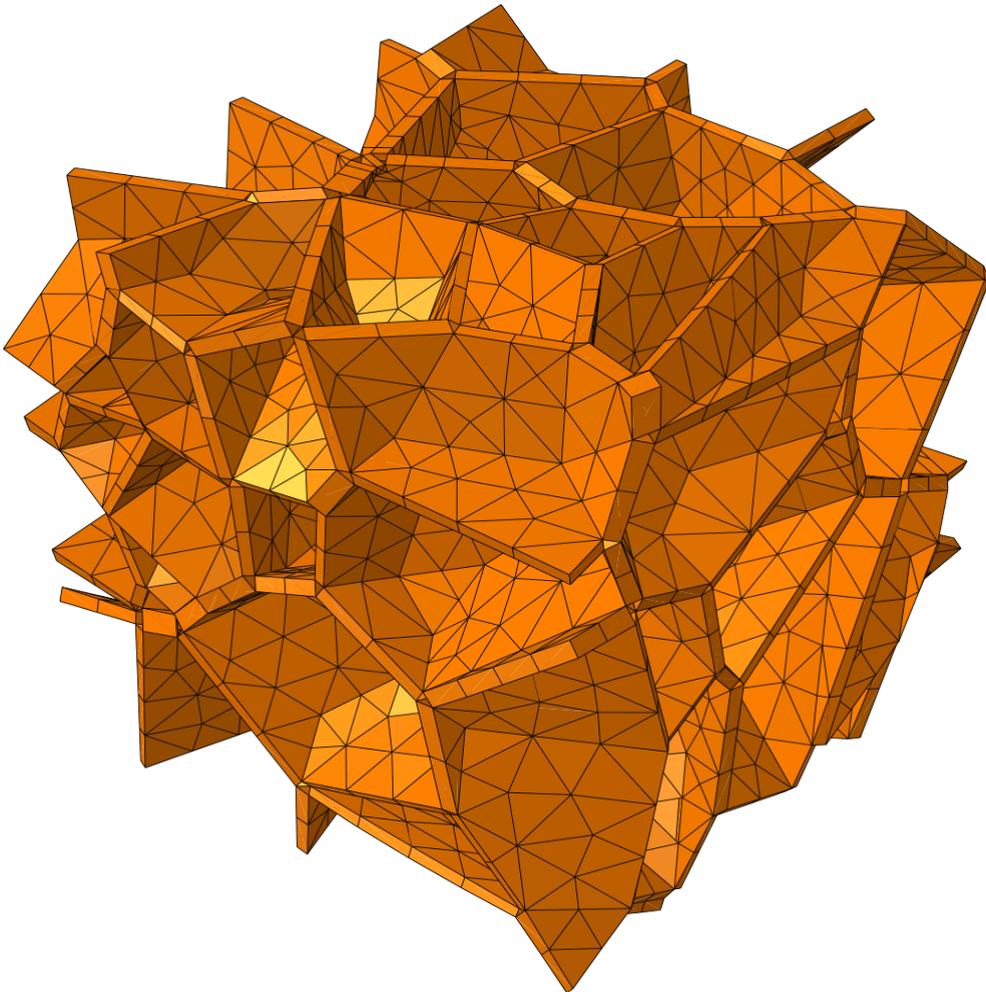


CHALMERS



Modeling of three dimensional microstructures including grain boundary mechanisms

Master's thesis in Engineering Mathematics

KRISTOFFER CARLSSON

Department of Applied Mechanics
Material and Computational Mechanics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2013
Master's thesis 2013:51

MASTER'S THESIS IN ENGINEERING MATHEMATICS

Modeling of three dimensional microstructures including grain boundary
mechanisms

KRISTOFFER CARLSSON

Department of Applied Mechanics
Material and Computational Mechanics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2013

Modeling of three dimensional microstructures including grain boundary mechanisms
KRISTOFFER CARLSSON

© KRISTOFFER CARLSSON, 2013

Master's thesis 2013:51
ISSN 1652-8557
Department of Applied Mechanics
Material and Computational Mechanics
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone: +46 (0)31-772 1000

Cover:

Cohesive zone elements between grains. The actual grains have been hidden and the cohesive elements been given a non zero width.

Chalmers Reproservice
Gothenburg, Sweden 2013

Modeling of three dimensional microstructures including grain boundary mechanisms
Master's thesis in Engineering Mathematics
KRISTOFFER CARLSSON
Department of Applied Mechanics
Material and Computational Mechanics
Chalmers University of Technology

ABSTRACT

The aim of this thesis is to give a method to include grain boundary mechanisms into models of microstructures containing grains of polyhedral shape. First, microstructures of this type is generated by Voronoi tessellation. Then FE analyses simulating uniaxial tensile testing of cubic RVEs with these microstructures were performed for a crystal plasticity model with grains of different sizes. The results of these analyses showed a decrease in the variance of the stress response when the number of grains was increased. An increase of the stress at the same strain could also be seen for microstructures with higher number of grains. The next part describes the method used to include grain boundary mechanisms in the model. This is done by inserting cohesive elements between the grains. The behavior of the cohesive elements is defined by a traction separation law. FE analyses on models containing cohesive elements were performed using both an implicit and explicit time integration technique to solve the FE equations. It was found that when using the implicit solver it was difficult to complete the analyses due to convergence problems. The explicit solver did not have these problems and gave the same results as the implicit solver at the convergence region for the implicit solver. Last, a traction separation law was adapted from atomistic simulations that represent the grain boundary mechanism between tungsten carbide infiltrated by cobalt. The three dimensional grain structure model was used to estimate the yield stress of this material. The results indicated a yield stress around 350 MPa but more extensive analyses are required to give confidence to this result.

Keywords: 3D microstructure, Voronoi tessellation, Cohesive zones, Grain boundary traction separation law.

PREFACE

This thesis concludes my Master of Science education in Engineering Mathematics at Chalmers University of Technology. The thesis was worked on throughout the spring of 2013 which was my tenth semester at Chalmers.

I would first want to greatly thank my supervisor Professor Magnus Ekh at the Division of Material and Computational Mechanics, Chalmers. He has been the person that I have approached with any issues I have had regarding the writing of the thesis. He has also given a lot of feedback regarding the language and grammar used. I would also like to thank Professor Göran Wahnström at the Division of Materials and Surface theory, Chalmers, for contacting me about the opportunity to do this thesis.

Thanks also go out to my friends and family who have supported me during not only the writing of this thesis, but during all of my five years studying at Chalmers. Special thanks goes to Martin Gren, Olof Ahlén, Petter Säterskog and Patric Holmvall who have been my main collaborators regarding school work during these five years. I would also like to give thanks to my brother Rikard Carlsson for the countless hours of Skype conversations that have been a constant source of laughter.

Before this thesis I had not had very much experience with Computational Mechanics. These six months have therefore exposed me to many new methods and tools that will certainly be of use later in my academic career. I have now been given the opportunity to continue my research performed in this thesis by working at the department this summer. I very much look forward to this and it feels like a great way to start off after completing the thesis.

Kristoffer Carlsson, Gothenburg, 2013.

CONTENTS

Abstract	i
Preface	iii
Contents	v
1 Introduction	1
2 Microstructure generation	2
3 Meshing of microstructure	3
4 Crystal plasticity in duplex steel	5
4.1 Loading case	5
4.2 Results	6
5 Grain boundary mechanisms	10
5.1 Cohesive elements	10
5.1.1 Undamaged behavior	11
5.1.2 Damage initiation	11
5.1.3 Damage evolution	12
5.1.4 Insertion of cohesive elements into mesh	13
5.2 FE analyses with cohesive elements	16
5.3 FE analyses of tungsten carbide	19
5.3.1 Test case	19
5.3.2 Full analysis	20
6 Conclusions	21
Appendices	25
A Introduction to finite element method	25
A.1 Galerkin method	25
A.2 FEM for nonlinear and linear elasticity	27

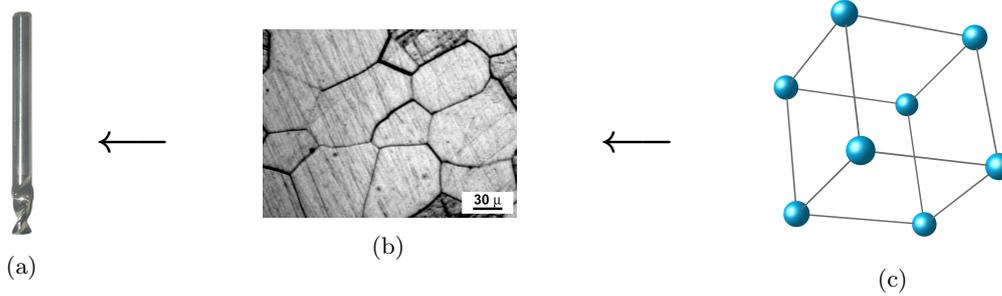


Figure 1.0.1: *Material properties at a certain scale are statistical averages of heterogeneous physical properties on a lower scale. The figure shows a polycrystalline material at its a) macro scale, b) meso scale[18] and c) micro scale[16]*

1 Introduction

Most materials that on a macroscopic scale can be considered to be homogeneous are usually heterogeneous on a lower scale. The heterogeneous properties of such a material tend to average out giving a homogeneous appearance on the macroscopic level. In applications, materials with certain macroscopic properties might be desirable and knowledge how to create a material with these specific properties is valuable.

By modelling the physical laws that are relevant at the mesoscale the correct macroscopic properties of the material should be possible to extract as the model gets large enough for the statistical differences in the heterogeneities to average out. Such a model is known as a *representative volume element* or in short *RVE*. Having such a model enables us to see how changes in mesoscopic properties affect the macroscopic material. Simulations can then be performed to predict which mesoscopic properties that gives the best macroscopic material for the given application. This is known as *material optimization*. This reasoning gives the motivation for investigating methods in accurately modelling materials at different scales and how the scales are coupled. This is known as *multiscale modelling* cf. [20].

In this thesis the Voronoi tessellation algorithm (see for example [9]) is used to generate three dimensional microstructures which approximate the distribution of grain size and grain shape that is found in many polycrystalline materials. To determine the mechanical behaviour of the microstructure model, i.e. the RVE, finite element analyses should be performed. These will give us information about the response of the RVE but also how stresses and strains distribute within the RVE. The Finite Element (FE) analyses require that the RVE is discretized in space i.e. meshed. The meshing is performed with the open source program Neper [19]. To investigate the performance of this methodology FE analyses of a two-phase stainless steel where the bulk behaviour of the two phases were modelled by crystal plasticity were performed. In these analyses the RVE was subjected to uniaxial tensile loading.

In order to have the capability of including grain boundary mechanisms in the model of the microstructure one method of doing so is examined. This method is the use of cohesive elements available in the commercial finite element analysis software Abaqus. Cohesive elements are inserted into the model of the microstructure at grain boundaries and similar finite element analyses as for the two phase steel are now performed for the prototype material tungsten carbide where cobalt has dissipated into the grain boundaries. The data for the traction separation law was taken from atomistic simulations performed in [7]. It should then be possible to investigate how the boundary mechanisms affect the macroscopic response of a WC-Co alloy.

