

## **Multiscale technics in a casting process**

Ana Paula de Oliveira Costa

Dissertation carried out within the scope of  
the Master's degree in Computational Mechanics

Orientador: Prof. Dr. José César De Sá  
Co-orientador: Prof. Dr. Abel Dias dos Santos  
Prof. Dra. Laura Ribeiro

2020







# Resumo

A modelagem em várias escalas está sendo usada em diferentes tipos de indústrias, porque pode levar à otimização do design e do comportamento do material. Tendo em vista isso a pesquisa foi desenvolvida na Faculdade de Engenharia da Universidade do Porto (FEUP) em conjunto com o Instituto de Ciência e Inovação em Engenharia Mecânica e Industrial (INEGI), no âmbito do projeto msCore, que tem como proposta a elaboração de um metodologia que permite análises em múltiplas escalas no projeto e fabricação de materiais avançados.

Para cumprir o objetivo foi feito uma revisão bibliográfica e implementado um modelo para prever a matriz de elasticidade a partir da microestrutura, através do método dos elementos finitos. Esse modelo foi desenvolvido recorrendo à utilização de códigos implementados em Python, para extrair as informações obtidas da análise da microestrutura e em MATLAB para obter a homogeneização das propriedades. Utilizaram-se imagens reais de microestruturas de partes de uma peça, obtida por fundição, com diferentes taxas de resfriamento que resultam em diferentes fases. Foram ainda testados efeitos da possível existência de poros e trincas ao nível da microestrutura.

Os resultados obtidos mostraram que a matriz de elasticidade homogeneizada, obtida a partir da microescala para as diferentes taxas de arrefecimento, se diferencia da matriz teórica e que a inclusão de poros e trincas também influencia as propriedades à microescala. A partir da análise térmica à macroescala foi possível verificar que na peça existem diferentes taxas de resfriamento em diferentes locais e que a atribuição de diferentes fases a esses locais, a partir das taxas de resfriamento consideradas, resulta no aparecimento de tensões residuais.

**Palavras-chave:** Modelo de Multiescala; Homogeneização; Propriedades elásticas; Materiais heterogêneos; micromecânica; tensões residuais, MEF, análise térmica, análise de tensões, microestrutura, OOF2; Abaqus; fundição.

# Abstract

Multi-scale modeling is being used in different types of industries because it can lead to the optimization of material design and behavior. In view of this, the research was developed at the Faculty of Engineering of the University of Porto (FEUP) in conjunction with the Institute of Science and Innovation in Mechanical and Industrial Engineering (INEGI), within the scope of the msCore project, which aims to develop a methodology that allows multi-scale analysis in the design and manufacture of advanced materials.

To meet the objective, a bibliographic review was made and a model was implemented to predict the elasticity matrix from the microstructure, through the finite element method. This model was developed using code implemented in Python, to extract the information obtained from the microstructure analysis and in MATLAB to obtain the homogenization of the properties. Real images of microstructures of different zones of a part obtained by casting were used, with different cooling rates that result in different phases. Effects of the possible existence of pores and cracks at the level of the microstructure were also tested.

The results showed that the homogenized elasticity matrix, obtained from the microscale for the different cooling rates, differs from the theoretical matrix and that the inclusion of pores and cracks also influences the properties to the microscale. From the thermal analysis to the macroscale it was possible to verify that in the part there are different cooling rates in different locations and that the assignment of different phases to these sites, from the cooling rates considered, results in the appearance of residual stresses.

**Keywords:** Multi-Scale models; Homogenization; Elasticity properties; Heterogeneous materials; Micromechanics; Residual Stress; FEM; Thermal analysis; Stress analysis; Microstructure; OOF2; Abaqus; Casting

# Acknowledgments

Gostaria de agradecer ao meu orientador Prof. César de Sá, pela atenção e o tempo dedicado a realização dessa tese, aos professores Abel Santos e Laura Ribeiro, pela oportunidade de fazer parte do projeto NORTE-01-0145-FEDER032419 – “msCORE - Multiscale methodology with model order reduction for advanced materials and processes and Project POCI-01-0145-FEDER-032466 – NanosFLiD” cofinanciado pelo Programa Operacional Regional do Norte (NORTE2020) e pelo Programa Operacional de Competitividade e Internacionalização (Compete2020), através do Fundo Europeu de Desenvolvimento Regional (FEDER) e pela Fundação para a Ciência e Tecnologia (FCT) através da sua componente do orçamento do estado.

Agradeço também ao INEGI e a FCT, por suportar financeiramente o desenvolvimento dessa pesquisa

Aos meus pais Lucilene e Humberto, que mesmo longe, nunca deixaram de me incentivar e apoiar nas minhas escolhas, as minhas irmãs Claudia, Laura e Marília pela paciência em sempre me ouvir. Aos meus amigos Levi, Val, André e Dani pela paciência e o suporte emocional. Muito obrigada a todos que de alguma maneira contribuíram na minha jornada.



# Contents

<b>Resumo .....</b>	<b>5</b>
<b>Abstract .....</b>	<b>6</b>
<b>Acknowledgments.....</b>	<b>7</b>
<b>List of figures .....</b>	<b>10</b>
<b>List of Tables.....</b>	<b>15</b>
<b>Abbreviations and Symbols.....</b>	<b>17</b>
<b>CHAPTER 1 Introduction.....</b>	<b>19</b>
1.1 Main objective.....	21
<b>CHAPTER 2 Materials characterization and Duplex 2205 Steel .....</b>	<b>24</b>
2.1 Nanoscale analysis and defects .....	24
2.2 Microscale analysis and defects .....	29
2.3 Steel.....	32
2.3.1 The Duplex Stainless Steel 2205.....	34
<b>CHAPTER 3 Continuum Mechanics and the Finite Element Method. ....</b>	<b>38</b>
3.1 Continuum mechanics .....	38
3.1.1 Kinematics of motion .....	39
3.1.2 Stress measures .....	43
3.1.2.1 Cauchy stress tensor .....	43
3.1.2.2 First Piola-Kirchhoff stress tensor.....	43
3.1.2.3 Kirchhoff stress tensor .....	44
3.1.3 Weak equilibrium - The principle of virtual work .....	44
3.2 Finite Element Method.....	45
<b>CHAPTER 4 Multiscale analysis – Microscale .....</b>	<b>50</b>
4.1 Micromechanics .....	50
4.2 Homogenization of elastic properties.....	52
4.2.1 Hooke’s Law .....	52



4.2.2 Thermal stress and strain .....	56
4.2 Numerical model.....	59
<b>CHAPTER 5 Influence of porosity and fractures in the micro and macrosctruture</b> .....	<b>80</b>
5.1 Elastic properties affected by the existence of pores.....	82
5.2 Elastic properties affected by the existence of fractures.....	85
<b>CHAPTER 6 Macroscale – Thermal/mechanical analysis. ....</b>	<b>89</b>
6.1 Thermal analysis .....	90
6.2 Mechanical analysis .....	98
<b>CHAPTER 7 Conclusion and future work.....</b>	<b>106</b>
<b>References.....</b>	<b>109</b>
<b>Appendix A: Tables of properties .....</b>	<b>113</b>
<b>Appendix B: Rectangle and tensile specimens dimension.....</b>	<b>117</b>
<b>Appendix C: MATLAB Scrip .....</b>	<b>118</b>
<b>Appendix D: Python Scrip .....</b>	<b>123</b>
<b>Appendix E: Abaqus file .....</b>	<b>129</b>

# List of figures

Figure 1.1 Part geometry, C-Ring dimensions,.....	22
Figure 2.1 Electronic structure of atoms (atoms, molecules and ions, 2020) .....	25
Figure 2.2 Common metallic Crystal structures (Britannica, 2020) .....	27
Figure 2.3 Two-dimensional representations of a vacancy (adapted from Callister, W.D. (2010) Materials Science and Engineering, p. 92) .....	28
Figure 2.4 Two-dimensional representations of a Self-interstitials (adapted from Callister, W.D. (2010) Materials Science and Engineering, p. 92] .....	28
Figure 2.5 Two-dimensional schematic representations of substitutional and interstitial impurity atoms. (adapted from Callister, W.D. (2010) Materials Science and Engineering, p. 94]). .....	29
Figure 2.6 (a) Crystalline structure of grains in the solidification process. (b) The grain structure as it would seem under the microscope; dark lines represent the grain boundaries. (Adapted from Callister, W.D. (2010) Materials Science and Engineering, p. 73]) .....	30
Figure 2.7 Schematic representation of a dislocation that has edge, screw, and mixed character. (Callister, W.D. (2010) Materials Science and Engineering, p. 100).....	31
Figure 2.8 micrograph of a titanium alloy in which the dark lines are dislocations with dislocation defect. (Callister, W.D. (2010) Materials Science and Engineering, p. 102.) .....	31
Figure 2.9 Schematic diagram displaying small and high-perspective grain obstacles and the adjacent atom positions. (Callister, W.D. (2010) Materials Science and Engineering, p. 103).....	32
Figure 2.10 Iron-Carbon Phase Diagram (FACTORY (2020) .....	33
Figure 2.11 Add the nickel content changes the microstructure of stainless-steel from ferritic (left) to duplex (middle) to austenitic (right)). (IMOA,2014 p. 9).....	35
Figure 2.12 - 2205 duplex stainless steel Isothermal precipitation diagram. (IMOA, 2014, p. 9).....	36

Figure 2.13 Cooling from the solution annealing temperature should be fast enough (cooling curve A) to avoid the sigma phase-field (cooling curve B). (IMOA,2014 p. 9) .....	36
Figure 2.14 Microstructure of 2205 sample aged at 850°C (1560°F) for 40 minutes showing sigma precipitation (arrows) on the austenite/ferrite grain boundaries. The ferrite (f) phase appears darker than the austenite (A) phase in the micrograph). (IMOA, 2014 p. 9) .....	37
Figure 3.1 Deformation of the body, (Neto, S., Peric, D., and Owen, D. R. J. (2008). Computational Methods for Plasticity p. 42).....	39
Figure 3.2 – Rigid deformation (Neto, S., Peric, D., and Owen, D. R. J. (2008). Computational Methods for Plasticity p. 42).....	40
Figure 3.3: Finite element interpolation. The element and global shape function. (adapted from Neto, S., Peric, D., and Owen, D. R. J. (2008). Computational Methods for Plasticity p. 86) .....	46
Figure 4.1: Macro and microscales and respective sizes (JReis,2015).....	52
Figure 4.2 Stress-strain curve - Representation of the elastic domain.....	53
Figure 4.3 Components of stress in three dimensions. ....	55
Figure 4.4- Expansion and contraction deformation caused by temperature variation. .	56
Figure 4.5 Representation of traction in x, y and shear in x and y .....	59
Figure 4.6 Boundary conditions of thermal simulation .....	61
Figure 4.7 Boundary conditions for 3D simulation .....	62
Figure 4.8 Sparse and refined triangle meshes .....	67
Figure 4.9 Unstructured and structured meshes.....	68
Figure 4.10 Tetrahedron mesh .....	69
Figure 4.11 Duplex microstructure first test.....	70
Figure 4.12 cooling rate of 6.6°Cmin-1 .....	71
Figure 4.13 cooling rate of 7.2°Cmin-1 .....	71
Figure 4.14 cooling rate of 156°Cmin-1 .....	72

Figure 4.15 Mesh biphasic microstructure first test .....	73
Figure 4.16 Mesh cooling rate of 6.6°Cmin-1 .....	74
Figure 4.17 Mesh cooling rate of 7.2°Cmin-1 .....	74
Figure 4.18 Mesh cooling rate of 156°Cmin-1 .....	75
Figure 4.19 cooling rate of 6.6°Cmin-1, deformation in the x-direction .....	77
Figure 4.20 cooling rate of 6.6°Cmin-1, deformation in the y-direction .....	77
Figure 4.21 cooling rate of 7.2°Cmin-1, deformation in the x-direction .....	78
Figure 4.22 cooling rate of 7.2°Cmin-1, deformation in the y-direction .....	78
Figure 4.23 cooling rate of 152°Cmin-1, deformation in the x-direction .....	79
Figure 4.24 cooling rate of 152°Cmin-1, deformation in the y-direction .....	79
Figure 5.1 Pressure and gas content along the mushy zone of ductile.....	81
Iron (From D. M. Stefanescu*1 and A. V. Catalina,20) .....	81
Figure 5.2 Microstructure Mesh cooling rate 156°Cmin-1, four pores inclusion, region A. .....	83
Figure 5.3 Microstructure Mesh cooling rate 156°Cmin-1, four pores inclusion, region B. .....	83
Figure 5.4 Microstructure Mesh cooling rate 156°Cmin-1, High pores inclusion.....	84
Figure 5.5 Microstructure Mesh cooling rate 7.2°Cmin-1, Horizontal Crack Region A, 30 nm.....	85
Figure 5.6 Microstructure Mesh cooling rate 7.2°Cmin-1, Horizontal Crack Region A, 60 nm.....	86
Figure 5.7 Microstructure Mesh cooling rate 7.2°Cmin-1, Vertical crack .....	86
Figure 5.8 Microstructure Mesh cooling rate 7.2°Cmin-1, Inclined Crack 30 nm .....	87
Figure 5.9 Microstructure Mesh cooling rate 7.2°Cmin-1, Inclined Crack 480 nm .....	87
Figure 6.1 Part geometry, C-Ring dimensions .....	90
Figure 6.2 Cooling down a) In mould; b) Out of the mould .....	91
Figure 6.3 Selected point at C- Ring. ....	92

Figure 6.4 Curves of cooling at different locations .....	92
Figure 6.5 Curves of cooling rates at different locations.....	93
Figure 6.6 C-Ring with the different phases.....	94
Figure 6.7 Vertex 3 was fixed in X and Y direction.....	94
Figure 6.8 Thermal stress analysis at C-Ring.....	95
Figure 6.9 Deformation analysis at the C-Ring, before and after cooling.....	95
Figure 6.10 Piece with the different phases.....	96
Figure 6.11 Middle point fixed in X and Y direction.....	96
Figure 6.12 Thermal stress analysis at the piece .....	97
Figure 6.13 Deformation analysis at the piece, before and after cooling.....	98
Figure 6.14- Specimen and boundary conditions.....	99
Figure 6.15 Curve of tensile test for Duplex microstructure specimen and the curve of stainless steel 2205.....	99
Figure 6.16 Curve of tensile test for 6.6°C/min microstructure specimen and the curve of stainless steel 2205.....	100
Figure 6.17 Curve of tensile test for 7.2°C/min microstructure specimen and the curve of stainless steel 2205.....	100
Figure 6.18 Curve of tensile test for 156°C/min microstructure specimen and the curve of stainless steel 2205.....	101
Figure 6.19 Curve of tensile test for low porosity 156°C/min microstructure specimen, region A, B and 156°C/min microstructure with no porosity.....	102
Figure 6.20 Curve of tensile test for High porosity 156°C/min microstructure specimen, region A, B and 156°C/min microstructure with no porosity.....	102
Figure 6.21 Curve of tensile test for crack 30nm and 60 nm in 7.2°C/min microstructure specimen and 7.2°C/min microstructure with no cracks.....	104
Figure 6.22 Curve of tensile test for vertical crack in 7.2°C/min microstructure specimen and 7.2°C/min microstructure with no cracks .....	104

Figure 6.23 Curve of tensile test for inclined crack 30nm and 480 nm in 7.2°C/min microstructure specimen and 7.2°C/min microstructure with no cracks .....104

# List of Tables

Table 4.1 Comparison of the elasticity matrices obtained, with sparse and refined meshes, with the theoretical matrix. ....	67
Table 4.2 Comparison of the elasticity matrices obtained, with structured and unstructured meshes, with the theoretical matrix .....	68
Table 4.3 Expansion coefficient comparison with theory coefficient, in x and y-direction. ....	69
Table 4.4 Elasticity Matrix comparison with theory matrix .....	69
Table 4.5 Austenite, ferrite and sigma properties .....	72
Table 4.6 subtitle corresponding the material colours at the simulation .....	73
Table 4.7 Results of elasticity matrix from different micrographic structures .....	76
Table 4.8 Results of expansion coefficient from different micrographics .....	76
Table 5.1 Elasticity Matrix comparison, Low porosity- Region A, B and high porosity, with matrix without pores. ....	84
Table 5.2 Elasticity Matrix comparison, horizontal crack, 30 and 60 mm, vertical and inclined crack, with matrix without cracks. ....	88
Table 6.1 Young Modules of tensile test microstructure specimen and the stainless steel 2205. ....	101
Table 6.2 Young Modules of tensile test for porosity microstructure specimen and the stainless steel 2205.....	103
Table 6.3 Young Modules of tensile test microstructure specimen and the stainless steel 2205. ....	105





# Abbreviations and Symbols

## Notation

$a$	Scalar value
$\mathbf{a}$	Vector
$\mathbf{A}$	Second-order tensor
$\mathbf{a}, \mathbf{A}$	Finite elements' related array (vector and tensor)
$\mathbf{A}$	Fourth-order tensor
$\mathcal{A}$	Space

## Acronyms

2D	Two dimensional;
3D	Three dimensional;
RVE	Representative Volume Element;
RCU	Repeating unit cell
OOF2	Object-Oriented Finite-Elements, version 2;
CT	Computed Tomography;
SEM	Scanning Electron Microscope;
nGP	Number of Gauss Points;

## Indices

$(\cdot)_t$	Time dependent; Transversal direction;
$(\cdot)_{iso}$	Isochoric component;
$(\cdot)_v$	Volumetric component;
$(\cdot)_d$	Discretized domain;
$(\cdot)_e$	Element domain;
$(\cdot)_g$	Global domain;
$(\cdot)_p$	Reference to the plane;
$(\cdot)_*$	Homogenized (Effective) property;

## Operators

$(\dot{\cdot})$  Time derivative;  
 $(\ddot{\cdot})$  Second time derivative;  
 $\partial(\cdot)$  Boundary domain;  
 $\partial(\cdot)/\partial a$  Derivative with respect to  $a$ ;  
 $\nabla(\cdot)$  Gradient;  
 $\nabla_s(\cdot)$  Symmetric gradient;  
 $\nabla_x(\cdot)$  Spatial gradient;  
 $\nabla_X(\cdot)$  Material gradient;  
 $\text{tr}(\cdot)$  Trace;  
 $\det(\cdot)$  Determinant;  
 $|\cdot|$  Scalar norm (absolute value);  
 $\|\cdot\|$  Euclidean norm of vectors or tensors;  
 $\ln(\cdot)$  Natural logarithm;  
 $\text{div}_x(\cdot)$  Spatial divergence;  
 $\text{div}_X(\cdot)$  Material divergence;

# CHAPTER 1

## Introduction

Most of the engineering materials are considered continuum homogeneous at the macroscale. However, at the micro or nanoscale, all real materials are heterogeneous. This heterogeneous nature has a significant impact on material properties at the macroscale as they depend strongly on the spatial distribution, size, shape, and direction of atoms, internal structures, and grains. Besides, thermal-mechanical loading may also affect the microstructural morphology.

