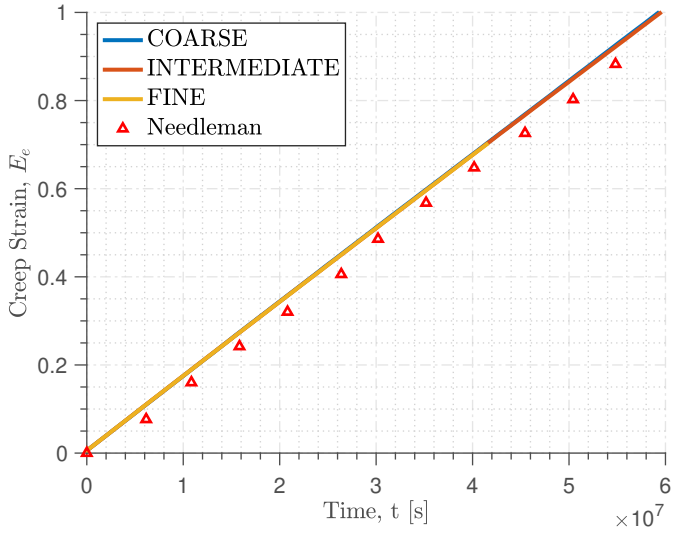


Figure 5.3: Equivalent creep strain, E_e over time for different mesh configurations with crystal orientation 100, $L = -1$ and $\chi = \frac{1}{3}$.

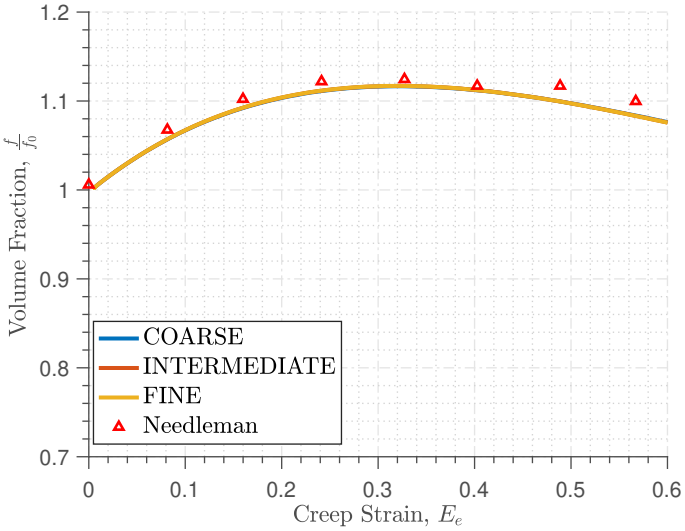


Figure 5.4: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different mesh configurations with crystal orientation 100, $L = -1$ and $\chi = \frac{1}{3}$.

However, if we look deeper into the previous curves, it can be observed how the Intermediate and the Fine mesh have reached convergence. This fact is depicted in Figures 5.5 and 5.6.

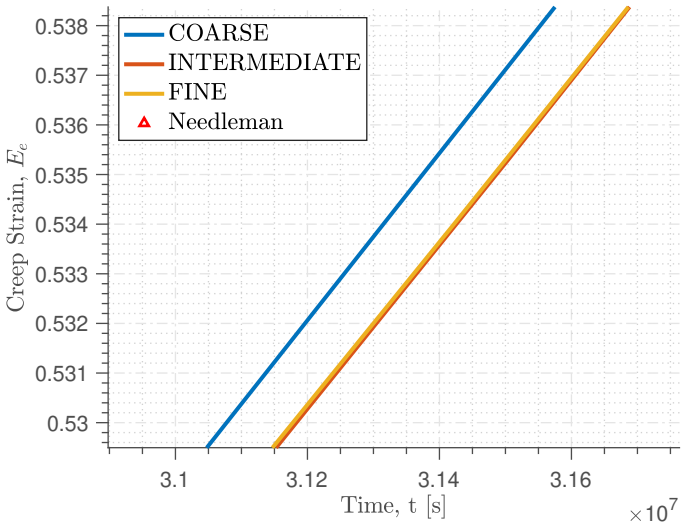


Figure 5.5: Zoom into the Equivalent creep strain, E_e time evolution for different mesh configurations with crystal orientation 100, $L = -1$ and $\chi = \frac{1}{3}$.

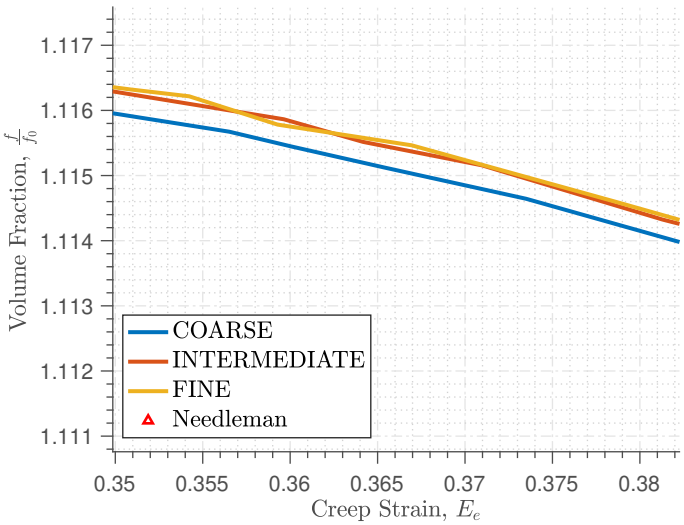


Figure 5.6: Zoom into the porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different mesh configurations with crystal orientation 100, $L = -1$ and $\chi = \frac{1}{3}$.

If now the attention is focused in the Equivalent strain rate, D_e, it can be clearly

seen how the Coarse mesh is not capturing accurately the softening behaviour (slope of the strain rate) due to the micro-structure evolution for the low stress triaxiality case. Again, the same conclusion can be drawn, both Intermediate and Fine mesh reached convergence.

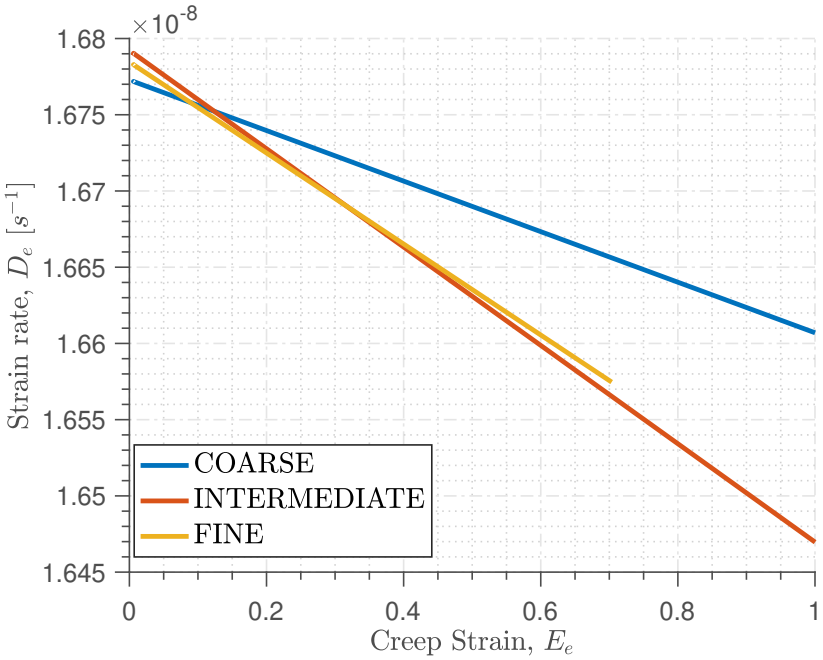


Figure 5.7: Equivalent Strain rate, D_e versus Equivalent creep strain, E_e for different mesh configurations with crystal orientation 100, $L = -1$ and $\chi = \frac{1}{3}$.

Stress Triaxiality, $\chi = 3$

The second case to look into for assessing mesh convergence is a case known for strong micro-structure evolution. It is known that for the high stress triaxiality case with Lode parameter, $L = -1$, the porosity will increase significantly.

The same procedure is followed in this section. Figures 5.8 and 5.9 show the evolution of the Equivalent Creep strain, E_e and the porosity, $\frac{f}{f_0}$. It can be observed how a further detail view is needed to analyse the difference for the different mesh cases.

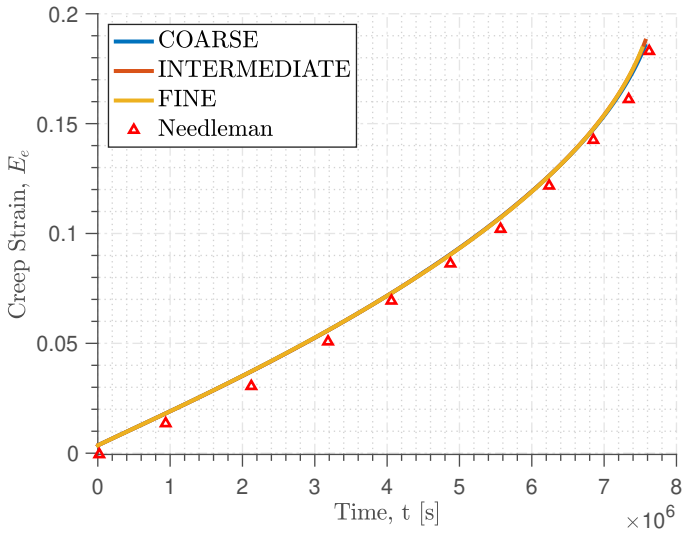


Figure 5.8: Equivalent creep strain, E_e over time for different mesh configurations with crystal orientation 110O1, $L = -1$ and $\chi = 3$.

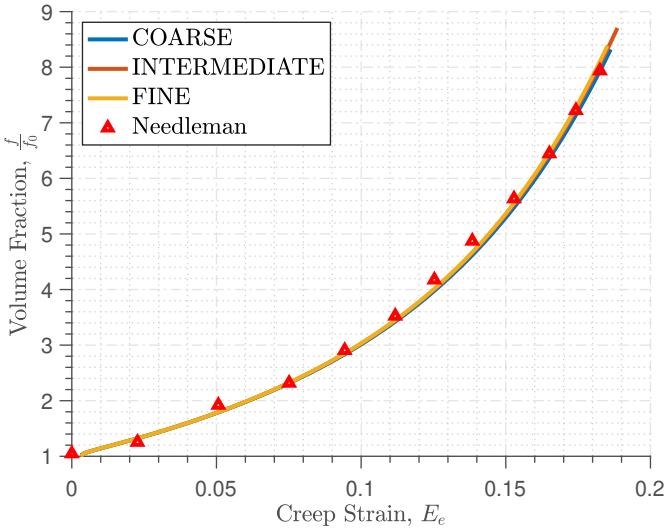


Figure 5.9: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different mesh configurations with crystal orientation 110O1, $L = -1$ and $\chi = 3$.

Same conclusions can be achieved after looking into Figures 5.10, 5.11 and 5.12. Both Intermediate and Fine mesh cases show convergence.

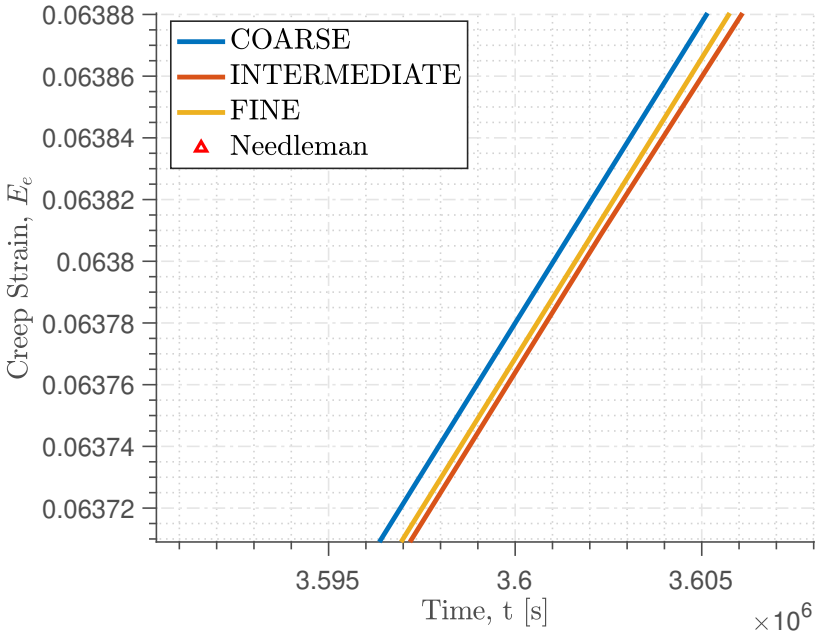


Figure 5.10: Zoom into the Equivalent creep strain, E_e time evolution for different mesh configurations with crystal orientation 110O1, $L = -1$ and $\chi = 3$.

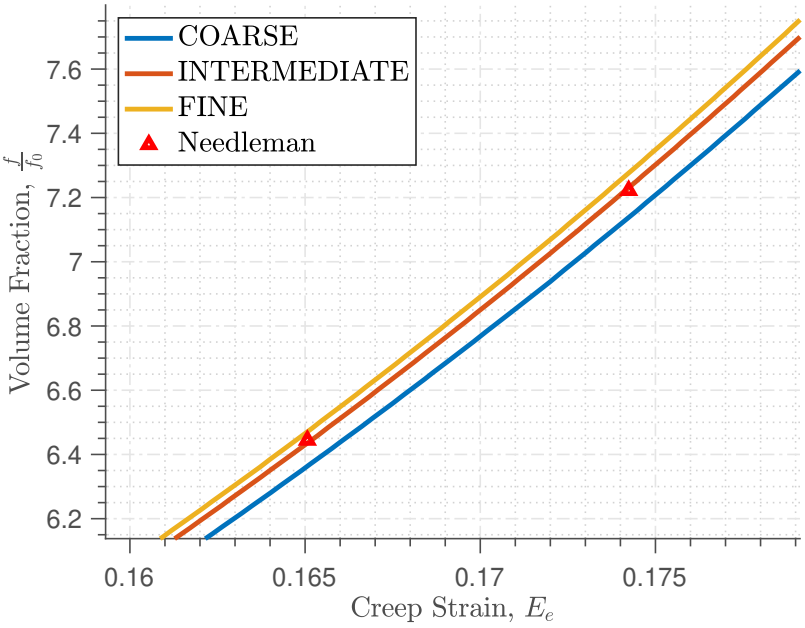


Figure 5.11: Zoom into the porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different mesh configurations with crystal orientation 110O1, $L = -1$ and $\chi = 3$.

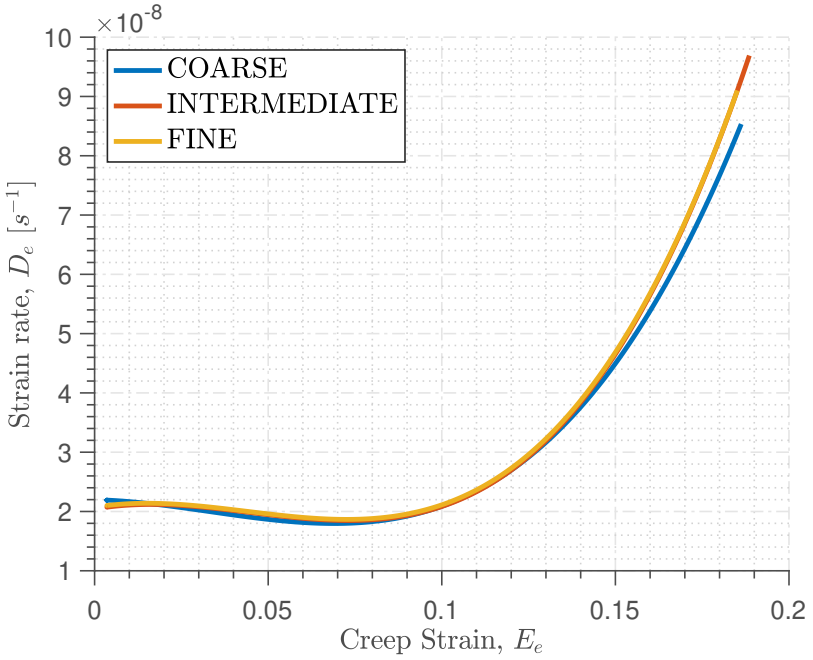


Figure 5.12: Equivalent Strain rate, D_e versus Equivalent creep strain, E_e for different mesh configurations with crystal orientation 110O1, $L = -1$ and $\chi = 3$.

The analysis of the strain rates evolution in Figure 5.12 throws the same conclusions: intermediate and fine mesh are showing converged results.

In summary, it has been observed how both Intermediate with 4208 mesh elements and Fine mesh cases with 6128 mesh elements ensure convergent results for the simulated cases and under the physical parameters specified in Table 5.2. However, a higher number of elements in the vicinity of the void is desired for characterizing the strain and stress concentration around the void, keeping an eye in the CPU time.

Given this fact, and taking into account that the CPU time associated to both Intermediate and Fine mesh differentiates in few hours (Intermediate Mesh $\sim 14h$ and Fine Mesh $\sim 18h$; it can be concluded that ***the Fine Mesh with a global aspect ratio of 0.15 with 40 elements distributed along the void perimeter and 10 elements in the ligament of the unit cell*** is necessary to capture accurately the creep behaviour and the micro-structure evolution a porous single crystal for both low and high stress triaxiality cases.

5.3 FCC Single Crystal - Finite strains and micro-structure evolution

This section gathers the parametric analysis for the FCC single crystal nickel-based super-alloy of concern.

For all the below cases, a first comment in the creep strain-time curve will be addressed. Secondly, the micro-structure evolution will be analyzed from different points of view, i.e. porosity evolution, stress concentration contour plots with void shape. Additionally, a look into the strain rate evolution will be linked with all the previous conclusions.

In terms of notation, AXT refers to Axisymmetric Tension, AXC to Axisymmetric Compression and PS denotes Pure Shear.

5.3.1 Effect of loading configuration

The first parametric analysis to conduct is related to the initial crystal orientation and Lode parameter, i.e. loading configuration. Two initial crystal orientations will be addressed: the symmetric [100] and the anisotropic [110O1] orientation, for three Lode parameter values. Table 5.3 summarises all the cases to consider.

The reader is referred to Figure 4.1 for recalling purposes in identifying the crystal orientation to be used during this parametric analysis.

At this point it is important to recall that a Lode parameter of $L = \pm 1$ reflects physically an axisymmetric loading, in compression and in tension, respectively. For the case of $L = 0$, the physical loading refers to a pure shear loading case.

Loading	χ	L	Σ_1 [MPa]	Σ_2 [MPa]	Σ_3 [MPa]
AXT [100]	1/3	-1	750	0	0
AXC [100]	1/3	+1	500	500	-250
PS [100]	1/3	0	683.01	250	-183.01
AXT [110]	1/3	-1	750	0	0
AXC [110]	1/3	+1	500	500	-250
PS [110]	1/3	0	683.01	250	-183.01

Table 5.3: FCC Loading Configuration analysis for 100 & 110O1 crystal orientations.

Figures 5.13, 5.14 and 5.15 show clearly the important effect of the loading configuration on the macroscopic response of the FCC single crystal as well as on the micro-structure evolution for fixed stress triaxiality value, $\chi = \frac{1}{3}$.

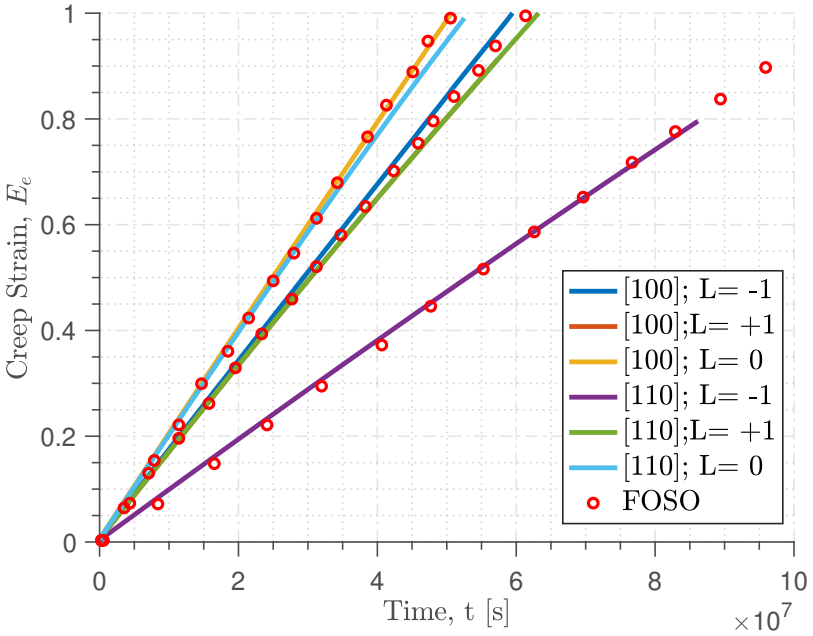


Figure 5.13: Time evolution of the Equivalent creep strain, E_e for different Lode parameters, stress triaxiality $\chi = \frac{1}{3}$ and two crystal orientations: 100 & 110O1. The same results have been included by using the FOSO homogenisation approach (red circles).