2 Microstructure generation

In order to conduct numerical analyses of materials with a polycrystalline structure the geometrical properties of the grains in the material must be modeled. Relevant properties to model are the distribution of grain size and the geometrical shape of the grains. One possibility is to extract the three dimensional microstructure from experiments by using X-ray microtomography. The images generated from such an experiment can be analyzed resulting in a digital representation of the microstructure, see for example[2]. However, this is a complicated process. The experiments required are extensive and expensive and the post processing of the data is also non trivial. In order to know if the size of the model that is used is large enough for it to be considered an RVE multiple runs have to be performed and the variance in the results have to be evaluated. It can therefore be said that it is necessary to be able to generate different realisations of microstructures of different sizes. Ideally, one would want a mathematically analytic description of a microstructure that is similar enough to the real structure of the material such that a meaningful analysis can be done. One common way of approximating the microstructure in polycrystalline materials is with the Voronoi tessellation.

A Voronoi tessellation is a partition of a domain $D \in \mathbb{R}^d$ into n regions R_i in D , each corresponding to one of n different *seed points* $\underline{\mathbf{P}}_i$. These regions consists of the set of all points that are closer to a particular seed point than to any other,

$$R_i = \{\underline{\mathbf{x}} \in D : \|\underline{\mathbf{P}}_i - \underline{\mathbf{x}}\| < \|\underline{\mathbf{P}}_j - \underline{\mathbf{x}}\| \quad \forall i \neq j, \quad i, j = 1, \dots, n\}.$$

In this thesis the norm used is the Euclidean distance, the dimension of the space is 3 and the bounding domain is a cube. In this case, the resulting regions will have the shape of convex polyhedrons which are referred to as *grains*. This is the grain structure that would exist in a material that forms in the following way:

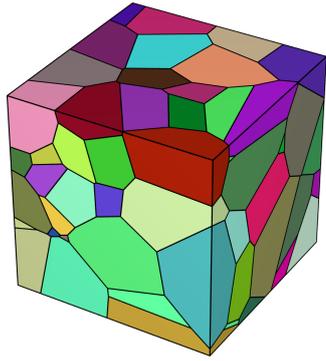
- Grains start to nucleate at all the seed points at the same time.
- The grains grow in all directions at the same rate.
- A grain boundary is created where the grains meet.

In such a Voronoi tessellation two grains will intersect over a plane called a *face*, three grains will intersect along a line called an *edge* and four grains will intersect in a point called a *vertex*. A common way to set the locations of the seed points is to randomly assign them to different positions in D . This is known as a *Poisson-Voronoi tessellation*. One extra requirement that can be used is that

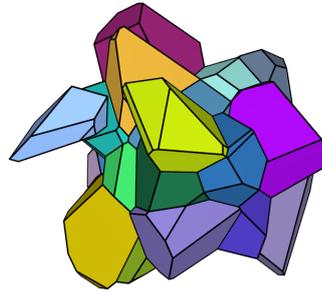
$$\min(|\underline{\mathbf{P}}_i - \underline{\mathbf{P}}_j|) < d, \quad \forall i \neq j, \quad d \in \mathbb{R}^+$$

This means that two seed points must at least be separated by a distance d . This is called *hardcore Voronoi tessellation*. The physical interpretation of this is that two grains can not start nucleate arbitrarily close to each other. This is however not used in this thesis. An example of a tessellation that has been obtained can be seen in figure 2.0.1 where 100 seed points have been used. A comparison between a cross section of a Voronoi tessellation and a polycrystalline material can be seen in figure 2.0.2.

The actual generation of the tessellation can easily be done in for example the commonly used software MATLAB [13]. The computational time required to generate such a tessellation is negligible for any number of grains that would be reasonable for an RVE of a polycrystalline material. In this thesis the open source software Neper [19] was used to generate the tessellation since this software also has a module capable of meshing the generated structure.

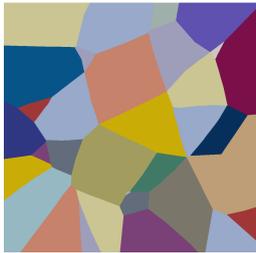


(a) All polyhedrons shown.

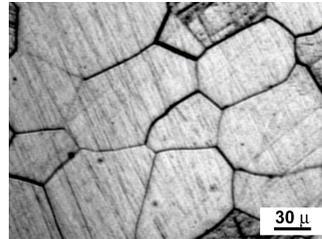


(b) Polyhedrons that are completely inside the domain.

Figure 2.0.1: Voronoi tessellation containing 100 polyhedrons bounded by a cube



(a)

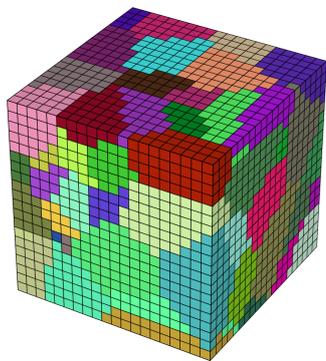


(b)

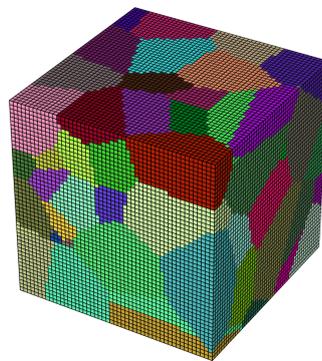
Figure 2.0.2: a) Slice through a voronoi tessellation. b) Micrograph image of a polycrystalline metal.[18]

3 Meshing of microstructure

In order to perform a finite element analysis the generated microstructure needs to be meshed. One way of creating such a mesh, which would be computationally fast and simple to implement, would be to ignore the shape of the grains and use a structured mesh of hexahedral elements. Each element would then be said to be a part of that grain which occupies most of the element's volume. With a reduction in the size of the hexahedrons the shapes of the grains are better approximated. Example of such meshes can be seen in figure 3.0.1.



(a)



(b)

Figure 3.0.1: Structured mesh of different coarseness.

A refinement of having each hexahedral finite element belonging to one single grain would be

allow the Gauss points in one element to belong to different grains. If an element is intersected by two or more grains the Gauss points inside that element could be assigned to the grain they are spatially located in. This method has been adopted [14], [5] and [1].

There are however drawbacks to using a structured mesh. In an unstructured the exact shape of the grains can be respected with a few number of elements. In some RVE:s a fine mesh of the grains are not needed but the shape of the grains might still be important to approximate well. If a structured mesh is used this puts an upper bound on the size of the elements and thus requires a longer computational time. In some cases, for example [4], the results are not significantly different when using a structured or unstructured mesh, even if the same number of degrees of freedom is used in the analyses. However, in other cases such as [10] it was found that when calculating the elasto-viscoplastic response of polycrystalline microstructures, the number of elements could be reduced and still get a convergent result when using a conforming unstructured mesh.

To generate the mesh the software Neper[19] was used. This is the same software that was used to generated the Voronoi tessellation. It has support of generating a conforming mesh which is what has been used in this thesis. Neper uses a scheme to generate a mesh which starts with meshing in the lowest dimension and then uses these elements as seeds to generate elements of a higher dimension. In practice, first nodes are defined at all vertices. Then these nodes are connected along the edges where new nodes are defined at desired intervals. By connecting the edges with each other the faces are created and these are meshed using triangles. The nodes in the edges surrounding the face is used as seeds for these triangles. Finally, the faces are connected, defining the grains, which are meshed with tetrahedrons using the triangles on the faces as seeds. A figure illustrating the process is shown in figure 3.0.2. There are many different algorithms used to do the actual meshing from the lower dimensional seeds. What Neper does is that it uses different algorithms in parallel and from the results chooses the one that generates the mesh with the overall highest quality.

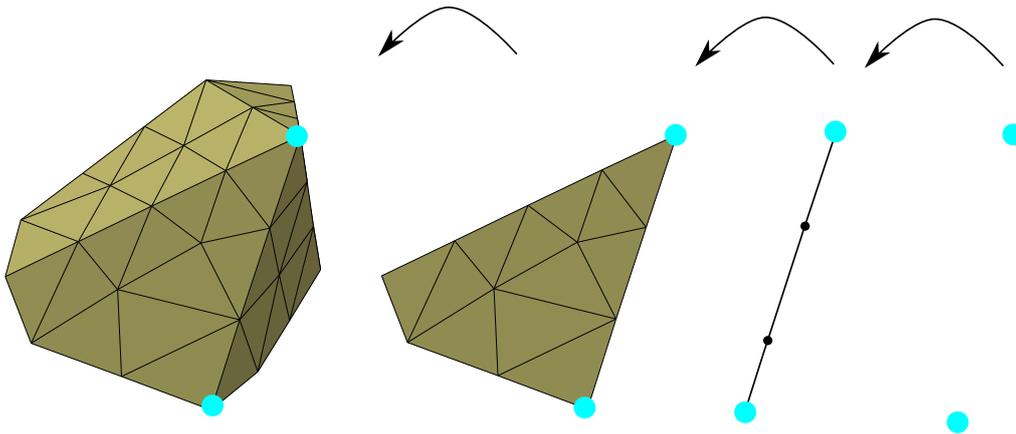


Figure 3.0.2: *Bottom up meshing strategy used by Neper. Grains are meshed from lower to higher dimensions.*

One problem in meshing a Voronoi tessellation is that the tessellation is likely to contain faces that are smaller than the desired characteristic size of the finite elements. One solution to this is to refine the mesh in those parts of the model but this will lead to a significant increase in the total number of finite elements. Another method is to slightly alter the shape of the grains in such a way that the small faces and edges are removed but the general shape of the grain is preserved. It is likely this will not significantly alter the results of the analysis. The

latter method is what the Neper software use. This can be observed if one carefully compares the unmeshed microstructure in figure 2.0.1 to the meshed microstructure in figure 3.0.1. Some of the small faces and edges in the unmeshed case have been removed during the meshing process. A detailed description of the algorithm used to reshape the grains is given in [19].

The output file from Neper lists all the nodes in the mesh and the elements in terms of the connectivity of the nodes. In addition, the output file defines different sets of nodes and elements. For example, for every grain there is a set with the elements that are part of that grain. This means that in the analysis later it is easy to assign certain material properties to each grain. The node sets are sets of the nodes that are found on different boundaries of the cube. The nodes on the bottom face of the RVE are for example listed in their own set. This means that boundary conditions on the boundaries of the RVE:s can easily be set.

4 Crystal plasticity in duplex steel

A material model based on crystal plasticity adopted for a two phase duplex steel has already been developed in [12]. Since it is well described there only the parameter values to the material model will be given here. They can be seen in table 4.0.1. The analyses previously performed with that material model have only been using a two dimensional RVE:s assuming plain stress conditions. It is therefore interesting to see how the same model behaves for a three dimensional RVE and study if significant changes in the stress strain curves are obtained for similar loading conditions.

Table 4.0.1: Material parameters used in the analysis.

Material	G [GPa]	λ [GPa]	τ^y [MPa]	q	ξ	h_0 [MPa]	h_∞ [MPa]	η	m	n	t_* [s]
Austenite	71	106	210	0.1	75	4000	0	0.0001	1.0	1.0	50
Ferrite	71	106	250	1.1	50	100	0	0.001	1.0	1.0	50

4.1 Loading case

The FE analyses performed simulated an RVE (cube) inside a larger structure experiencing a uniaxial tensile testing. This is done by prescribing suitable boundary conditions on the RVE. The boundary conditions are defined by prescribing the displacements of certain nodes of the model. To be specific, the nodes at the bottom face of the cube are fixed such that they cannot move in the y -direction. The nodes at the bottom face on two of the edges have the additional boundary condition that they can only move parallel to their edge, i.e. $u_x = 0$ at $z = 0$ and $u_z = 0$ at $x = 0$. Stresses in the model are induced by having the nodes in the top face of the cube move at a fixed rate in the y -direction. An illustration of the boundary conditions is shown in figure 4.1.1.