The multiscale materials modeling lies in predicting relations between microstructural morphology, processing, and the final properties. Its use encompasses a variety of computational techniques as molecular dynamics, multiscale finite element methods, computational homogenization, to name a few. Multiscale Science and Engineering is a relatively new field. The advance of multiscale modeling in the industrial sector was mostly due to financial motivations. Once industry realized that the notions of multiscale modeling and simulation-based design were invariant to the type of product

and that effective multiscale simulations could lead to design optimization, a paradigm shift began to occur, in various measures within different industries, as cost savings and accuracy in product warranty estimates were rationalized. (Horstemeyer, Mark., 2009).

The advantages of using multiscale modeling, according to (Horstemeyer, Mark., 2009), are: to reduce product cost through innovation in material, product, and process designs; to reduce the product development time by alleviating costly trial-and-error iterations, to reduce the number of costly large systems scale experiments; to increase product quality and performance by providing more accurate predictions of response to design loads; to help developing new materials.

In this work, a casting process, involving a stainless-steel duplex 2205 will be addressed. Residual stress and defects as micro-cracks and micro porosities, which are non-visual at the macroscale, are some aspects that often occur in a casting process that should be minimized, as all of them affect the properties of materials directly. The residual stresses are the system of stresses which can exist in a body when it is free from external forces (Dieter, George Ellwood, and D. J. Bacon, 1988). They are produced when a body is submitted to nonuniform plastic deformations or thermal variations and maybe the cause of the initiation of internal cracks. Microporosity locates mainly in places in which solidification lastly occurs. Alloys that have a wide solidification interval are more susceptible to the formation of micro porosities, independently of the molding process.

The stainless-steel duplex 2205, which will be addressed in this study, is a common alloy used in the casting process. Its mechanical resistance is two times more than normal austenitic steel. It has excellent tensile strength, good weldability, and moderate impact resistance with temperature up to 80 ° C. It is typically used in offshore oil and gas processes, chemical processing, desalination plants, cellulose and paper, sculptures in coastal locations, brewing, food processing, bridges, tensioning systems, nuclear reprocessing, tunnel infrastructure, chemical tankers, among many other applications. (Aprodinox, 2020)

A new framework has been developed aiming to understand the behavior of alloys is stainless steel in casting problems and the formation of defects through multiscale modeling and finite element method

## **1.1 Main objective**

Different microstructure distributions and the existence of porosities result in different homogenized properties of a material at the macro scale in different regions of a part, influencing its overall response to mechanical and thermal loads. Having that in view, the main objective of this work is to create simple predictive tools, using homogenization techniques and a finite element method framework, to find the different properties of the material in part, ranging from elasticity and thermal properties to the detection of possible residual stresses and fracture locations.

## **1.2 Main tasks performed**

Throughout the work, a bibliography study on the different issues addressed was carried out. Initially, MATLAB and Python scripts were created to obtain a local homogenized elasticity matrix and thermal properties from sample heterogeneous material representative regions. Simple validation tests were performed. The effect at the macroscale of the inclusion of pores and fractured locations was tested. Finally, an analysis of a part, obtained in a casting process, C-Ring, figure 1.1, was made, attributing different material properties to different regions of the part, according to the knowledge of their cooling rates distribution and their corresponding effect on the final local microstructure.

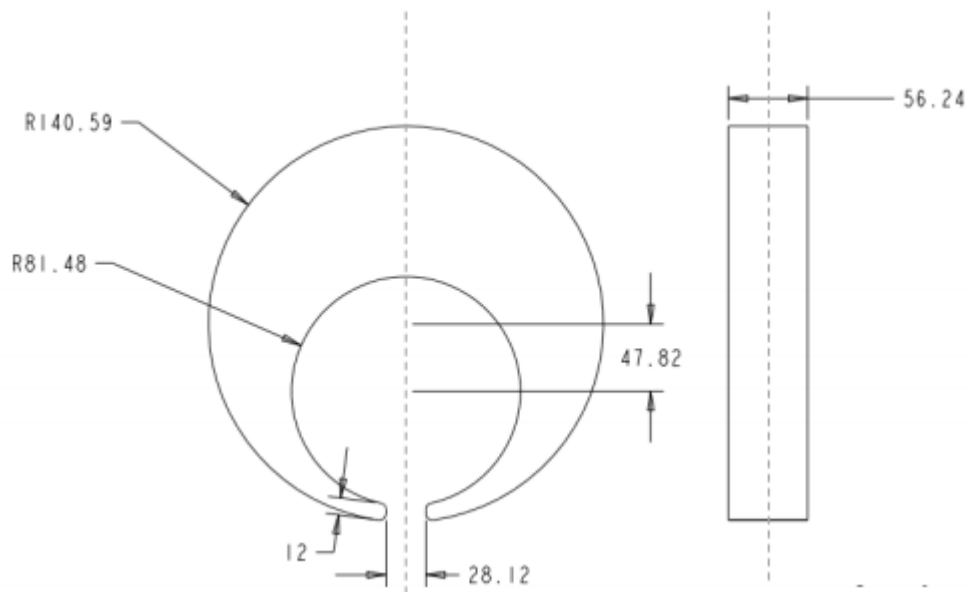


Figure 1.1 Part geometry, C-Ring dimensions,

### 1.3 Structure of the document

This thesis is divided into seven chapters and includes, in the end, a reference section and appendix. Brief information on chapters 2 to 7 is shown below.

#### Chapter 2

In chapter 2, the concepts regarding materials characterization and, in particular, on the material used, steel duplex 2205, are presented. The relation between the electron distribution, atoms, molecules, and internal structures and grains, in general, and all these concepts applied to the steel. Besides, the phases diagrams transformation, and the time-temperature transformation (TTT) of steel duplex 2205, are presented. It is an important topic to understand and justify the next chapters.

#### Chapter 3

A brief introduction of continuum mechanics, the concept regarding a homogenization and a brief reference to the finite element method, are presented in this chapter, those concepts are essential to understand the type of approach that was

undertaken to obtain homogenized properties at the macroscale.

#### **Chapter 4**

The multiscale analysis study is shown in this chapter, emphasizing the use of the microscale, because the first step was to consider the different microstructures as heterogeneous, which means that different homogenized properties will be found at the macroscale. A simple example is used to validate the code developed and the error associated are presented. The results also are presented for four microstructures with different phase distributions and obtained at different cooling rates.

#### **Chapter 5**

Porosity and Cracks are a common defect from the casting process, and the goal of this chapter is to understand how they are formed and how they could affect the material properties at the macroscale. Some tests are performed, including some porosity at the microscale. The XFEM (extended finite element method) was used the microscale level to take into account the crack effect.

#### **Chapter 6**

In this chapter departing from a set of concepts and results described in the previous chapters, an application on the simulation of the behavior of a part, at the macrostructure level, namely a C-ring part under thermal and mechanical loading, is undertaken, paying special attention to the evaluation of resulting residual stresses.

#### **Chapter 7**

The main conclusions of the work described in the previous chapters are outlined in this chapter. The importance of this study is discussed, and suggestions for future work are made.

## **CHAPTER 2**

# **Materials characterization and Duplex 2205 Steel**

The knowledge on how the atoms, molecules, chemical bonds and the internal structures of materials, grains and their orientation, can affect the properties of a part at the macroscale, or in other words, how the characterization of material at the micro and nanoscale is foremost for the characterization of material thermal, mechanical, and electric properties at the macroscale, is an important subject on Materials Science study.

For better understanding, aspects on this topic were divided here into three parts: firstly, a reference to nanoscale aspects, secondly, the other important aspects at the microscale and finally their influence at the macroscale properties of steel, in particular the duplex 2205 steel, utilized in this work.

### **2.1 Nanoscale analysis and defects**



Some of the critical properties of solid materials depend on geometric arrangements of atoms, and of interactions that exist between atoms or molecules constituents as well. (Callister, W.D, 2010). There are some important topics to discuss, and they are electronic configuration, atoms, bonds, and internal structures.

Atoms consist of a tiny nucleus composed of protons and neutrons, which are surrounded by moving electrons. Protons and electrons are electrically loaded, and the different elements are characterized per number of protons that exist in them. This electronic structure that is shown in figure 2.1 is that determines the type of bond the element may have. The existent electrons characterize which is gain or loss in a chemical bond. There are many studies to predict the behavior of the electrons and the most famous and used of them is the Bohr model, that divides the electrons in layers of less or more energy, determining the ease of gaining or losing electrons that are in valence layer, and the Pauli model that divides the electronic cloud into sublayers and with a maximum of 2 electrons sublevels with opposite spin.

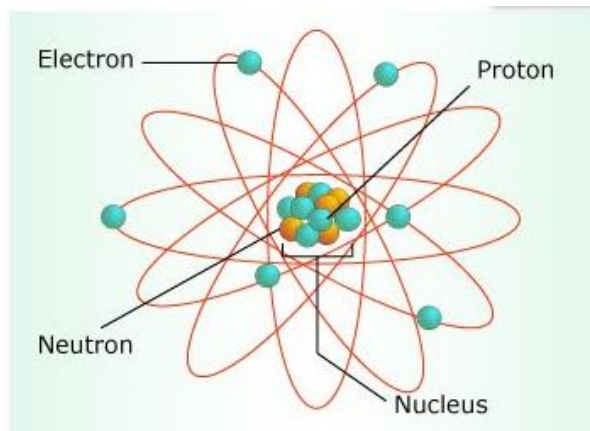


Figure 2.1 Electronic structure of atoms (atoms, molecules and ions, 2020)

There are two types of chemical bonds, they are the strong (primary) bond as ionic, covalent, or metallic, which are found in molecules and crystals, and the weak bond (secondary), that is the Van der Waals bonds or hydrogen bonds. The most common bond found in metals is the metallic bond.

The metallic bonding is found in metals and their alloys. A simple model has been proposed that approximates the bonding scheme. There are one, two, or three valence

electrons at metallic materials. The valence electrons have more or less freedom to drift throughout the entire metal because they are not bonded to any particular atom in the solid. They may be thought of as belonging to the metal as a whole or forming a “sea of electrons” or an “electron cloud.” The remaining nonvalenced electrons and atomic nuclei form the ion cores, which possess a net positive charge equivalent in magnitude to the total valence electron charge per atom. (Callister, W.D, 2010).

After the bonding in metal, the atoms get organized in two different ways, forming either a crystalline structure or a noncrystalline (amorphous) structure. A crystalline material has a regularity distribution of atoms over vast atomic distances; at solidification and the atoms organize themselves in a repetitive three-dimensional shape, in which each atom is bonded to its nearest-neighbor atoms. Otherwise, in a non-crystalline material, the distribution of the atoms does not have any regularity or order. In all metallic materials, there is a crystalline structure.

The small groups of atoms form repetitive patterns, which indicates the atomic order in crystalline solids. Hence, it is often appropriate to subdivide the structure into small repeat entities to describe the crystal structures, which is called unit cells (Callister, W.D, 2010).

There are three types of typical metallic crystalline structure. They are:

- the Face-Centered Cubic Crystal Structure (FCC), in which the crystal structure has a unit cell of cubic geometry, with atoms positioned at each of the corners and the centers of all the cube faces and that is typically found at metals (represented by the first cube of figure 2.2);
- Body-centered cubic (BCC) – that has also a cubic unit cell with atoms situated at all eight corners and a single atom at the cube center (represented by the second cube of figure 2.2);
- Hexagonal close-packed (HCP) - which has a hexagonal unit cell; it is formed by six atoms with a regular hexagon arrangement and a single atom in the center at the top and bottom planes and has another level between those planes with three additional atoms, which have as nearest neighbors atoms in both of those adjoining two planes; the equivalent of six atoms is

contained in every unit cell; one-sixth of each of the 12 top and bottom face corner atoms, one-half of each of the two centers face atoms, and all three midplane central atoms (Callister, W.D, 2010) (represented in the hexagonal prism of figure 2.2).

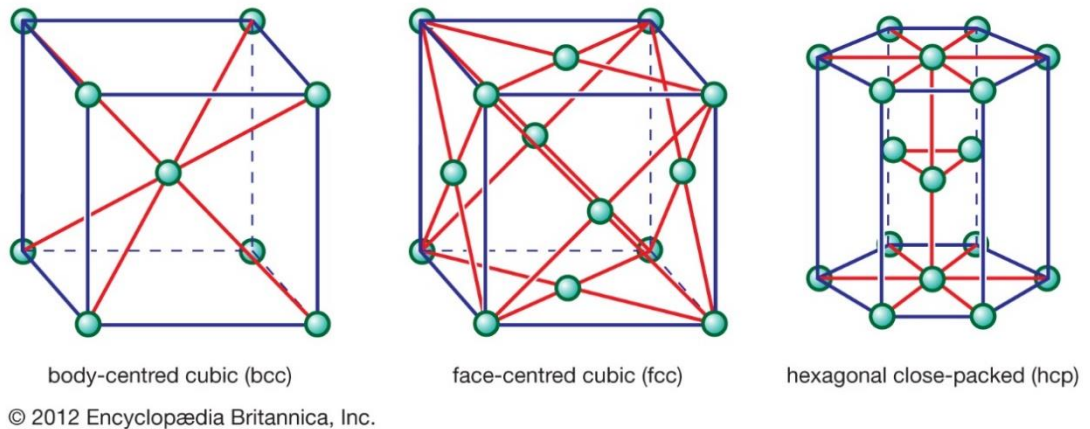


Figure 2.2 Common metallic Crystal structures (Britannica, 2020)

On the atomic scale, we can have a perfect crystalline structure in some pieces of the part, but it is difficult to have in all material a structure without any imperfections. It is common to have defects, in atomic sizes, like vacancies, self-interstitials, or impurities.

The vacancies refer cases where in the crystalline structure one or more atoms are missing, letting a vacancy. This imperfection is shown in figure 2.3; they are formed during solidification and could be formed as a result of vibrations, at the atomic level. The presence of vacancies is necessary, because they increase the entropy of the crystal. The equilibrium number of vacancies for a given quantity of material increase as increases the temperature.