From Figure 5.13 it is important to note that some loading cases overlap between each other, as result of crystal symmetry and slip phenomena. For instance, the red line for the crystal orientation 100 and $L = +1$ is overlapping with the green one, corresponding to 110O1 orientation and $L = -1$. For the FOSO homogenization results, the same can be said, only three main lines can be observed, but six cases are represented.

In the time histories of the equivalent creep strain, E_e for both crystal orientations, it can be seen how E_e increases monotonically within time at a constant rate. It is important to note that the calculations were terminated at $E_e = 1$ or when mesh distortion starts to affect the results (Srivastava and Needleman, 2015). As it can be seen in Figure 5.13 for the symmetric crystal orientation [100], the largest steady state creep is achieved for $L = 0$, being slightly greater than for $L = \pm 1$. The same can be said for the 110O1 crystal orientation. Perfect agreement can be observed in

terms of creep strain evolution with the homogenization approach for all the loading configuration cases.

The evolution of the porosity strongly depends on the Lode parameter, as it is depicted in Figures 5.14 and 5.15.

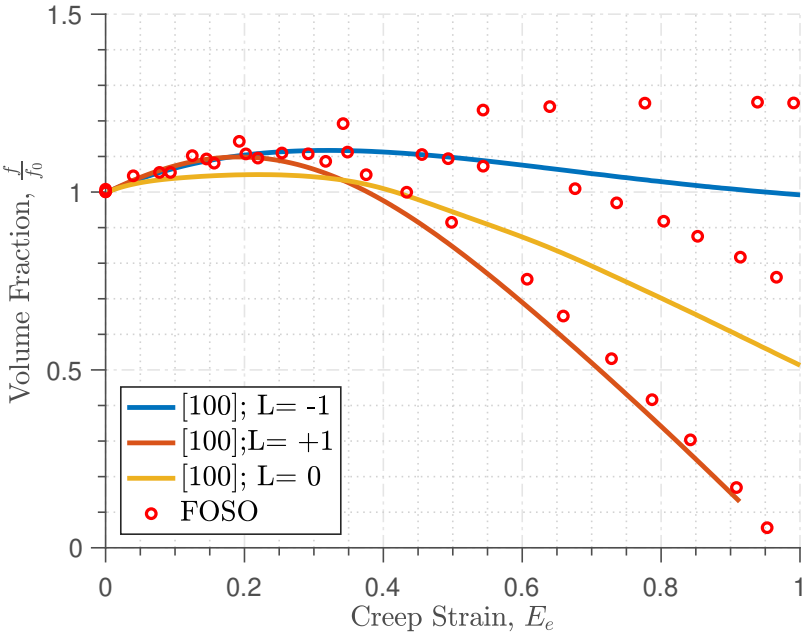


Figure 5.14: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different Lode parameter, stress triaxiality $\chi = \frac{1}{3}$ and crystal orientations 100. The same results have been included by using the FOSO homogenisation approach (red circles).

The porosity evolution, $\frac{f}{f_0}$ with E_e for the [100] crystal orientation for the three Lode parameters timidly increases showing a peak ($\frac{f}{f_0} > 1$). Specifically, for $L = -1$, the pore size saturates to the initial porosity level, while for the $L = 0$ and $L = +1$, the porosity decreases resulting in the collapse of the void.

On the other hand, for the 11O01 crystal orientation and $L = 0$ and $L = -1$ the porosity level remains almost constant to the initial porosity level. However, the case of $L = +1$ shows a initial peak in the pore growth to a further monotonic decrease, i.e. void collapse. Here it can be seen the differences of having a periodic or a random micro-structure distribution. The differences between both methodologies rise in terms of micro-structure evolution when the initial crystal orientation is more anisotropic, i.e. 110O1 orientation.

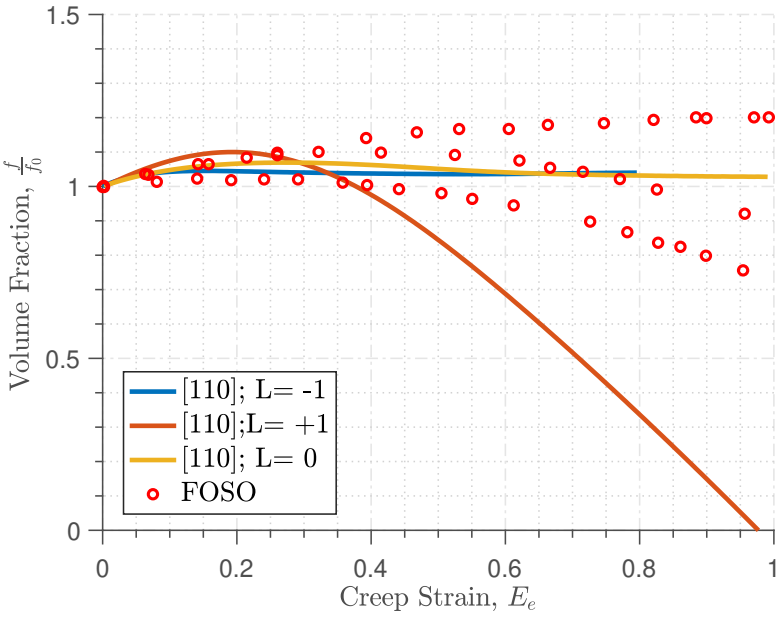


Figure 5.15: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different Lode parameter, stress triaxiality $\chi = \frac{1}{3}$ and crystal orientations 110O1. The same results have been included by using the FOSO homogenisation approach (red circles).

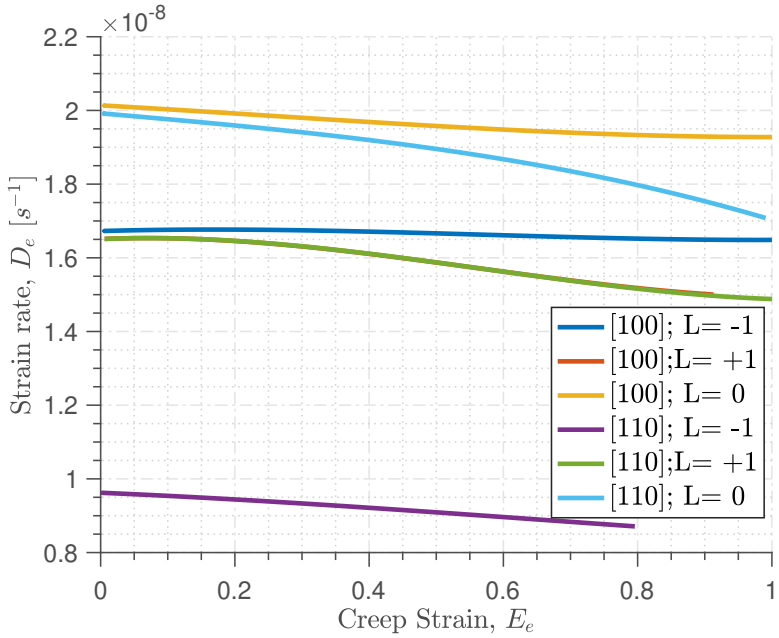


Figure 5.16: Equivalent strain rate, D_e vs Equivalent creep strain, E_e for different Lode parameters, stress triaxiality $\chi = \frac{1}{3}$ and two crystal orientations: 100 & 110O1.

If now the attention is focused in the evolution of the equivalent strain rate, D_e, it can be concluded how the loading configuration indeed is playing an important role in the creep life of the FCC single crystal.

A higher strain rate is related to an earlier tertiary creep and indeed creep failure. It can be seen how the most detrimental loading cases corresponds to the ones with the Lode parameter $L = 0$, as it has been previously exposed in Figure 5.13. Additionally, the case with the highest creep enhancement is the one corresponding to the 110O1 crystal orientation with $L = -1$, where the porosity remains constant to its initial value, as it was observed in Figure 5.15.

In Figures 5.17 and 5.18 it can be seen the stress concentration contours for the unit cell for the range of loading configurations. The micro-structure evolution in terms of void shape is highly influenced by the crystal orientation and the Lode parameter.

For the 100 crystal orientation and $L = -1$, it can be seen how the void evolves into a stretched spheroidal shape with a circular cross section. This evolution is in agreement with the results observed in Figure 5.14 and 5.15 where the pore is slightly growing, just stretching, keeping its initial porosity level. The stress concentrations are located in the surrounding of the void where the local radius is smaller.

In the case of axisymmetric compression ($L = +1$) for 100 orientation, the void evolves into a penny-shape leading to collapse. On the other hand, for the $L = 0$ case with the symmetric crystal orientation, the void evolves into an ellipsoidal shape. The void collapse will also occur, into a ellipsoidal crushed shape but for higher associated creep strains than the previous case. A compressing stress concentration appears in the perimeter of the void.

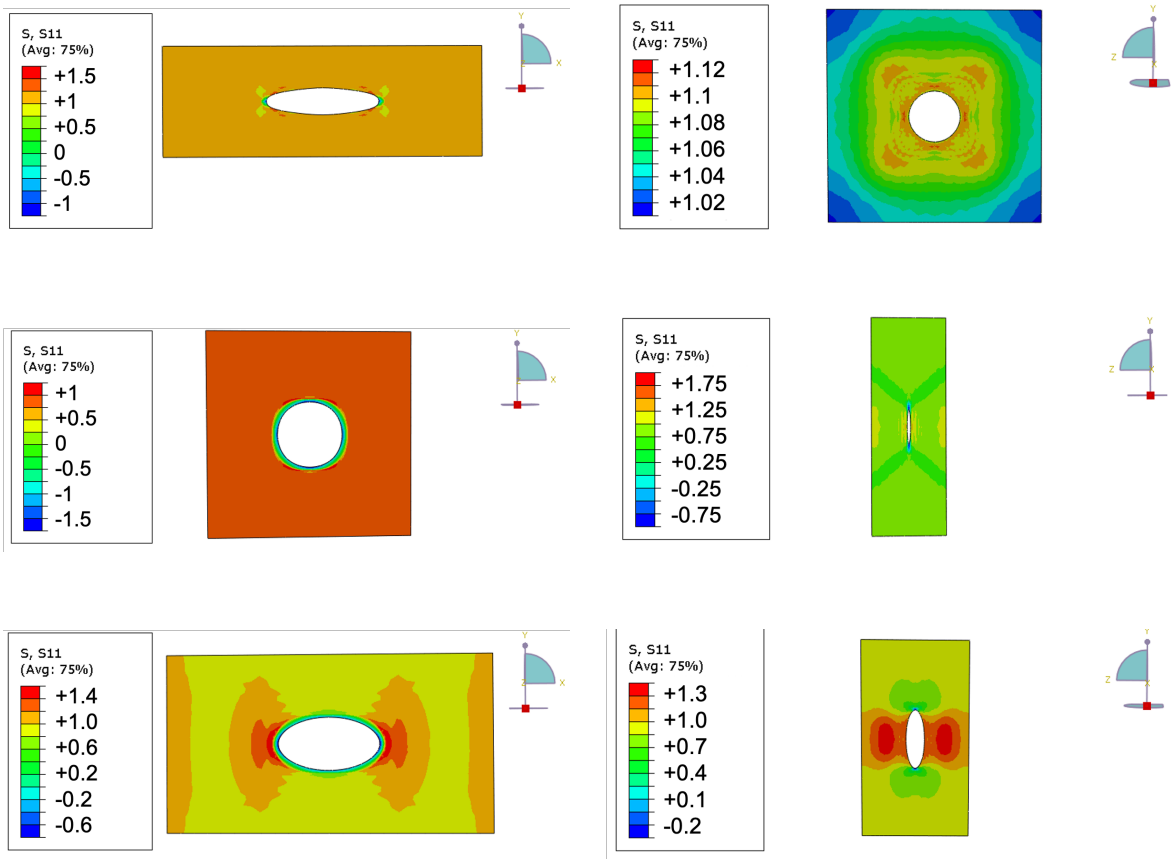


Figure 5.17: Cross sections of the stress, $\frac{\sigma_{11}}{\Sigma_e}$ for $\chi = \frac{1}{3}$ and $E_e \sim 0.7$. From top to bottom: 100 orientation $L = -1$; 100 orientation $L = +1$; 100 orientation $L = 0$.

Finally, the 110O1 crystal orientation shows two different behaviours in terms of pore shape evolution: for $L = +1$, in compression, the pore evolves into the aforementioned "penny-shape" leading to collapse; and for $L = 0$; $L = -1$, the void is stretched into an ellipsoidal-like void having almost identical cross section for the same equivalent creep strain. This fact has been observed previously in Figure 5.15.

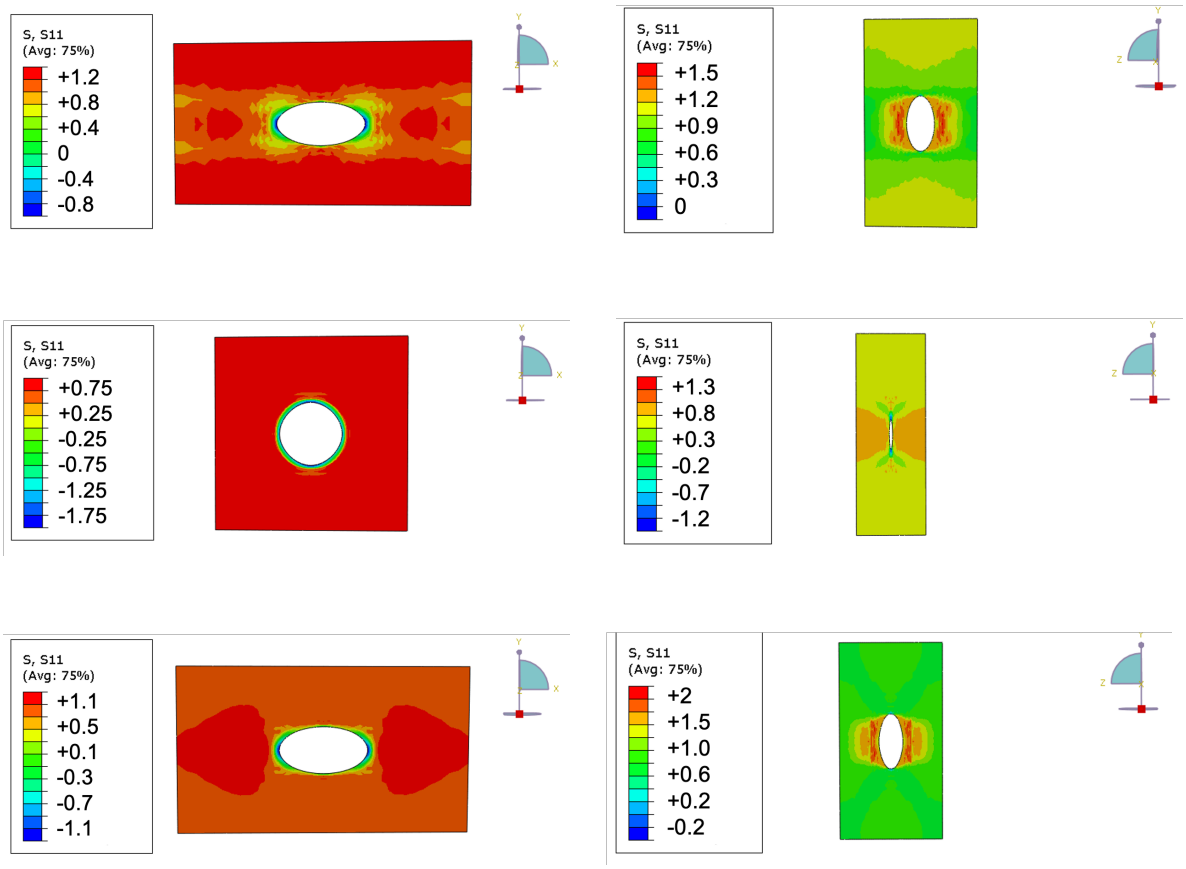


Figure 5.18: Cross sections of the stress, $\frac{\sigma_{11}}{\Sigma_e}$ for $\chi = \frac{1}{3}$ and $E_e \sim 0.7$. From top to bottom: 110O1 orientation $L = -1$; 110O1 orientation $L = +1$; 110O1 orientation $L = 0$.

A last comparison can be plotted for the ISO homogenization approach, the void aspect aspect ratio evolution, in Figure 5.19.

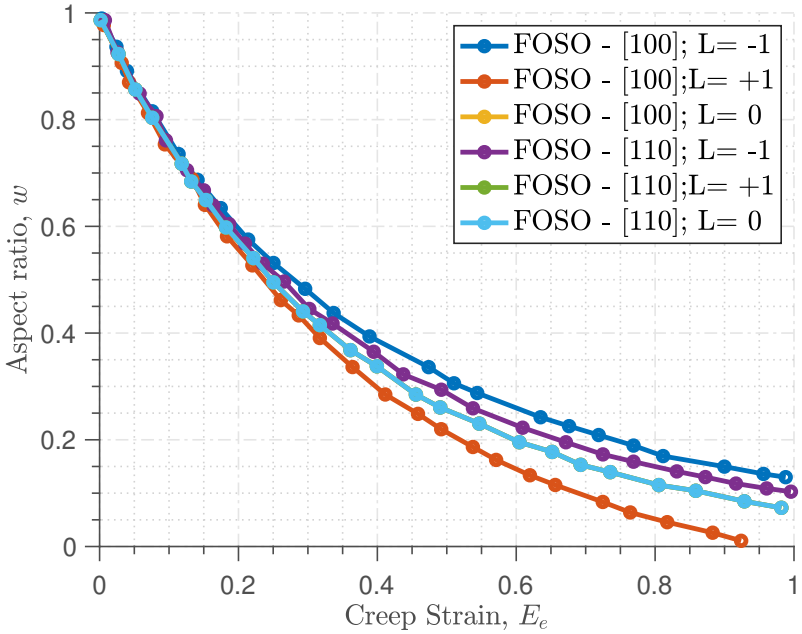


Figure 5.19: Aspect ratio, w vs Equivalent Creep strain, E_e for different Lode parameters, stress triaxiality $\chi = \frac{1}{3}$ and two crystal orientations: 100 & 110O1 (Song and Ponte Castañeda, 2017a).

In Figure 5.19, the cases corresponding to: 100 orientation and $L = 0$, 110O1 orientation and $L = +1$; $L = 0$ (yellow, green and light blue curves) are overlapping and the representation reduces to a single curve, denoted in a light blue color.

It is important to remember that the exposed aspect ratio, w, related the minimum aspect ratio of the void. Comparing with Figures 5.17 and 5.18, it can be seen how the aspect ratio decreases for all the cases, being more accused for the cases leading to void collapse, such as 100 and 110O1 crystal orientations and $L = +1$. No further information can be extracted from the previous graph as the minimum aspect ratio values are not available from the Finite Element computations.

5.3.2 Effect of stress triaxiality

The loading configuration has been shown to be of crucial importance on creep behaviour in a porous FCC single crystal. Another important parameter to account for is the stress triaxiality, which is related to the loading level.

Two loading configurations will be considered in this analysis: 110O1 crystal orientation with Lode parameters $L = -1$ and $L = +1$. Six different stress triaxiality levels will be computed for each loading case. A summary of the computed cases can be found in Tables 5.4 and 5.5.