In order to perform an FE analysis each element in the model needs to be assigned properties that governs the stress strain behavior of the specific element. In the FE analysis to be performed we want to be able to assign certain material properties depending on what grain the element belongs to. In the meshed model a grain is just a collection of finite elements. These collections are available as element sets so that instead of having to assign the material to each and every finite element it is sufficient to assign the properties to the element set that represent one grain.

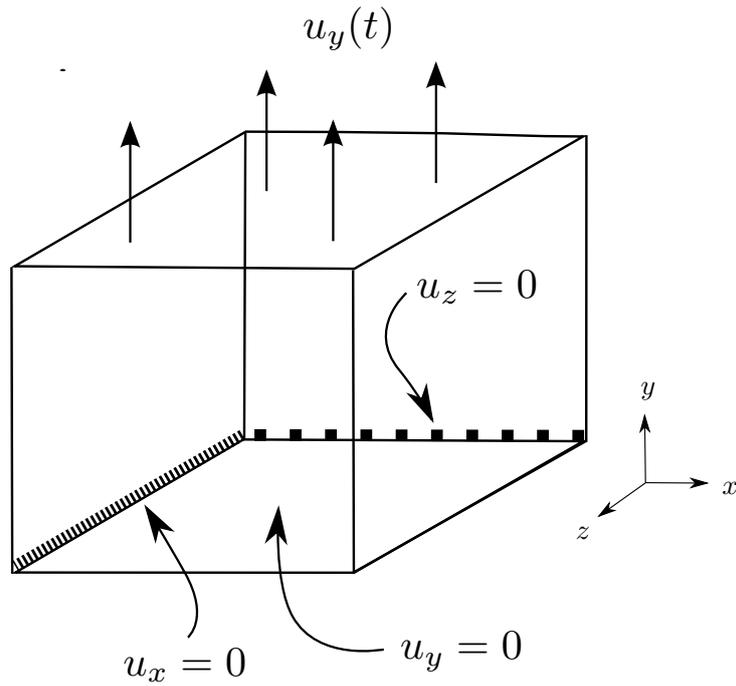


Figure 4.1.1: *Illustration of the boundary conditions applied to the RVE.*

In this thesis the two phases are assigned to random grains such that equal volume fractions are obtained. Each grain is then assigned random slip directions. A figure of the RVE after the material properties have been assigned to the grains is given in figure 4.1.2.

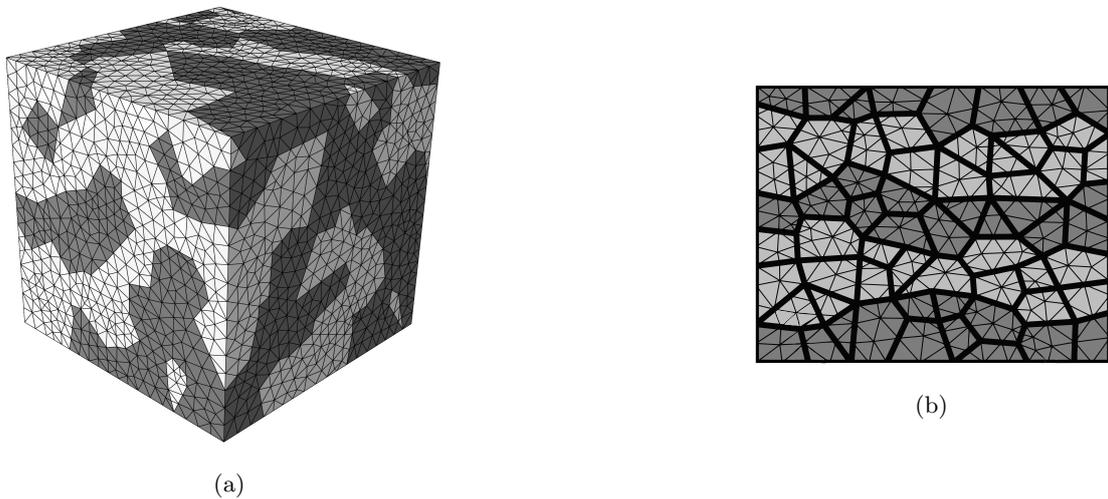
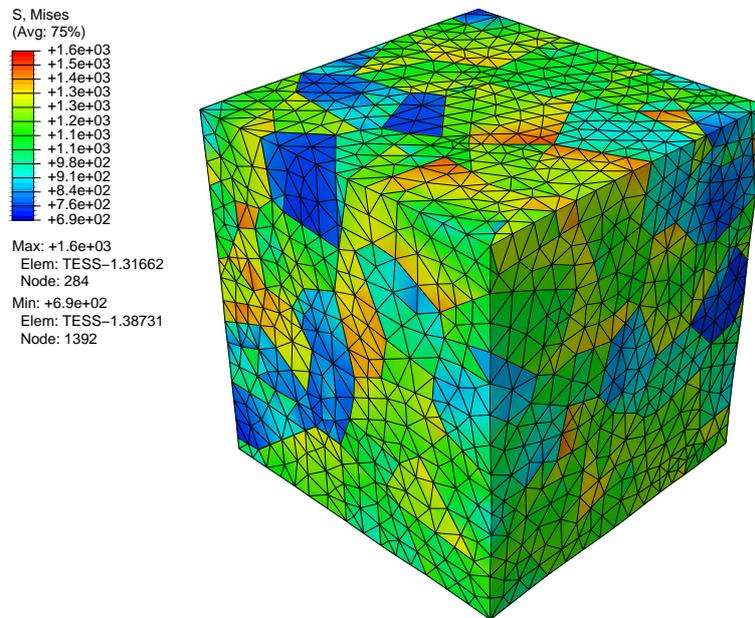


Figure 4.1.2: *a) An example of a three dimensional microstructure with austenite and ferrite assigned to the grains. b) Two dimensional grain structure used in [11], reprinted with permission.*

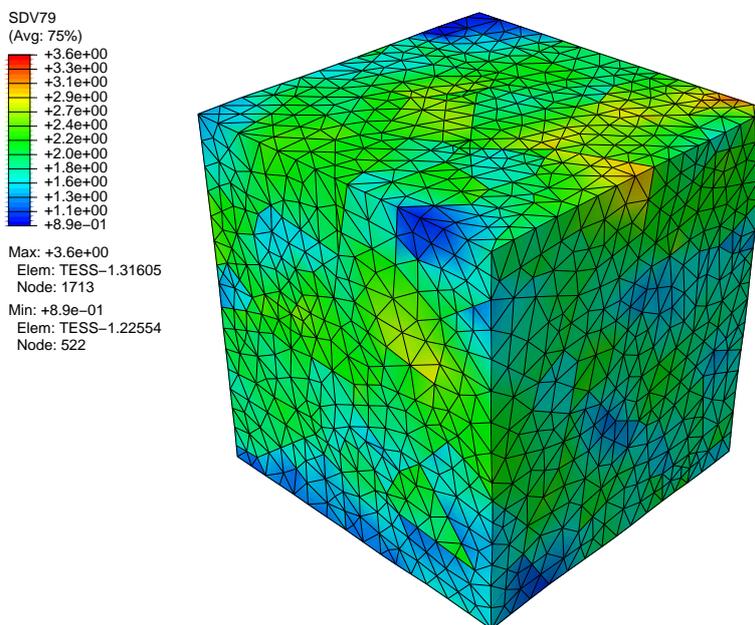
4.2 Results

Five finite element analyses were conducted for models with different number of grains. The number of grains used in the different models were 5, 20, 50, 100, 250 and 500. The difference between analyses with the same number of grains is in the randomness of the seed points in

the Voronoi tessellation and the random assignment of crystal slip directions in each grain. The nodes on the upper boundary of the cube were displaced at a rate that corresponds to a change in strain of $3.125 \cdot 10^{-3} \text{ s}^{-1}$ until a total macroscopic strain of 0.05 was obtained. In order to highlight the heterogeneity at the mesoscopic level the accumulated plastic slip and the von Mises equivalent stress field of an RVE is shown for one analysis in figure 4.2.1.



(a)



(b)

Figure 4.2.1: (a) The von Mises equivalent stress field and (b) the accumulated plastic slip for a mesomodel with 100 grains.

The resulting stress strain curves can be seen for the different analyses in figure 4.2.2. From

this figure there are a few observations that can be made. Before any plastic strain develops all the different analyses have exactly the same stress strain response. This is to be expected since at this amount of strain, the grains behave elastically and the crystal orientation has no effect. Therefore the whole model can be seen as one homogeneous cube. It can also be noted that the variance between the analyses with the same number of grains is reduced when the number of grains is increased. In a model with more grains, the heterogeneity in the model tends to cancel out giving a more representative result for a macroscopic model.

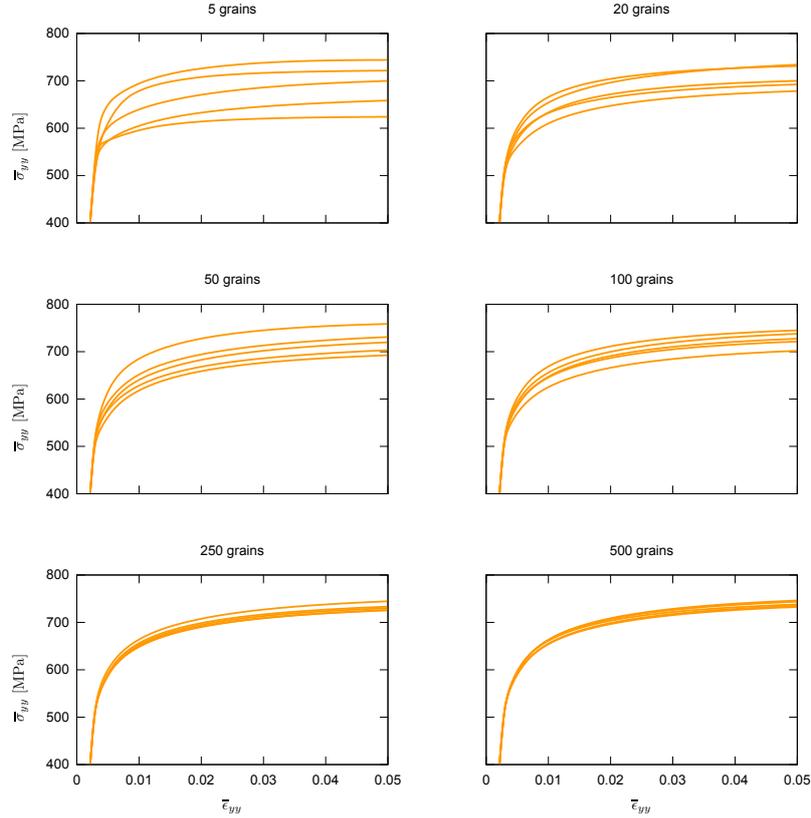


Figure 4.2.2: 5 realizations (analyses) where slip directions are randomly varied of mesomodels with 5, 20, 50, 100, 250 and 500 grains.