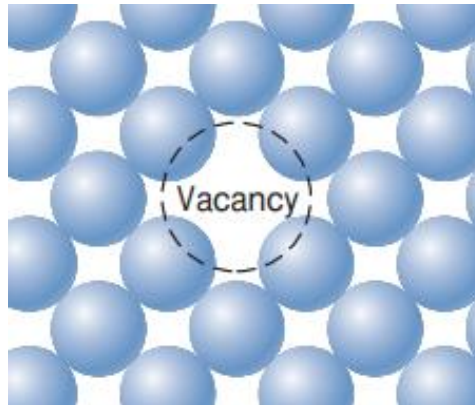


Figure 2.3 Two-dimensional representations of a vacancy (adapted from Callister, W.D. (2010) *Materials Science and Engineering*, p. 92)

Self-interstitials are when the atoms are compressed in a space that in reasonable condition is not occupied, that cause disturbs in crystallography grip, as the space is minor then the atoms. This type of imperfection is less probable to happen. It showed in figure 2.4.

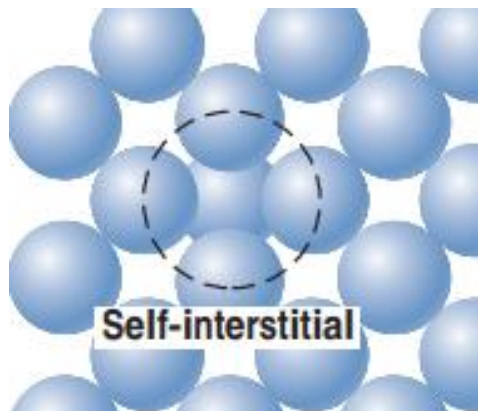


Figure 2.4 Two-dimensional representations of a Self-interstitials (adapted from Callister, W.D. (2010) *Materials Science and Engineering*, p. 92)

Impurities in solids occur when an atom of a different element appears on the crystallite grid. This imperfection of crystallite can occur when the atom of different elements replaces the original atom or when an atom appears in the middle of original atoms, interstitial. Figure 2.5 represents both problems.

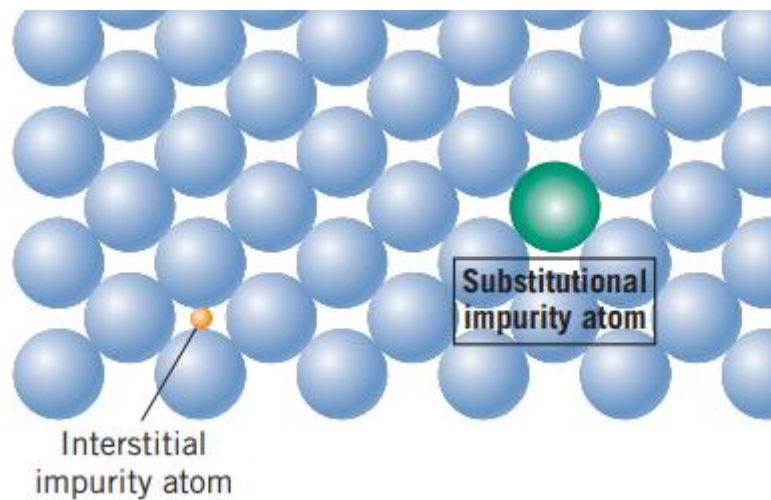


Figure 2.5 Two-dimensional schematic representations of substitutional and interstitial impurity atoms. (adapted from Callister, W.D. (2010) *Materials Science and Engineering*, p. 94).

## 2.2 Microscale analysis and defects

The crystalline solid may have a unique crystal, as garnet, or polycrystalline structure as most of the metals.

In a metal micrography, the first things we can observe are the grains and their boundaries. The grains in metal are nothing more than a conjunction of crystalline structures. These grains could have different orientations and size. Different grains in the same material could be named as phases and all properties found in a grain affect all material. Figure 2.6 is an example of that we can see in the crystalline structure and the grains and their boundaries.

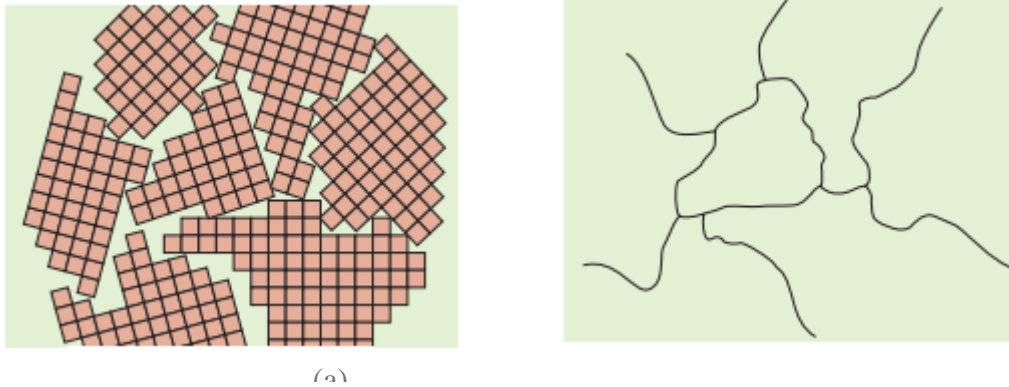


Figure 2.6 (a) Crystalline structure of grains in the solidification process. (b) The grain structure as it would seem under the microscope; dark lines represent the grain boundaries. (Adapted from Callister, W.D. (2010) *Materials Science and Engineering*, p. 73])

Properties in anisotropic materials depend on crystallographic direction, physical properties of the single crystals, which is not the case for isotropic materials.

Nevertheless, in many polycrystalline materials, the crystallographic orientations of the individual grains are random. Under these circumstances, despite the fact that every grain may be anisotropic, a specimen composed of the grain combination behaves isotropically. (Callister, W.D, 2010).

Some of the defects we can see in microscale are dislocations, interfacial defects, bulk or volume defects.

A dislocation is a linear or one-dimensional imperfection around which the number of the atoms are misaligned. Figure 2.7 shows the illustration of a dislocation that has the edge, screw, and blended character.

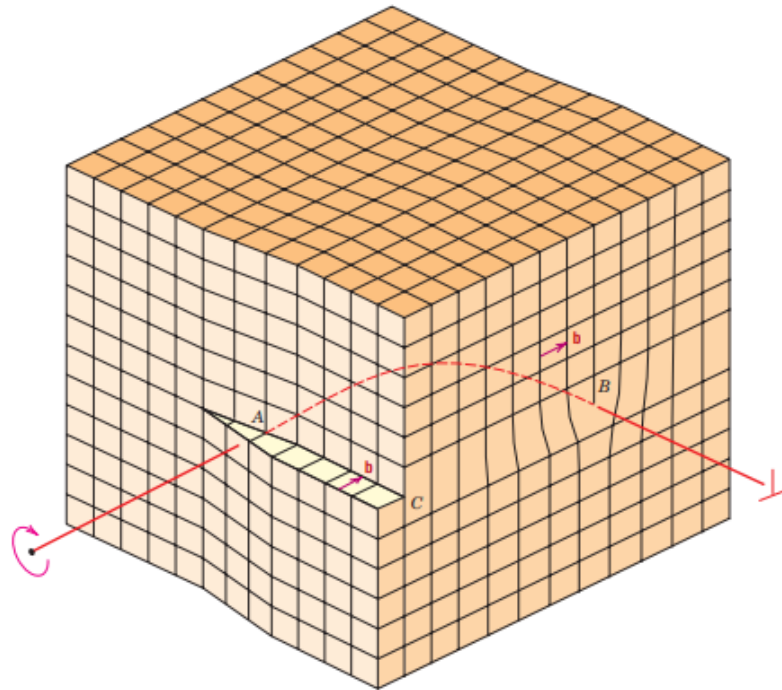


Figure 2.7 Schematic representation of a dislocation that has edge, screw, and mixed character. (Callister, W.D. (2010) *Materials Science and Engineering*, p. 100)

Figure 2.8 shows a micrograph of a titanium alloy in which the dark lines are dislocations with dislocation defect.

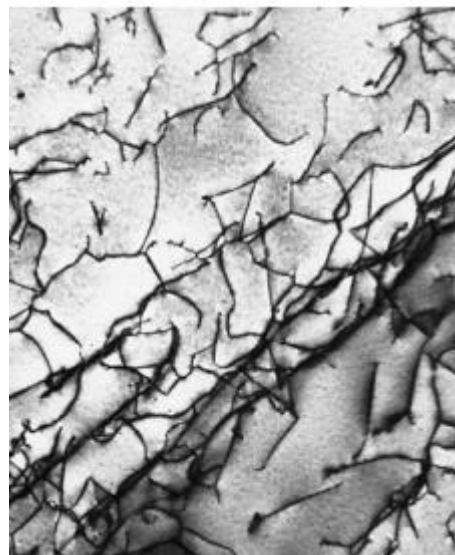


Figure 2.8 micrograph of a titanium alloy in which the dark lines are dislocations with dislocation defect. (Callister, W.D. (2010) *Materials Science and Engineering*, p. 102.)



Interfacial defects are found between two different crystal structures or crystallographic orientations and it is separated usually by boundaries that have two dimensions. These imperfections include external surfaces, grain boundaries, phase boundaries, twin boundaries and stacking faults. A schematic of this defect is exposed in figure 2.9.

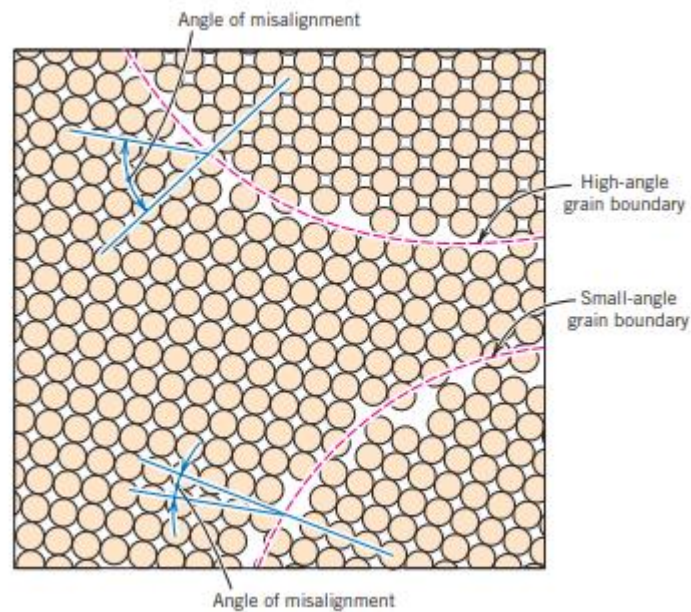


Figure 2.9 Schematic diagram displaying small and high-perspective grain obstacles and the adjacent atom positions. (Callister, W.D. (2010) *Materials Science and Engineering*, p. 103)

Bulk or volume defects are most prominent than the other discussed before. They include the cracks, porosity, foreign inclusions, and other phases.

### 2.3 Steel

The steel is composed of iron with around 2 per cent of carbon and it is normally used for infrastructure and industries. Pure iron, upon heating, has two changes in crystal structure before it melts. At room temperature, the solid form, called ferrite, or  $\alpha$ -iron, has a BCC crystal structure. Ferrite experiences a polymorphic transformation to FCC austenite, or  $\gamma$ -iron, at 912C (1674F). This austenite persists to 1394C (2541F), at which temperature the FCC austenite reverts to a BCC phase known as  $\delta$ -ferrite, which in the end melts at 1538C (2800F). (Callister, W.D, 2010)



### Effects of carbon

The carbon can be found in the steel in two forms, as carbide or in a solid solution, as austenite and ferrite. The carbide form can be found as iron carbide, cementite ( $\text{Fe}_3\text{C}$ ) or it can be a carbide of an alloying element. The cementite has up to 6.7% wt of carbon is a solid solution and is not deformable due to a high hardness, which makes it fragile. The other ways that the carbon is found is in ferrite ( $\alpha - \text{Fe}$ ) that has up to 0.035% wt of Carbon, BCC crystal structure, it is ductile, resistant and easily deformable, it is the main phase of steel and is easily found at iron at pure state. The Austenite ( $\gamma - \text{Fe}$ ) that has up to 2.1% wt of carbon, it is also ductile, easily deformable and resistant. Some other phases are found at the steel as martensite, bainite, perlite and sigma phase. At the equilibrium diagram, figure 2.10, it is possible to see the influence of carbon.

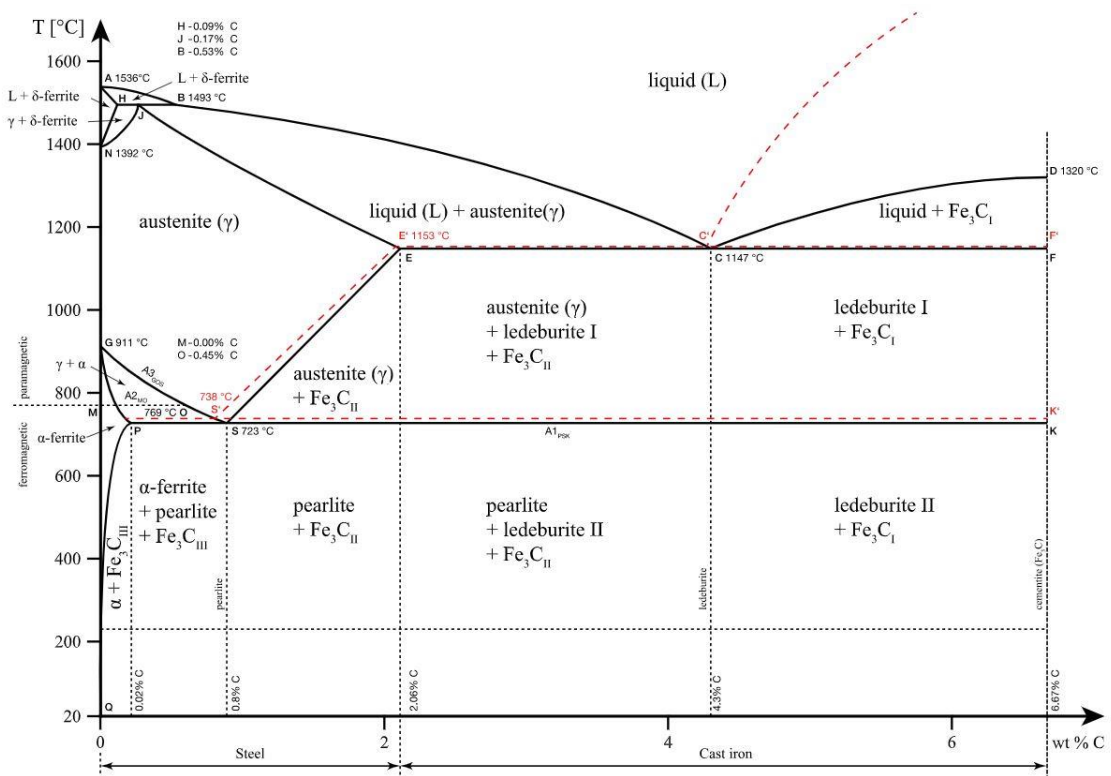


Figure 2.10 Iron-Carbon Phase Diagram (FRACTORY (2020))

### Effects of alloying

The temperature, cooling rate and holding time at the microstructures could be changed by adding alloying elements, which alter the transformation points between ferrite and austenite, regulate solution and diffusion rates, and compete with other factors in forming intermetallic compounds together with carbides and nitrides.

### **2.3.1 The Duplex Stainless Steel 2205**

Duplex stainless steel has excellent and high strength corrosion resistance, what explains why it is widely used. It is constituted by 22% Cr 5% Ni 3% Mo 0,15%N. Each element has an important role and this will be briefly addressed in the next paragraphs. For a more complete information the following reference may be consulted:(IMO, 2014).

The minimum chromium is set 10.5%, which is vital to form a stable passive film enough to defend steel against mild atmospheric corrosion. The corrosion resistance of a stainless-steel increases with increasing chromium content. Chromium may well be a ferrite producer, meaning that the addition of chromium promotes the body-centred cubic structure. At higher chromium contents, more nickel is critical to make an austenitic or duplex (austenitic-ferritic) structure. Chromium also grows the oxidation resistance at elevated temperatures. The chromium effect is critical because it influences the formation and removal of oxide scale or heat tint consequential from heat treatment or welding.