Loading	χ	L	Σ_1 [MPa]	Σ_2 [MPa]	Σ_3 [MPa]
AXT [110]	1/3	-1	750	0	0
AXT [110]	2/3	-1	1000	250	250
AXT [110]	3	-1	2750	2000	2000
AXT [110]	-1/3	-1	-750	0	0
AXT [110]	-2/3	-1	-1000	-250	-250
AXT [110]	-3	-1	-2750	-2000	-2000

Table 5.4: FCC Stress Triaxiality Analysis (Axisymmetric Tension)

Loading	χ	L	Σ_1 [MPa]	Σ_2 [MPa]	Σ_3 [MPa]
AXC [110]	1/3	+1	500	500	-250
AXC [110]	2/3	+1	750	750	0
AXC [110]	3	+1	2500	2500	1750
AXC [110]	-1/3	+1	-500	-500	250
AXC [110]	-2/3	+1	-750	-750	0
AXC [110]	-3	+1	-2500	-2500	-1750

Table 5.5: FCC Stress Triaxiality Analysis (Axisymmetric Compression)

Starting with the creep strain evolution for the 110O1 orientation in axisymmetric tension ($L = -1$) in Figure 5.20, it can be concluded that the creep strain evolution is nearly independent of the stress triaxiality level from negative values up to $\chi = \frac{2}{3}$. For the highest stress triaxiality $\chi = 3$, it can be seen how there is an "onset on the creep strain evolution" (rapid increase of the strain in short time), which will lead to creep failure.

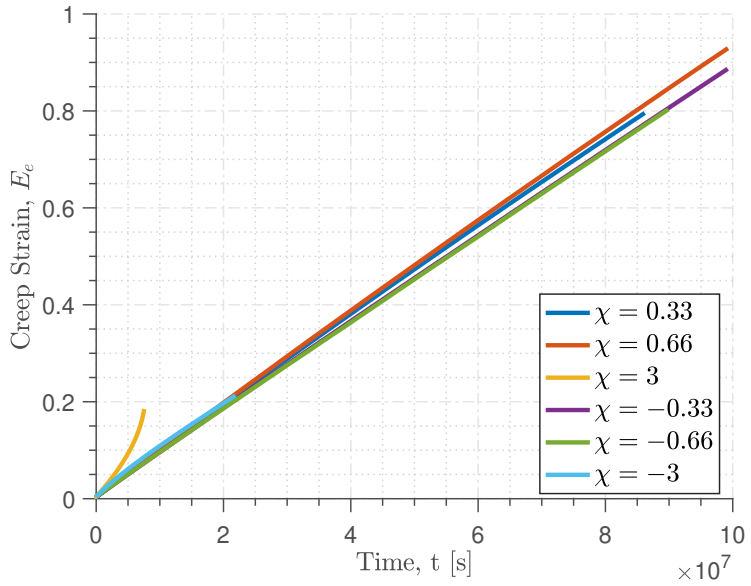


Figure 5.20: Time evolution of the Equivalent creep strain, E_e for different stress triaxiality values, Lode Parameter $L = -1$ and crystal orientation 110O1.

Looking now to Figure 5.21, it is clearly observed the strong influence of the stress triaxiality level on the micro-structure evolution of the single crystal.

For negative values of the stress triaxiality, the porosity level decreases monotonically leading to void collapse. The decrease is more abrupt as the stress triaxiality value is smaller, leading to void collapse with an earlier associated equivalent creep strain, E_e.

In turn, for $\chi = \frac{1}{3}$ the porosity slightly evolves, remaining constant to its initial value. Although, for $\chi = \frac{2}{3}$, the pore size increases until reaching a porosity level of $\frac{f}{f_0} = 1.5$. The highest stress triaxiality case, $\chi = 3$ causes an abrupt increase of the pore size for relatively low associated creep strains. This fact explains the appearance of the "onset on the creep evolution" around $t = 8 \cdot 10^6 s$ observed in Figure 5.20.

Figure 5.22 allows once again to confirm the previous conclusions in terms of creep evolution. For values of stress triaxiality, $-\frac{2}{3} < \chi < \frac{2}{3}$ the strain rate keeps an equally constant value along deformation. For the most negative stress triaxiality value, it can be seen how the strain rate decreases abruptly due to the accentuated decrease in porosity level leading to void collapse, for a very low associate creep strain. Moreover, for the highest value of stress triaxiality, the strain rate experiences an abrupt increase, having a marked softening effect in the material creep life.

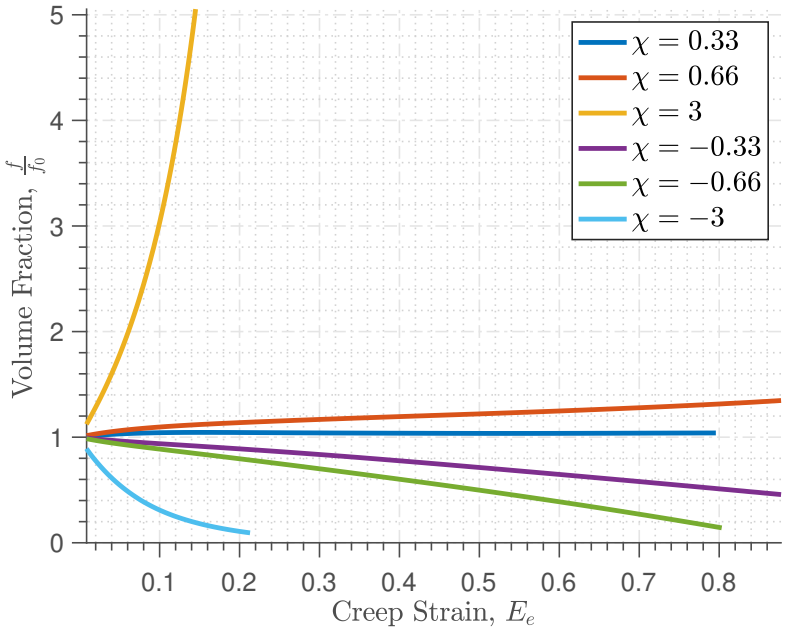


Figure 5.21: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different stress triaxiality values, Lode Parameter $L = -1$ and crystal orientation 110O1.

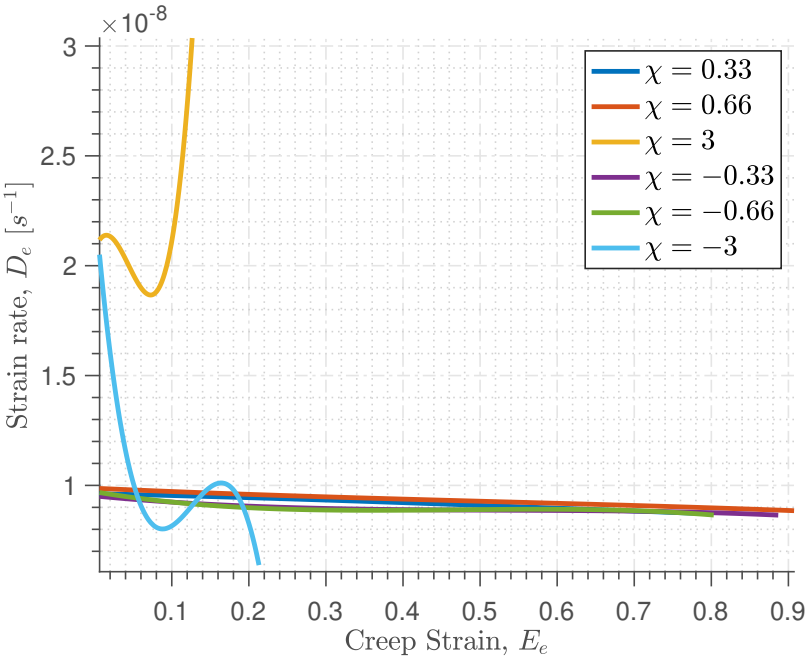


Figure 5.22: Equivalent strain rate, D_e vs Equivalent creep strain, E_e for different stress triaxiality values, Lode Parameter $L = -1$ and crystal orientation 110O1.

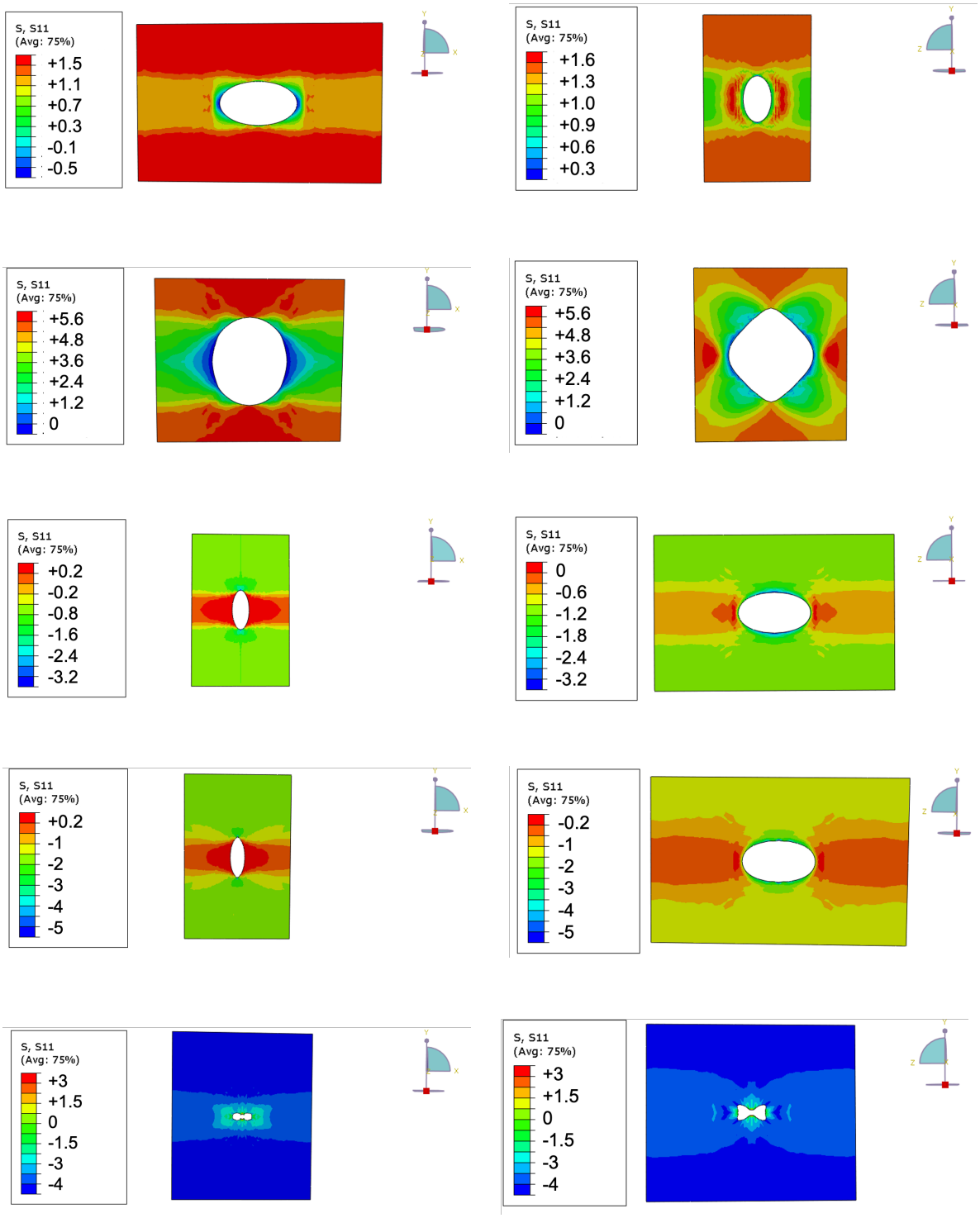


Figure 5.23: Cross sections of the stress, $\frac{\sigma_{11}}{\Sigma_e}$ for 110O1 crystal orientation, $L = -1$. From top to bottom: $\chi = \frac{2}{3}$ and $E_e = 0.5$; $\chi = 3$ and $E_e = 0.15$; $\chi = -\frac{1}{3}$ and $E_e = 0.5$; $\chi = -\frac{2}{3}$ and $E_e = 0.5$; $\chi = -3$ and $E_e = 0.2$.

The stress triaxiality, χ has a huge influence in the void shape evolution as depicted in Figure 5.23.

For the highest value of stress triaxiality, $\chi = 3$, the void growth is abrupt, adopting a final characteristic polygonal shape with rounded corners.

For negative stress triaxiality values up to $\chi = -\frac{2}{3}$, the void shape is evolving into a ellipsoidal shape, as described previously but with the characteristic cross-sections shifted.

For the most negative value, $\chi = -3$, the pore evolves into a "nutshell-like" shape. For relative low creep strains, the void tends to split in four individual voids. Along deformation, the pore size is decreasing too.

Additionally, the stress concentration increases as the stress triaxiality does.

Focusing now in the case where the Lode parameters is fixed to $L = +1$, similar phenomena can be observed. The history of the creep strain in Figure 5.24 is fairly independent of the stress triaxiality for $\chi < \frac{2}{3}$, while for $\chi = 3$, the onset on the creep strain appears again. In this case the onset happens around $t = 5 \cdot 10^6 s$.

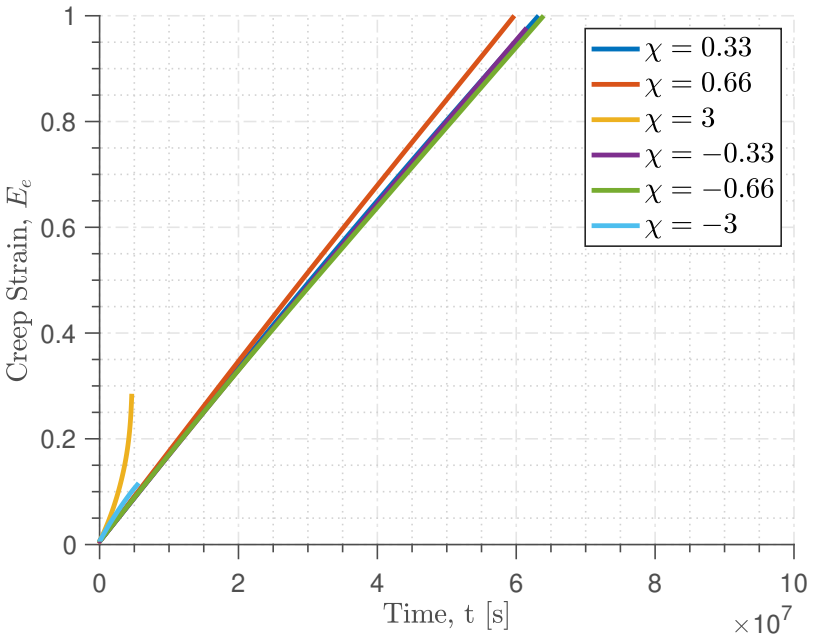


Figure 5.24: Time evolution of the Equivalent creep strain, E_e for different stress triaxiality values, Lode Parameter $L = +1$ and crystal orientation 110O1.

For the micro-structure evolution in Figure 5.25, three effects can be observed. First, for $\chi < \frac{1}{3}$ the porosity evolves with different decreasing rates, higher as the stress triaxiality is reduced, leading to void collapse. For $\chi = \frac{2}{3}$, the micro-structure evolution is characterized for a pore growth at different rates along deformation, to reach a value of $\frac{f}{f_0} = 2$ for $E_e = 1$. The last behaviour is the aforementioned "abrupt pore growth", for $\chi = 3$.

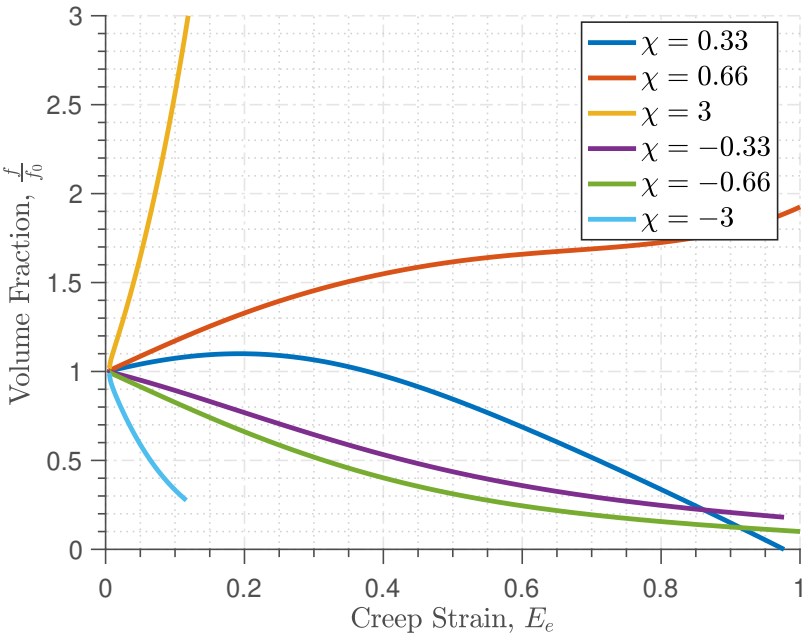


Figure 5.25: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different stress triaxiality values, Lode Parameter $L = +1$ and crystal orientation 110O1.

The comments in the evolution equivalent strain rate, D_e in Figure 5.26 are the same as for the last case, with 110O1 crystal orientation and $L = -1$. For brevity, the void shape analysis is not included for this case as all the needed remarks have been commented in Figure 5.23.

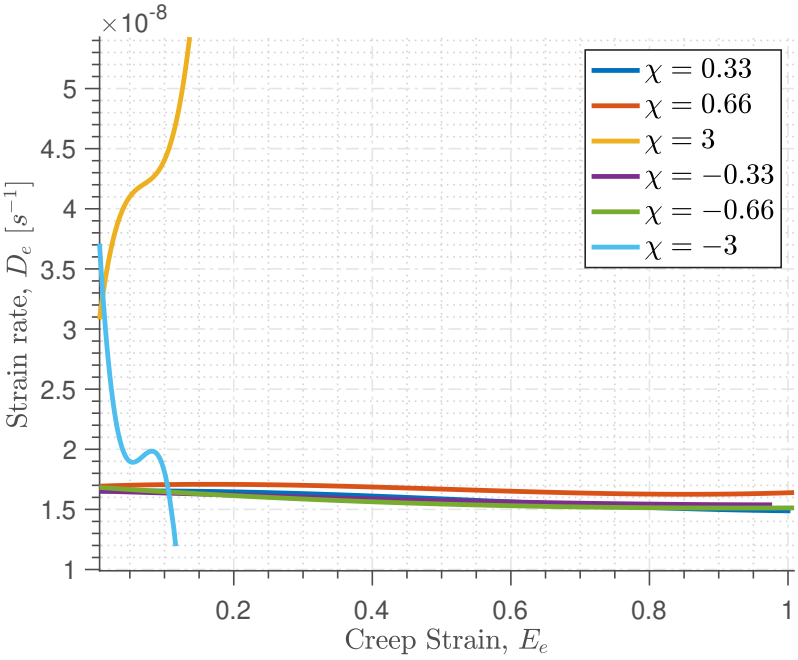


Figure 5.26: Equivalent strain rate, D_e vs Equivalent creep strain, E_e for different stress triaxiality values, Lode Parameter $L = +1$ and crystal orientation 110O1.

5.3.3 Influence of initial porosity level

Another important parameter to look into concerning single crystal turbine blades is the initial porosity level, f_0. Many researchers have focused their efforts recently on trying to understand better the complex manufacturing behind a single crystal and how the mentioned initial porosity could be reduced (Yu et al., 2017) (Caron and Lavigne, 2011).

Single crystal turbine blades concerns very low porosity levels, well below the lowest case represented here, i.e. 0.25%. Four cases will be compared in this section. Table 5.6 gathers a summary of the simulated cases.

Loading	χ	L	Σ_1 [MPa]	Σ_2 [MPa]	Σ_3 [MPa]
AXT [100] $f = 0.25\%$	1/3	-1	750	0	0
AXT [100] $f = 0.5\%$	1/3	-1	750	0	0
AXT [100]$f = 1\%$	1/3	-1	750	0	0
AXT [100] $f = 2\%$	1/3	-1	750	0	0

Table 5.6: FCC Porosity analysis

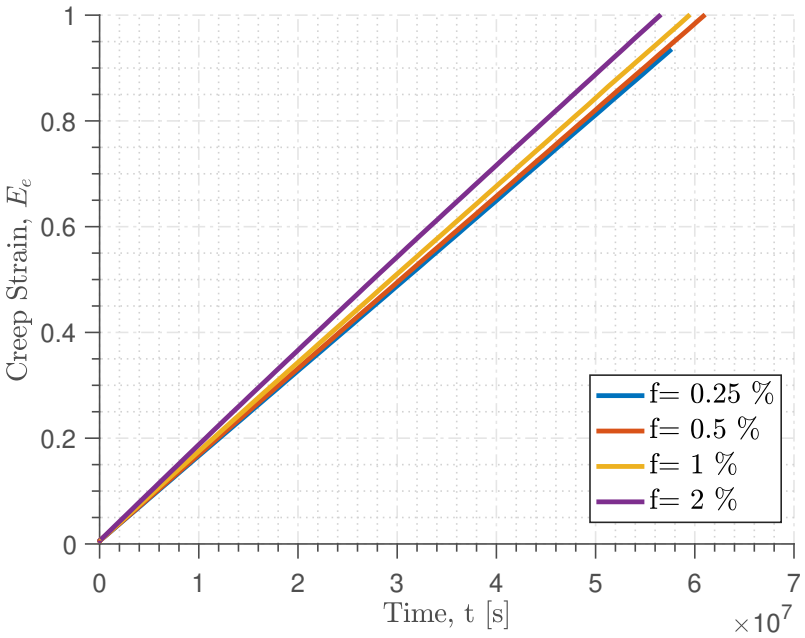


Figure 5.27: Time evolution of the Equivalent creep strain, E_e for different initial porosity levels, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = -1$ and crystal orientation 100.