The average stress strain responses for the analyses with the same number of grains are compared for different number of grains figure 4.2.3. It can be seen that the stress values tend to increase with more grains in the model and a satisfying convergence is not achieved. In order to determine the cause of this more analyses were conducted. To reduce the heterogeneity these analyses were performed with only ferrite in the RVE. Analyses with up to 2000 grains were conducted. The boundary conditions were the same as in the analyses that used a 50% austenite and ferrite ratio. The same general trend could be seen in these FE analyses, the stress response is slightly stiffer with increasing number of grains. To quantify this effect the stress at a macroscopic strain of 0.05 was plotted for the different analyses with respect to the effective sphere diameter \bar{D}_g defined as

$$\bar{D}_g = 2 \cdot \left(\frac{3\bar{V}}{4\pi} \right)^{1/3}, \quad (4.2.1)$$

where \bar{V} is the average volume of a grain. The result is shown in figure 4.2.4. A line is also

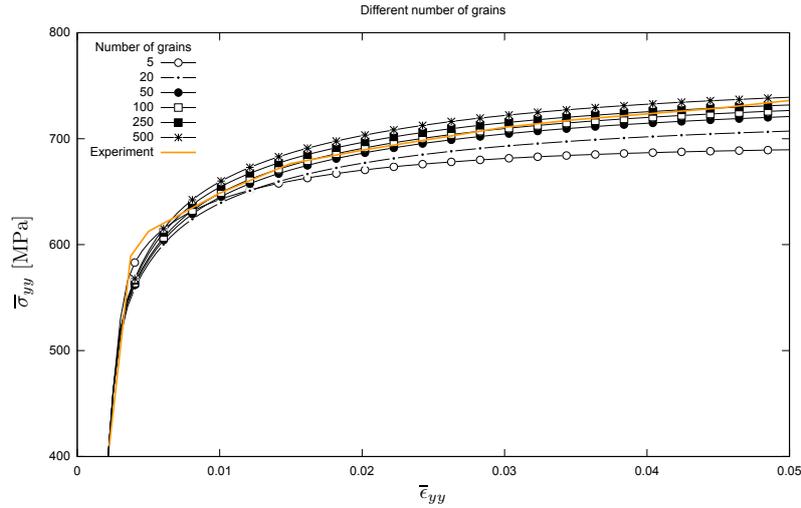


Figure 4.2.3: Average responses of 5 different realizations for 5, 20, 50, 100, 250 and 500 grains. The experimental result found in [12] for the two-phase stainless steel is given.

shown in the figure where the two parameters σ_0 and k_y have been fitted to the data to best fit the curve

$$\sigma_{zz} = \sigma_0 + \frac{k_y}{\sqrt{D}} \quad (4.2.2)$$

with the results $\sigma_0 \approx 650$ and $k_y \approx 27$. Equation 4.2.2 is the Hall-Petch relation. It is not yet clear if the material model follows this relation but what is clear is that there is a relation between the grain size and the stress in the model for equal macroscopic strains. Further investigations must be conducted to understand the reason why the averaged stress do not converge when increasing the number of grains.

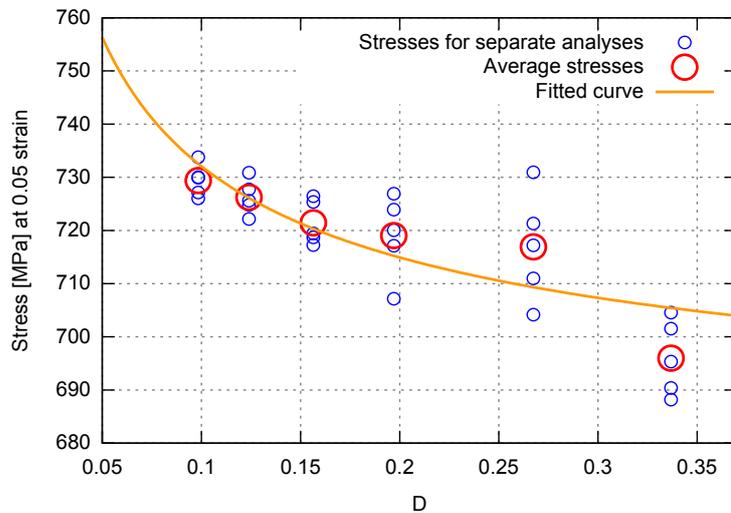


Figure 4.2.4: The stress response at a macroscopic strain of 0.05 against the equivalent sphere diameter.

5 Grain boundary mechanisms

In polycrystalline materials different mechanisms in the boundary between two grains can have significant effects on the macroscopic behavior of the material. Investigations of grain boundary effects in different materials be found in e.g [8], [15]. In order to improve the modeling of materials on a mesoscopic scale there is a need to include these effects into for example a finite element model. This is done in this thesis by something called *cohesive elements*.

5.1 Cohesive elements

The finite element analysis software Abaqus, used in this thesis, has support for cohesive elements. In the manual for Abaqus it states that these "are primarily intended for bonded interfaces where the interface thickness is negligibly small.". In this thesis they are used to model the grain boundary deformation mechanisms that can exist between grains.

The difference between cohesive elements and solid elements is that cohesive elements do not model any material but instead they model the actual interface between two material surfaces. Since the cohesive elements do not contain any material there is no stress tensor like in solid elements. Instead, when the cohesive element is subjected to deformations a vector of the traction components, the normal traction t_n and two shear tractions t_s and t_t is the output. What is considered the normal direction is defined from the order of the nodes defining the vertices of the element as shown in figure 5.1.1. Corresponding separations are denoted by $\delta_n, \delta_s, \delta_t$. Nominal strain in the element are defined as

$$\epsilon_n = \frac{\delta_n}{T_0}, \quad \epsilon_s = \frac{\delta_s}{T_0}, \quad \epsilon_t = \frac{\delta_t}{T_0}$$

The parameter T_0 is not the thickness of the cohesive element but a constitutive thickness that when set to 1.0 yields the strain and the separation become equal. For a more detailed description of how the constitutive thickness works see section 31.5.6 in the Abaqus manual.

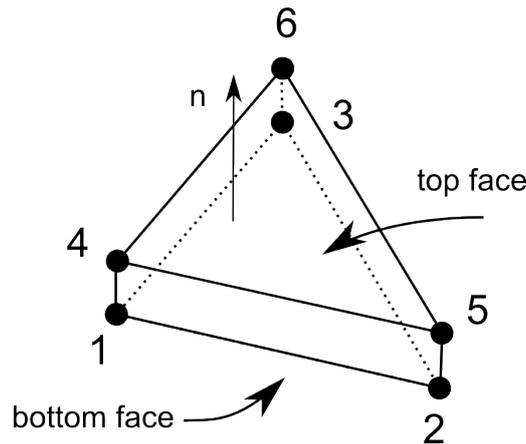


Figure 5.1.1: *Orientation of a cohesive element. The order of the nodes is used to determine the bottom and top face as well as the positive normal direction.*

When modeling damage and failure of a material, or rather an interface surface, by using cohesive element there are three main areas of interest:

1. The properties of the material interface before damage has taken place. Abaqus only supports a linear elastic traction-separation law prior to damage. This is however not a

major limitation since many materials have a linear response before damage occurs.

2. The criterion that defines when the material starts experiencing damage. This is called the *damage initiation criterion*.
3. The properties of the material interface after that the damage initiation criteria has been fulfilled need to be prescribed. This is denoted as *damage evolution* which describes how the degradation of the material stiffness develops as the material interface is separated more and more.

A possible scenario for the damage initiation and evolution can be seen in figure 5.1.2.

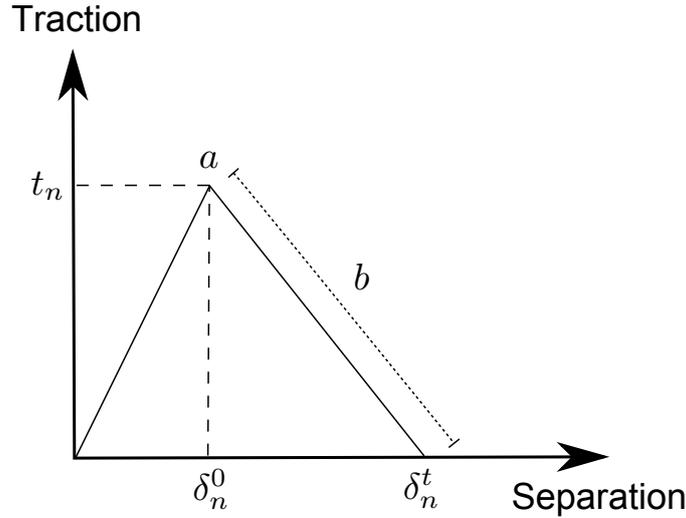


Figure 5.1.2: An example of a traction separation law. The point a is the damage initiation point where the stiffness of the material interface start to degrade. The damage evolution is the interval covered by b showing how the damage is evolving with increasing separation. In this figure a linear damage evolution is assumed. At the separation point δ_n^t the material interface experience a complete failure and fracture occurs.

5.1.1 Undamaged behavior

The traction separation law before damage initiation is assumed to be linear and is written in term of a constitutive matrix as

$$\underline{\mathbf{t}} = \begin{bmatrix} t_n \\ t_s \\ t_t \end{bmatrix} = \begin{bmatrix} K_{nn} & K_{nt} & K_{ns} \\ K_{tn} & K_{tt} & K_{ts} \\ K_{sn} & K_{st} & K_{ss} \end{bmatrix} \begin{bmatrix} \epsilon_n \\ \epsilon_s \\ \epsilon_t \end{bmatrix} = \underline{\mathbf{K}}\underline{\boldsymbol{\epsilon}}.$$

As can be seen from the matrix it is possible to allow for a coupling between normal and shear components. However, in this thesis only uncoupled behavior will be used and thus only the diagonal elements of \mathbf{K} will be non zero in this analysis. Any difference between the shear directions will not be considered which means that $K_{tt} = K_{ss}$.

5.1.2 Damage initiation

The damage initiation criterion is the point when the degradation of the material interface starts to occur. This criterion can either be described as a function of the strain or the traction.

There are two possible functions that can be used in Abaqus, one that is based on the maximum strain or traction component and one that is based on length of the strain or traction vector. For the traction criteria these functions can be expressed as

$$\max \left(\frac{\langle t_n \rangle}{t_n^0}, \frac{t_s}{t_s^0}, \frac{t_t}{t_t^0} \right) = 1,$$

$$\left(\frac{\langle t_n \rangle}{t_n^0} \right)^2 + \left(\frac{t_s}{t_s^0} \right)^2 + \left(\frac{t_t}{t_t^0} \right)^2 = 1.$$

Here t_n^0, t_s^0, t_t^0 are three user defined parameters. The symbol $\langle \rangle$ denotes the Macaulay brackets which is non zero only if the quantity inside the bracket is positive. This has the effect that pure compression (negative normal traction) do not lead to any damage. The corresponding functions for the strains are found by trivial modifications to the above functions. In the analysis used in this thesis the quadratic strain criterion is used.