Nitrogen grows the corrosion resistance of austenitic and duplex stainless steels. It also substantially increases their strength and it is a low-cost alloying element and stable austenite producer, able to replace a variety of the nickel content for austenite stabilization. The improved sturdiness of nitrogen-bearing duplex stainless steel is a way to their higher austenite content and reduced intermetallic content. Nitrogen does not prevent the precipitation of intermetallic phases but delays their formation sufficient to allow processing and fabrication of the duplex grades. It is added to highly corrosion resistant austenitic and duplex stainless steels that contain high chromium and molybdenum content to counterpoise their propensity to form sigma phase. Nitrogen makes the strength of the austenite phase to grow by primary solution strengthening and also increases its work hardening rate. In duplex stainless steels, nitrogen is commonly added and, thus, the quantity of nickel is adjusted to achieve the desired phase

equilibrium. The ferrite producer, chromium and molybdenum, are stable by austenite producer nickel and nitrogen, to develop the duplex structure.

Nickel is a stabilizer of austenite, since it promotes a change of the crystal structure of stainless-steel from body-centred cubic (ferritic) to face-centred cubic (austenitic). Figure 2.11 has an exemplification of that. The ferritic stainless steels contain little or no nickel, duplex stainless steels contain low to average total of nickel from 1.5 to 7 %. The addition of nickel postpones the appearance of unfavorable intermetallic levels in austenitic stainless steels but is much less powerful than nitrogen in delaying their formation in duplex stainless steels. The face-center cubic structure is responsible for the outstanding toughness of the austenitic stainless steels. Its presence, in about 1/2 the microstructure of duplex grades, significantly increases their toughness compared to ferritic stainless steels.

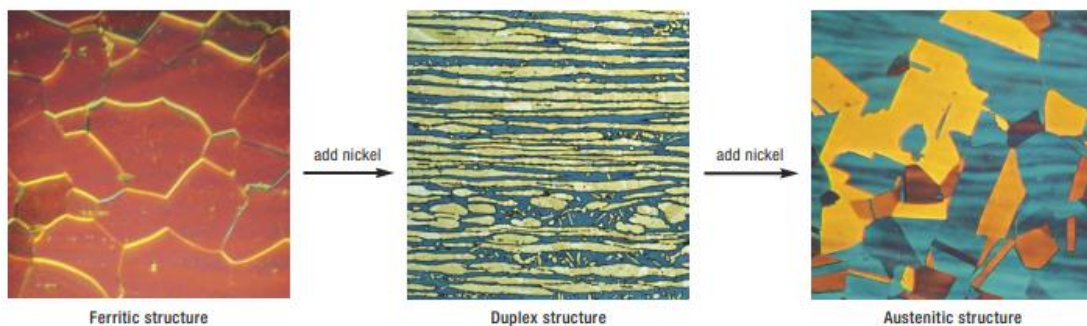


Figure 2.11 Add the nickel content changes the microstructure of stainless-steel from ferritic (left) to duplex (middle) to austenitic (right)). (IMO, 2014 p. 9)

In figures 2.12 and 2.13 we can find a time-temperature-transformation diagram, through which we can check or predict the microstructure that can be found in a part, according to knowledge of the cooling curve. The sigma phase can be avoided if we have a fast cooling rate.

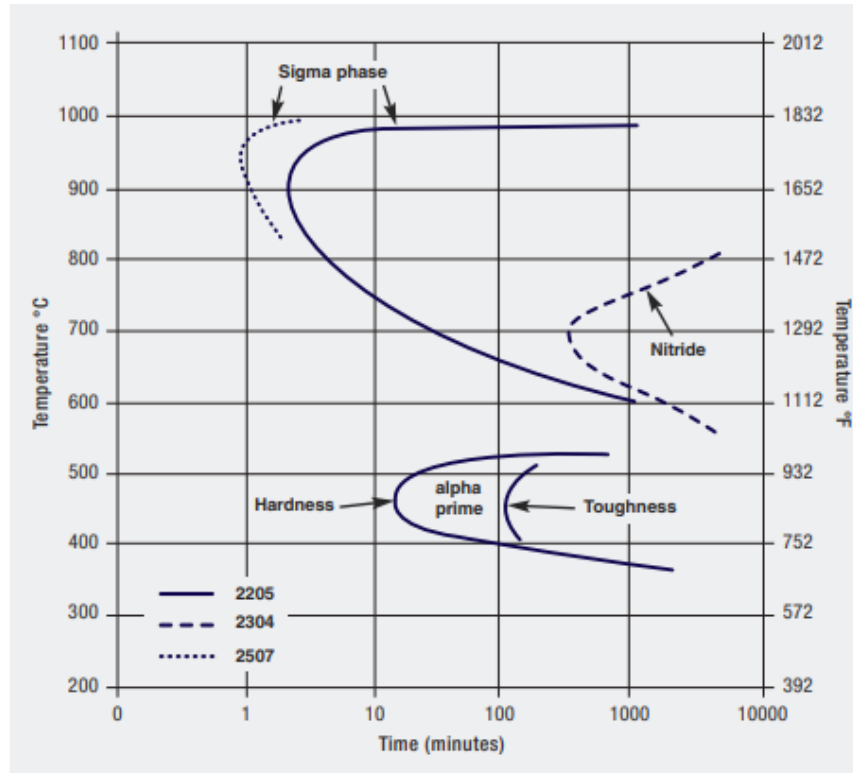


Figure 2.12 - 2205 duplex stainless steel Isothermal precipitation diagram. (IMO, 2014, p. 9)

In figure 2.13, we have two different cooling curves, the curve A which does not have a sigma phase, and the curve B which has the sigma phase in its microstructure. Figure 2.14 shows the microstructure for the case of curve B.

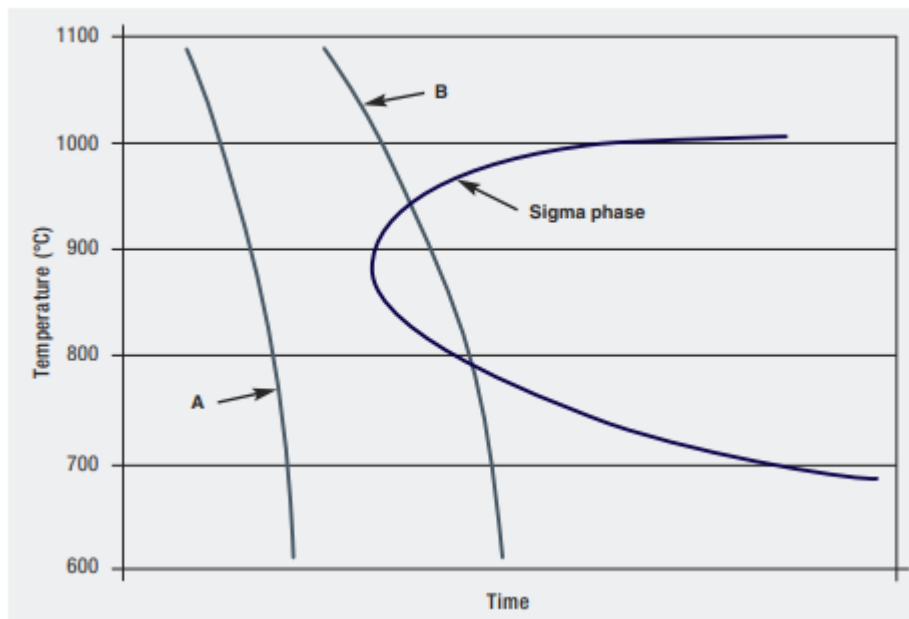
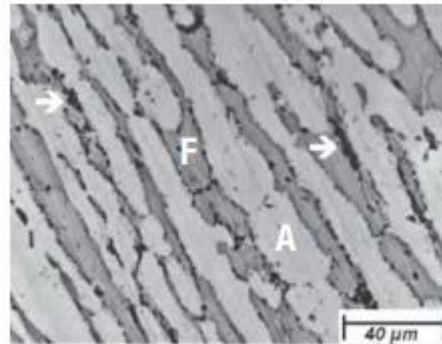


Figure 2.13 Cooling from the solution annealing temperature should be fast enough (cooling curve A) to avoid the sigma phase-field (cooling curve B). (IMO, 2014 p. 9)



*Figure 2.14 Microstructure of 2205 sample aged at 850°C (1560°F) for 40 minutes showing sigma precipitation (arrows) on the austenite/ferrite grain boundaries. The ferrite (f) phase appears darker than the austenite (A) phase in the micrograph). (IMOA, 2014 p. 9)*

## **CHAPTER 3**

# **Continuum Mechanics and the Finite Element Method.**

A brief overview of concepts of continuum mechanics, as well as, of the finite element method that are essential to comprehend following aspects and developments of this work will be undertaken in this chapter. Therefore, the chapter is divided into two main parts. The first addresses basic continuum mechanics concepts relating to the kinematics of motion and strain and stress measures, and the second is addresses the finite element method main concepts.

### **3.1 Continuum mechanics**

According to the continuum mechanics concept an homogenous continuum solid could be divided and subdivided into infinitesimal subvolumes, each of which retaining the properties of the bulk material.

The mathematician Augustin-Louis Cauchy was the first to formulate this concept that lies in a combination of kinematic and mechanical behavior of the materials. The kinematics is the geometry of motion with velocity and acceleration as the most important concepts. Two approaches of the description of these movements can be found: Lagrange and Eulerian descriptions or material and spatial descriptions. The Lagrangian or material description, has uses a reference or initial configuration of the body and motion is analysed following any particular material particle, whilst in the Eulerian or spatial description motion is analysed in a fixed space configuration observing any particular particle passing through every space point.

### 3.1.1 Kinematics of motion

Let us consider a body  $\mathcal{B}$ , which in the Euclidean three-dimensional space, occupies the region  $\Omega$  with a regular boundary defined by  $\partial\Omega$  on its initial configuration.

Consider the position  $\mathbf{X}$  of a particle  $P$  of  $\mathcal{B}$  in the initial configuration coordinate system, in time  $t$ . After given a motion  $\varphi$ , the new coordinates are given by:

$$\mathbf{x} = \varphi\{\mathbf{X}, t\} \quad (3.1)$$

Figure 3.1 shows the motion of a particle and its new coordinates.

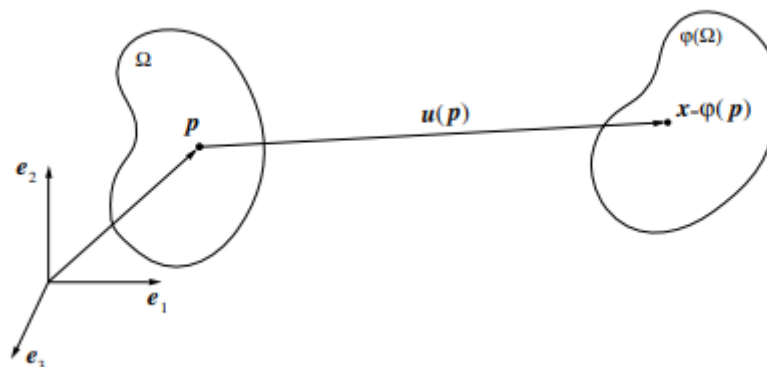


Figure 3.1 Deformation of the body, (Neto, S., Peric, D., and Owen, D. R. J. (2008). *Computational Methods for Plasticity* p. 42)

It's possible to have two different types of motion; a rigid body motion, that may involve a translation and/or a rotation as in Figure 3.2, and, more generally, a motion involving also a deformation of the body, as in Figure 3.1.

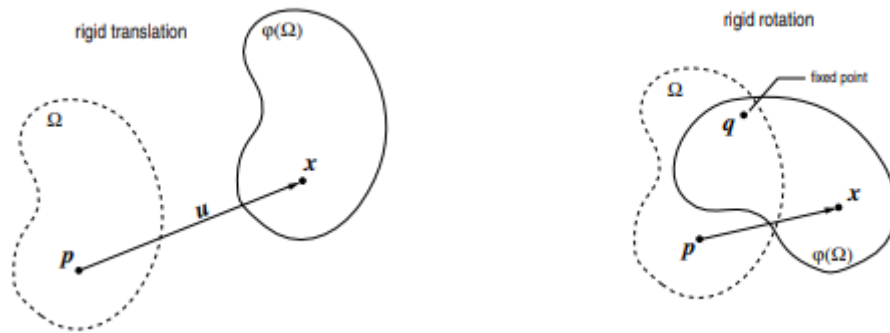


Figure 3.2 – Rigid deformation (Neto, S., Peric, D., and Owen, D. R. J. (2008). *Computational Methods for Plasticity* p. 42)

The displacement field,  $\mathbf{u}$ , of a material particle, is given by:

$$\mathbf{u}\{\mathbf{X}\} = \varphi\{\mathbf{X}\} - \mathbf{X} \quad (3.2)$$

Thus, the expression of the position of particle  $P$  in a deformed configuration of solid is given by:

$$\mathbf{x} = \mathbf{X} + \mathbf{u}\{\mathbf{X}\}. \quad (3.3)$$

The derivative of the displacement field in order to time  $t$  is the velocity field.

$$\dot{\mathbf{x}}\{\mathbf{X}, t\} = \frac{\partial \varphi\{\mathbf{X}, t\}}{\partial t} \quad (3.4)$$

The coordinates  $\mathbf{X}$  can be related to the coordinates on the deformed configuration,  $\mathbf{x}$ , assuming that  $\varphi\{\mathbf{X}, t\}$  is invertible, for all instant  $t$ .

$$\mathbf{X} = \varphi^{-1}\{\mathbf{x}, t\} = \mathbf{x} - \mathbf{u}\{\varphi^{-1}(\mathbf{x}, t), t\}. \quad (3.5)$$

The function below define the spatial velocity field  $\mathbf{v}$ .  $\dot{\mathbf{x}}$  and  $\mathbf{v}$ , both represent the velocity of solid during the deformation, although they have a different referential While



$\dot{x}$  has material particle and time as arguments, the  $v$  has the spatial position and time as argument.

$$v\{x, t\} = \dot{x}\{\varphi^{-1}(x, t), t\} \quad (3.6)$$

### Deformation gradient

The second-order tensor  $\mathbf{F}$  is the deformation gradient of the motion  $\varphi$  given by:

$$\mathbf{F}\{X, t\} = \nabla_X \varphi(X, t) = \frac{\partial x_t}{\partial X} \quad (3.7)$$

With algebraic manipulation of equation 3.3 and 3.7, the gradient deformation can be rewritten as:

$$\mathbf{F}\{X, t\} = \mathbf{I} + \nabla_x \mathbf{u} \quad (3.8)$$

where  $\mathbf{I}$  is the second-order identity. The the spatial description the gradient is given by:

$$\mathbf{F}\{X, t\} = [\nabla_X \varphi^{-1}(X, t)]^{-1} = [\mathbf{I} + \nabla_x \mathbf{u}]^{-1} \quad (3.9)$$

The operators  $\nabla_X$  and  $\nabla_x$  define, respectively, the gradient operator in the material and spatial description.

The determinant of the deformation gradient is given by:

$$J = \det \mathbf{F} \quad (3.10)$$

### Polar decomposition of the deformation gradient

Applying the polar decomposition, the deformation gradient can be decomposed by a pure rotation and an elongation, as shown in following expression:

$$\mathbf{F} = \mathbf{R}\mathbf{U} = \mathbf{V}\mathbf{R} \quad (3.11)$$

The rotation tensor is defined by  $\mathbf{R}$ , the right and left stretch tensors,  $\mathbf{U}$  and  $\mathbf{V}$  are positive and symmetric, and they can be expressed by:

$$\mathbf{U} = \sqrt{\mathbf{C}} \quad (3.12)$$

$$\mathbf{V} = \sqrt{\mathbf{B}} \quad (3.13)$$

$\mathbf{B}$  and  $\mathbf{C}$  are respectively the right and left Cauchy-Green strain tensors, that are expressed by:

$$\mathbf{C} = \mathbf{F}^T \mathbf{F} \quad (3.14)$$

$$\mathbf{B} = \mathbf{F} \mathbf{F}^T \quad (3.15)$$

### Strain measures

There are different ways to quantify the straining phenomenon of solid, even though the different strain quantities represent the same physic phenomenon. This phenomenon is related to a commitment with the mathematical framework.

The most used strain measures in theoretical and numerical formulations are the *Lagrangian* and the *Eulerian* strain tensors. The *Lagrangian* strain tensors are expressed by:

$$\mathbf{E}^m = \begin{cases} \frac{1}{m}(\mathbf{U} - \mathbf{I}), & \text{if } m \neq 0 \\ \ln[\mathbf{U}], & \text{if } m = 0 \end{cases} \quad (3.16)$$

The function of the left stretch tensor, the *Eulerian* strain tensors, is expressed by:

$$\mathbf{E}^m = \begin{cases} \frac{1}{m}(\mathbf{V} - \mathbf{I}), & \text{if } m \neq 0 \\ \ln[\mathbf{V}], & \text{if } m = 0 \end{cases} \quad (3.17)$$

In these expressions, the “m” is a real number, and  $\ln[*]$  denotes the logarithm function. At the case when  $m = 0$  in Eulerian logarithmic strain tensor is used in the formulation of material constitutive models defined at the micro-scale level, in this document.