From Figure 5.27 it can be seen clearly the detrimental effect of a higher initial porosity level. Higher values of porosity levels leads to a monotonically stepper increase of the creep strain with time. In fact, as Finite element computation were only carried out until $E_e = 1$ for obvious reason of mesh distortion, it is predicted that the onset for rapid increase in the creep strain, i.e. leading to creep failure, will happen earlier for the highest level of porosity studied, $f = 2\%$.

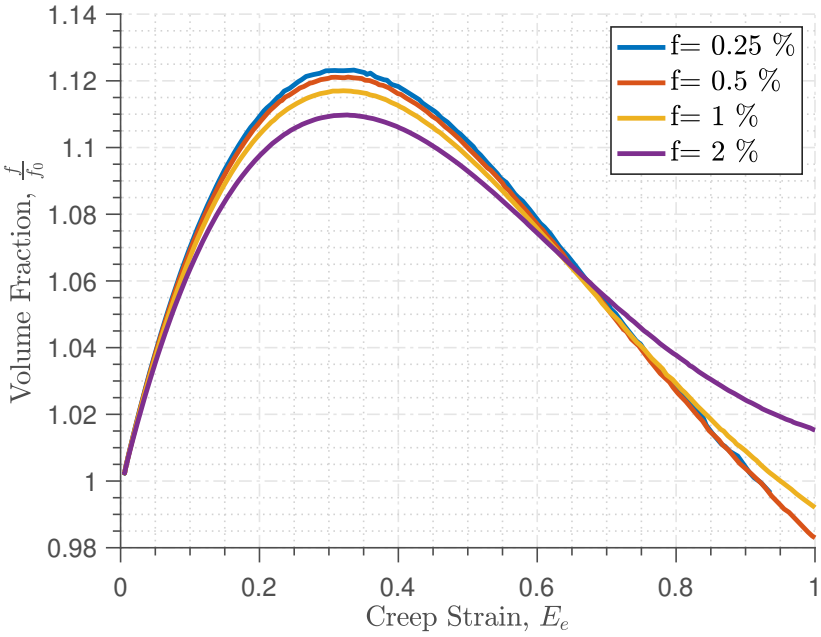


Figure 5.28: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different initial porosity levels, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = -1$ and crystal orientation 100.

Analysing now the micro-structure evolution in terms of pore size in Figure 5.28, it can be concluded that qualitatively the micro-structure evolution is independent of the porosity level, as far as the porosity levels are kept in the same order of magnitude. As it has been commented in previous sections, for the initial crystal orientation [100], $\chi = \frac{1}{3}$ and $L = -1$, the porosity level first increases until $E_e \sim 0.35$, to decrease for further deformation levels, below the initial porosity level.

However, it can be seen a small effect linked to the initial porosity: higher values of initial porosity leads to lower peak in the initial pore growth, and a softer decrease in the porosity level than for lower initial porosity levels.

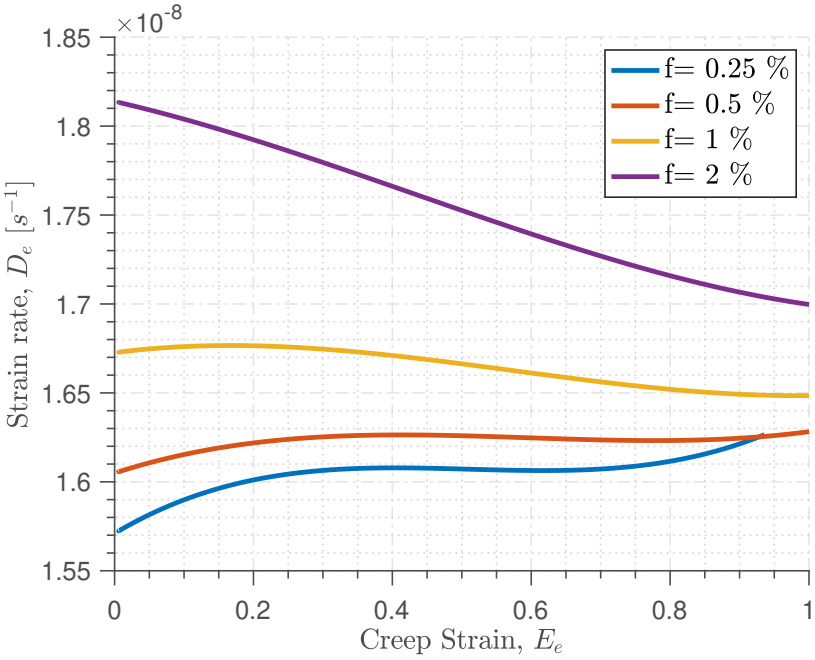


Figure 5.29: Equivalent strain rate, D_e vs Equivalent creep strain, E_e for different initial porosity levels, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = -1$ and crystal orientation 100.

The strain rates in Figure 5.29 for different porosity levels reflect what has been commented before. The higher the initial porosity level, the higher is the associated strain rate. A higher strain rate leads to earlier creep failure.

It can be seen how the slope of the strain rate, in turn, shows a slight decreasing tendency for all the cases. This fact needs to be further analysed, for larger equivalent creep strain values.

Regarding void shapes and stress concentration in Figure 5.30, it can be clearly seen how the pore shape evolution is independent of the initial porosity level; and the stress concentration around the void slightly increase as the porosity level does.

A last word must be said regarding the mesh modification for these cases. As the initial porosity changes, the size of the spherical inclusion changes and so the number mesh elements around it. While the author is aware of the need of a new mesh convergence for each case, the porosity levels involved are in the same order of magnitude and it is assumed that convergence is also achieved.

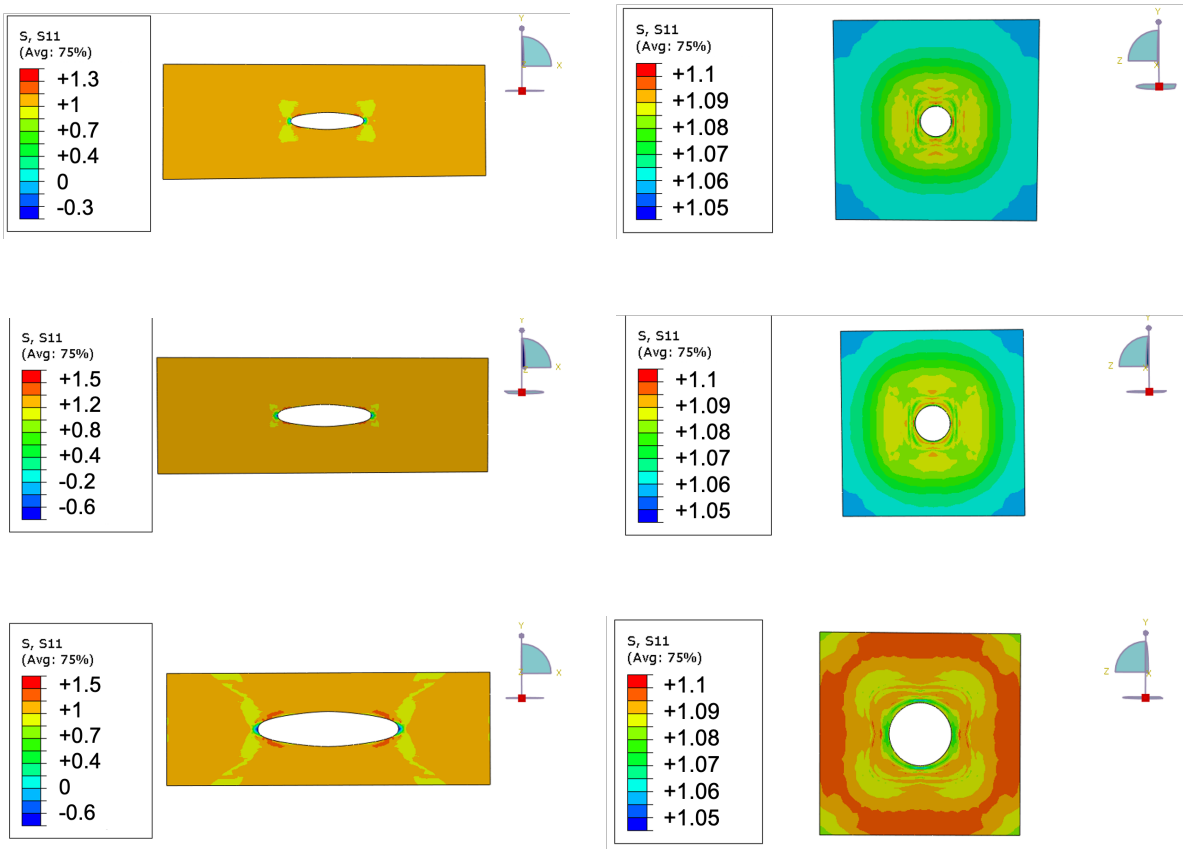


Figure 5.30: Cross sections of the stress, $\frac{\sigma_{11}}{\Sigma_e}$ for $\chi = \frac{1}{3}$, $L = -1$ and $E_e \sim 0.7$. From top to bottom: 100 orientation $f = 0.25\%$; 100 orientation $f = 0.5\%$; 100 orientation $f = 2\%$.

5.3.4 Influence of creep parameters

Another parameter to look into regarding creep behaviour is the empirical creep exponent, n. Here the effect of varying the creep exponent will be exposed for a crystal orientation [100], $\chi = \frac{1}{3}$ and $L = -1$. A summary of the computed cases can be found in Table 5.7.

Loading	χ	L	Σ_1 [MPa]	Σ_2 [MPa]	Σ_3 [MPa]
AXT [100] n=1	1/3	-1	750	0	0
AXT [100] n=2	1/3	-1	750	0	0
AXT [100] n=5	1/3	-1	750	0	0
AXT [100] n=10	1/3	-1	750	0	0

Table 5.7: FCC Creep Exponent analysis

An increase in the creep exponent leads to an earlier creep failure. This fact is clearly depicted in Figure 5.31. The slope of the equivalent creep strain over time multiplies as the creep exponent rises. A higher n exponent has a big impact in creep life.

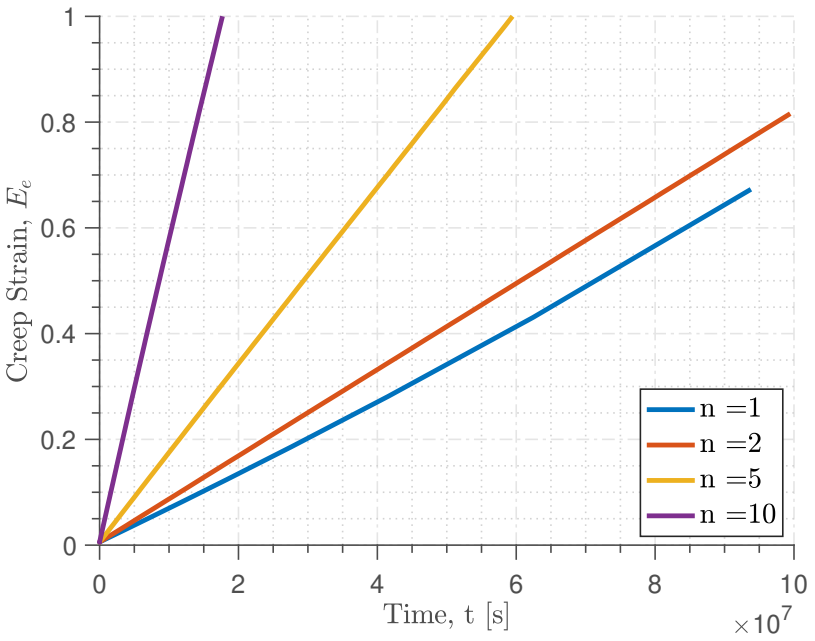


Figure 5.31: Time evolution of the Equivalent creep strain, E_e for different creep exponents, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = -1$ and crystal orientation 100.

Another effect of the creep exponent concerns the micro-structure evolution in Figure 5.32. A higher creep exponent is linked to higher pore growth until a certain E_e; and a latter larger decrease in the porosity level, below the initial value. The porosity evolution curve flattens as the creep exponent decreases.

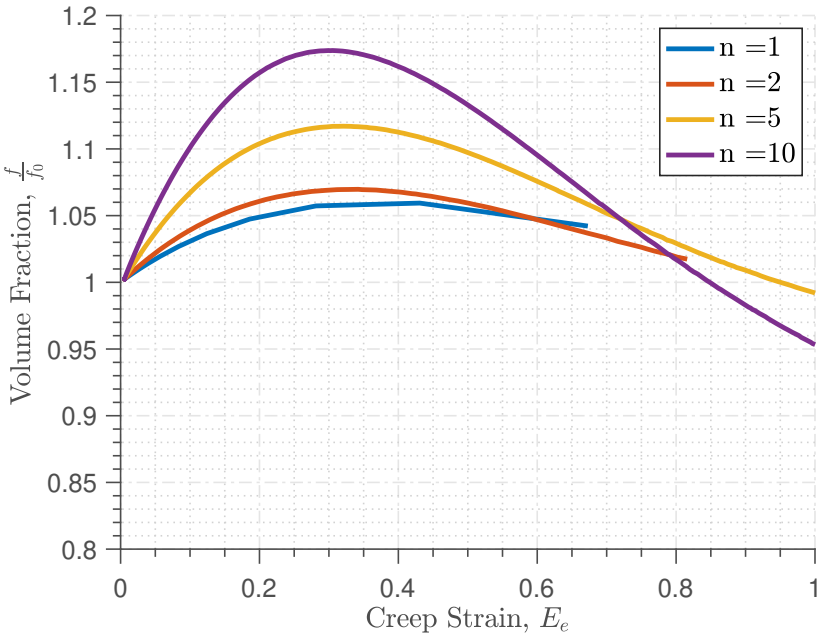


Figure 5.32: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different creep exponents, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = -1$ and crystal orientation 100.

Additionally, the equivalent strain rate vs E_e does not give any further information and it is not showed here. Higher creep exponent leads to higher initial equivalent strain rate. The strain rate curves for all the creep exponent analysed remains constant during deformation in the range and loading case studied.

From Figure 5.33, it can be observed how the void shape evolution is almost independent of the creep exponent, n.

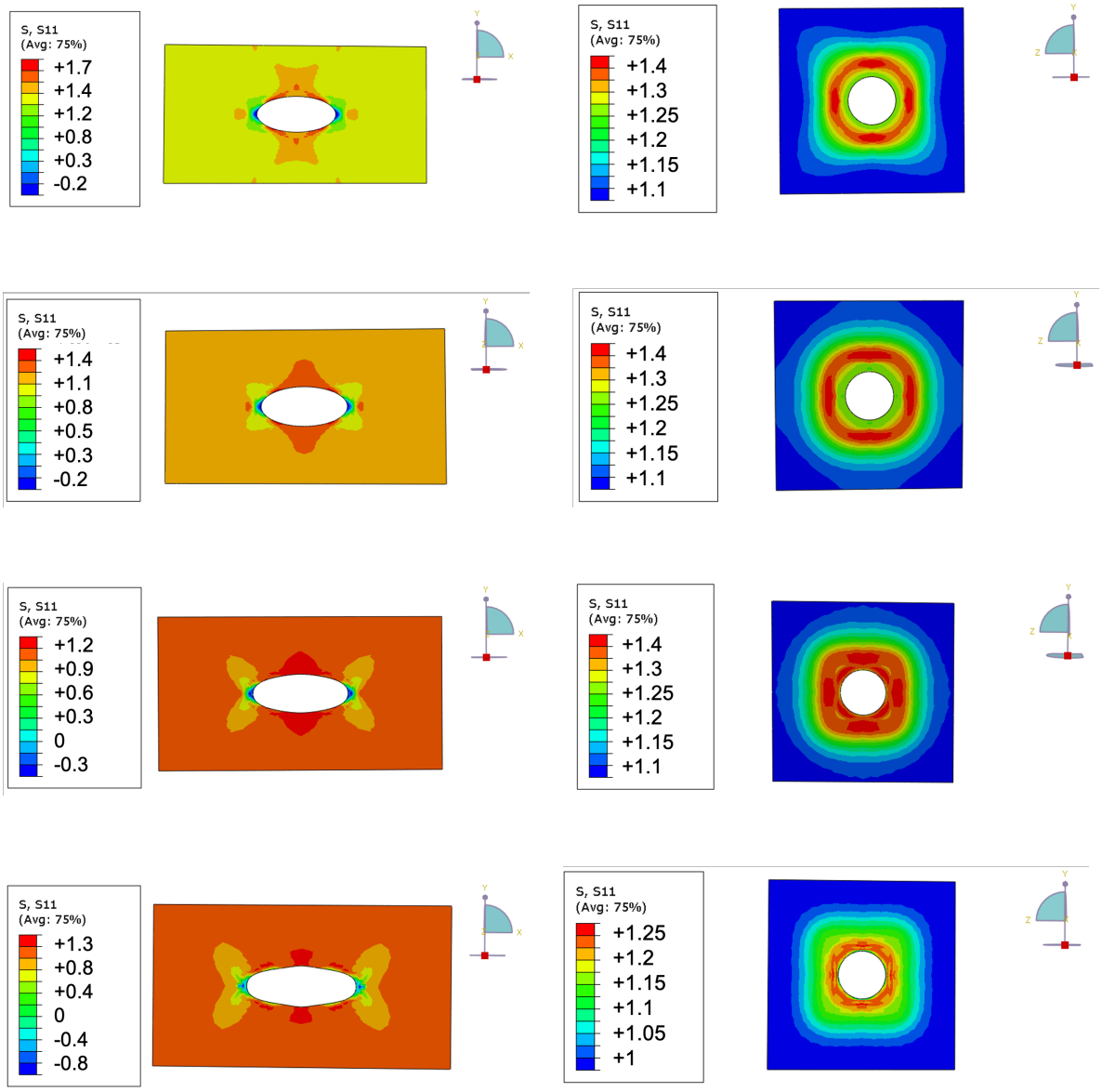


Figure 5.33: Cross sections of the stress, $\frac{\sigma_{11}}{\Sigma_e}$ for $\chi = \frac{1}{3}$, $L = -1$ and $E_e \sim 0.4$. From top to bottom: 100 orientation $n = 1$; 100 orientation $n = 2$; 100 orientation $n = 5$; 100 orientation $n = 10$.

5.3.5 Effect of initial void shape

The last analysis for FCC single crystal concerns the initial void shape. While many researchers have focused on understanding how the pores are formed during the casting of single crystals, others tried to obtain a micro-photography of them.

For this reason, different initial ellipsoidal void shapes will be considered. Here, w represents the aspect ratio between major and minor axis, so that $w = \frac{r_2}{r_1}$. r_1 represents the radius in the X-direction and r_2 represents the radius in the Y-direction, i.e. main loading directions. It is also considered $r_3 = r_1$. Axysimmetric tension and axysimemtric compression for an initial crystal orientation [100], $\chi = \frac{1}{3}$ will be conducted. Table 5.8 summarizes the cases where $L = -1$.

Loading	χ	L	Σ_1 [MPa]	Σ_2 [MPa]	Σ_3 [MPa]
AXT [100] $w = 0.25$	1/3	-1	750	0	0
AXT [100] $w = 0.5$	1/3	-1	750	0	0
AXT [100] $w = 1$	1/3	-1	750	0	0
AXT [100] $w = 2$	1/3	-1	750	0	0

Table 5.8: FCC Initial void shape analysis (I).

From Figure 5.34 it can be clearly observed how the initial void shape has no influence in the creep strain evolution of the porous single crystal for the cases studied.