5.1.3 Damage evolution

The damage evolution describes the degradation of the material interface as a function of the separation. The current degradation of the material is completely captured by one scalar damage variable D . The traction vector is then calculated as

$$\underline{\mathbf{t}} = (1 - D)\bar{\underline{\mathbf{t}}},$$

where $\bar{\underline{\mathbf{t}}}$ is the traction in the element if there was no damage in the material interface. Before the damage initiation criterion has been fulfilled, $D = 0$. Having the damage completely described by only a scalar is one limitation of the cohesive elements that are available in Abaqus. It has the effect that it is hard to capture eventual differences in the traction separation law for normal and shear separations. It would be useful if the damage in the material interface could be described by a three component vector or more generally by a matrix. This is likely possible by writing a user defined element where one codes how the traction is calculated from the separations (known in Abaqus as a **UEL**) but doing this out of scope for this thesis.

If we first assume that we only have separations in the normal direction and want to give D in table form as an input to Abaqus, the following procedure would be used. K_{nn} is calculated as the slope of the initial elastic response. For separations larger than the damage initiation δ_n^0 , at a specific point (δ_n, σ_n) , D is calculated as $D_f(K_{nn}, \delta_n, \sigma_n)$ where D_f is the function:

$$D_f(K, \delta, \sigma) \mapsto 1 - \frac{\sigma}{K\delta}.$$

However, in general, we do not have pure normal or pure shear separations. From Abaqus we can get a measure of the relation in magnitude between the normal and shearing traction components. This measure is a number ϕ_1 between 0 and 1 that is defined as

$$\phi_1 = \frac{2}{\pi} \arctan \left(\frac{\tau}{\langle t_n \rangle} \right)$$

where τ is the effective shear traction $\sqrt{t_s^2 + t_t^2}$. It is also possible to get a number ϕ_2 that gives a relation between the shear tractions but since we will not assume any difference between the shear directions in this thesis, this number is not used. Using the equation for ϕ_1 together with the equation for the effective separation,

$$d_m = \sqrt{d_n^2 + d_s^2 + d_t^2},$$

we can solve for the normal separation d_n and the effective shear separation $d_\tau = \sqrt{d_s^2 + d_t^2}$. in terms of the known ϕ_1 and d . The following results are obtained:

$$d_n = \frac{d_m}{\sqrt{\left(\frac{PK_{nn}}{K_{tt}}\right)^2 + 1}} \quad (5.1.1)$$

$$d_\tau = d_m \sqrt{1 - \frac{1}{\left(\frac{PK_{nn}}{K_{tt}}\right)^2 + 1}}. \quad (5.1.2)$$

where $P = \tan(2\phi_1/2)$. We now calculate two different damage scalars, $D_n = D_f(K_{nn}, \delta_n, \sigma_n)$ and $D_\tau = D_f(K_{tt}, d_\tau, \sigma_\tau)$. These are the correct damage scalars for the case of pure shear or pure normal separation by a distance of δ_n or δ_τ respectively. There is a level of arbitrariness now as to how these two damage scalars should be combined to give an overall damage of the interface. One way of doing this is by adopting the formula,

$$D = 1 - (1 - D_n)(1 - D_t).$$

For mixed separation conditions, D is always smaller than the individual D_n and D_t . This formula is similar to how Abaqus calculates the total damage when multiple damage criteria are active, see section 23.2.3 in the Abaqus manual.

5.1.4 Insertion of cohesive elements into mesh

From Neper we get a mesh of the generated microstructure but Neper does not have the capability of inserting cohesive elements between the grains. Hence, this must be done as a post processing of the generated mesh. A script was written in the programming language Python that parsed the mesh file and inserted the cohesive elements. In order to facilitate the description of the algorithm the script uses to insert the cohesive elements, some notation will be given here.

- A node n is represented by a unique identifier $i \in \mathbb{N}$ among the nodes and three coordinates indicating its location in space x_i, y_i, z_i . It is denoted as $n_i(x_i, y_i, z_i)$.
- A first order triangle element T is represented by a unique identifier $j \in \mathbb{N}$ among all the elements and the three nodes that are used as its vertices. It is denoted as $T_j(n_{j_1}, n_{j_2}, n_{j_3})$.
- A cohesive element C also has a unique identifier but instead of being defined by nodes it is defined in terms of the two triangles which prescribes the two faces perpendicular to the normal of the cohesive element, it is written as $C_j(T_{j_1}, T_{j_2})$.

The algorithm can now be described in two main steps. The algorithm is described for one grain boundary but the same method is used for all boundaries.

1. *Disjoin the shared face in a grain boundary.* - In the generated mesh each face is shared by two grains except the faces on the boundary of the cube which are ignored. An example of two grains with a shared face can be seen in 5.1.3a. What we want to do is create two separate faces, one for each grain, like in figure 5.1.3a. From the Neper mesh file, element sets with the triangles in the current face can be extracted. To identify the grains they are assigned a number $G = 1, \dots, n_g$ where n_g is the total number of grains in the microstructure. The identifiers for the two grains on the current grain boundary will be denoted G_1 and G_2 . The following method is used to separate the faces between two grains.

From each node $n_i(x_i, y_i, z_i)$ in the face two new nodes are created, $n_{f(G_1,i)}(x_i, y_i, z_i)$ and $n_{f(G_2,i)}(x_i, y_i, z_i)$. These new nodes need unique identifiers and this is the purpose of the function f . The function f should take an identifier for a grain and a node such that:

$$f : (\mathbb{N}, \mathbb{N}) \rightarrow \mathbb{N} \quad (5.1.3)$$

$$f(G_1, i) = f(G_2, j) \Leftrightarrow (G_1, i) = (G_2, j) \quad (5.1.4)$$

If this property does not hold then the newly created nodes might have the same identifier as other nodes which would cause problems since each identifier needs to be unique. One simpler way of always obtaining a unique identifier for the new nodes is to increment the identifier by one each time a new node is added. However, this does not work since nodes in the edges of grains are part of multiple faces. Duplicated nodes for these faces would have different identifiers even though they are part of the same grain and originating from the same node. This would cause hanging nodes. A suitable function which was used in this thesis is

$$f(G, i) = G \cdot n_{\text{tot}} + i, \quad (5.1.5)$$

where n_{tot} is the total number of nodes in the original mesh. This has the effect that some of the identifiers for the newly created node can become quite large. However, the identifiers are sparse in the sense that there are many numbers between different identifiers that are not used by any other node as an identifier. This means that after the insertion of cohesive elements are done, then the identifiers can be renumbered in such a way that they are dense and at this point they are no longer a higher number than necessary.

With the new nodes in the mesh, new triangles can be defined. From the set of triangles in the face $\{T_j(n_{j_1}, n_{j_2}, n_{j_3}) | \forall j : T_j \in \text{face}\}$, two new sets are created, one for each grain

$$\{T_{j_1}(n_{f(G_1,j_1)}, n_{f(G_1,j_2)}, n_{f(G_1,j_3)}) | \forall j : T_j \in \text{face}\}$$

and

$$\{T_{j_2}(n_{f(G_2,j_1)}, n_{f(G_2,j_2)}, n_{f(G_2,j_3)}) | \forall j : T_j \in \text{face}\}.$$

These two sets of triangles represent two new faces that do not share any nodes but are at the same location in space as the original face. What needs to be done now is to connect the tetrahedral element that where part of the original face to the newly created faces. This is done by searching each tetrahedral inside the first and second grain, with identifier G_1 and G_2 respectively, for nodes $n_i \in \text{face}$ and swap these nodes for the newly created nodes $n_{f(G_1,i)}$ and $n_{f(G_2,i)}$ respectively.

2. *Connect the new faces with cohesive elements.* At this point we have two faces that have no connection of elements to each other. These are now easily connected by creating the set of cohesive elements $\{C_j(T_{j_1}, T_{j_2}) | \forall j : T_j \in \text{face}\}$. This is illustrated in figure 5.1.3c.

After this algorithm has been executed for all faces the cohesive elements are inserted. The new mesh can then be exported and used in FE analyses. A figure where cohesive elements have been inserted between grains in a full microstructure can be seen in figure 5.1.4. In the figure the grains have been pulled apart from each other in order for the cohesive elements to have a finite width and be visible.

The algorithms and numerical methods presented in this section have been implemented as an open source software package for Python: Phon [6].

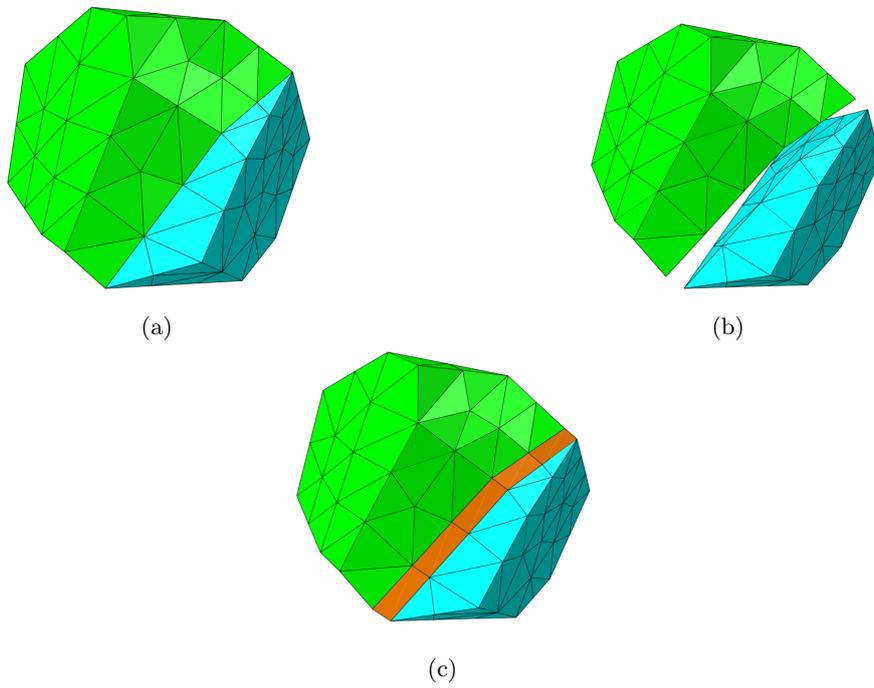


Figure 5.1.3: *Insertion of cohesive elements between two grains. (a) The two grains. (b) Duplication of nodes in the face. Grains have been separated for clarity of the figure. (c) Cohesive elements inserted between the two faces.*

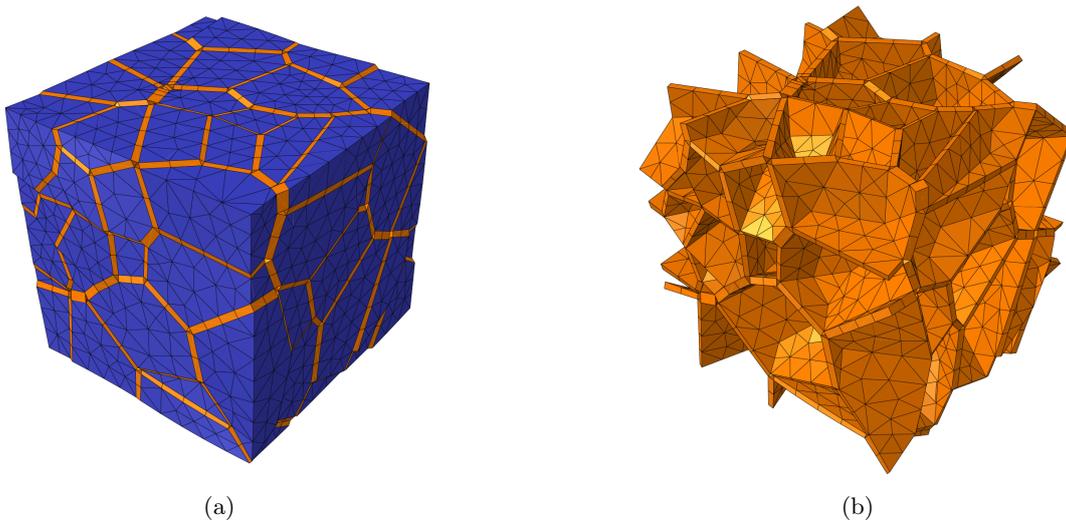


Figure 5.1.4: *Cohesive elements with a) grains shown and b) grains hidden.*

5.2 FE analyses with cohesive elements

In order to test the generated mesh that includes cohesive elements, FE analyses were performed on a micro structure with material properties that were not set to simulate a real case.