### 3.1.2 Stress measures

The body deforms under applied forces that can be classified into three categories, The Boundary forces, resulted of contact between bodies; Body forces, as gravitation and magnetic forces and the last one the internal forces, the interaction between adjacent parts of a body. The concepts of stress, to describe surface forces are presented in this section.

#### 3.1.2.1 Cauchy stress tensor

The Cauchy's theorem establishes that the dependence of the surface force  $t$  upon the normal  $n$  is linear, i.e. there exists a second-order tensor field  $\sigma(x)$  such that the Cauchy stress vector is given by

$$t\{n, t\} = \sigma n \quad (3.18)$$

Additionally,  $\sigma$  is symmetric,

$$\sigma = \sigma^T \quad (3.19)$$

The deviatoric stress,  $s$ , is given by:

$$s = \sigma + pI. \quad (3.20)$$

The stress tensor invariant,  $p$  is hydrostatic pressure given by:

$$p = -\frac{1}{3} \text{tr} \sigma \quad (3.21)$$

#### 3.1.2.2 First Piola-Kirchhoff stress tensor

While the Cauchy stress tensor characterizes the stress in a generic point of a continuum solid in the deformed configuration the first Piola stress tensor, denoted by  $P$ , is related to the reference undeformed configuration, and it can be unsymmetric.  $P$  is given by:

$$P = J\sigma F^{-T} \quad (3.22)$$

### 3.1.2.3 Kirchhoff stress tensor

The Kirchhoff stress tensor is defined using the Cauchy stress tensor and the determinant of the deformation gradient,  $J$ , as:

$$\boldsymbol{\tau} = J\boldsymbol{\sigma}. \quad (3.23)$$

### 3.1.3 Weak equilibrium - The principle of virtual work

The weak form of strong equilibrium equations is the consequence of the application of the Principle of Virtual Work. It is a typical formulation used to solve problems by finite elements method in stress and strain analysis.

As at the beginning of this chapter, section 3.1.1, let's consider a generic solid  $\mathcal{B}$  which occupies the region  $\Omega$  in the Euclidean three-dimensional space with a boundary  $\partial\Omega$  in its initial configuration. The body is subjected to forces in its interior and surface tractions on its boundary.

#### Spatial version

In the spatial or Eulerian description of the principle Virtual Work, the body,  $\mathcal{B}$  in equilibrium if, and only if, its Cauchy stress field,  $\boldsymbol{\sigma}$ , satisfies the equation

$$\int_{\varphi(\Omega)} [\boldsymbol{\sigma} : \nabla_x \boldsymbol{\eta} - \mathbf{b} \cdot \boldsymbol{\eta}] dV - \int_{\varphi(\partial\Omega)} [\mathbf{t} \cdot \boldsymbol{\eta}] dA = 0 \quad \forall \boldsymbol{\eta} \in \mathcal{V} \quad (3.24)$$

At the equation above  $\boldsymbol{\eta}$  denotes the virtual displacements that belong to the space of admissible virtual displacements  $\mathcal{V}$ .

At this expression, the body force per unit deformed volume and the boundary forces per unit area, are denoted by  $\mathbf{b}$  and  $\mathbf{t}$ . The space of virtual displacements of  $\mathcal{B}$  is  $\mathcal{V}$ .

#### Material version

For the material or Lagrangian description, the principle of virtual work is in equilibrium with  $\mathcal{B}$  if and only if its first Piola-Kirchhoff stress field satisfies the equation:

$$\int_{\varphi(\Omega)} [\mathbf{P} : \nabla_x \boldsymbol{\eta} - \mathbf{b} \cdot \boldsymbol{\eta}] dV - \int_{\varphi(\Omega)} [\mathbf{t} \cdot \boldsymbol{\eta}] dA = 0 \quad \forall \boldsymbol{\eta} \in \mathcal{V} \quad (3.25)$$

### 3.2 Finite Element Method

The Finite Element Method is a numerical method for solving differential and partial differential equations or integral equations. The basic idea of the FEM is to divide the domain of the problem into finite elements connected by nodes where the values of the unknown variables are sought. The set of finite element and nodes defines a mesh. The problems to solve may be linear or non-linear and the application of the method leads to a system of linear or non-linear equations that must be solved. Depending on the complexity of the problem addressed thousands of nodes may be required in order to obtain a reasonably precise solution, so computers are essential for solving these equations. Usually, as the number of elements (and nodes) expands the accuracy of the solution improve, but the computer time, and hence the cost, also increases.

In this section the main concepts of the FEM are presented. For more information the following references may be consulted: (Fish and T. Belytschko, 2007), (Irgens, Fridtjov, 2008) and (Neto, S., Peric, D., and Owen, D. R. J., 2008)

#### Integral Formulation of the Problem

The integral formulation for a structural problem, already presented in section 3.1.3, associated with the principle of virtual work for both the spatial and material descriptions, will be used here to illustrate the application of the FEM.

## Spatial Discretization

The spatial domain,  $\Omega^d$ , is related to the discretization by the finite element method of the solid in analysis through a finite number, “ $n_e$ ”, of subdomains,  $\Omega^e$ , called finite elements.

$$\Omega \approx \Omega^d = \bigcup_{e=1}^{n_e} \Omega^e \quad (3.26)$$

In each sub-domain, the variables, like displacements, stress and strain, of the problem are determined by interpolation the nodal values using shape functions, which should satisfy some restrictions. Figure 3.3 is shown the finite element interpolation and the element and global shape function.

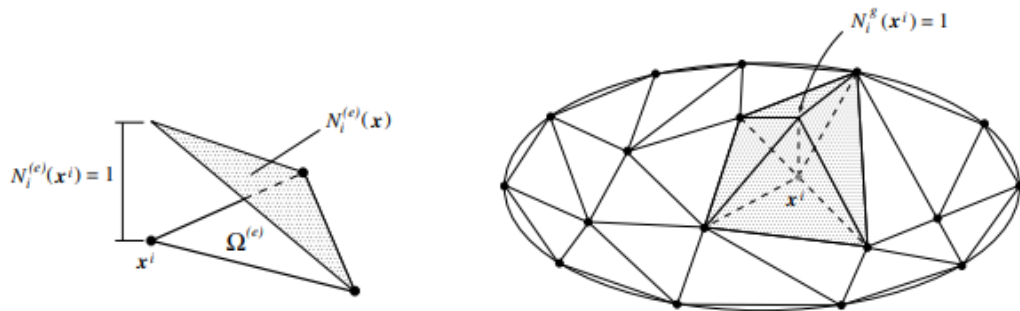


Figure 3.3: Finite element interpolation. The element and global shape function. (adapted from Neto, S., Peric, D., and Owen, D. R. J. (2008). *Computational Methods for Plasticity* p. 86)

Let us consider a generic field “ $a(x)$ ” over the domain  $\Omega^e$  of the element. The process of interpolation can be expressed by:

$$a(x) = \sum_{i=1}^{n_e} N_i^e(x) a(x_i) \quad (3.27)$$

At each  $a(x_i)$  represent the nodal value of the generic field the nodal and

the  $N_i^e(x)$  represent the interpolation function or shape function.

If considered the whole discretized domain  $\Omega_d$ , the interpolation process can be expressed, in the same way, by:

$$a(x) = \sum_{i=1}^{n_p} N_i(x) a(x_i) \quad (3.28)$$

The number of nodes of the whole domain is denoted by  $n_p$ , and the global shapes functions are denoted  $N_i(\mathbf{x})$ :

A global interpolation matrix can be expressed as:

$$N(x) = \left[ \text{diag}[N_1(x)] \mid \text{diag}[N_2(x)] \mid \dots \mid \text{diag}[N_{n_p}(x)] \right] \quad (3.29)$$

The  $\text{diag}[N_i(x)]$  represent the diagonal of a matrix, and the size is determinate with the number of degrees of freedom.

$$\text{diag}[N_i(x)] = \begin{bmatrix} N_i^e(x) & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & N_i^e(x) \end{bmatrix} \quad (3.30)$$

Two operators should be introduced to complete the spatial discretization, the global discrete material gradient operator,  $G^g$  and the global discrete spatial gradient operator,  $B^g$ , which are the discretized versions of operators  $\nabla_{\mathbf{x}}(*)$  and  $\nabla_{\mathbf{x}}(*)$ , respectively. Both matrices have dimensions related to the type of problem that is going to be solved, The matrix below, 3.31a and 3.31b, are used to solve the 2D problems, plane strain and stress.

$$\mathbf{B}^g = \left[ \begin{array}{cc|cc| \dots \dots} \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \dots & \dots & \frac{\partial N_n}{\partial x} & 0 \\ 0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & \dots & \dots & 0 & \frac{\partial N_n}{\partial y} \\ \hline \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} & \dots & \dots & \frac{\partial N_n}{\partial y} & \frac{\partial N_n}{\partial x} \end{array} \right] \quad (3.31a)$$

$$\mathbf{G}^g = \left[ \begin{array}{cc|cc| \dots \dots} \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \dots & \dots & \frac{\partial N_n}{\partial x} & 0 \\ 0 & \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & \dots & \dots & 0 & \frac{\partial N_n}{\partial x} \\ \hline \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & 0 & \dots & \dots & \frac{\partial N_n}{\partial y} & 0 \\ 0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & \dots & \dots & 0 & \frac{\partial N_n}{\partial y} \end{array} \right] \quad (3.31b)$$

### Spatial Discretization of the Equilibrium Equations

With the equation introduced at section 3.1.3, the weak form of the equilibrium equations can be discretized where  $v^d$  represents the discretized virtual displacement space and  $\boldsymbol{\sigma}$  is the vector representation of the Cauchy stress tensor  $\boldsymbol{\sigma}$ . Considering the Eulerian formulation, the discretized version of Equation (3.24) is:

$$\int_{\varphi(\Omega^d)} [(\mathbf{B}^g)^T \boldsymbol{\sigma} - \mathbf{N}^T \mathbf{b}] dV - \int_{\varphi(\partial\Omega^d)} \mathbf{N}^T \mathbf{t} dA \}^T \cdot \boldsymbol{\eta} = 0 \quad \forall \boldsymbol{\eta} \in v^d \quad (3.32)$$

Considering the Lagrangian formulation, the discretized version of Equation (3.25) is:

$$\int_{(\Omega^d)} [(\mathbf{G}^g)^T \mathbf{P} - \mathbf{N}^T \bar{\mathbf{b}}] dV - \int_{\varphi(\partial\Omega^d)} \mathbf{N}^T \bar{\mathbf{t}} dA \}^T \cdot \boldsymbol{\eta} = 0 \quad \forall \boldsymbol{\eta} \in v^d \quad (3.33)$$

The vector configuration of the first stress tensor of Piola-Kirchhoff is denoted by  $\mathbf{P}$ .

The previously equations must be satisfied for all virtual displacements  $\boldsymbol{\eta}$ . Consequently, the expression inside the brackets must be null. Therefore, it is possible to rewrite them as:

$$\mathbf{f}^{int} - \mathbf{f}^{ext} = \mathbf{0} \quad (3.34)$$

The global internal,  $\mathbf{f}^{int}$  and external force vectors  $\mathbf{f}^{ext}$  in the spatial form are denoted by:

$$\mathbf{f}^{int} = \int_{\varphi(\Omega^d)} [(\mathbf{B}^g)^T \boldsymbol{\sigma}] dV \quad (3.35a)$$

$$\mathbf{f}^{ext} = \int_{\varphi(\Omega^d)} [\mathbf{N}^T \mathbf{b}] dA + \int_{\varphi(\partial\Omega^d)} [\mathbf{N}^T \mathbf{t}] dA \quad (3.35b)$$



The element force is obtained by integration over its domain.

$$\mathbf{f}_e^{int} = \int_{\varphi(\Omega^e)} [(\mathbf{B}^e)^T \boldsymbol{\sigma}] dV \quad (3.36a)$$

$$\mathbf{f}_e^{ext} = \int_{\varphi(\Omega^e)} [(\mathbf{N}^e)^T \mathbf{b}] dA + \int_{\varphi(\partial\Omega^e)} [(\mathbf{N}^e)^T \mathbf{t}] dA \quad (3.36b)$$

In the Lagrangian formulation, the force vectors are given by:

$$\mathbf{f}_e^{int} = \int_{\Omega^e} [(\mathbf{G}^e)^T \mathbf{P}] dV \quad (3.37a)$$

$$\mathbf{f}_e^{ext} = \int_{\Omega^e} [(\mathbf{N}^e)^T \mathbf{b}] dA + \int_{\partial\Omega^e} [(\mathbf{N}^e)^T \mathbf{t}] dA \quad (3.37b)$$

The global force vectors are structured from the element force vectors.

$$\mathbf{f}^{int} = \bigwedge_{e=1}^{n_e} \mathbf{f}_e^{int} \quad (3.38a)$$

$$\mathbf{f}^{ext} = \bigwedge_{e=1}^{n_e} \mathbf{f}_e^{ext} \quad (3.38b)$$

In the previous equations, the symbol  $\bigwedge$  denotes an appropriated assembly operator.

# **CHAPTER 4**

## **Multiscale analysis – Microscale**

The multi-scale study consists in the analysis of properties of the different components of the heterogeneous material at the microscale and their effect on the behavior at the macroscale, through homogenization procedures. There are some methods used to do it as analytical methods, for example, the Hill Bounds, Hashin and Shtrikman Bounds, self-consistent method, Mori-Tanaka Methods among others, that can be find at Carvalho M. (2016), and numerical methods, as the finite element method, that was chosen to develop this research. Thus, this chapter introduces the basic concepts of micromechanics analysis and its application in obtaining thermoelastic properties at the macroscale using a numerical method to homogenize the properties found at the microscale.

### **4.1 Micromechanics**

The purpose of micromechanics is to account explicitly for a material's heterogeneous microstructure while allowing it to be treated as a valid (pseudo-

homogeneous) continuum at a higher length scale (e.g., within a structure). In order to explain this microstructure, micromechanics relies on either a representative volume element (RVE) or a repeating unit cell (RUC). (Jacob Aboudi, Steven M. Arnold, Brett A. Bednarczyk, 2013)

RVE is a dimension of the matter that represents the behavior of the material as a whole. If the matter is considered perfectly homogeneous, the RVE would be infinitely small. The representative volume element must contain a large enough dimension so that it captures the essence of the microstructure from a statistical standpoint. It means that if you apply any boundary conditions to the RVE, this could obtain an identical response that genuinely represents the material.

The repeating unit cell (RUC) is an equally essential concept, that applies to a heterogeneous microstructure that has a certain periodicity. The RUC is the volume of the material that repeats itself to generate the overall microstructure. To impose the repeating nature of the RUC periodic boundary conditions must be applied. To have a more accurate result it is essential to consider as many as possible unit cells at heterogeneous materials.

In figure 4.1 is illustrated the relationship between the scale of an RVE or RUC and structure. The nanoscale is represented by  $d$ , the  $l$  represents the microstructural size, and  $L_{\text{macro}}$  is a typical length of the component or structure composed of the heterogeneous material. RVE should include a sufficient number of microstructural units to allow the heterogeneous nature of the material to be homogenised, and for this, the ratio of these dimensions ( $d/l$  and  $l/L_{\text{macro}}$ ) should be much less than 1.

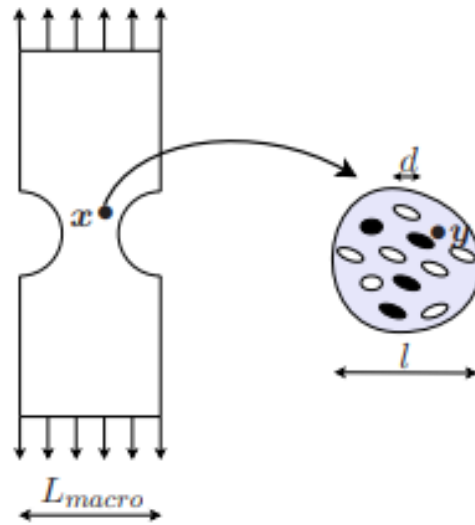


Figure 4.1: Macro and microscales and respective sizes (JReis,2015).