However, the porosity evolution, depicted in Figure 5.35 shows how the pore growth is highly affected by the initial void shape. A larger radius in the direction of loading, r_1 or X-direction, is linked to a smaller peak in the porosity evolution, $\frac{f}{f_0}$. The peak in the porosity occurs for $\frac{f}{f_0} > 1$ and higher as the aspect ratio increases. Additionally, higher aspect ratios retard the peak on pore size to latter equivalent creep strains, i.e. time.

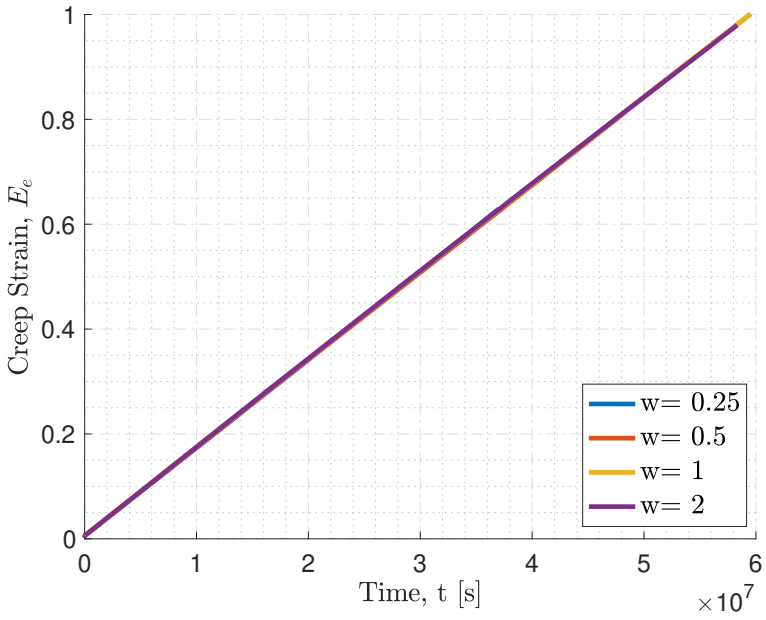


Figure 5.34: Time evolution of the Equivalent creep strain, E_e for different initial ellipsoidal void shapes, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = -1$ and crystal orientation 100.

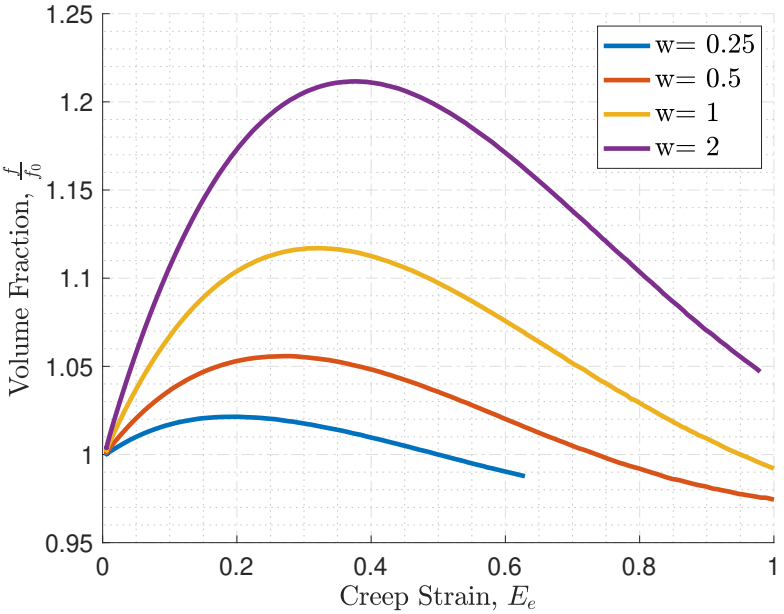


Figure 5.35: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different initial ellipsoidal void shapes, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = -1$ and crystal orientation 100.

The strain rates, highly related to the creep life of the material, do not show big differences between each other. It is important to notice the case of $w = 0.25$ where the strain rate shows an abrupt trend. This fact is linked to excessive mesh distortion achieved during simulation for a certain creep strain.

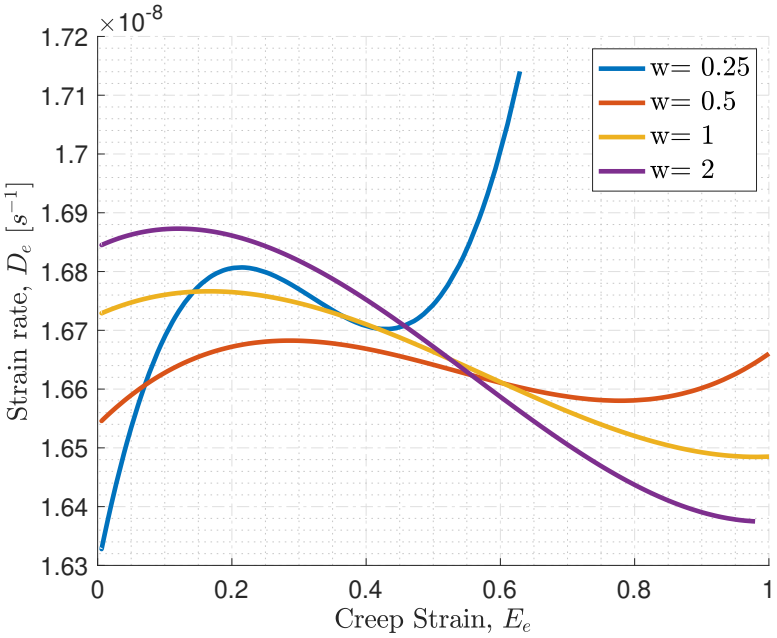


Figure 5.36: Equivalent strain rate, D_e vs Equivalent creep strain, E_e for different initial ellipsoidal void shape, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = -1$ and crystal orientation 100.

If now we have a look into the stress contour plots in Figure 5.37, it may appear that the void shape evolution strongly depends on the initial void shape. However, the evolution is the same for all the cases, and equal to the one described in the first analysis for $w = 1$: the void is stretching in the main direction of loading, i.e. X-direction. What it is changing is the initial pore shape, which in fact, will define the pore characteristic shape at a certain equivalent creep strain to be different for each case.

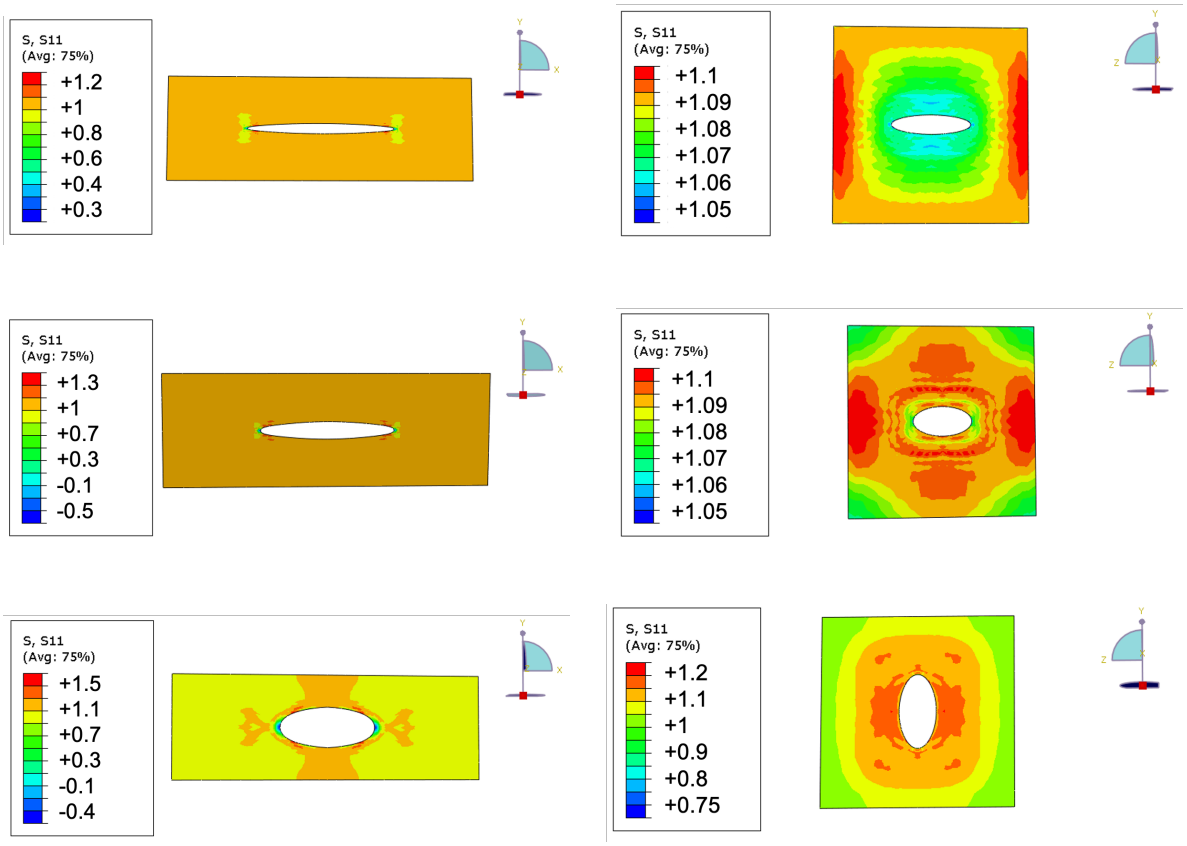


Figure 5.37: Cross sections of the stress, $\frac{\sigma_{11}}{\Sigma_e}$ for $\chi = \frac{1}{3}$, $L = -1$ and $E_e \sim 0.7$. From top to bottom: 100 orientation $w = 0.25$; 100 orientation $w = 0.5$; 100 orientation $w = 2$.

Regarding the axisymmetric compression case ($L = +1$) in Table 5.9, the conclusions to be drawn in terms of creep strain history and porosity evolution are equal. The creep life of the porous single crystal is nearly independent of the initial void shape as it can be seen in Figure 5.38.

Loading	χ	L	Σ_1 [MPa]	Σ_2 [MPa]	Σ_3 [MPa]
AXC [100] $w = 0.25$	1/3	+1	500	500	-250
AXC [100] $w = 0.5$	1/3	+1	500	500	-250
AXC [100] $w = 1$	1/3	+1	500	500	-250
AXC [100] $w = 2$	1/3	+1	500	500	-250

Table 5.9: FCC Initial void shape analysis (II).

In this case, the direction of loading shifts to Y-direction, or r_2. Lower values of w will lead to higher peak in the pore growth as in Figure 5.39.

If now the attention is shifted to the strain rate evolution, lower aspect ratios shows a slight detrimental effect in the creep evolution. Nevertheless, the differences are negligible. Additionally, as expected from a case of void collapse, previously linked to a "hardening effect". the strain rate evolution shows a decreasing tendency along deformation.

The conclusions in terms of micro-structure for the pore shape evolution are similar to the one described in previous case, the evolution along a given quantity of creep strain is the same, although the final pore shape will differ due to the initial different void aspect ratios. This results in terms of stress concentration are not showed for brevity.

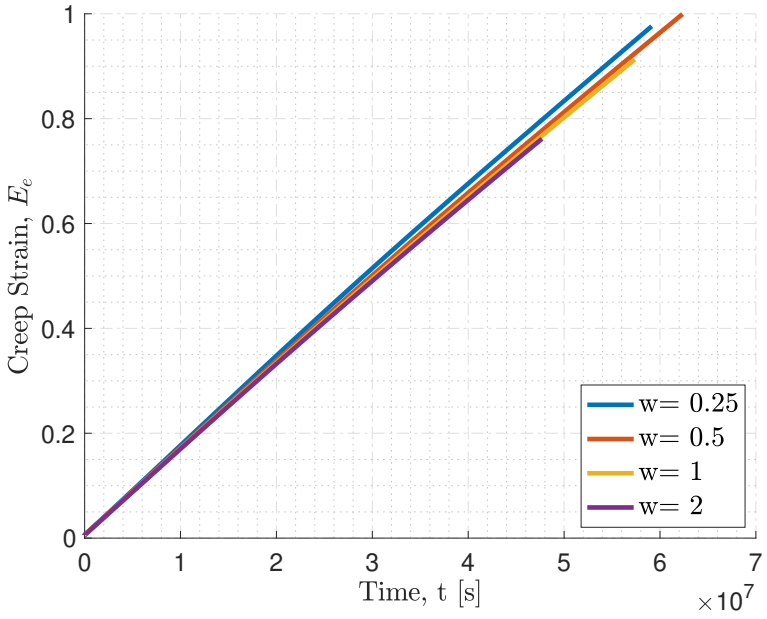


Figure 5.38: Time evolution of the Equivalent creep strain, E_e for different initial ellipsoidal void shapes, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = +1$ and crystal orientation 100.

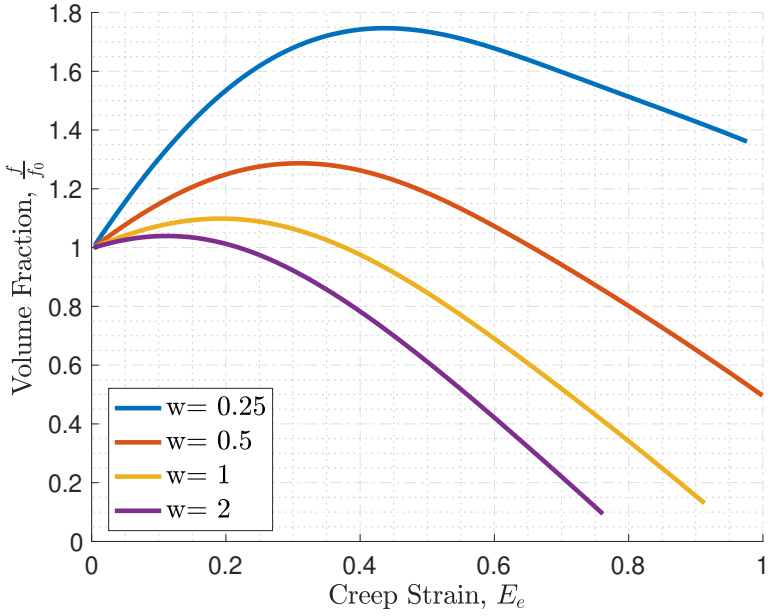


Figure 5.39: Porosity evolution, $\frac{f}{f_0}$ vs Equivalent creep strain, E_e for different initial ellipsoidal void shapes, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = +1$ and crystal orientation 100.

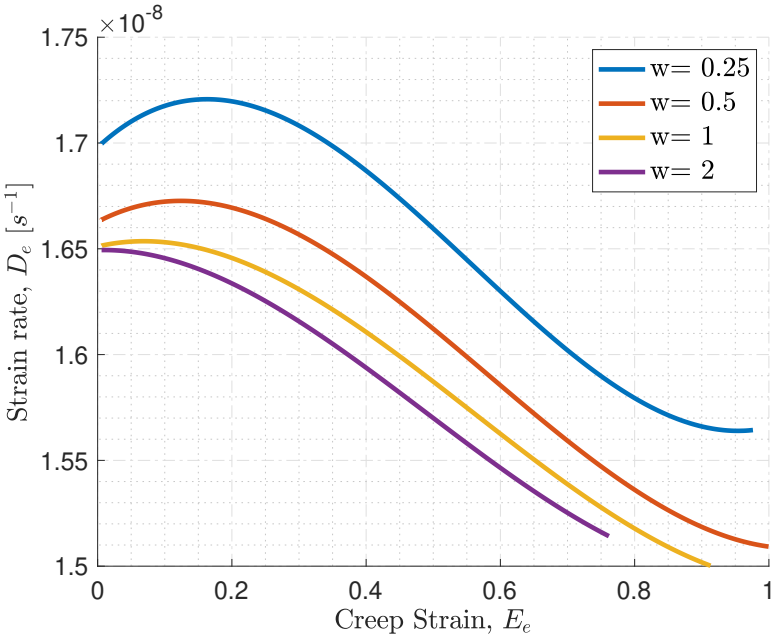


Figure 5.40: Equivalent strain rate, D_e vs Equivalent creep strain, E_e for different initial ellipsoidal void shapes, stress triaxiality $\chi = \frac{1}{3}$, Lode Parameter $L = +1$ and crystal orientation 100.

6 Conclusions

A general Unit Cell with Periodic Boundary conditions (PBC) Finite Element model has been developed. The Unit Cell Finite Element model allows to compute the finite-strain and finite-rotations macroscopic response and the porosity evolution of a porous single crystal with an arbitrary initial crystal orientation under general loading conditions with periodic micro-structure. This works focused on the creep behaviour of a classical FCC single crystal turbine blade. The capabilities of the Unit Cell model may be extended to a wide range of engineering applications, where the micro-structure evolution plays an important role.

A complete overview of the finite-strain macroscopic response and micro-structure evolution of a non-linear porous FCC single crystal has been presented by making use of two different approaches: a second order variational homogenization approach and a numerical three dimensional Finite Element computation.

The strong influence of the micro-structure evolution on a porous FCC single crystal creep behaviour has been widely proven.

The implementation of the User-defined Amplitude (UAMP) subroutine allows to define a true stress or Cauchy stress loading condition, such that the values of the Lode parameter, L and stress triaxiality, χ were kept constant during deformation.

The accuracy of the variational second-order (FOSO) homogenization approach together with the iterative homogenization method of (Ponte Castañeda, 2015)(Agoras and Ponte Castañeda, 2013) has been assessed along the extensive creep parametric study conducted for FCC single crystals in terms of creep strain evolution. Fairly good agreement in terms of micro-structure evolution was found for the range of loading configurations for the symmetric crystal orientation [100]. The misalignment reflected in the comparison reminds the reader the fact of the different micro-structure definition in both methods: periodic array of voids for the Unit Cell Finite Element Results versus the random distribution considered in the homogenization approach.

The 100 crystal orientation show a more symmetric behaviour whereas the initial crystal orientation 11O01 shows a markedly anisotropic behaviour for different loading configurations. In turn, for the 110O1 crystal orientation an overall improvement in the creep strain can be achieved relate to no inclusion growth, or even if void collapse occurs.

For the 110O1 anisotropic orientation, the stress triaxiality plays an important role in the micro-structure evolution, specifically in the porosity evolution and the pore shape. For the same value of Lode parameter, the stress triaxility value will determine whether the void growth occurs or not; or even if the void evolves leading to collapse. Moreover, for $L = -1$ high stress triaxiality test showed how the micro-structure evolve with a significant increase in size and adopting a polygonal void shape with rounded corners. In the case of the most negative stress triaxiality case, $\chi = -3$, the void evolves into a nut-shell shape, leading to void splitting.

The creep exponent has a strong effect in the creep life of the specimen. A higher value of n has a detrimental effect on the creep strain evolution of the porous single crystal.

Furthermore, creep strain evolution has been proven to be nearly independent of the initial porosity level and the initial void shape. Higher effects were found in the micro-structure evolution in terms of void growth and shape.

Stress concentration around the void has been proven to be highly dependent on the crystal orientation and loading scenario.

7 Future work

A complete overview of the creep response of a porous FCC single crystal under different loading cases has been addressed in this work. Face center cubic (FCC) structure is characteristic of an aeronautical type single crystal turbine blade, such as the ***PWA1484 Ni-based single crystal super-alloy***.

However, the constructed Unit Cell Finite Element model with Periodic Boundary conditions can be exploited for a vast range of applications. Given the capability of the User-defined Material subroutine (UMAT), it would be interesting to extend it for Hexagonal Closest Pack (HCP) structures.

This fact can be carefully implemented by solely modifying two subroutines inside the aforementioned UMAT code, the *SLIPSYSTEM* and the *ROTATION* functions, to account for the effect of aspect ratio of non-cubic crystals and the relative orientation of the basal vectors for non-orthotropic crystal. It is important to recall that the slip phenomena associated with Hexagonal Closest Pack (HCP) is more complex that the Face Center Cubic (FCC) one.

From creep behaviour in big ice plaques located in Antarctica to understand better the "ice flow" in a long-time scale; to understand the possibility of using HCP single crystal materials for turbine blade applications.

The capabilities of the Unit Cell PBC problem allow in fact to understand deeply the physic behind the micro-structure evolution of a porous single crystal.

Bibliography

ABAQUS. (2014). Abaqus / cae 6.14 user's manual, 1–1146.