One Voronoi tessellation of 50 grains was generated and meshed. Cohesive elements were then inserted into the mesh. The grains were set to behave elastically with a Young's modulus of $E = 200$ GPa and Poisson ratio $\nu = 0.3$. Cohesive elements were set to initially be mode independent with an elasticity equal to the one in the solid elements. The damage initiation criteria used was $\sigma_n^0 = 0.001$ and a linear damage evolution law was used with fracture at $\sigma_n^t = 0.04$. The same method of inducing strain as in section 4 was used. The analysis was run for meshes of different coarseness to see if there was any mesh dependence in the results. The resulting stress strain curves can be seen in figure 5.2.1. The analyses were set to run

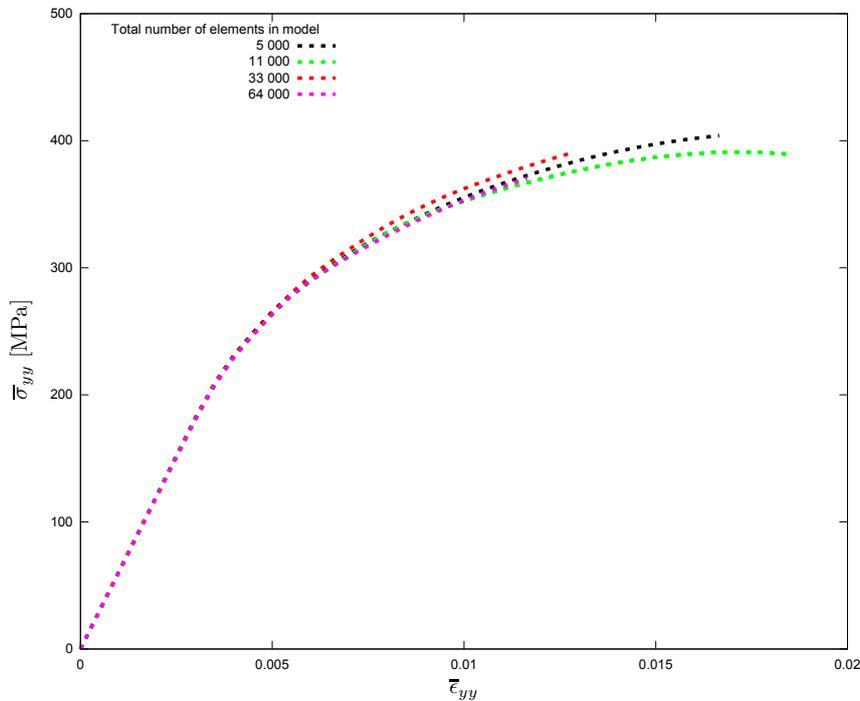


Figure 5.2.1: *The analyses were performed for the same microstructure but with different mesh coarseness. This showed slight differences in the resulting stress strain curve and also different points of convergence failure in the implicit integration technique.*

until a macroscopic strain of 0.05 of the RVE. However, at different points in the analysis convergence problems with the implicit solver caused the analysis to abort. This happened at different points during the analysis. Implicit solvers generally have difficulties with softening in materials and the discontinuous nature in the gradient of the traction separation law in cohesive elements probably makes things worse. It is likely that there is a discontinuity in the force displacement curve causing the Newton Raphson method to not converge to an equilibrium point. A common advice given to overcome the convergence problems with implicit solvers associated with the softening of materials due to cohesive elements is to introduce viscous regularization in the cohesive elements. The viscous stiffness degradation variable D_v is defined

as

$$\dot{D}_v = \frac{1}{\mu}(D - D_v) \quad (5.2.1)$$

where μ is the viscosity parameter. The response of the damaged cohesive interface is then given as

$$\underline{\mathbf{t}} = (1 - D_v)\bar{\underline{\mathbf{t}}}. \quad (5.2.2)$$

This should have the effect of smoothing out the damage in the cohesive element making it less likely that the implicit solver fails in finding an equilibrium point. This was attempted but it had little to no effect in terms of convergence. The exact cause for the convergence problem is yet to be pinpointed.

There is however another way of solving the FE problem. In an implicit solver the equilibrium equation that is solved is

$$\underline{\mathbf{f}}_{\text{int}}(u) - \underline{\mathbf{f}}_{\text{ext}}(u) = 0. \quad (5.2.3)$$

For notation see section A.2. Instead of solving the equilibrium equation we can take into account acceleration giving the equation

$$\underline{\mathbf{M}}\ddot{\mathbf{u}} + \underline{\mathbf{f}}_{\text{int}}(u) - \underline{\mathbf{f}}_{\text{ext}}(u) = 0., \quad (5.2.4)$$

where $\underline{\mathbf{M}}$ is known as the *mass matrix* defined as

$$M_{ij} = \int_{\Omega} \varphi v_i v_j \, dx. \quad (5.2.5)$$

This can be solved by e.g. the central different method as

$$\frac{1}{\Delta t^2} \underline{\mathbf{M}}\mathbf{u}^{t+\Delta t} = \underline{\mathbf{f}}_{\text{ext}}^t(u) - \underline{\mathbf{f}}_{\text{int}}^t(u) + \frac{1}{\Delta t^2} (2\underline{\mathbf{M}}\mathbf{u}^t - \underline{\mathbf{u}}^{t-\Delta t}) \quad (5.2.6)$$

This is however conditionally stable depending on the time step Δt used, c.f. [17]. A necessary condition for convergence is the Courant–Friedrichs–Lewy condition. In FE analyses this condition can be stated that the distance traveled by the fastest wave in the material in one time step should be smaller than the characteristic size of the element L . The speed of the waves is related to the stiffness k and mass m of the material. The condition can be written as

$$\Delta t \leq \alpha \frac{L\sqrt{m}}{\sqrt{k}} \quad (5.2.7)$$

where α is a proportionality constant.

This condition can be quite strict and a small time step might thus be needed for stability which leads to extensive computational time required. However, the problem we are trying to solve is static and should be independent of the mass. We should therefore be able to scale the mass in the FE model in order to allow for larger time steps to respect the convergence condition without significantly affecting the result. In order to study this analyses where the mass was scaled in order for the stable time step to change were performed and the results are shown in figure 5.2.2. It can be seen that if the mass is scaled high enough the resulting stress is irregular and seem to have some high frequency oscillations. However, with smaller time steps the curves seem to converge nicely.

In order to test the validity of the results from the explicit solver the results are compared to the implicit solver for identical mesh and material properties. In figure 5.2.3 the results are shown and very good agreement between the explicit and implicit solutions can be seen. From this we draw the conclusion that it is possible to use the explicit solver in Abaqus in order to circumvent the convergence problems associated with the implicit solver.

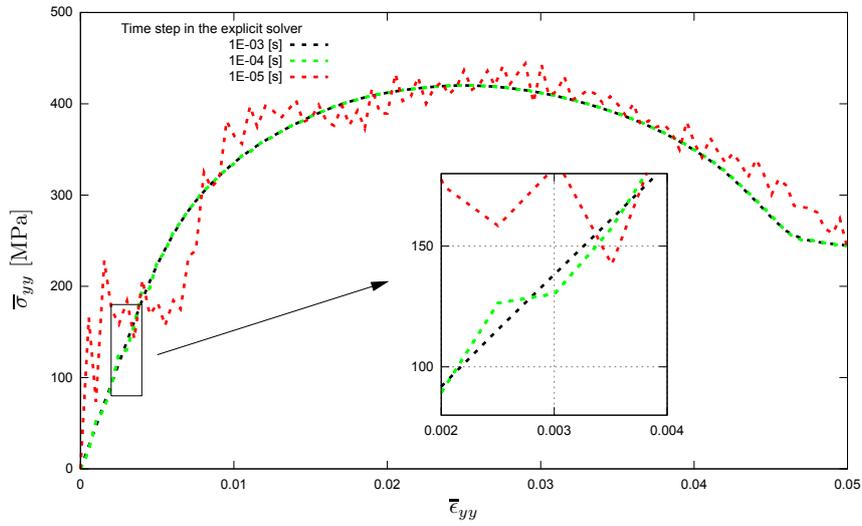


Figure 5.2.2: Results of the analyses where the mass of the model has been scaled to give a stable time increment of 10^{-3} s, 10^{-4} s and 10^{-5} s .

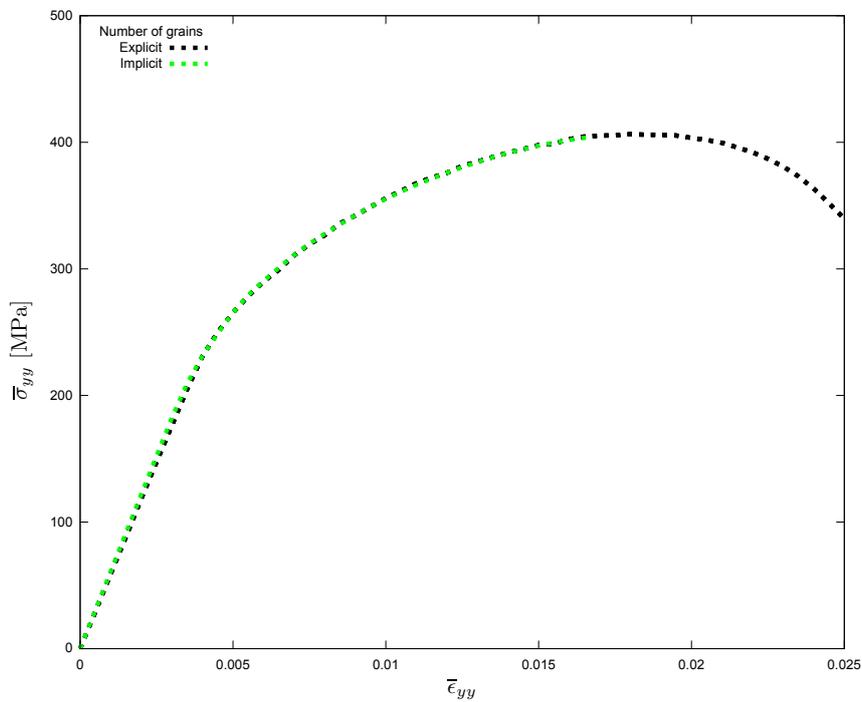


Figure 5.2.3: Comparison between the solution from an explicit and an implicit solver. This shows that for the region where the implicit solver converges the explicit and implicit solution agree well.