### Volume averaging

In a RVE or a RUC subjected to a stress/strain field the average stress and tension tensors are defined macroscopically by:

$$\bar{\sigma}_{ij} = \frac{1}{V} \int_V \sigma_{ij} dV \quad (4.1)$$

$$\bar{\varepsilon}_{ij} = \frac{1}{V} \int_V \varepsilon_{ij} dV \quad (4.2)$$

where the volume of the RVE or RUC is denoted by  $V$ .

## 4.2 Homogenization of elastic properties

### 4.2.1 Hooke's Law

Hooke's law relates strain and stress states in the elastic regime for most of materials (Figure 4.2).

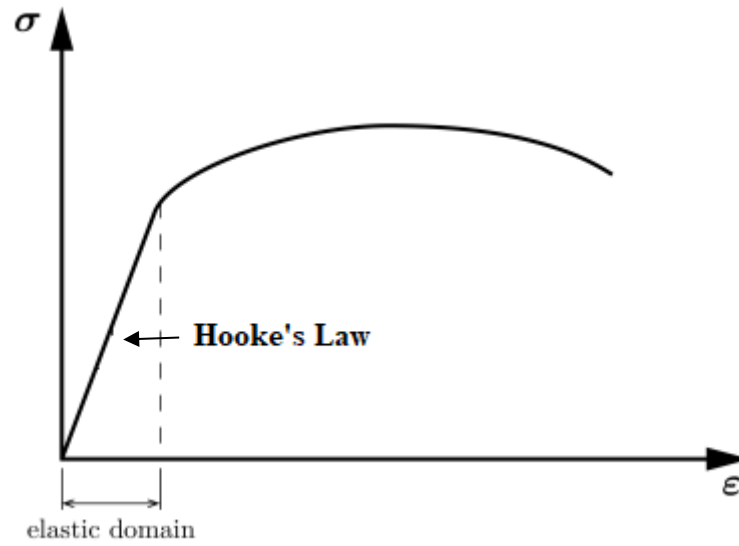


Figure 4.2 Stress-strain curve - Representation of the elastic domain

The constitutive equations can be expressed mathematically by relating of stress and strain tensors by the fourth-order tensor  $\mathbf{C}$ , called the elasticity tensor or stiffness tensor or by the fourth-order tensor  $\mathbf{S}$ , called the compliance tensor or flexibility tensor, as:

$$\varepsilon_{ij} = S_{ijkl} \sigma_{kl}, \quad (4.3)$$

$$\sigma_{kl} = C_{ijkl} \varepsilon_{ij}, \quad (4.4)$$

Being  $\mathbf{C}$  the inverse tensor of  $\mathbf{S}$

$$\mathbf{C} = \mathbf{S}^{-1} \quad (4.5)$$

The 4th-order elasticity tensor has symmetries resulting from the symmetry of the stress and strain tensors, making it possible to reduce the 81 components relating 9 components of stress and to only 36 independent constants:

$$\sigma_{ij} = \sigma_{ji} \Rightarrow C_{jikl} = C_{ijlk} \quad (4.6)$$

$$\varepsilon_{ij} = \varepsilon_{ji} \Rightarrow C_{jikl} = C_{ijlk} \quad (4.7)$$

The strain energy function is related to the capacity of the materials to store energy in the form of elastic energy per unit volume, which is denoted as,

$$U = \frac{1}{2} \boldsymbol{\varepsilon} : \mathbf{C} : \boldsymbol{\varepsilon} \quad (4.8)$$

and the stress tensor can be obtained from the strain energy by

$$\boldsymbol{\sigma} = \frac{\partial U}{\partial \boldsymbol{\varepsilon}} \quad (4.9)$$

Thus, the number of independent elastic constants decreases to 21 if it assumes the material has an anisotropic behavior

Following the Voigt notation it is possible to describe the Hooke's Law in a matrix form as:

$$\begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{12} \\ \sigma_{13} \end{Bmatrix} = \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & C_{1123} & C_{1112} & C_{1113} \\ & C_{2222} & C_{2233} & C_{2223} & C_{2212} & C_{2213} \\ & & C_{3333} & C_{3323} & C_{3312} & C_{3313} \\ & & & C_{2323} & C_{2312} & C_{2313} \\ & & & & C_{1212} & C_{1312} \\ & & & & & C_{1313} \end{bmatrix} \begin{Bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \gamma_{23} \\ \gamma_{12} \\ \gamma_{13} \end{Bmatrix} \quad (4.10)$$

where,  $\gamma_{23} = 2\varepsilon_{23}$ ,  $\gamma_{12} = 2\varepsilon_{12}$ ,  $\gamma_{13} = 2\varepsilon_{13}$  are the engineering shear strains.

In order to simplify the notation this can be written as:

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{Bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ & & C_{33} & C_{34} & C_{35} & C_{36} \\ & & & C_{44} & C_{45} & C_{46} \\ & & & & C_{55} & C_{56} \\ & & & & & C_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \end{Bmatrix} \quad (4.11)$$

The number of independent elastic constants is reduced to 13, to usual for linear elastic solids that have at least one plane of symmetry,. This phenomenon is known as elastic symmetry, and a material that has one plane of symmetry is called a monoclinic material:

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{Bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & 0 & 0 \\ & C_{22} & C_{23} & C_{24} & 0 & 0 \\ & & C_{33} & C_{34} & 0 & 0 \\ & & & C_{44} & 0 & 0 \\ & & & & C_{55} & C_{56} \\ sym & & & & & C_{66} \end{bmatrix} \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \end{Bmatrix} \quad (4.12)$$

The number of independent elastic constants is reduced to only 2 in an isotropic material.

### Plane stress and plane strain

The stress and strain of structures and components are typically defined in three dimensions (Figure 4.3), but in some cases, two-dimensional models are enough to have significant results.

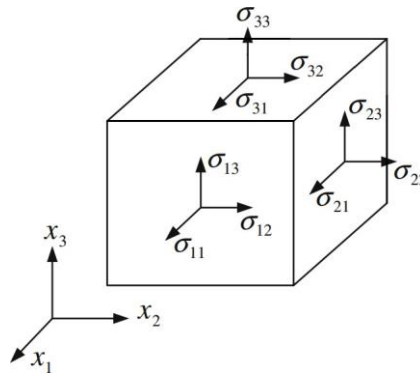


Figure 4.3 Components of stress in three dimensions.

In these cases the Hooke's Law may be written in a matrix form as

$$\begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{12} \end{Bmatrix} = \begin{bmatrix} C_{1111} & C_{1122} & C_{1112} \\ C_{2211} & C_{2222} & C_{2212} \\ C_{1211} & C_{1222} & C_{1212} \end{bmatrix} \begin{Bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \gamma_{12} \end{Bmatrix} \quad (4.13)$$

This is the case of plane stress, and plane strain states. For an isotropic material the elasticity matrix for a plane stress state is

$$D = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \quad (4.15)$$

and for plane strain state is

$$D = \frac{E}{(1-\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & (1-2\nu)/2 \end{bmatrix} \quad (4.16)$$

### 4.2.2 Thermal stress and strain

Materials expand and contract owing to temperature changes causing a deformation, that in an unidimensional setting, can be denoted as

$$\varepsilon^T = \alpha(\Delta T) \quad (4.17)$$

where  $\varepsilon^T$  is the thermal deformation,  $\alpha$  the expansion coefficient and  $\Delta T$  the temperature variation.

Figure 4.4 shows an example of the expansion and contraction deformation caused by temperature variation. The body increases the length when it is submitting the  $+\Delta T$  and in likewise when the  $-\Delta T$  is found the length get lower.

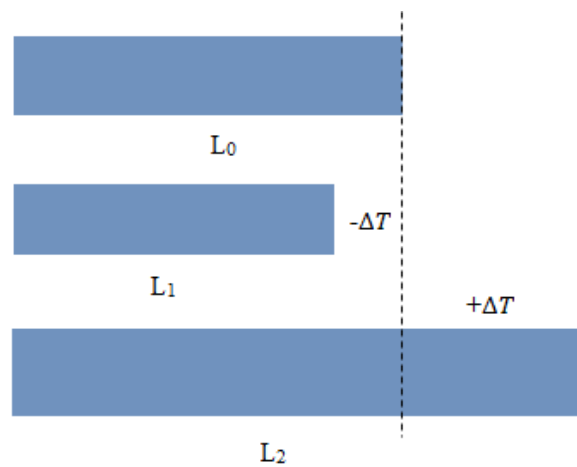


Figure 4.4- Expansion and contraction deformation caused by temperature variation.



In some cases , there may be an external force that makes the body keep the same length, but a thermal stress is ‘created’, nonetheless.

If we consider a 3D generic material then

$$\varepsilon_{ij}^T = \alpha_{ij}(\Delta T) \quad (4.18)$$

where  $\alpha_{ij}$  is a 2<sup>nd</sup> order tensor.

The total strain tensor of the material as the sum of the mechanical strain and thermal strain is

$$\varepsilon_{ij} = \varepsilon_{ij}^M + \varepsilon_{ij}^T \quad (4.19)$$

where  $\varepsilon_{ij}^M$  is the ‘mechanical’ deformation

$$\varepsilon_{ij}^M = S_{ijkl}\sigma_{kl} \quad (4.20)$$

where  $S_{ijkl}$  is the compliance tensor

Substituting the expression 4.12 and 4.14 into 4.13

$$\varepsilon_{ij} = S_{ijkl}\sigma_{kl} + \alpha_{ij}(\Delta T) \quad (4.21)$$

$$\text{with } S_{ijkl}^{-1} = E_{ijkl} \quad (4.22)$$

If multiply both sides by the inverse of the compliance the expression becomes

$$\sigma_{kl} = E_{ijkl}\varepsilon_{ij} - E_{ijkl}\alpha_{ij}(\Delta T) \quad (4.23)$$

In anisotropic materials there may be six possibilities for the expansion coefficients:  $\alpha_{11}$ ,  $\alpha_{22}$ ,  $\alpha_{33}$ ,  $\alpha_{12}$ ,  $\alpha_{13}$ ,  $\alpha_{23}$  and in orthotropic materials these possibilities reduce to three:  $\alpha_{11}$ ,  $\alpha_{22}$ ,  $\alpha_{33}$ .

The  $\Delta T$  should be determined by evaluating the heat flux through the material. Three types of heat transfer can be considered: conduction, radiation and convection.

**Heat Conduction** - the heat flux happens through contact and could be the direct contact between the excited molecules, in solids, liquids and gases, with their neighbors, which transferring part of their motion. It can occur between two different objects in contact or the same object when there is a temperature differential through its volume.

The physical law that governs this mechanism is known as Fourier's Law (F. Incropera, 2006).

$$q'' = -k \frac{dT}{dx} \quad (4.24)$$

where:  $q''$  represent the heat flux ( $W/m^2$ );  $k$  is the thermal conductivity ( $W/m \cdot K$ ) from the material considered or of the crossing point between two materials;  $\frac{dT}{dx}$  is the gradient of temperature in the direction of  $x$ .

**Heat Convection** - The flux depends on the mass's motion of a fluid such as air. The convection occurs when there is a temperature differential between the surface and the surrounding fluid

The physical law that governs this mechanism is Newton's law of cooling (F. Incropera, 2006).

$$q'' = h \cdot (T_s - T_\infty) \quad (4.25)$$

where:  $q''$  represent the heat flux ( $W/m^2$ );  $h$  is the heat transfer coefficient by convection ( $W/m^2 \cdot K$ ),  $T_s$  is the temperature on the surface and  $T_\infty$  is the temperature of the fluid.

**Heat Radiation** -This flux occurs between two bodies. The one with the high temperature transfer the heat to the body with lower temperature, without the contact, these are separated in space.

The physical law that governs this mechanism is the Steffan-Boltzmann law (F. Incropera, 2006):

$$q'' = \varepsilon \cdot \sigma \cdot (T_s^4 - T_{sur}^4) \quad (4.26)$$

Where  $q''$  represent the heat flux ( $W/m^2$ ); the emissivity of the surface is denoted by  $\varepsilon$ , with  $0 \leq \varepsilon \leq 1$ , indicating how efficient the surface of the object is when emitting radiation, in comparison to that of a black body ( $\varepsilon = 1$ ),  $\sigma$  is the Stefan-Boltzmann constant:  $\sigma = 5.67 \cdot 10^{-8} W/m^2 \cdot K^2$ ,  $T_s$  is the temperature on the surface and  $T_{sur}$  is the temperature of the surroundings of the object.

#### 4.2 Numerical model.

Through of continuum mechanics theory and the finite element method, approached in chapter 3, and concepts of micromechanics and homogenization, approached in this chapter a finite element model was created and used for different boundary condition to predict the elasticity matrix and thermal expansion coefficients at the macro scale, through of micrography studies.

First of all, in order to extract the outcomes of the simulation in-plane stress and plane strain of traction/compression in x, y directions and shear in x and y plane, a model was made in Abaqus, figure 4.5, that intends to represent an RVE subject to appropriate boundary conditions in order to obtain the averaged terms of the elasticity matrix. Strain, stress, and the coordinates of each element from the f.e. mesh, was collected using a the python script, appendix D, that generated an excel file with all necessary information. According the concepts of homogenization and the equation 4.1, 4.2, 4.14a, 4.14.b, 4.14.c, a MATLAB script was created, that calculates the coefficients of the elasticity matrix.

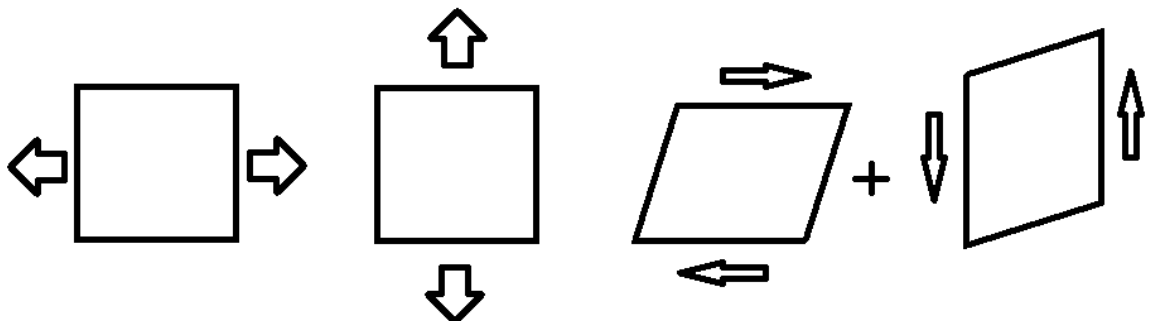


Figure 4.5 Representation of traction in x, y and shear in x and y

At the first simulation of plane stress and strain, 2D – triangle element – traction in X direction, the  $\varepsilon_{22} = \gamma_{12} = 0$ , and  $\varepsilon_{11} = 1$ , using the equations 4.14a, 4.14.b, 4.14.c, and gaussian quadrature, the coefficient  $C_{ij11}$ , are denoted by;

$$C_{1111} = \sum_1^n (((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3) * w * J) / V \quad (4.27.a)$$

$$C_{2211} = \sum_1^n (((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3) * w * J) / V \quad (4.27.b)$$

$$C_{1211} = \sum_1^n (((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3) * w * J) / V \quad (4.27.c)$$

Where n represent the numbers of elements;  $C_{11ij}$  are the coefficient of elasticity matrix corresponding to the first column, and the  $(\sigma_{ij})_k$  are the stress in each node of the triangle element, the notation k in this equation are used to denoted the first, second and third triangle node; the  $N_k$  are the shape functions;  $w$  is the weight of the nodes, and the  $J$  is the Jacobin matrix;  $V$  is the volume of part.

In second simulation of plane stress and strain – traction in Y direction, the  $\varepsilon_{11} = \gamma_{12} = 0$ , and  $\varepsilon_{22} = 1$ , the coefficient  $C_{ij22}$ , are denoted by;

$$C_{1122} = \sum_1^n (((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3) * w * J) / V \quad (4.28.a)$$

$$C_{2222} = \sum_1^n (((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3) * w * J) / V \quad (4.28.b)$$

$$C_{1222} = \sum_1^n (((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3) * w * J) / V \quad (4.28.c)$$

Where  $C_{22ij}$  are the coefficient of elasticity matrix corresponding to the second column.

In third and fourth simulation of plane stress and strain – shear X and Y direction, the  $\varepsilon_{11} = \varepsilon_{22} = 0$ , and  $\gamma_{12} = 1$ , using the equation 4.14a, 4.14.b, 4.14.c, and the coefficient  $C_{ij12}$ , are denoted by;

$$C_{1112} = \sum_1^n (((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3) + ((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3)) * w * J) / V \quad (4.29.a)$$

$$C_{2212} = \frac{\sum_1^n (((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3) + ((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3)) * w * J) / V}{(4.29.b)}$$

$$C_{1212} = \frac{(\sum_1^n (((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3) + ((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3)) * w * J) / V}{(4.29.c)}$$

Where  $C_{12ij}$  are the coefficient of elasticity matrix corresponding to the third column.