Agoras, M., & Ponte Castañeda, P. (2013). Iterated linear comparison bounds for viscoplastic porous materials with ellipsoidal microstructures. *Journal of the Mechanics and Physics of Solids*, *61*(3), 701–725. https://doi.org/10.1016/j.jmps.2012.11.003

Agoras, M., & Ponte Castañeda, P. (2014). Anisotropic finite-strain models for porous viscoplastic materials with microstructure evolution. *International Journal of Solids and Structures*, *51*(5), 981–1002. https://doi.org/10.1016/j.ijsolstr.2013.11.025

Akca, E., & Gürsel, A. (2015). A Review on Superalloys and IN718 Nickel-Based INCONEL Superalloy. *Period. Eng. Nat. Sci.*, *3*(1). https://doi.org/10.21533/pen.v3i1.43

Anton, D. L., & Giamei, A. F. (1985). Porosity distribution and growth during homogenization in single crystals of a nickel-base superalloy. *Mater. Sci. Eng.*, *76*(100), 173–180. https://doi.org/10.1016/0025-5416(85)90091-6

Aravas, N., & Castañeda, P. P. (2004). Numerical methods for porous metals with deformation-induced anisotropy. *Computer Methods in Applied Mechanics and Engineering*, *193*(36), 3767–3805. https://doi.org/10.1016/j.cma.2004.02.009

Asaro, R., & Needleman, A. (1985). Overview no. 42 texture development and strain hardening in rate dependent polycrystals. *Acta Metallurgica*, *33*(6), 923–953. https://doi.org/10.1016/0001-6160(85)90188-9

Asaro, R. J. (1983). Micromechanics of crystals and polycrystals (J. W. Hutchinson & T. Y. Wu, Eds.). In J. W. Hutchinson & T. Y. Wu (Eds.). Elsevier. https://doi.org/10.1016/S0065-2156(08)70242-4

Bassani, J. L., & Wu, T.-Y. (1991). Latent hardening in single crystals ii. analytical characterization and predictions. *Proceedings: Mathematical and Physical Sciences*, *435*(1893), 21–41. http://www.jstor.org/stable/52000

Bohm, H. J. (2004). *Mechanics of Microstructured Materials*. https://doi.org/10.1007/978-3-7091-2776-6

Bokstein, B. S., Epishin, A. I., Link, T., Esin, V. A., Rodin, A. O., & Svetlov, I. L. (2007). Model for the porosity growth in single-crystal nickel-base superalloys

during homogenization. *Scr. Mater.*, *57*(9), 801–804. https://doi.org/10.1016/j.scriptamat.2007.07.012

Caron, P., & Lavigne, O. (2011). Recent studies at onera on superalloys for single crystal turbine blades. *Aerosp. lab*, *AL03-02*(3), 1–14.

Castañeda, P. P., & Willis, J. R. (1995). The effect of spatial distribution on the effective behavior of composite materials and cracked media. *J. Mech. Phys. Solids*, *43*(12), 1919–1951. https://doi.org/10.1016/0022-5096(95)00058-Q

Castañeda, P. P., & Suquet, P. (1997). Nonlinear composites (E. [der Giessen] & T. Y. Wu, Eds.). In E. [der Giessen] & T. Y. Wu (Eds.). Elsevier. https://doi.org/10.1016/S0065-2156(08)70321-1

Castañeda, P. (1991). The effective mechanical properties of nonlinear isotropic composites. *Journal of the Mechanics and Physics of Solids*, *39*(1), 45–71. https://doi.org/10.1016/0022-5096(91)90030-R

Castañeda, P., & Zaidman, M. (1994a). Constitutive models for porous materials with evolving microstructure. *Journal of the Mechanics and Physics of Solids*, *42*(9), 1459–1497. https://doi.org/10.1016/0022-5096(94)90005-1

Castañeda, P., & Zaidman, M. (1994b). Constitutive models for porous materials with evolving microstructure. *Journal of the Mechanics and Physics of Solids*, *42*(9), 1459–1497. https://doi.org/10.1016/0022-5096(94)90005-1

Chen, I., & Argon, A. (1979). Steady state power-law creep in heterogeneous alloys with coarse microstructures. *Acta Metallurgica*, *27*(5), 785–791. https://doi.org/10.1016/0001-6160(79)90112-3

Danas, K., & Castañeda], P. [(2009a). A finite-strain model for anisotropic viscoplastic porous media: I – theory. *European Journal of Mechanics - A/Solids*, *28*(3), 387–401. https://doi.org/10.1016/j.euromechsol.2008.11.002

Danas, K., & Castañeda], P. [(2009b). A finite-strain model for anisotropic viscoplastic porous media: Ii – applications. *European Journal of Mechanics - A/Solids*, *28*(3), 402–416. https://doi.org/10.1016/j.euromechsol.2008.11.003

DeBotton, G., & Ponte Castaneda, P. (1995). Variational estimates for the creep behaviour of polycrystals. *Proceedings of The Royal Society of London, Series A: Mathematical and Physical Sciences*, *448*(1932), 121–142. https://doi.org/10.1098/rspa.1995.0009

Galantucci, L., & Tricarico, L. (1998). A computer-aided approach for the simulation of the directional-solidification process for gas turbine blades. *Journal of Materials*

Processing Technology, *77*(1), 160–165. https://doi.org/10.1016/S0924-0136(97)00413-5

Gan, Y. X., Kysar, J. W., & Morse, T. L. (2006). Cylindrical void in a rigid-ideally plastic single crystal II: Experiments and simulations. *International Journal of Plasticity*, *22*(1), 39–72. https://doi.org/10.1016/j.ijplas.2005.01.009

Gell, M., & Duhl, D. N. (1986). Development of Single Crystal Superalloy Turbine Blades., 41–49.

Gurson, A. L. (1977). Continuum Theory of Ductile Rupture by Void Nucleation and Growth: Part I—Yield Criteria and Flow Rules for Porous Ductile Media. *Journal of Engineering Materials and Technology*, *99*(1), 2–15. https://doi.org/10.1115/1.3443401

Hill, R., & Rice, J. (1972). Constitutive analysis of elastic-plastic crystals at arbitrary strain. *Journal of the Mechanics and Physics of Solids*, *20*(6), 401–413. https://doi.org/10.1016/0022-5096(72)90017-8

Huang, Y. (1991). Harvard university report mech-178.

Idiart, M. I., & Ponte Castañeda, P. (2007). Variational linear comparison bounds for nonlinear composites with anisotropic phases. I. General results. *Proc. R. Soc. A Math. Phys. Eng. Sci.*, *463*(2080), 907–924. https://doi.org/10.1098/rspa.2006.1797

Jin, N., Wang, F., Wang, Y., Zhang, B., Cheng, H., & Zhang, H. (2019). Failure and energy absorption characteristics of four lattice structures under dynamic loading. *Materials Design*, *169*, 107655. https://doi.org/10.1016/j.matdes.2019.107655

Joshi, A., & Stein, D. F. (1972). Chemistry of grain boundaries and its relation to intergranular corrosion of austenitic stainless steel. *CORROSION*, *28*(9), 321–330. https://doi.org/10.5006/0010-9312-28.9.321

Kailasam, M., & Castañeda, P. (1998). A general constitutive theory for linear and nonlinear particulate media with microstructure evolution. *Journal of the Mechanics and Physics of Solids*, *46*(3), 427–465. https://doi.org/10.1016/S0022-5096(97)00095-1

Kailasam, M., Ponte Castañeda, P., & Willis, J. (1997). The effect of particle size, shape, distribution and their evolution on the constitutive response of nonlinearly viscous composites. i. theory [cited By 34]. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *355*(1730), 1835–1852. https://doi.org/10.1098/rsta.1997.0092

Kermanpur, A., Varahraam, N., Engilehei, E., Mohammadzadeh, M., & Davami, P. (2000). Directional solidification of ni base superalloy in738lc to improve creep properties. *Materials Science and Technology*, *16*(5), 579–586. https://doi.org/10.1179/026708300101508117

Kouchmeshky, B., & Zabaras, N. (2009). Modeling the response of hcp polycrystals deforming by slip and twinning using a finite element representation of the orientation space. *Computational Materials Science*, *45*(4), 1043–1051. https://doi.org/10.1016/j.commatsci.2009.01.009

Liu, Y., Gilormini, P., & Castañeda], P. [(2003). Variational self-consistent estimates for texture evolution in viscoplastic polycrystals. *Acta Materialia*, *51*(18), 5425–5437. https://doi.org/10.1016/S1359-6454(03)00409-9

Lou, Y., & Schapery, R. (1971). Viscoelastic characterization of a nonlinear fiber-reinforced plastic. *Journal of Composite Materials*, *5*(2), 208–234. https://doi.org/10.1177/002199837100500206

Marin, E. (2006). On the Formulation of a Crystal Plasticity Model. https://doi.org/10.1098/rspa.2006.1797

Mbiakop, A., Constantinescu, A., & Danas, K. (2015). A model for porous single crystals with cylindrical voids of elliptical cross-section. *International Journal of Solids and Structures*, *64-65*, 100–119. https://doi.org/10.1016/j.ijsolstr.2015.03.017

Noels, L. (n.d.). Toward stochastic multi-scale methods in continuum solid mechanics.

Ogden, R. W. (1984). *Non-linear elastic deformations*. New York, NY, Dover Publications. http://cds.cern.ch/record/2278985

Peirce, D., Asaro, R., & Needleman, A. (1982). An analysis of nonuniform and localized deformation in ductile single crystals. *Acta Metallurgica*, *30*(6), 1087–1119. https://doi.org/10.1016/0001-6160(82)90005-0

Peirce, D., Shih, C., & Needleman, A. (1984). A tangent modulus method for rate dependent solids. *Computers Structures*, *18*(5), 875–887. https://doi.org/10.1016/0045-7949(84)90033-6

Ponte Castañeda, P. (2015). Fully optimized second-order variational estimates for the macroscopic response and field statistics in viscoplastic crystalline composites. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *471*(2184). https://doi.org/10.1098/rspa.2015.0665

Ponte Castañeda, P., & Willis, J. R. (1999). Variational second-order estimates for nonlinear composites. *Proceedings of the Royal Society A: Mathematical, Physical*

and Engineering Sciences, *455*(1985), 1799–1811. https://doi.org/10.1098/rspa.1999.0380

Rice, J. (1971). Inelastic constitutive relations for solids: An internal-variable theory and its application to metal plasticity. *Journal of the Mechanics and Physics of Solids*, *19*(6), 433–455. https://doi.org/10.1016/0022-5096(71)90010-X

Roters, F., Eisenlohr, P., Hantcherli, L., Tjahjanto, D. D., Bieler, T. R., & Raabe, D. (2010). Overview of constitutive laws, kinematics, homogenization and multi-scale methods in crystal plasticity finite-element modeling: Theory, experiments, applications. *Acta Mater.*, *58*(4), 1152–1211. https://doi.org/10.1016/j.actamat.2009.10.058

Seetharaman, V., & Pratt, A. D. C. (2003). Thickness Debit in Creep Properties of PWA 1484, (Icf 14). https://doi.org/10.7449/2004/Superalloys_2004_207_214

Segurado, J., Lebensohn, R. A., & Llorca, J. (2018). *Computational Homogenization of Polycrystals* (1st ed., Vol. 51). Elsevier Inc. https://doi.org/10.1016/bs.aams.2018.07.001

Song, D., Agoras, M., & Castañeda], P. [(2015). The evolution of pore shape and orientation in plastically deforming metals: Implications for macroscopic response and shear localization [Proceedings of the IUTAM Symposium on Micromechanics of Defects in Solids]. *Mechanics of Materials*, *90*, 47–68. https://doi.org/10.1016/j.mechmat.2015.01.011

Song, D., & Ponte Castañeda, P. (2017a). A finite-strain homogenization model for viscoplastic porous single crystals: I – Theory. *Journal of the Mechanics and Physics of Solids*, *107*, 560–579. https://doi.org/10.1016/j.jmps.2017.06.008

Song, D., & Ponte Castañeda, P. (2017b). A finite-strain homogenization model for viscoplastic porous single crystals: II – Applications. *Journal of the Mechanics and Physics of Solids*, *107*, 580–602. https://doi.org/10.1016/j.jmps.2017.06.003

Srivastava, A., & Needleman, A. (2012). Porosity evolution in a creeping single crystal. *Modelling and Simulation in Materials Science and Engineering*, *20*(3). https://doi.org/10.1088/0965-0393/20/3/035010

Srivastava, A., & Needleman, A. (2015). Effect of crystal orientation on porosity evolution in a creeping single crystal. *Mechanics of Materials*, *90*, 10–29. https://doi.org/10.1016/j.mechmat.2015.01.015

Staroselsky, A., & Cassenti, B. N. (2011). Creep, plasticity, and fatigue of single crystal superalloy. *Int. J. Solids Struct.*, *48*(13), 2060–2075. https://doi.org/10.1016/j.ijsolstr.2011.03.011

Tvergaard, V. (1989). Material failure by void growth to coalescence (J. W. Hutchinson & T. Y. Wu, Eds.). In J. W. Hutchinson & T. Y. Wu (Eds.). Elsevier. https://doi.org/10.1016/S0065-2156(08)70195-9

Willis, J. (1991). On methods for bounding the overall properties of nonlinear composites. *Journal of the Mechanics and Physics of Solids*, *39*(1), 73–86. https://doi.org/10.1016/0022-5096(91)90031-I

Wu, L., Adam, L., Doghri, I., & Noels, L. (2017). An incremental-secant mean-field homogenization method with second statistical moments for elasto-visco-plastic composite materials. *Mech. Mater.*, *114*, 180–200. https://doi.org/10.1016/j.mechmat.2017.08.006

Yu, Z. Y., Wang, X. Z., Yue, Z. F., & Wang, X. M. (2017). Investigation on microstructure evolution and fracture morphology of single crystal nickel-base superalloys under creep-fatigue interaction loading. *Materials Science and Engineering A*, *697*(April), 126–131. https://doi.org/10.1016/j.msea.2017.05.018

Zhang, S., Song, X., Dini, D., & Korsunsky, A. (2005). Formulation and calibration of a crystal plasticity finite element model for hcp polycrystals.

Appendix A

```
### Author: Simon Carrillo Segura
### University of Pennsylvania, Philadelphia
### Department of Mechanical Engineering and Applied Mechanics (MEAM)

from part import *
from material import *
from section import *
from assembly import *
from step import *
from interaction import *
from load import *
from mesh import *
from optimization import *
from job import *
from sketch import *
from visualization import *
from connectorBehavior import *

## Subroutine for 3D Periodic Boundary Conditions to be supplied
Mdb()
execfile('PBC_script.py') # calling the subroutine for the PBC computation

##############################
#INPUTS (user defined)
##############################
ODBname = 'AXT_001_w' #set the name for the job

#Microstructure properties
pi=acos(-1.) # pi number definition
a_0 = 0.5 # semi-length cube
Porosity = 0.01 # Porosity level

ellipsoid = 0 # 1 if ellipsoid; 0 if spheroid
asperity = 2 # major axis

if ellipsoid == 1 :