5.3 FE analyses of tungsten carbide

To test the cohesive element method on a real world material FE analyses were performed on a model where the characteristics of the traction separation law were based on results from atomistic simulations in [7]. The traction separation law for the grain boundaries in tungsten carbide was entered in the cohesive elements and FE analyses were performed.

5.3.1 Test case

In order to confirm that the data was entered correctly into Abaqus a simple test was performed. The test was defined by a model of two grains with the shape of rectangular prism. The grains were connected with cohesive elements and the nodes in the top grain was displaced in different directions. In this test the grains were made very stiff such that only deformation in the cohesive elements took place. The model after displacement can be seen in figure 5.3.1. The two white grains can be seen as well as the green cohesive elements connecting them. The displacement has been exaggerated for clarity of the figure.

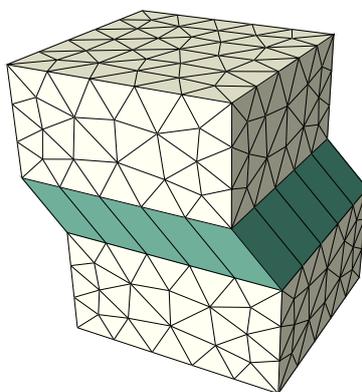


Figure 5.3.1: *The model that was used for the test case.*

The atomistic simulations were only performed with displacements of the bulk grains in pure normal and shear directions. There is therefore a need to interpolate the traction separation law between these two cases and to do this the method described in section 5.1.3 was used. The top grain in the test case was displaced along the pure normal and shear directions as well as in a mixed direction. The traction in the cohesive elements for the different directions is shown in figure 5.3.2. The traction separation law for the edge cases are exactly those that was entered into the cohesive element and the result for the mixed case is sensible. This confirms that the method used to interpolate the traction separation law is reasonable and that the pure normal and shear responses are correct.

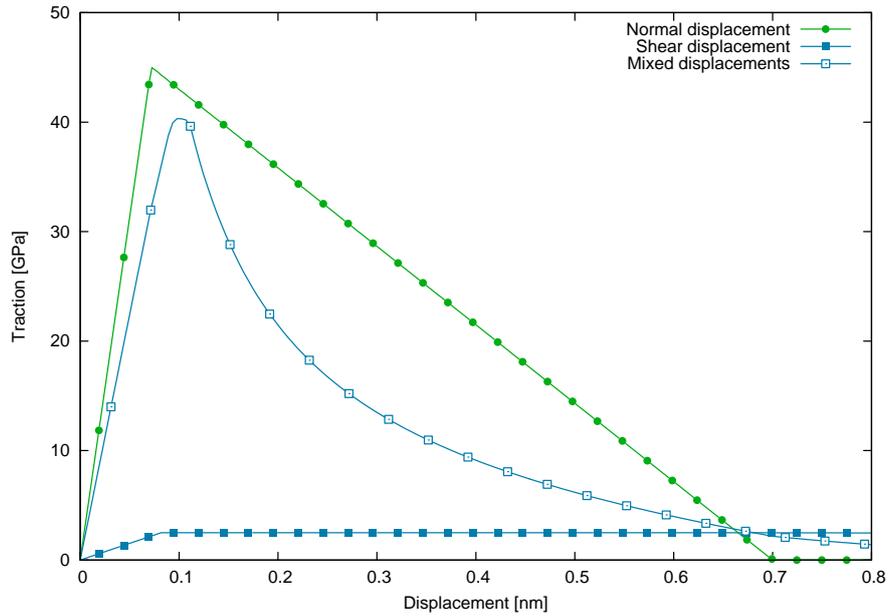


Figure 5.3.2: Results for the traction in the cohesive zone elements for three different directions. The traction law with pure normal and pure shear displacement is respected and reasonable results are shown for mixed separations.

5.3.2 Full analysis

The same traction separation law as in the test case above was used for these analyses of three dimensional grain structures. The grains of tungsten carbide were modeled to deform elastically with $E \approx 250$ GPa. This is consistent with results in [3]. This effects of the cobalt skeleton that usually exist in this material is not taken into account in this thesis. The size of the grains were set to average 1 micron. Simulations were done with 50, 100, 250, 500 grains. Results of the average stresses between the analyses of equal number of grains can be seen in figure 5.3.3

An animation over the deformation in the model showed that a majority of the cohesive element completely fracture in a small time span. This might indicate that the analyses were not performed at a fine enough time resolution and the model might thus not capture the correct interface law.

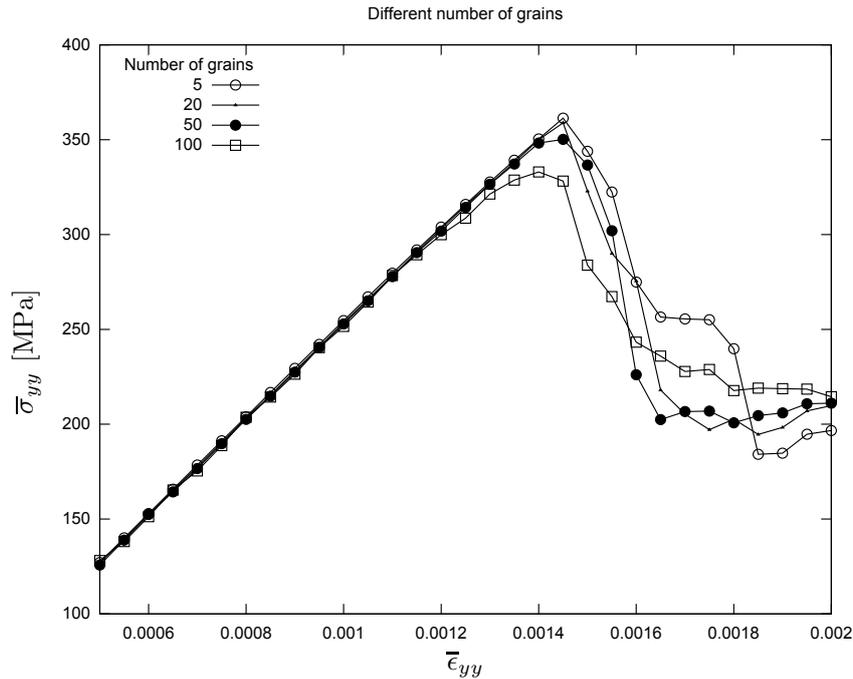


Figure 5.3.3: Results for wolfram carbide for different number of grains

6 Conclusions

This thesis has investigated a method of modelling polycrystalline method as well as one method to include grain boundary effects. The microstructures were generated using the Voronoi tessellation and meshed into a mesh conforming to the grain boundaries. Both the generation of the microstructure and the meshing were performed with the software Neper [19]. This meshed microstructure was then used to perform analyses of a crystal plasticity model presented in [12]. Grain boundary mechanisms were included in the model by inserting cohesive elements in the interface between grains. FE analyses were performed on this model, firstly with a non realistic material interface law by both an explicit and implicit integration technique and secondly with a traction separation law based on atomistic simulations performed in [7]. The code that imports the mesh from Neper and inserts cohesive elements was released as a free open source package from Python: Phon [6].

The FE analyses of the crystal plasticity model showed similar results for the three dimensional RVE as the two dimensional used in the previous work. However, it was found that the stress at a certain macroscopic strain increased with the number of grains. It is possible that the parameters for the material model was tuned to agree with experiments for a certain number of grains. This would imply that the material model needs more refinement to take into account the effect of grain size to accurately model the material.

The results of the analyses with cohesive elements showed that the implicit integration method might be unsuitable for analyses of that type. Convergence problems were encountered even for simple models. It was however shown that an explicit integration method could be used since it yielded the same results as with the implicit method but did not have problems with convergence. Analyses of a microstructure that tried to model tungsten carbide grain boundaries were performed.

In order to be able to say if the cohesive element method is a useful method to model grain boundary mechanisms more extensive analyses must be performed. The analyses performed in this thesis had not access to very detailed bulk behavior of the material nor was the modeling of the grain structure very accurate. With more detailed data of the bulk material better results are likely achievable.

The method presented that describes the insertion of the cohesive elements between interfaces should be usable on other types of microstructure than those generated using Voronoi tessellation. This is useful if one wants to tune a model of the microstructure in cases when Voronoi tessellation is not an accurate enough approximation to the real grain microstructure.

References

- [1] F. Barbe et al. “Intergranular and intragranular behavior of polycrystalline aggregates. Part 1: F.E. model”. In: *International Journal of Plasticity* 17.4 (2001), pp. 513–536. DOI: 10.1016/S0749-6419(00)00061-9. URL: <http://www.sciencedirect.com/science/article/pii/S0749641900000619>.
- [2] Y. Bhandari et al. “3D polycrystalline microstructure reconstruction from {FIB} generated serial sections for {FE} analysis”. In: *Computational Materials Science* 41.2 (2007), pp. 222–235. DOI: 10.1016/j.commatsci.2007.04.007. URL: <http://www.sciencedirect.com/science/article/pii/S0927025607000997>.
- [3] K. Buss. “High temperature deformation mechanisms of cemented carbides and cermets”. PhD thesis. Lausanne: SB, 2004. DOI: 10.5075/epfl-thesis-3095.
- [4] T. Böhlke et al. “Elastic properties of polycrystalline microcomponents”. In: *Mechanics of Materials* 42.1 (2010), pp. 11–23. DOI: 10.1016/j.mechmat.2009.08.007. URL: <http://www.sciencedirect.com/science/article/pii/S016766360900132X>.
- [5] G. Cailletaud et al. “Some elements of microstructural mechanics”. In: *Computational Materials Science* 27.3 (2003), pp. 351–374. DOI: 10.1016/S0927-0256(03)00041-7. URL: <http://www.sciencedirect.com/science/article/pii/S0927025603000417>.
- [6] K. Carlsson. *Phon*. URL: <https://github.com/KristofferC/phon>.
- [7] M. Gren. “Molecular dynamics simulations of grain boundaries in cemented carbides”. In: *To be published* ().
- [8] N. Kheradmand, A. Barnoush, and H. Vehoff. “Investigation of the role of grain boundary on the mechanical properties of metals”. In: *Journal of Physics: Conference Series* 240.1 (2010), p. 012017. URL: <http://stacks.iop.org/1742-6596/240/i=1/a=012017>.
- [9] S. Kumar et al. “Properties of a three-dimensional Poisson-Voronoi tessellation: A Monte Carlo study”. English. In: *Journal of Statistical Physics* 67.3-4 (1992), pp. 523–551. DOI: 10.1007/BF01049719. URL: <http://dx.doi.org/10.1007/BF01049719>.
- [10] W. Li and N. Zabaras. “A virtual environment for the interrogation of 3D polycrystalline microstructures including grain size effects”. In: *Computational Materials Science* 44.4 (2009), pp. 1163–1177. DOI: 10.1016/j.commatsci.2008.07.034. URL: <http://www.sciencedirect.com/science/article/pii/S0927025608003716>.
- [11] R. Lillbacka. *On the Multiscale Modeling of Duplex Stainless Steel: Thesis for the Degree of Doctor of Philosophy in Solid Mechanics*. Doktorsavhandlingar vid Chalmers tekniska högskola. Chalmers University of Technology, Department of Applied Mechanics, 2007. URL: <http://books.google.se/books?id=uf10NQAACAAJ>.
- [12] R. Lillbacka and M. Ekh. “On the parameter identification of a crystal plasticity model via the calibration of a multiscale material model for duplex stainless steel”. In: (2007).
- [13] *MATLAB manual. Ordinary Differential Equations*. Version R2013a. Mathworks, 2013. URL: <http://www.mathworks.se/help/matlab/ref/voronoi.html>.
- [14] M. Nygård. “Number of grains necessary to homogenize elastic materials with cubic symmetry”. In: *Mechanics of Materials* 35.11 (2003), pp. 1049–1057. DOI: 10.1016/S0167-6636(02)00325-3. URL: <http://www.sciencedirect.com/science/article/pii/S0167663602003253>.
- [15] T. Ohmura and K. Tsuzaki. “Analysis of grain boundary effect of bulk polycrystalline materials through nanomechanical characterization”. In: *Journal of Physics D: Applied Physics* 41.7 (2008), p. 074015. URL: <http://stacks.iop.org/0022-3727/41/i=7/a=074015>.
- [16] T. C. Ozawa and S. J. Kang. “Balls & Sticks: easy-to-use structure visualization and animation program”. In: *Journal of Applied Crystallography* 37.4 (Aug. 2004), p. 679. DOI: 10.1107/S0021889804015456. URL: <http://dx.doi.org/10.1107/S0021889804015456>.
- [17] K. Park. “Practical aspects of numerical time integration”. In: *Computers and Structures* 7.3 (1977), pp. 343–353. DOI: 10.1016/0045-7949(77)90072-4. URL: <http://www.sciencedirect.com/science/article/pii/0045794977900724>.
- [18] E. Pleshakov. *Microstructure of VT22*. [Online; accessed 8-May-2013]. 2008. URL: <http://en.wikipedia.org/wiki/File:CrystalGrain.jpg>.
- [19] R. Quey, P. Dawson, and F. Barbe. “Large-scale 3D random polycrystals for the finite element method: Generation, meshing and remeshing”. In: *Computer Methods in Applied Mechanics and Engineering*