To calculate the expansion coefficients was simulated the temperature variation in X and Y direction.as shown in figure 4.6, also was shown the boundaries condition used.

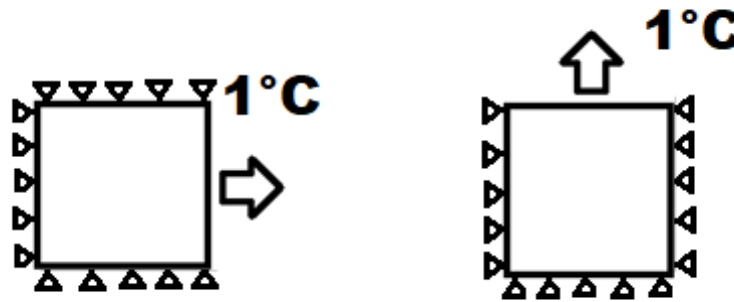


Figure 4.6 Boundary conditions of thermal simulation

In the first simulation of plane stress and strain, 2D – temperature variation in X direction, there isn't imposed dislocation, then  $\varepsilon = 0$ , and  $\varepsilon^T = \alpha(\Delta T)$ , for the 1°C of variation, the coefficient  $\alpha_{ii}$ , is denoted by;

$$\alpha_{11} = \frac{\sum_1^n ((\varepsilon_{11} * N_1 + \varepsilon_{11} * N_2 + \varepsilon_{11} * N_3) * w * J) / V}{(4.30)}$$

Where  $\alpha_{11}$  is the expansion coefficient in X direction

The second simulation of plane stress and strain, 2D – temperature variation in Y direction, the,  $\varepsilon = 0$ , and  $\varepsilon^T = \alpha(\Delta T)$  for the 1°C of variation, the coefficient  $\alpha$ , is denoted by;

$$\alpha_{22} = (\sum_1^n (\varepsilon_{22} * N_1 + \varepsilon_{22} * N_2 + \varepsilon_{22} * N_3) * w * J) / V \quad (4.31)$$

Where  $\alpha_{22}$  is the expansion coefficient in Y direction

To obtain the elasticity matrix coefficient to 3D problem was necessary did simulated the traction in X, Y, and Z direction, and shear in XY, XZ, YZ direction. As shown in figure 4.7

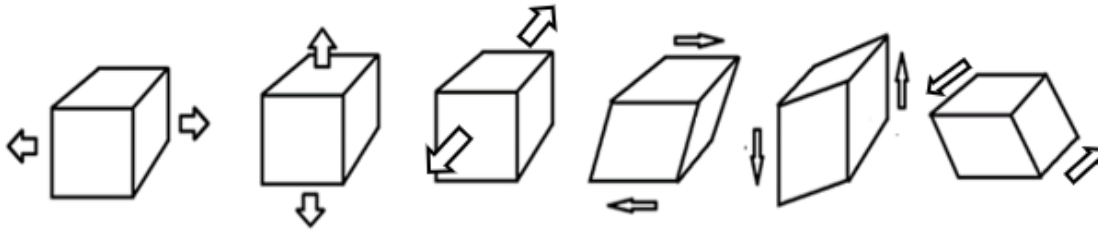


Figure 4.7 Boundary conditions for 3D simulation

At the first simulation – tetrahedron element – traction in X direction, the  $\varepsilon_{22} = \varepsilon_{33} = \gamma_{23} = \gamma_{12} = \gamma_{13} = 0$ , and  $\varepsilon_{11} = 1$ , the coefficient  $C_{ij11}$ , are denoted by;

$$C_{1111} = \sum_1^n (((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) * w * J) / V \quad (4.32.a)$$

$$C_{2211} = \sum_1^n (((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) * w * J) / V \quad (4.32.b)$$

$$C_{3311} = \sum_1^n (((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) * w * J) / V \quad (4.32.c)$$

$$C_{2311} = \sum_1^n (((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) * w * J) / V \quad (4.32.d)$$

$$V \quad C_{1211} = \sum_1^n (((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3 + (\sigma_{12})_4 * N_4) * w * J) / \quad (4.32.e)$$

$$V \quad C_{1311} = \sum_1^n (((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) * w * J) / \quad (4.32.f)$$

At the second simulation – tetrahedron element – traction in Y direction, the  $\varepsilon_{11} = \varepsilon_{33} = \gamma_{23} = \gamma_{12} = \gamma_{13} = 0$ , and  $\varepsilon_{22} = 1$ , the coefficient  $C_{ij22}$ , are denoted by;

$$V \quad C_{1122} = \sum_1^n (((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) * w * J) / \quad (4.33.a)$$

$$V \quad C_{2222} = \sum_1^n (((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) * w * J) / \quad (4.33.b)$$

$$V \quad C_{3322} = \sum_1^n (((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) * w * J) / \quad (4.33.c)$$

$$V \quad C_{2322} = \sum_1^n (((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) * w * J) / \quad (4.33.d)$$

$$V \quad C_{1222} = \sum_1^n (((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3 + (\sigma_{12})_4 * N_4) * w * J) / \quad (4.33.e)$$

$$V \quad C_{1322} = \sum_1^n (((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) * w * J) / \quad (4.33.f)$$

At the third simulation – tetrahedron element – traction in Z direction, the  $\varepsilon_{11} = \varepsilon_{22} = \gamma_{23} = \gamma_{12} = \gamma_{13} = 0$ , and  $\varepsilon_{33} = 1$ , the coefficient  $C_{ij33}$ , are denoted by;

$$V \quad C_{1133} = \sum_1^n (((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) * w * J) / \quad (4.34.a)$$

$$V \quad C_{2233} = \sum_1^n (((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) * w * J) / \quad (4.34.b)$$

$$J) / V \quad C_{3333} = \sum_1^n (((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) * w * \quad (4.34.c)$$

$$V \quad C_{2333} = \sum_1^n (((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) * w * J) / \quad (4.34.d)$$

$$V \quad C_{1233} = \sum_1^n (((\sigma_{12})_1 * N_1 + (\sigma_{12})_2 * N_2 + (\sigma_{12})_3 * N_3 + (\sigma_{12})_4 * N_4) * w * J) / \quad (4.34.e)$$

$$V \quad C_{1333} = \sum_1^n (((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) * w * J) / \quad (4.34.f)$$

At the fourth, fifth, and sixth simulation – tetrahedron element – shear in YZ direction, the  $\varepsilon_{11} = \varepsilon_{22} = \varepsilon_{33} = \gamma_{23} = \gamma_{13} = 0$ , and  $\gamma_{23} = 1$ , the coefficient  $C_{ij23}$ , are denoted by;

$$C_{1123} = \sum_1^n (((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) + ((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) + ((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4)) * w * J) / V \quad (4.35.a)$$

$$C_{2223} = \sum_1^n (((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) + ((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) + ((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4)) * w * J) / V \quad (4.35.b)$$



$$C_{3323} = \sum_1^n \left( ((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) + ((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) + ((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) \right) * w * J / V \quad (4.35.c)$$

$$C_{2323} = \sum_1^n \left( ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) \right) * w * J / V \quad (4.35.d)$$

$$C_{1223} = \sum_1^n \left( ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) \right) * w * J / V \quad (4.35.e)$$

$$C_{1323} = \sum_1^n \left( ((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) + ((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) + ((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) \right) * w * J / V \quad (4.35.f)$$

At the seventh, eighth, ninth simulation – tetrahedron element – shear in XY direction, the  $\varepsilon_{11} = \varepsilon_{22} = \varepsilon_{33} = \gamma_{23} = \gamma_{13} = 0$ , and  $\gamma_{12} = 1$ , the coefficient  $C_{ij12}$ , are denoted by;

$$C_{1112} = \sum_1^n \left( ((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) + ((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) + ((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) \right) * w * J / V \quad (4.36.a)$$

$$C_{2212} = \sum_1^n \left( ((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) + ((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) + ((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) \right) * w * J / V \quad (4.36.b)$$

$$C_{3312} = \sum_1^n \left( ((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) + ((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) + ((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) \right) * w * J / V \quad (4.36.c)$$

$$C_{2312} = \frac{\sum_1^n (((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4)) * w * J) / V}{(4.36.d)}$$

$$C_{1212} = \frac{\sum_1^n (((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4)) * w * J) / V}{(4.36.e)}$$

$$C_{1312} = \frac{\sum_1^n (((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) + ((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) + ((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4)) * w * J) / V}{(4.36.f)}$$

At the tenth, eleventh, and twelfth simulation – tetrahedron element – shear in XY direction, the  $\varepsilon_{11} = \varepsilon_{22} = \varepsilon_{33} = \gamma_{23} = \gamma_{12} = 0$ , and  $\gamma_{13} = 1$ , the coefficient  $C_{ij13}$ , are denoted by;

$$C_{1113} = \frac{\sum_1^n (((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) + ((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4) + ((\sigma_{11})_1 * N_1 + (\sigma_{11})_2 * N_2 + (\sigma_{11})_3 * N_3 + (\sigma_{11})_4 * N_4)) * w * J) / V}{(4.37.a)}$$

$$C_{2213} = \frac{\sum_1^n (((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) + ((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4) + ((\sigma_{22})_1 * N_1 + (\sigma_{22})_2 * N_2 + (\sigma_{22})_3 * N_3 + (\sigma_{22})_4 * N_4)) * w * J) / V}{(4.37.b)}$$

$$C_{3313} = \frac{\sum_1^n (((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) + ((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4) + ((\sigma_{33})_1 * N_1 + (\sigma_{33})_2 * N_2 + (\sigma_{33})_3 * N_3 + (\sigma_{33})_4 * N_4)) * w * J) / V}{(4.37.c)}$$

$$C_{2313} = \frac{\sum_1^n (((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4)) * w * J) / V}{(4.37.d)}$$

$$C_{1213} = \frac{\sum_1^n (((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4) + ((\sigma_{23})_1 * N_1 + (\sigma_{23})_2 * N_2 + (\sigma_{23})_3 * N_3 + (\sigma_{23})_4 * N_4)) * w * J) / V}{(4.37.e)}$$

$$C_{1313} = \frac{\sum_1^n (((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) + ((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4) + ((\sigma_{13})_1 * N_1 + (\sigma_{13})_2 * N_2 + (\sigma_{13})_3 * N_3 + (\sigma_{13})_4 * N_4)) * w * J) / V}{(4.37.f)}$$

In order to validate the MATLAB script an example of a square with 20x20 mm, with triangle element mesh was used assigning the same 200GPa young modulus and 0.3 Poisson rate to all the elements, using a sparse and a more refined mesh, shown in figure 4.8. In Table 4.1 the comparison of the elasticity matrices obtained, with sparse and refined meshes, with the theoretical matrix is shown, depicting a very small error and showing that the script was adequate to use even in the sparse mesh case in this homogenous material case. However, if different materials are involved, the accuracy of results may depend on the mesh. Therefore structured and unstructured meshes were also tested.

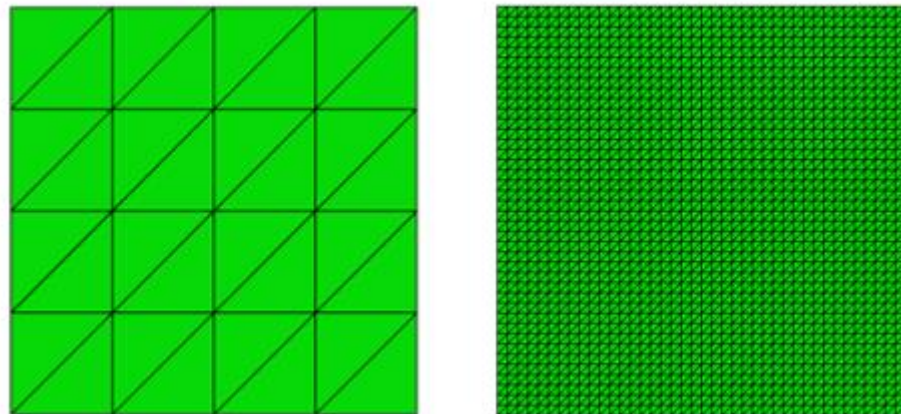


Figure 4.8 Sparse and refined triangle meshes

Example	Elasticity Matrix found	Elasticity Matrix theory	Error (%)
32 elements	$\begin{bmatrix} 219780.21 & 65934.6 & 0 \\ 65934.06 & 219780.21 & 0 \\ 0 & 0 & 76923.07 \end{bmatrix}$	$\begin{bmatrix} 219780 & 65930 & 0 \\ 65930 & 219780 & 0 \\ 0 & 0 & 76920 \end{bmatrix}$	0.0033
3200 elements	$\begin{bmatrix} 219780.21 & 65934.6 & 0 \\ 65934.06 & 219780.21 & 0 \\ 0 & 0 & 76923.07 \end{bmatrix}$	$\begin{bmatrix} 219780 & 65930 & 0 \\ 65930 & 219780 & 0 \\ 0 & 0 & 76920 \end{bmatrix}$	0.0033

Table 4.1 Comparison of the elasticity matrices obtained, with sparse and refined meshes, with the theoretical matrix.

The last test was made in a RVE, 648x864 mm, supposing that it is an homogeneous material with  $E=200\text{GPa}$ , Poisson rate 0.32, with structured unstructured meshes , as shown in figure 4.9. The results found are shown below in Table 4.2, where it can be seen a much smaller error and time of evaluation for the solution with the structured mesh.

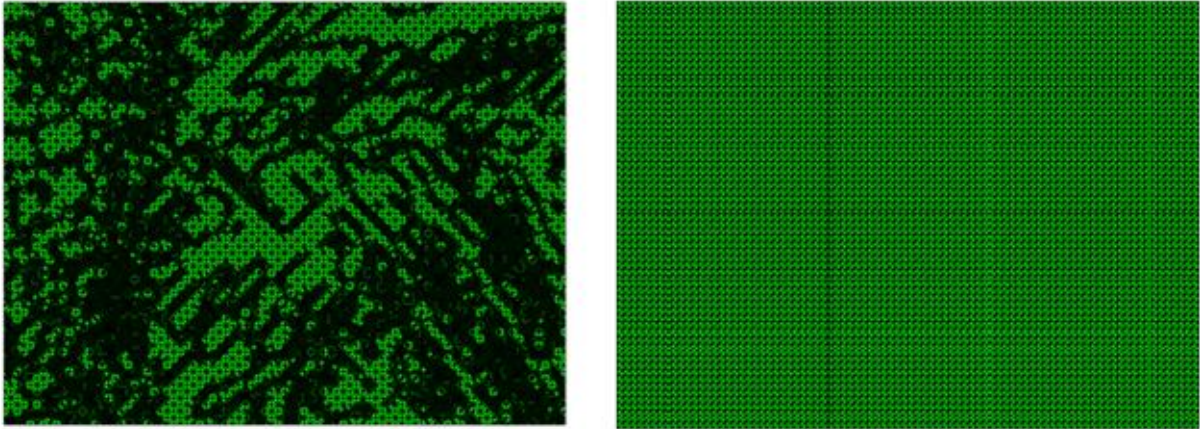


Figure 4.9 Unstructured and structured meshes

Example	Elasticity Matrix found	Elasticity Matrix theory	Error (%)	time (min)	Element number
Unstructured mesh	$\begin{bmatrix} 222820 & 71300 & 0 \\ 71300 & 222820 & 0 \\ 0 & 0 & 80130 \end{bmatrix}$	$\begin{bmatrix} 222820 & 71300 & 0 \\ 71300 & 222820 & 0 \\ 0 & 0 & 75760 \end{bmatrix}$	$8.3 \times 10^{-2}$	360	52414
Structured mesh	$\begin{bmatrix} 222820 & 71300 & 0 \\ 71300 & 222820 & 0 \\ 0 & 0 & 75760 \end{bmatrix}$				

Table 4.2 Comparison of the elasticity matrices obtained, with structured and unstructured meshes, with the theoretical matrix

A MATLAB script to predict the expansion coefficients in x, and y-directions in a 2.D problem was also a homogenous material the result obtained is shown in the table 3.3.

Direction	Expansion coefficient ( $\times 10^{-6} \text{ } ^\circ\text{C}^{-1}$ ) <sup>1)</sup>	Expansion coefficient ( $\times 10^{-6} \text{ } ^\circ\text{C}^{-1}$ ) theory	Error (%)
X	12	12	0.00
Y	12	12	0.00

Table 4.3 Expansion coefficient comparison with theory coefficient, in x and y-direction.