```

```
38      r_0 = a_0*(6.*Porosity/(pi*asperity))**(1./3.) #major axis
39      r_1  = r_0 *asperity; # minor axis
40
41  elif ellipsoid == 0 :
42
43      r_0 = a_0*(6.*Porosity/pi)**(1./3.); # radius of the
            inclusion
44      r_1 = r_0
45
46  # mesh parameters
47  n_lig = 10 #number of elements in the ligament 10
48  n_void = 10 # number of elements in a quarter of void 10
49  sides = 6 # number of cells in the semi-cube 6
50
51  #Material properties (single crystal)
52  C11= 283300.
53  C22= 197500.
54  C44= 112000.
55  n = 5 #creep exponent
56  gammadot = 0.00000000153 #gammadot_zero
57  tau_0 = 245. # initial shear stress
58  crystal_orientation1=  ( 1.,0.,0.,1.,0.,0.) # vector relating
        local and global crystal orientation
59  crystal_orientation2=  ( 0.,1.,0.,0.,1.,0.) # vector relating
        local and global crystal orientation
60
61
62  # Loading parameters
63  F11=750. # sigma11
64  F12=0. # tau12
65  F13=0. # tau13
66
67  F21=0. # tau21
68  F22=2000. # sigma22
69  F23=0. # tau23
70
71  F31=0. # tau31
72  F32=0. # tau32
73  F33=2000. # sigma33
74
75
76  total_time = 200000000. # max total sim time
77
```

```
78
79 ##########################################
80 ## DO NOT TOUCH BELOW HERE!!!!!! ###
81 ##########################################
82 #CREATE PART
83 ##########################################
84 mdb.models['Model-1'].ConstrainedSketch(name='__profile__',
       sheetSize=200.0) # define the work-area
85
86 mdb.models['Model-1'].sketches['__profile__'].rectangle(point1
       =(0.0, 0.0), # create the rectangule (cubic face)
87        point2=(a_0, a_0))
88 mdb.models['Model-1'].Part(dimensionality=THREE_D, name='Part
       -2-1', type=
89        DEFORMABLE_BODY)
90 mdb.models['Model-1'].parts['Part-2-1'].BaseSolidExtrude(depth
       =a_0, sketch=
91        mdb.models['Model-1'].sketches['__profile__']) # 3d cube
            constructed here
92
93
94 mdb.models['Model-1'].ConstrainedSketch(name='__profile2__',
       sheetSize=200.0) # define the work-area
95 g = mdb.models['Model-1'].sketches['__profile2__'].
       EllipseByCenterPerimeter(center=(0.0, 0.0), axisPoint1=(r_0,
        0.0),
96             axisPoint2=(0.0, r_1)) #ellipse
97 mdb.models['Model-1'].sketches['__profile2__'].Line(point1
       =(0.0, r_1), point2=(0.0, -r_1)) #line to join the arc end
       points
98 mdb.models['Model-1'].sketches['__profile2__'].
       ConstructionLine(point1=(0.0,-r_1),point2=(0.0,r_1)) #
       oonstruction line for the axis of revolution
99
100 mdb.models['Model-1'].sketches['__profile2__'].autoTrimCurve(
       curve1=g, point1=(-r_0, 0.0)) # obtain half ellipse (
       trimming)
101
102 mdb.models['Model-1'].Part(dimensionality=THREE_D, name='Part
       -3', type=
103        DEFORMABLE_BODY)
104 mdb.models['Model-1'].parts['Part-3'].BaseSolidRevolve(sketch=
105        mdb.models['Model-1'].sketches['__profile2__'], angle=
```

```
        360.0) # 3d sphere constructed here
106
107
108 mdb.models['Model-1'].rootAssembly.Instance(name='Part-2-2',
        part= mdb.models['Model-1'].parts['Part-2-1'])  #define
        instances for the cut procedure
109 mdb.models['Model-1'].rootAssembly.Instance(name='Part-3-2',
        part= mdb.models['Model-1'].parts['Part-3'])  #define
        instances for the cut procedure
110     # redefine Part-2 as the intersection of the bodies
111 mdb.models['Model-1'].rootAssembly.PartFromBooleanCut(name='
        Part-2', instanceToBeCut= mdb.models['Model-1'].rootAssembly
        .instances['Part-2-2'], cuttingInstances= (mdb.models['Model
        -1'].rootAssembly.instances['Part-3-2'],))
112
113 ##########################################
114 # Definition of the mesh parameters
115 ##########################################
116
117 mdb.models['Model-1'].parts['Part-2'].seedPart(deviationFactor
        =0.1,
118       minSizeFactor=0.1, size=0.2) # 0.15
119 mdb.models['Model-1'].parts['Part-2'].setMeshControls(
        elemShape=HEX_DOMINATED, regions=
120       mdb.models['Model-1'].parts['Part-2'].cells, technique=
            STRUCTURED)
121
122 mdb.models['Model-1'].parts['Part-2'].setElementType(elemTypes
        =(ElemType(
123       elemCode=C3D20H, elemLibrary=STANDARD), ElemType(elemCode=
            C3D20H,
124       elemLibrary=STANDARD), ElemType(elemCode=C3D20H,
            elemLibrary=STANDARD)),
125       regions=(mdb.models['Model-1'].parts['Part-2'].cells, ))
126
127       ##Mesh refinements in the edges
128
129 mdb.models['Model-1'].parts['Part-2'].seedEdgeByNumber(edges=
        (mdb.models['Model-1'].parts['Part-2-1'].edges.findAt
        (((0.3,0.0,0.0),))[0],)
130 , number= 14)
131
132 mdb.models['Model-1'].parts['Part-2'].seedEdgeByNumber(edges=
```

```
    (mdb.models['Model-1'].parts['Part-2-1'].edges.findAt
    (((0.0,0.3,0.0),)))[0],)
, number= 14)

mdb.models['Model-1'].parts['Part-2'].seedEdgeByNumber(edges=
    (mdb.models['Model-1'].parts['Part-2-1'].edges.findAt
    (((0.5,0.3,0.0),)))[0],)
, number= 6)

mdb.models['Model-1'].parts['Part-2'].seedEdgeByNumber(edges=
    (mdb.models['Model-1'].parts['Part-2-1'].edges.findAt
    (((0.5,0.3,0.0),)))[0],)
, number= 6)

mdb.models['Model-1'].parts['Part-2'].seedEdgeByNumber(edges=
    (mdb.models['Model-1'].parts['Part-3'].edges.findAt(((0.0,
    r_1,0.0),(r_0, 0.0, 0.0),)))[0],)
, number= 24)

session.viewports['Viewport: 1'].view.setValues(nearPlane
    =1.1417,
    farPlane=2.49912, width=1.09434, height=0.535756,
        cameraPosition=(
    -0.94545, 0.976212, 1.41532), cameraUpVector=(0.489371,
        0.724344,
    -0.485635), cameraTarget=(0.26049, 0.229021, 0.26049))
session.viewports['Viewport: 1'].view.setValues(nearPlane
    =1.20614,
    farPlane=2.40382, width=1.1561, height=0.565994,
        cameraPosition=(
    -1.18899, 0.458363, -0.819664), cameraUpVector=(0.341186,
        0.893802,
    0.29105), cameraTarget=(0.261675, 0.231541, 0.271364))
session.viewports['Viewport: 1'].view.setValues(nearPlane
    =1.21185,
    farPlane=2.39966, width=1.16157, height=0.568671,
        cameraPosition=(
    -1.00771, 0.179191, -1.04392), cameraUpVector=(0.21639,
        0.952558,
    0.214031), cameraTarget=(0.259236, 0.235297, 0.274381))
p = mdb.models['Model-1'].parts['Part-2']
c = p.cells
pickedRegions = c.getSequenceFromMask(mask=('[#1 ]', ), )
```

```
159 p.deleteMesh(regions=pickedRegions)
160 p = mdb.models['Model-1'].parts['Part-2']
161 e = p.edges
162 pickedEdges1 = e.getSequenceFromMask(mask=('[#8 ]', ), )
163 p.seedEdgeByBias(biasMethod=SINGLE, end1Edges=pickedEdges1,
    ratio=5.0,
164     number=n_lig, constraint=FINER)
165 p = mdb.models['Model-1'].parts['Part-2']
166 e = p.edges
167 pickedEdges1 = e.getSequenceFromMask(mask=('[#80 ]', ), )
168 p.seedEdgeByBias(biasMethod=SINGLE, end1Edges=pickedEdges1,
    ratio=5.0,
169     number=n_lig, constraint=FINER)
170 p = mdb.models['Model-1'].parts['Part-2']
171 e = p.edges
172 pickedEdges2 = e.getSequenceFromMask(mask=('[#40 ]', ), )
173 p.seedEdgeByBias(biasMethod=SINGLE, end2Edges=pickedEdges2,
    ratio=5.0,
174     number=n_lig)
175 p = mdb.models['Model-1'].parts['Part-2']
176 e = p.edges
177 pickedEdges = e.getSequenceFromMask(mask=('[#2 ]', ), )
178 p.seedEdgeByNumber(edges=pickedEdges, number=n_void)
179 p = mdb.models['Model-1'].parts['Part-2']
180 e = p.edges
181 pickedEdges = e.getSequenceFromMask(mask=('[#1 ]', ), )
182 p.seedEdgeByNumber(edges=pickedEdges, number=n_void)
183
184
185 if ellipsoid == 1 :
186     p = mdb.models['Model-1'].parts['Part-2']
187     pickedRegions = c.getSequenceFromMask(mask=('[#1 ]', ), )
188     p.deleteMesh(regions=pickedRegions)
189     p = mdb.models['Model-1'].parts['Part-2']
190     e = p.edges
191     pickedEdges1 = e.getSequenceFromMask(mask=('[#40 ]', ), )
192     p.seedEdgeByBias(biasMethod=SINGLE, end1Edges=pickedEdges1
        , ratio=5.0,
193     number=n_lig)
194     p = mdb.models['Model-1'].parts['Part-2']
195     e = p.edges
196     pickedEdges2 = e.getSequenceFromMask(mask=('[#2000 ]', ),
        )
```

```
197       p.seedEdgeByBias(biasMethod=SINGLE, end2Edges=pickedEdges2
              , ratio=5.0,
198       number=n_lig)
199       p = mdb.models['Model-1'].parts['Part-2']
200       e = p.edges
201       pickedEdges = e.getSequenceFromMask(mask=('[#20 ]', ), )
202       p.seedEdgeByNumber(edges=pickedEdges, number=sides)
203       p = mdb.models['Model-1'].parts['Part-2']
204       e = p.edges
205       pickedEdges = e.getSequenceFromMask(mask=('[#10 ]', ), )
206       p.seedEdgeByNumber(edges=pickedEdges, number=sides)
207
208       e = p.edges
209       pickedEdges = e.getSequenceFromMask(mask=('[#80 ]', ), )
210       p.seedEdgeByNumber(edges=pickedEdges, number=sides,
              constraint=FINER)
211       p = mdb.models['Model-1'].parts['Part-2']
212       e = p.edges
213       pickedEdges = e.getSequenceFromMask(mask=('[#1000 ]', ), )
214       p.seedEdgeByNumber(edges=pickedEdges, number=sides,
              constraint=FINER)
215
216
217 mdb.models['Model-1'].parts['Part-2'].generateMesh()
218
219
220 #### FULL MODEL
221
222 # Symmetry to extend the mesh
223 mdb.models['Model-1'].parts['Part-2'].PartFromMesh(copySets=
        True, name=
224       'Part-2-mesh')
225 mdb.models['Model-1'].Part(compressFeatureList=ON, mirrorPlane
        =XYPLANE, name=
226       'Part-3-mesh', objectToCopy=
227       mdb.models['Model-1'].parts['Part-2-mesh'])
228 mdb.models['Model-1'].Part(compressFeatureList=ON, mirrorPlane
        =YZPLANE, name=
229       'Part-4-mesh', objectToCopy=
230       mdb.models['Model-1'].parts['Part-2-mesh'])
231 mdb.models['Model-1'].Part(compressFeatureList=ON, mirrorPlane
        =XYPLANE, name=
232       'Part-5-mesh', objectToCopy=
```

```
233         mdb.models['Model-1'].parts['Part-4-mesh'])
234 mdb.models['Model-1'].Part(compressFeatureList=ON, mirrorPlane
      =XZPLANE, name=
235         'Part-6-mesh', objectToCopy=
236         mdb.models['Model-1'].parts['Part-2-mesh'])
237 mdb.models['Model-1'].Part(compressFeatureList=ON, mirrorPlane
      =XYPLANE, name=
238         'Part-7-mesh', objectToCopy=
239         mdb.models['Model-1'].parts['Part-6-mesh'])
240 mdb.models['Model-1'].Part(compressFeatureList=ON, mirrorPlane
      =YZPLANE, name=
241         'Part-8-mesh', objectToCopy=
242         mdb.models['Model-1'].parts['Part-6-mesh'])
243 mdb.models['Model-1'].Part(compressFeatureList=ON, mirrorPlane
      =XYPLANE, name=
244         'Part-9-mesh', objectToCopy=
245         mdb.models['Model-1'].parts['Part-8-mesh'])
246
247 #Assembly
248 mdb.models['Model-1'].rootAssembly.Instance(dependent=ON, name
      ='Part-2-mesh-1',
249         part=mdb.models['Model-1'].parts['Part-2-mesh'])
250 mdb.models['Model-1'].rootAssembly.Instance(dependent=ON, name
      ='Part-3-mesh-1',
251         part=mdb.models['Model-1'].parts['Part-3-mesh'])
252 mdb.models['Model-1'].rootAssembly.Instance(dependent=ON, name
      ='Part-4-mesh-1',
253         part=mdb.models['Model-1'].parts['Part-4-mesh'])
254 mdb.models['Model-1'].rootAssembly.Instance(dependent=ON, name
      ='Part-5-mesh-1',
255         part=mdb.models['Model-1'].parts['Part-5-mesh'])
256 mdb.models['Model-1'].rootAssembly.Instance(dependent=ON, name
      ='Part-6-mesh-1',
257         part=mdb.models['Model-1'].parts['Part-6-mesh'])
258 mdb.models['Model-1'].rootAssembly.Instance(dependent=ON, name
      ='Part-7-mesh-1',
259         part=mdb.models['Model-1'].parts['Part-7-mesh'])
260 mdb.models['Model-1'].rootAssembly.Instance(dependent=ON, name
      ='Part-8-mesh-1',
261         part=mdb.models['Model-1'].parts['Part-8-mesh'])
262 mdb.models['Model-1'].rootAssembly.Instance(dependent=ON, name
      ='Part-9-mesh-1',
263         part=mdb.models['Model-1'].parts['Part-9-mesh'])
```

```
264
265 mdb.models['Model-1'].rootAssembly.InstanceFromBooleanMerge(
        domain=MESH,
266         instances=(mdb.models['Model-1'].rootAssembly.instances['
              Part-2-mesh-1'],
267         mdb.models['Model-1'].rootAssembly.instances['Part-3-mesh
              -1'],
268         mdb.models['Model-1'].rootAssembly.instances['Part-4-mesh
              -1'],
269         mdb.models['Model-1'].rootAssembly.instances['Part-5-mesh
              -1'],
270         mdb.models['Model-1'].rootAssembly.instances['Part-6-mesh
              -1'],
271         mdb.models['Model-1'].rootAssembly.instances['Part-7-mesh
              -1'],
272         mdb.models['Model-1'].rootAssembly.instances['Part-8-mesh
              -1'],
273         mdb.models['Model-1'].rootAssembly.instances['Part-9-mesh
              -1']),
274         mergeNodes=BOUNDARY_ONLY, name='Part-1',
              nodeMergingTolerance=1e-06,
275         originalInstances=SUPPRESS)
276
277 ######################################
278 #DEFINE MATERIAL AND SECTION
279 ######################################
280 mdb.models['Model-1'].Material(name='CRYSTAL')
281
282 mdb.models['Model-1'].materials['CRYSTAL'].Depvar(n=125)
283
284 mdb.models['Model-1'].materials['CRYSTAL'].UserMaterial(type=
        MECHANICAL, unsymm=ON, mechanicalConstants=(C11,C22,C44
        ,0.,0.,0.,0.,0.,
285         0.,0.,0.,0.,0.,0.,0.,0.,
286         0.,0.,0.,0.,0.,0.,0.,0.,1.,0.,0.,
287         0.,0.,0.,0.,0.,1.,1.,1.,1.,1.,0.,0.,0.,
288         0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,
289         crystal_orientation1[0],crystal_orientation1[1],
              crystal_orientation1[2],crystal_orientation1[3],
              crystal_orientation1[4],crystal_orientation1[5], #
              local/global crystal orientation
290         0.,0.,
291         crystal_orientation2[0],crystal_orientation2[1],
```

```
            crystal_orientation2[2],crystal_orientation2[3],
            crystal_orientation2[4],crystal_orientation2[5], #
            local/global crystal orientation
292         0.,0., n,gammadot,0.,0.,0.,0.,0.,0.,
293         0.,0.,0.,0.,0.,0.,0.,0.,
294         0.,0.,0.,0.,0.,0.,0.,0.,0.0,109.5,tau_0,
295         0.,0.,0.,0.,0., 0.0,0.0,0.,0.,0.,0.,0.,0.,
296         0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,0.,
297         0.,0.,0.,0.,0.,0.,0.,0.,
298         0.,0.,0.,0.,0.,0.,0.,0., 0.5,1.,0.,0.,
299         0.,0.,0.,0., 1.,10.,1e-05,0.,0.,0.,0.,0.))
300
301 mdb.models['Model-1'].HomogeneousSolidSection(material='
    CRYSTAL', name=
302       'Section-1', thickness=None)
303
304 mdb.models['Model-1'].parts['Part-1'].SectionAssignment(offset
    =0.0,
305       offsetField='', offsetType=MIDDLE_SURFACE, region=Region(
306       elements=mdb.models['Model-1'].parts['Part-1'].elements),
            sectionName='Section-1',
307       thicknessAssignment=FROM_SECTION)
308
309 ##################################################
310 #CREATE PERIODIC BOUNDARY CONDITIONS
311 ##################################################
312 mdb.models['Model-1'].rootAssembly.Set(nodes=(
313       mdb.models['Model-1'].rootAssembly.instances['Part-1-1'].
            nodes,), name='PBC') ## change this to the set of nodes
            where the PCB will be applied!!
314
315 (CoorFixNode,NameRef1, NameRef2,NameRef3)=PeriodicBound3D(mdb,
    'Model-1','PBC',[(1.0,0.0,0.0),(0.0,1.0,0.0),(0.0,0.0,1.0)
    ],)
316
317 ##################################################
318 #CREATE STEP AND APPLY BC
319 ##################################################
320 mdb.models['Model-1'].ViscoStep(name='Step-1', timePeriod=
    total_time, nlgeom=ON, previous='Initial', initialInc
    =0.0001, maxNumInc=1000, minInc=0.000001, maxInc=total_time,
     cetol=0.00001)
321 #Apply boundary conditions on reference nodes
```

```
322 DefMat=[(UNSET,0.0,0.0),(0.0,UNSET,0.0), (0.0,0.0,UNSET)]
323 mdb.models['Model-1'].DisplacementBC(amplitude=UNSET,
    createStepName='Step-1',
324     distributionType=UNIFORM, fieldName='', fixed=OFF,
        localCsys=None, name=
325     'BC-REF-1', region=Region(referencePoints=(
326     mdb.models['Model-1'].rootAssembly.instances[NameRef1].
        referencePoints[1],
327     )), u1=DefMat[0][0], u2=DefMat[0][1], u3=DefMat[0][2], ur1
        =UNSET, ur2=UNSET, ur3=UNSET)
328 mdb.models['Model-1'].DisplacementBC(amplitude=UNSET,
    createStepName='Step-1',
329     distributionType=UNIFORM, fieldName='', fixed=OFF,
        localCsys=None, name=
330     'BC-REF-2', region=Region(referencePoints=(
331     mdb.models['Model-1'].rootAssembly.instances[NameRef2].
        referencePoints[1],
332     )), u1=DefMat[1][0], u2=DefMat[1][1], u3=DefMat[1][2], ur1
        =UNSET, ur2=UNSET, ur3=UNSET)
333 mdb.models['Model-1'].DisplacementBC(amplitude=UNSET,
    createStepName='Step-1',
334     distributionType=UNIFORM, fieldName='', fixed=OFF,
        localCsys=None, name=
335     'BC-REF-3', region=Region(referencePoints=(
336     mdb.models['Model-1'].rootAssembly.instances[NameRef3].
        referencePoints[1],
337     )), u1=DefMat[2][0], u2=DefMat[2][1], u3=DefMat[2][2], ur1
        =UNSET, ur2=UNSET, ur3=UNSET)
338
339 mdb.models['Model-1'].DisplacementBC(amplitude=UNSET,
    createStepName='Step-1',
340     distributionType=UNIFORM, fieldName='', fixed=OFF,
        localCsys=None, name=
341     'BC-FIXNODE', region=Region(
342     nodes=mdb.models['Model-1'].rootAssembly.instances['Part
        -1-1'].nodes.getByBoundingSphere(center=CoorFixNode,
        radius=0.001)), u1=0.0, u2=0.0, u3=0.0, ur1=UNSET, ur2=
        UNSET, ur3=UNSET)
343
344
345 ##########################
346 ## APPLY LOADING
347 ##########################
```

```
348
349 #Create the user defined amplitudes
350 mdb.models['Model-1'].UserAmplitude(name='LOADAMP1',
        numVariables=0,
351       timeSpan=STEP)
352 mdb.models['Model-1'].UserAmplitude(name='LOADAMP2',
        numVariables=0,
353       timeSpan=STEP)
354 mdb.models['Model-1'].UserAmplitude(name='LOADAMP3',
        numVariables=0,
355       timeSpan=STEP)
356
357 #loading
358 mdb.models['Model-1'].ConcentratedForce(name='Load-1',
        createStepName='Step-1',
359       distributionType=UNIFORM, region=Region(referencePoints=(
            mdb.models['Model-1'].rootAssembly.instances[NameRef1].
            referencePoints[1], )), cf1=F11, cf2=F12, cf3=F13,
            amplitude=UNSET)
360
361 mdb.models['Model-1'].ConcentratedForce(name='Load-2',
        createStepName='Step-1',
362       distributionType=UNIFORM, region=Region(referencePoints=(
            mdb.models['Model-1'].rootAssembly.instances[NameRef2].
            referencePoints[1], )), cf1=F21, cf2=F22, cf3=F23,
            amplitude=UNSET)
363
364 mdb.models['Model-1'].ConcentratedForce(name='Load-3',
        createStepName='Step-1',
365       distributionType=UNIFORM, region=Region(referencePoints=(
            mdb.models['Model-1'].rootAssembly.instances[NameRef3].
            referencePoints[1], )), cf1=F31, cf2=F32, cf3=F33,
            amplitude=UNSET)
366
367 #assign UAMP subroutine to the 3 loads
368 mdb.models['Model-1'].loads['Load-1'].setValues(amplitude='
    LOADAMP1')
369 mdb.models['Model-1'].loads['Load-2'].setValues(amplitude='
    LOADAMP2')
370 mdb.models['Model-1'].loads['Load-3'].setValues(amplitude='
    LOADAMP3')
371
372
```

```
373 ##########################################################
374 #### SENSOR OUTPUT REQUEST & FIELD OUTPUTS
375 ##########################################################
376
377 mdb.models['Model-1'].fieldOutputRequests['F-Output-1'].
        setValues(variables=(
378       'S', 'PE', 'ER', 'PEEQ', 'PEMAG', 'CE', 'CEEQ', 'CEMAG', '
              LE', 'U', 'RF',
379       'CF', 'CSTRESS', 'CDISP', 'SVOL', 'EVOL', 'ESOL', 'IVOL',
              'STH',
380       'COORD'))
381
382 #sensors
383 regionDef=mdb.models['Model-1'].rootAssembly.sets['RefPoint-0'
        ]
384 mdb.models['Model-1'].HistoryOutputRequest(name='X1',
        createStepName='Step-1',
385       variables=('U1', ), region=regionDef, sectionPoints=
              DEFAULT,
386       rebar=EXCLUDE, sensor=ON)
387 regionDef=mdb.models['Model-1'].rootAssembly.sets['RefPoint-1'
        ]
388 mdb.models['Model-1'].HistoryOutputRequest(name='Y1',
        createStepName='Step-1',
389       variables=('U2', ), region=regionDef, sectionPoints=
              DEFAULT,
390       rebar=EXCLUDE, sensor=ON)
391 regionDef=mdb.models['Model-1'].rootAssembly.sets['RefPoint-2'
        ]
392 mdb.models['Model-1'].HistoryOutputRequest(name='Z1',
        createStepName='Step-1',
393       variables=('U3', ), region=regionDef, sectionPoints=
              DEFAULT,
394       rebar=EXCLUDE, sensor=ON)
395
396 ############################
397 #### JOB CREATION
398 ############################
399
400 mdb.Job(name=ODBname, model='Model-1', description='', type=
        ANALYSIS,
401       atTime=None, waitMinutes=0, waitHours=0, queue=None,
              memory=90,
```

```
402         memoryUnits=PERCENTAGE, getMemoryFromAnalysis=True,
403         explicitPrecision=SINGLE, nodalOutputPrecision=SINGLE,
                echoPrint=OFF,
404         modelPrint=OFF, contactPrint=OFF, historyPrint=OFF,
405         userSubroutine='C:\\temp\\UMAT_UAMP_REFPOINTS.for',
                scratch='',
406         resultsFormat=ODB, multiprocessingMode=DEFAULT, numCpus=4,
407         numDomains=4, numGPUs=0)
408
409
410 # Auto-launch of the ABAQUS job
411 #mdb.jobs['Tri3_L-1_100'].submit(consistencyChecking=OFF)
```