200.17-20 (2011), pp. 1729–1745. DOI: 10.1016/j.cma.2011.01.002. URL: <http://www.sciencedirect.com/science/article/pii/S004578251100003X>.

- [20] T. Zohdi and P. Wriggers. “Computational micro-macro material testing”. English. In: *Archives of Computational Methods in Engineering* 8.2 (2001), pp. 131–228. DOI: 10.1007/BF02897871. URL: <http://dx.doi.org/10.1007/BF02897871>.

A Introduction to finite element method

Here follows a short introduction to the basic principles behind the Galerkin finite element method for elliptic problems. This method is later exemplified by applying it to a linear elasticity problem.

A.1 Galerkin method

In general the steps in the Galerkin method are:

1. *Find the weak formulation of the problem.* The weak formulation does not require the solution to the equation to hold pointwise but instead it should hold with respect to certain arbitrary test functions. An example to illustrate this is useful. Consider the Poisson equation on a bounded domain $\Omega \subset \mathbb{R}^d$ with a boundary Γ , and

$$\begin{cases} u_{,ii} = f, & \text{in } \Omega, \\ u = 0, & \text{on } \Gamma. \end{cases} \quad (\text{A.1.1})$$

Let $v \in V$ be an arbitrary function where V is the Hilbert space $H_0^1(\Omega)$ where the zero in the subscript denotes that the functions should vanish on the boundary. Multiplying (A.1.1) by v , integrating over Ω and using Green's identity gives:

$$\underbrace{\int_{\Omega} u_{,i} v_{,i}}_{a(u,v)} = \cancel{\int_{\Gamma} v u_{,i} n_i} - \underbrace{\int_{\Omega} f v}_{f(v)}, \quad \forall v \in V. \quad (\text{A.1.2})$$

The integral over the boundary is zero due to v being zero everywhere on it. A function u which satisfies this relation is called a weak solution to the original problem. In general the weak form can be written as

$$a(u, v) = f(v), \quad \forall v \in V \quad (\text{A.1.3})$$

where $a(.,.)$ is of bilinear form and $f(.)$ is of linear form. The purpose of rewriting the equation in weak form is that the requirement on the smoothness of the function u has been reduced.

2. *Discretize the problem.* This is done by constructing a finite dimensional subspace of V denoted V_h . Let $\{N_1, \dots, N_D\}$ be a basis for the subspace. The function v can now be written as a linear combination of the basis

$$v = c_{\beta} N_{\beta}, \quad c_{\beta} \in \mathbb{R} \quad (\text{A.1.4})$$

Here we introduce the notation that Greek letter indices are taken from the range 1 to D . The discrete analogue of the weak form (A.1.3) is then written as: Find $u_h \in V_h$ such that

$$a(u_h, v) = f(v) \quad \forall v \in V_h. \quad (\text{A.1.5})$$

Since this should hold for all admissible v it should hold true componentwise:

$$a(u_h, N_{\beta}) = f(N_{\beta}). \quad (\text{A.1.6})$$

Representing u_h in the new basis gives

$$a(a_{\alpha} N_{\alpha}, N_{\beta}) = f(N_{\beta}) \quad a_{\alpha} \in \mathbb{R} \quad (\text{A.1.7})$$

Since a is bilinear this can be rewritten as

$$a(N_\alpha, N_\beta) a_\alpha = f(N_\beta) \quad (\text{A.1.8})$$

This can in condensed form be written as

$$\mathbf{K}\mathbf{a} = \mathbf{b} \quad (\text{A.1.9})$$

where $K_{\alpha\beta} = a(N_\alpha, N_\beta)$ is a $D \times D$ matrix and $b_\beta = f(N_\beta)$ is a vector with D elements. \mathbf{K} is commonly denoted as the *stiffness matrix* and b the load vector.

3. *Solve the discrete problem.* In practice this means to solve the system of equations A.1.9. This can be solved by direct (Gauss elimination) or iterative methods (Jacobi method, Gauss-Seidel method). Generally the functions constituting the basis are chosen such that they are non zero only in a small region of V_h . This means that the matrix \mathbf{K} will be sparse and can be solved much faster than a general dense $D \times D$ system.

The way the subspace V_h is constructed deserves some more attention. A common way is to divide the domain Ω into a set (called a *mesh*) of geometrical solids K_i such that

$$\Omega = \bigcup_{i=1}^D K_i \quad (\text{A.1.10})$$

If the dimension of Ω is 2 and the boundary Γ is piecewise linear, one way of constructing such a mesh is by triangles where no vertex lies on the edge of another triangle. The vertices of the triangles are called *nodes*. These nodes are assigned a unique number $P_i \in \mathbb{N}$ as an identifier. P_i can then be used as a parameter to describe the function v . The space V_h can be constructed as for example the set of all linear functions on every triangle P_i satisfying the boundary conditions on Γ . The basis can be written as

$$N_i(P_j) = \delta_{ij}, \quad i, j = 1, \dots, D \quad (\text{A.1.11})$$

We can see that N_i is only non zero for triangles having the node P_i as one of its vertices. Figure A.1.1 shows an example of this where only the shaded triangles gets a contribution from N_i . This means that the stiffness matrix will in general be sparse.

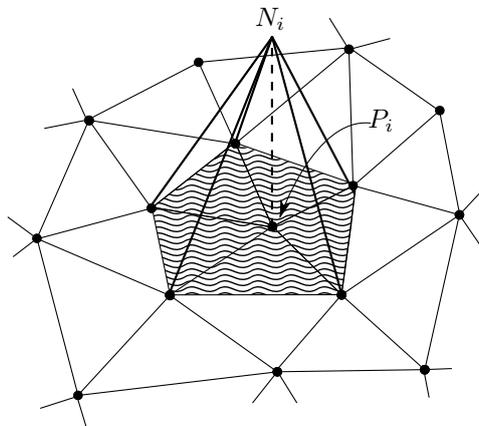


Figure A.1.1: *Basis function.*

A.2 FEM for nonlinear and linear elasticity

As a relevant example, the derivation of the stiffness matrix and load vector for nonlinear and linear elastic mechanics problem is given. The equation that is to be solved is Cauchy's momentum equation on a domain Ω with a given traction $t_i = \sigma_{ij}\hat{n}_j$ on the boundary Γ ,

$$\begin{cases} \sigma_{ji,j} + f_i = 0 & \text{on } \Omega \\ t_i = g_i & \text{on } \Gamma \end{cases} \quad (\text{A.2.1})$$

As described before the equation is multiplied by a kinetically admissible function v and an integration over the domain is performed,

$$\int_{\Omega} v_i \sigma_{ji,j} dx + \int_{\Omega} v_i f_i dx = 0, \quad (\text{A.2.2})$$

where $\sigma_{ji} = \sigma(\epsilon_{ij})$ and $\epsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i})$. Using Gauss identity and the boundary condition of the problem gives

$$\underbrace{\int_{\Omega} v_{i,j} \sigma_{ji} dx}_{f_{\text{int}}(\bar{u})} - \underbrace{\int_{\Omega} v_i f_i dx + \int_{\Gamma} v_i g_i ds}_{f_{\text{ext}}} = 0 \quad (\text{A.2.3})$$

This is the weak formulation of the original problem. It can be interpreted as equilibrium equations in all the nodes of the FE mesh. It is commonly solved with the Newton-Raphsons method. If linear elasticity is assumed the constitutive relations between the stress and strain can be rewritten as

$$\sigma_{ij} = C_{ijkl} \epsilon_{kl} \quad (\text{A.2.4})$$

Due to symmetry in σ_{ij} and ϵ_{ij} the stiffness tensor must satisfy $C_{ijkl} = C_{ijlk}$ and thus we have the following relation.

$$C_{ijkl} \epsilon_{kl}(u) = C_{ijkl} \frac{1}{2}(u_{k,l} + u_{l,k}) = C_{ijkl} u_{k,l} \quad (\text{A.2.5})$$

Inserting (A.2.4) and (A.2.5) into (A.2.3) gives

$$\int_{\Omega} v_{i,j} C_{ijkl} u_{k,l} dx = \int_{\Gamma} v_i g_i ds + \int_{\Omega} v_i f_i dx \quad (\text{A.2.6})$$

By moving to the finite dimensional subspace V_h , and rewriting v_h and u_h into the basis of that subspace we end up with the expression:

$$\int_{\Omega} N_{\beta,j} v_{i\beta} C_{ijkl} N_{\alpha,l} u_{k\alpha} dx - \int_{\Omega} N_{\beta} v_{i\beta} f_i - \int_{\Gamma} N_{\beta} v_{i\beta} g_i = 0 \quad (\text{A.2.7})$$

This can be rewritten as

$$(K_{\beta i \alpha k} u_{\alpha k} - f_{i\beta}) v_{i\beta} = 0, \quad (\text{A.2.8})$$

where

$$K_{\beta i \alpha k} = \int_{\Omega} N_{\beta,j} C_{ijkl} N_{\alpha,l} dx, \quad f_{i\beta} = \int_{\Gamma} N_{\beta} g_i + \int_{\Omega} N_{\beta} f_i. \quad (\text{A.2.9})$$

Since this should hold for every kinematically admissible v_i we have the following linear equations to solve:

$$K_{\beta i \alpha k} u_{\alpha k} - f_{i\beta} = 0. \quad (\text{A.2.10})$$