Appendix B

```
### Author: Simon Carrillo Segura
### University of Pennsylvania, Philadelphia
### Department of Mechanical Engineering and Applied Mechanics (MEAM)
#
#################################################
#PERIODIC BOUNDARY CONDITIONS IN 3D
#mdb: Model.
#NameModel: A string with the name of your model.
#NameSet: A string with the name of your set
#LatticeVec:An array with the lattice vectors, for example [(1.0,0.0,0.0),(0.0,1.0,0.0),(0.0,0.0,1.0)] for a cubic lattice
#################################################

def PeriodicBound3D(mdb,NameModel,NameSet,LatticeVec):
    import time
    start1 = time.time()

    from part import THREE_D, DEFORMABLE_BODY
    #Create reference parts and assemble
    NameRef1='RefPoint-0'; NameRef2='RefPoint-1'; NameRef3='RefPoint-2'
    mdb.models[NameModel].Part(dimensionality=THREE_D, name=NameRef1, type=
        DEFORMABLE_BODY)
    mdb.models[NameModel].parts[NameRef1].ReferencePoint(point=(0.0, 0.0, 0.0))
    mdb.models[NameModel].Part(dimensionality=THREE_D, name=NameRef2, type=
        DEFORMABLE_BODY)
    mdb.models[NameModel].parts[NameRef2].ReferencePoint(point=(0.0, 0.0, 0.0))
    mdb.models[NameModel].Part(dimensionality=THREE_D, name=NameRef3, type=
        DEFORMABLE_BODY)
    mdb.models[NameModel].parts[NameRef3].ReferencePoint(point=(0.0, 0.0, 0.0))
    mdb.models[NameModel].rootAssembly.Instance(dependent=ON, name=NameRef1,
```

```
30          part=mdb.models[NameModel].parts[NameRef1])
31      mdb.models[NameModel].rootAssembly.Instance(dependent=ON,
            name=NameRef2,
32          part=mdb.models[NameModel].parts[NameRef2])
33      mdb.models[NameModel].rootAssembly.Instance(dependent=ON,
            name=NameRef3,
34          part=mdb.models[NameModel].parts[NameRef3])
35
36      #Create set of reference points
37      mdb.models[NameModel].rootAssembly.Set(name=NameRef1,
            referencePoints=(
38          mdb.models[NameModel].rootAssembly.instances[NameRef1
                ].referencePoints[1],))
39      mdb.models[NameModel].rootAssembly.Set(name=NameRef2,
            referencePoints=(
40          mdb.models[NameModel].rootAssembly.instances[NameRef2
                ].referencePoints[1],))
41      mdb.models[NameModel].rootAssembly.Set(name=NameRef3,
            referencePoints=(
42          mdb.models[NameModel].rootAssembly.instances[NameRef3
                ].referencePoints[1],))
43      end1 = time.time()
44      print end1 - start1
45      start2 = time.time()
46      #Get all nodes
47      nodesAll=mdb.models[NameModel].rootAssembly.sets[NameSet].
            nodes
48      nodesAllCoor=[]
49      for nod in mdb.models[NameModel].rootAssembly.sets[NameSet
            ].nodes:
50          nodesAllCoor.append(nod.coordinates)
51      end2 = time.time()
52      print end2 - start2
53      start3 = time.time()
54      repConst=0
55      #Find periodically located nodes and apply equation
            constraints
56      ranNodes=range(0,len(nodesAll))      #Index array of nodes
            not used in equations constraint
57      print len(nodesAll)
58      for repnod1 in range(0,len(nodesAll)):
59          stop=False                       #Stop will become true
                when equation constraint is made between nodes
```

```
60          Coor1=nodesAllCoor[repnod1]                 #Coordinates
               of Node 1
61          for repnod2 in ranNodes:          #Loop over all
               available nodes
62           Coor2=nodesAllCoor[repnod2] #Coordinates of Node 2
63           for comb in range(0,len(LatticeVec)):       #Check
                 if nodes are located exactly the vector lattice
                 apart
64               if int(1000.0*(LatticeVec[comb][0]-Coor2[0]+
                    Coor1[0]))==0 and int(1000.0*(LatticeVec[
                    comb][1]-Coor2[1]+Coor1[1]))==0 and int
                    (1000.0*(LatticeVec[comb][2]-Coor2[2]+Coor1
                    [2]))==0:
65                  #Correct combination found
66                  #Create sets for use in equations
                       constraints
67                  mdb.models[NameModel].rootAssembly.Set(
                       name='Node-1-'+str(repConst), nodes=
68                     mdb.models[NameModel].rootAssembly.sets
                          [NameSet].nodes[repnod1:repnod1+1])
69                  mdb.models[NameModel].rootAssembly.Set(
                       name='Node-2-'+str(repConst), nodes=
70                     mdb.models[NameModel].rootAssembly.sets
                          [NameSet].nodes[repnod2:repnod2+1])
71                  #Create equations constraints for each dof
72                  for Dim1 in [1,2,3]:
73                     mdb.models[NameModel].Equation(name='
                          PerConst'+str(Dim1)+'-'+str(repConst)
                          ,
74                               terms=((1.0,'Node-1-'+str(
                                   repConst),Dim1),(-1.0, 'Node
                                   -2-'+str(repConst),Dim1) ,
75                             (1.0, 'RefPoint-'+str(comb),Dim1)
                                ))
76                  repConst=repConst+1 #Increase integer for
                       naming equation constraint
77                  ranNodes.remove(repnod1)#Remove used node
                       from available list
78                  stop=True              #Don't look further,
                       go to following node.
79                  break
80           if stop:
81               break
```

```
82      end3 = time.time()
83      print end3 - start3
84      #Return coordinates of free node so that it can be fixed
85      return (nodesAll[ranNodes[0]].coordinates, NameRef1,
            NameRef2,NameRef3)
```

Appendix C

```
# -*- coding: mbcs -*-
# Output Request file for RVE PBC
### Author: Simon Carrillo Segura
### University of Pennsylvania, Philadelphia
### Department of Mechanical Engineering and Applied Mechanics (MEAM)

from abaqus import *
from abaqusConstants import *
import __main__

import section
import regionToolset
import displayGroupMdbToolset as dgm
import part
import material
import assembly
import step
import interaction
import load
import mesh
import optimization
import job
import sketch
import visualization
import xyPlot
import displayGroupOdbToolset as dgo
import connectorBehavior

##############
##INPUTS
##############

ODBname='C:/temp/AXC_001_w2.odb' # name of the ODB to extract values from
n_steps= 240 # number of steps to extract

#############################################################
######### DO NOT TOUCH BELOW HERE ###########
#############################################################
```

```
39
40 #openning ODB session
41 session.mdbData.summary()
42 o1 = session.openOdb(name=ODBname)
43 session.viewports['Viewport: 1'].setValues(displayedObject=o1)
44
45 ## Extract elemental volume of the cell per time step
46
47 for x in range(0,n_steps):
48         session.viewports['Viewport: 1'].odbDisplay.setFrame(
              step=0, frame=x)
49         odb = session.odbs[ODBname]
50         session.fieldReportOptions.setValues(printTotal=OFF,
              printMinMax=OFF)
51         session.writeFieldReport(fileName='vol.rpt', append=ON
              ,
52           sortItem='Element Label', odb=odb, step=0, frame=x
              ,
53           outputPosition=WHOLE_ELEMENT, variable=(('EVOL',
              WHOLE_ELEMENT), ))
54
55
56 ## Extract logarithmic strains components in the centroid per
     time step
57
58 for x in range(0,n_steps):
59         session.viewports['Viewport: 1'].odbDisplay.setFrame(
              step=0, frame=x)
60         session.fieldReportOptions.setValues(printTotal=OFF,
              printMinMax=OFF)
61         session.writeFieldReport(fileName='strains.rpt',
              append=ON,
62           sortItem='Element Label', odb=odb, step=0, frame=x
              ,
63           outputPosition=ELEMENT_CENTROID, variable=(('LE',
              INTEGRATION_POINT, ((
64           COMPONENT, 'LE11'), (COMPONENT, 'LE22'), (
              COMPONENT, 'LE33'), (
65           COMPONENT, 'LE12'), (COMPONENT, 'LE13'), (
              COMPONENT, 'LE23'), )), ))
66
67 ## Extract strains rate components in the centroid per time
     step
```

```
68
69 for x in range(0,n_steps):
70         session.viewports['Viewport: 1'].odbDisplay.setFrame(
               step=0, frame=x)
71         session.fieldReportOptions.setValues(printTotal=OFF,
               printMinMax=OFF)
72         session.writeFieldReport(fileName='strains_rate.rpt',
               append=ON,
73             sortItem='Element Label', odb=odb, step=0, frame=x
                   ,
74             outputPosition=ELEMENT_CENTROID, variable=(('ER',
                   INTEGRATION_POINT, ((
75             COMPONENT, 'ER11'), (COMPONENT, 'ER22'), (
                   COMPONENT, 'ER33'), (
76             COMPONENT, 'ER12'), (COMPONENT, 'ER13'), (
                   COMPONENT, 'ER23'), )), ))
77
78
79 ## Extract sensors information to compute V_cell (FULL CUBE)
80
81 xy1 = xyPlot.XYDataFromHistory(odb=odb,
82     outputVariableName='Spatial displacement: U1 PI: REFPOINT
           -0 Node 1 in NSET REFPOINT-0',
83     suppressQuery=True, __linkedVpName__='Viewport: 1')
84 c1 = session.Curve(xyData=xy1)
85 xy2 = xyPlot.XYDataFromHistory(odb=odb,
86     outputVariableName='Spatial displacement: U2 PI: REFPOINT
           -1 Node 1 in NSET REFPOINT-1',
87     suppressQuery=True, __linkedVpName__='Viewport: 1')
88 c2 = session.Curve(xyData=xy2)
89 xy3 = xyPlot.XYDataFromHistory(odb=odb,
90     outputVariableName='Spatial displacement: U3 PI: REFPOINT
           -2 Node 1 in NSET REFPOINT-2',
91     suppressQuery=True, __linkedVpName__='Viewport: 1')
92 c3 = session.Curve(xyData=xy3)
93 xyp = session.XYPlot('chart')
94 chartName = xyp.charts.keys()[0]
95 chart = xyp.charts[chartName]
96 chart.setValues(curvesToPlot=(c1, c2, c3, ), )
97 session.viewports['Viewport: 1'].setValues(displayedObject=xyp
       )
98 x0 = session.xyDataObjects['_temp_1']
99 x1 = session.xyDataObjects['_temp_2']
```

```
100 x2 = session.xyDataObjects['_temp_3']
101 session.writeXYReport(fileName='volume_cell.rpt', xyData=(x0,
        x1, x2))
```

Appendix D

```
clc
clear all

%% Postprocessing subroutine for RVE problem
%%% Author: Simon Carrillo Segura
%%% Department of Mechanical Engineering and Applied Mechanics
%%% University of Pennsylvania, Philadelphia

%% Load inputs
f_0 = 0.01; % initial porosity level
n_elements =6992;  % number of mesh elements in the FEM model

n_steps =232; % number of steps extracted from the ABAQUS
    result file
ODB_name = 'AXC_001_w2';

%% strains input reading

for i = 1:n_steps
    AA = importdata('strains.rpt', ' ' ,19+(i-1)*(n_elements
        +21));

    strains(i) = {AA.data};
end

%% strains rate

for i = 1:n_steps
    EE = importdata('strains_rate.rpt', ' ' ,19+(i-1)*(
        n_elements+21));    % ,19+(i-1)*7493);    ,19+(i-1)*2469);
         %,19+(i-1)*955);

    strain_rate(i) = {EE.data};
end

%% elemental volumes input reading

for i = 1:n_steps
    BB = importdata('vol.rpt', ' ' ,19+(i-1)*(n_elements+21));

```

```
37      vol(i) = {BB.data};
38  end
39
40  %% Creep strain computation (per time step)
41
42  for i = 1:n_steps
43
44      strains11_vol(i,:) = strains{i}(:,2).*vol{i}(:,2);
45      strains11(i) = sum(strains11_vol(i,:)) / sum(vol{i}(:,2));
46
47      strains22_vol(i,:) = strains{i}(:,3).*vol{i}(:,2);
48      strains22(i) = sum(strains22_vol(i,:)) / sum(vol{i}(:,2));
49
50      strains33_vol(i,:) = strains{i}(:,4).*vol{i}(:,2);
51      strains33(i) = sum(strains33_vol(i,:)) / sum(vol{i}(:,2));
52
53      strains12_vol(i,:) = strains{i}(:,5).*vol{i}(:,2);
54      strains12(i) = sum(strains12_vol(i,:)) / sum(vol{i}(:,2));
55
56      strains23_vol(i,:) = strains{i}(:,6).*vol{i}(:,2);
57      strains23(i) = sum(strains23_vol(i,:)) / sum(vol{i}(:,2));
58
59      strains13_vol(i,:) = strains{i}(:,7).*vol{i}(:,2);
60      strains13(i) = sum(strains13_vol(i,:)) / sum(vol{i}(:,2));
61
62      true_strains(i) = sqrt(2)/3 * sqrt( (strains11(i)-
            strains22(i)).^2 + (strains22(i)-strains33(i)).^2 + (
            strains11(i)-strains33(i)).^2 +3/2*(strains12(i).^2 +
            strains23(i).^2 + strains13(i).^2))
63
64  end
65
66  %% strain rate computation (per time step)
67
68  for i = 1:n_steps
69
70      strain_rate11_vol(i,:) = strain_rate{i}(:,2).*vol{i}(:,2);
71      strain_rate11(i) = sum(strain_rate11_vol(i,:)) / sum(vol{i
            }(:,2));
72
73      strain_rate22_vol(i,:) = strain_rate{i}(:,3).*vol{i}(:,2);
74      strain_rate22(i) = sum(strain_rate22_vol(i,:)) / sum(vol{i
            }(:,2));
```

```
75
76      strain_rate33_vol(i,:) = strain_rate{i}(:,4).*vol{i}(:,2);
77      strain_rate33(i) = sum(strain_rate33_vol(i,:)) / sum(vol{i
            }(:,2));
78
79      strain_rate12_vol(i,:) = strain_rate{i}(:,5).*vol{i}(:,2);
80      strain_rate12(i) = sum(strain_rate12_vol(i,:)) / sum(vol{i
            }(:,2));
81
82      strain_rate23_vol(i,:) = strain_rate{i}(:,6).*vol{i}(:,2);
83      strain_rate23(i) = sum(strain_rate23_vol(i,:)) / sum(vol{i
            }(:,2));
84
85      strain_rate13_vol(i,:) = strain_rate{i}(:,7).*vol{i}(:,2);
86      strain_rate13(i) = sum(strain_rate13_vol(i,:)) / sum(vol{i
            }(:,2));
87
88      true_strain_rate(i) = sqrt(2)/3 * sqrt( (strain_rate11(i)-
            strain_rate22(i)).^2 + (strain_rate22(i)-strain_rate33(i
            )).^2 + (strain_rate33(i)-strain_rate11(i)).^2  +3/2*(
            strain_rate12(i).^2 + strain_rate23(i).^2 +
            strain_rate13(i).^2))
89
90  end
91
92  %% Cell volume vs time
93
94  % reference points displacement input reading
95  CC = importdata('volume_cell.rpt', ' ' ,3);
96
97  for i=1:4
98
99      vol_cell(:,i) = CC.data(:,i);
100
101 end
102
103     for i=1:4
104
105         vol_cell(:,i) = CC.data(:,i);
106
107     end
108
109     for i= 1: n_steps
```

```
110
111         V_cell(i) = (1+vol_cell(i,2)).*(1+vol_cell(i,3)).*(1+
                vol_cell(i,4)); %ref points volume
112
113     end
114
115 %% Volume fraction computation (per time step)
116
117 for i=1:n_steps
118
119     v_cell(i,1) = vol_cell(i,1);
120     v_cell(i,2) = V_cell(i);
121
122 end
123
124 for i=1:n_steps
125
126     v_matrix(i) = sum(vol{i}(:,2));
127
128 end
129
130 for i=1:n_steps
131
132     f(i) = (v_cell(i,2) - v_matrix(i))/v_cell(i,2);
133
134 end
135
136 vol_fraction = f/f(1)
137
138
139 %% write results to file
140
141 A = [v_cell(:,1)'; true_strains];
142 fileID = fopen(['Creep_strain_',ODB_name,'.txt'],'w');
143 fprintf(fileID,'%6s %12s\n','Time','Equivalent Creep strains')
        ;
144 fprintf(fileID,'%2.12f %2.12f\n',A);
145 fclose(fileID);
146
147 B = [true_strains; vol_fraction];
148 fileID = fopen(['Porosity_',ODB_name,'.txt'],'w');
149 fprintf(fileID,'%6s %12s\n','Equivalent Creep Strain','
        Porosity');
```

```
150 fprintf(fileID,'%2.12f %2.12f\n',B);
151 fclose(fileID);
152
153 C = [true_strains; true_strain_rate];
154 fileID = fopen(['Strain_rate_',ODB_name,'.txt'],'w');
155 fprintf(fileID,'%6s %12s\n','Equivalent Creep Strain','Strain
      Rate');
156 fprintf(fileID,'%2.12f %2.12f\n',C);
157 fclose(fileID);
```

Appendix E

```
1  c Author: Simon Carrillo Segura
2  c University of Pennsylvania, Philadelphia
3  c Department of Mechanical Engineering and Applied Mechanics (
       MEAM)
4  c
5  c
6  c     user amplitude subroutine
7        Subroutine uamp(
8  c           passed in for information and state variables
9        *     ampName, time, ampValueOld, dt, nProps, props,
              nSvars, svars,
10       *     lFlagsInfo, nSensor, sensorValues, sensorNames,
11       *     jSensorLookUpTable,
12 c           to be defined
13       *     ampValueNew,
14       *     lFlagsDefine,
15       *     AmpDerivative, AmpSecDerivative, AmpIncIntegral,
16       *     AmpIncDoubleIntegral)
17
18       include 'aba_param.inc'
19
20 c     svars - additional state variables, similar to (V)UEL
21       dimension sensorValues(nSensor), svars(nSvars), props(
              nProps)
22       character*80 sensorNames(nSensor)
23       character*80 ampName
24
25 c     time indices
26       parameter (iStepTime         = 1,
27       *           iTotalTime       = 2,
28       *           nTime            = 2)
29 c     flags passed in for information
30       parameter (iInitialization   = 1,
31       *           iRegularInc      = 2,
32       *           nFlagsInfo       = 2)
33 c     optional flags to be defined
34       parameter (iComputeDeriv       = 1,
35       *           iComputeSecDeriv   = 2,
36       *           iComputeInteg      = 3,
37       *           iComputeDoubleInteg = 4,
```

```
38      *            iStopAnalysis        = 5,
39      *            iConcludeStep        = 6,
40      *            nFlagsDefine         = 6)
41       dimension time(nTime), lFlagsInfo(nFlagsInfo),
42      *           lFlagsDefine(nFlagsDefine)
43       dimension jSensorLookUpTable(*)
44
45          VALUE1 = GetSensorValue('X1', jSensorLookUpTable,
                sensorValues)
46          VALUE2 = GetSensorValue('Y1', jSensorLookUpTable,
                sensorValues)
47          VALUE3 = GetSensorValue('Z1', jSensorLookUpTable,
                sensorValues)
48
49          ampValueOld = 1d0
50
51          if (ampName(1:8) .eq. 'LOADAMP1') then
52
53  C       User code to compute  ampValue = F(sensors)
54          if (lFlagsInfo(iInitialization).eq.1) then
55             ampValueNew       = ampValueOld
56          else
57  c          Example: f(t) = t
58             tim = time(iStepTime)
59             tStart = tim - dt
60             tEnd   = tim
61
62             ampValueNew = (1+VALUE2)*(1+VALUE3)
63
64
65       end if
66
67              end if
68
69                          if (ampName(1:8) .eq. 'LOADAMP2')
                               then
70
71  C       User code to compute  ampValue = F(sensors)
72          if (lFlagsInfo(iInitialization).eq.1) then
73             ampValueNew       = ampValueOld
74          else
75  c          Example: f(t) = t
76             tim = time(iStepTime)
```

```
77                tStart = tim - dt
78                tEnd   = tim
79
80              ampValueNew = (1+VALUE1)*(1+VALUE3)
81
82
83          end if
84
85                end if
86
87                              if (ampName(1:8) .eq. 'LOADAMP3')
                                   then
88
89  C           User code to compute  ampValue = F(sensors)
90              if (lFlagsInfo(iInitialization).eq.1) then
91                 ampValueNew       = ampValueOld
92              else
93  c              Example: f(t) = t
94                 tim = time(iStepTime)
95                 tStart = tim - dt
96                 tEnd   = tim
97
98                 ampValueNew = (1+VALUE1)*(1+VALUE2)
99
100
101         end if
102
103               end if
104         return
105         end
106 c
```