Tests were also performed for a 3D REM. A cube of 20x20x20 mm with 200GPa young modulus and 0.33 Poisson rate was considered for an homegeous material. The mesh used was tetrahedra elements, as is in figure 4.10. A very small error of almost zero per cent of was found, independently of mesh used.

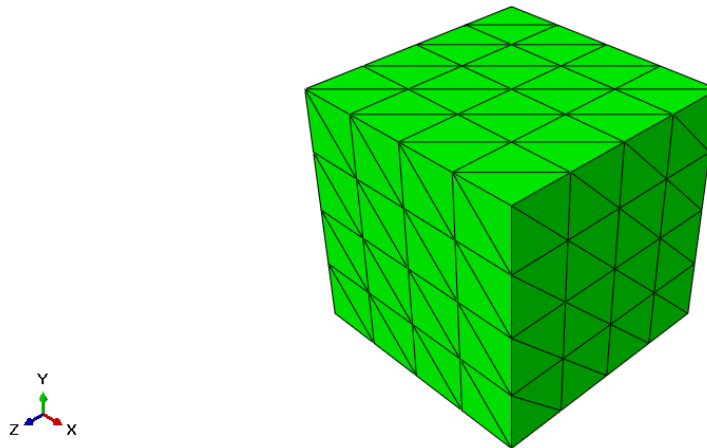


Figure 4.10 Tetrahedron mesh

Result						Theory						Error (%)
269230	115380	115380	0	0	0	269230	115380	115380	0	0	0	10 <sup>-5</sup>
115380	269230	115380	0	0	0	115380	269230	115380	0	0	0	
115380	115380	269230	0	0	0	115380	115380	269230	0	0	0	
0	0	0	7692	0	0	0	0	0	7692	0	0	
0	0	0	0	51282	0	0	0	0	0	51282	0	
0	0	0	0	0	7692	0	0	0	0	0	76920	

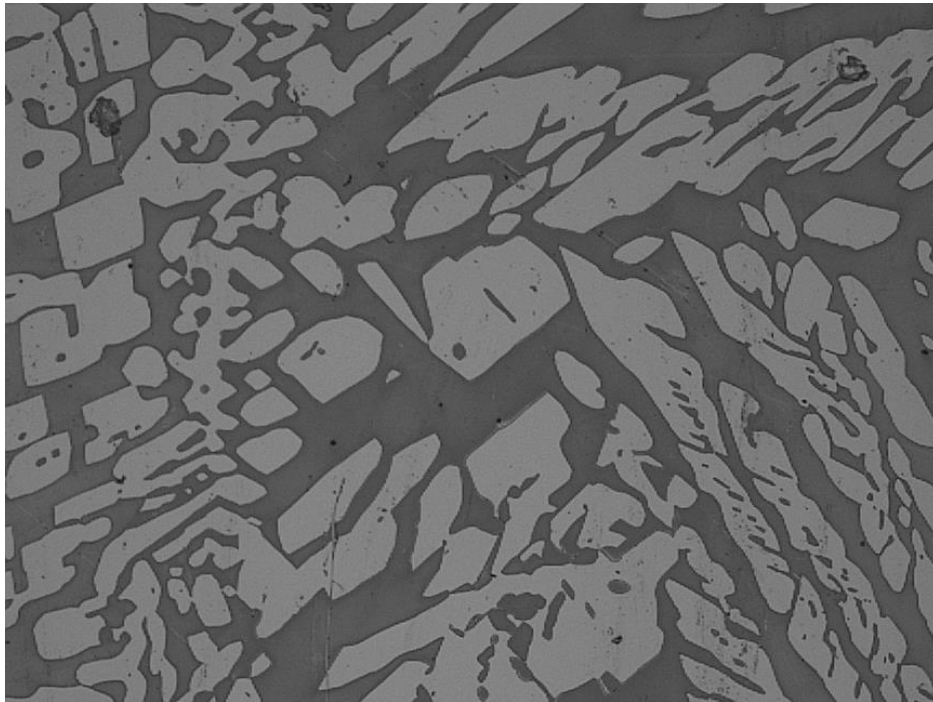
Table 4.4 Elasticity Matrix comparison with theory matrix

The real microstructure image can be obtained from optical microscopy or scanning electron microscopy, 2D, and micro-CT (Computed tomography), 3D. There are also different ways to generate stochastic microstructure 2D and 3D images using

commercial, like Altair Multiscale design or VGrain. It is possible to obtain results closer to reality, if we could use a real microstructure as it was the case in this research .

Assuming that properties at the microscale of all components and phases are not the same, the homogenised properties may be different, at different locations , from those found in the literature for the macroscale.

Therefore a multi-scale analysis was performed in this research. For the same material, a Duplex steel ASTM 2205, four different micrograph images, obtained for different cooling rates and different phases were used. The images were obtained in optical microscopy, in the Department of the Materials of the University of Porto and are shown in figures 4.11, 4.12, 4.13 and 4.14



*Figure 4.11 Duplex microstructure first test*

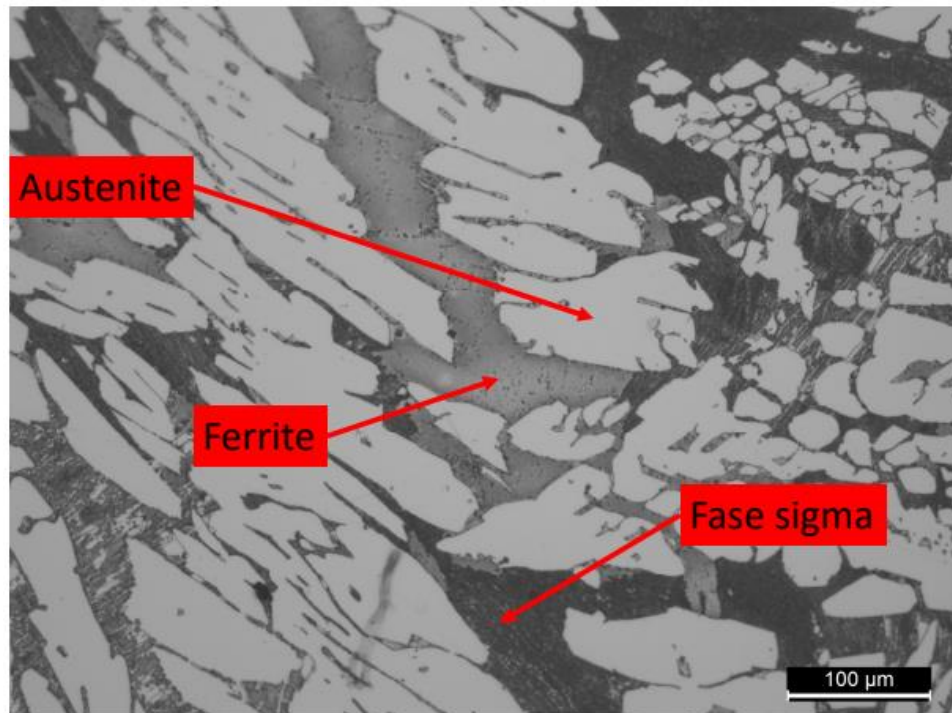


Figure 4.12 cooling rate of 6.6°Cmin-1

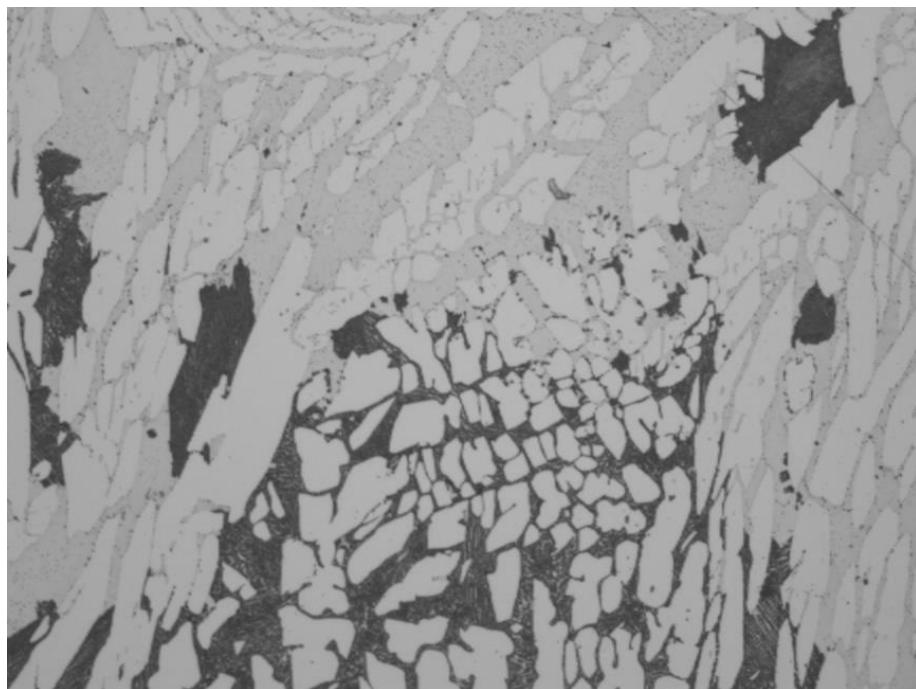


Figure 4.13 cooling rate of 7.2°Cmin-1



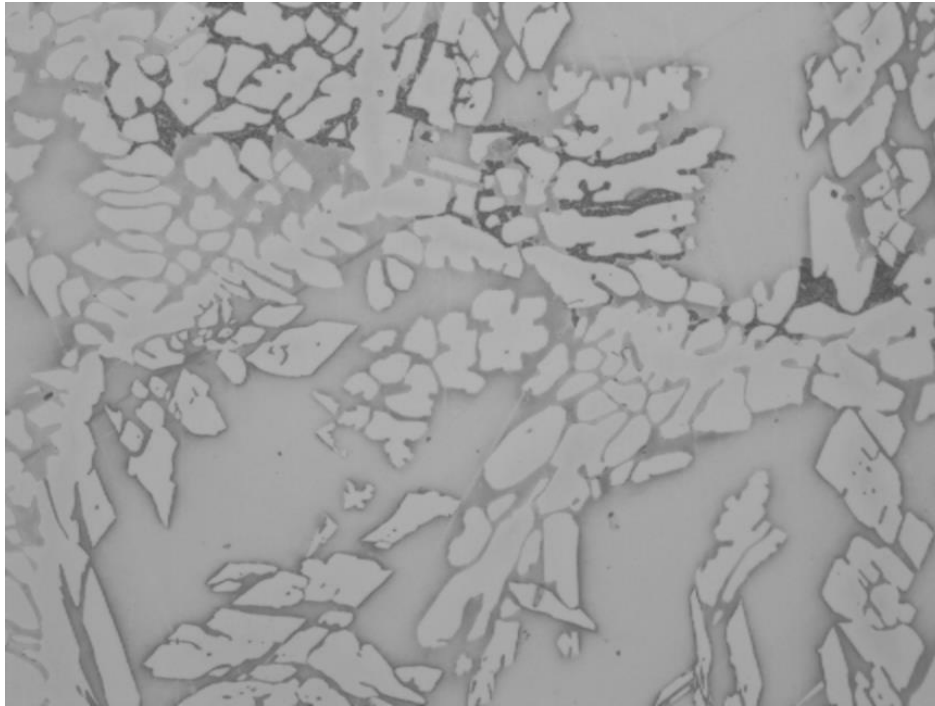


Figure 4.14 cooling rate of 156°Cmin-1

In these micrographs, in figures 4.12, 4.13 and 4.14 it is possible to see three different phases, austenitic, ferritic and sigma. Their properties are shown below in table 4.5.

<b>Phase</b>	<b>Young's Modulus (GPa)</b>	<b>Poisson rate</b>	<b>Expansion coefficient (<math>\times 10^{-6} \text{ } ^\circ\text{C}^{-1}</math>)</b>
<b>Austenite</b>	189	0.3	17.2
<b>Ferrite</b>	220	0.3	10.4
<b>Sigma</b>	201	0.3	17.6

Table 4.5 Austenite, ferrite and sigma properties

The OOF2 software was used to obtain the f.e. mesh, by a microstructural map. This software uses the pixel color to distinguish materials and generate the mesh. Subsequently, the file with the mesh information was exported to be used in Abaqus software. The meshes from the four micrographs are shown in figures 4.15, 4.16, 4.17, 4.18 and Table 4.6 below. There is a subtitle for the color of which phase in Abaqus.






<b>Materials</b>	<b>Color</b>
<b>AUSTENITE</b>	
<b>FERRITE</b>	
<b>SIGMA</b>	

Table 4.6 subtitle corresponding the material colours at the simulation

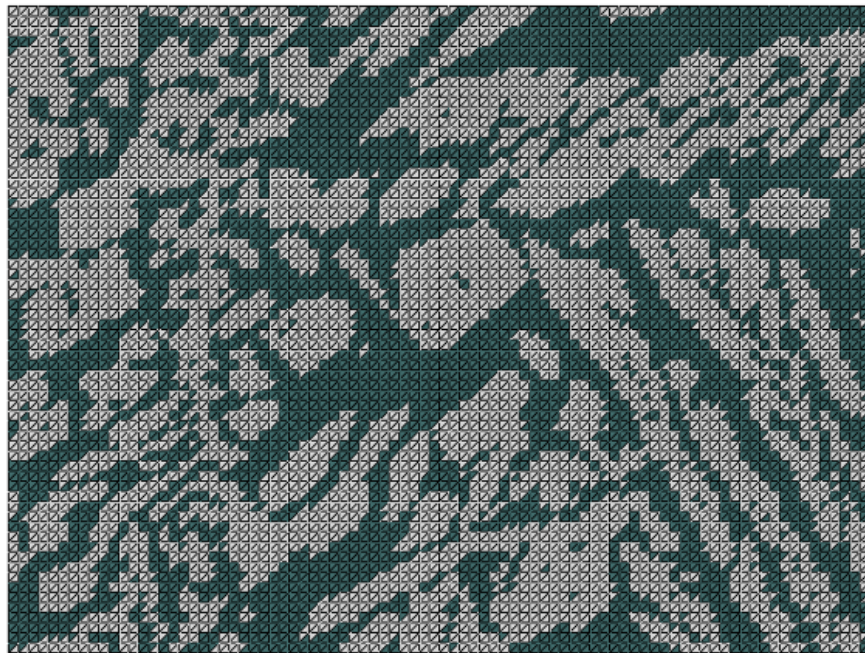


Figure 4.15 Mesh biphasic microstructure first test

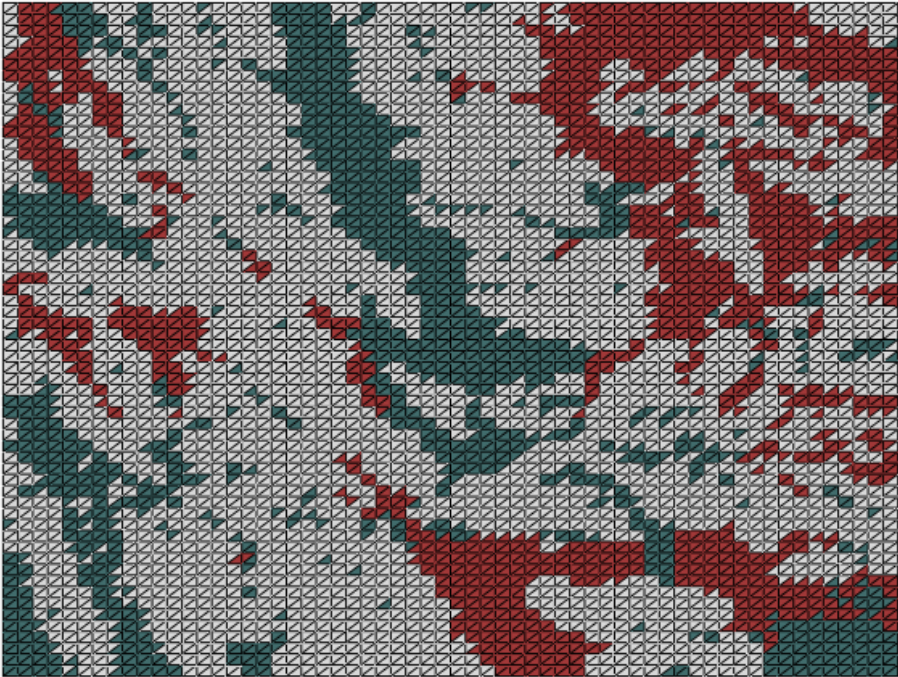


Figure 4.16 Mesh cooling rate of 6.6°Cmin-1

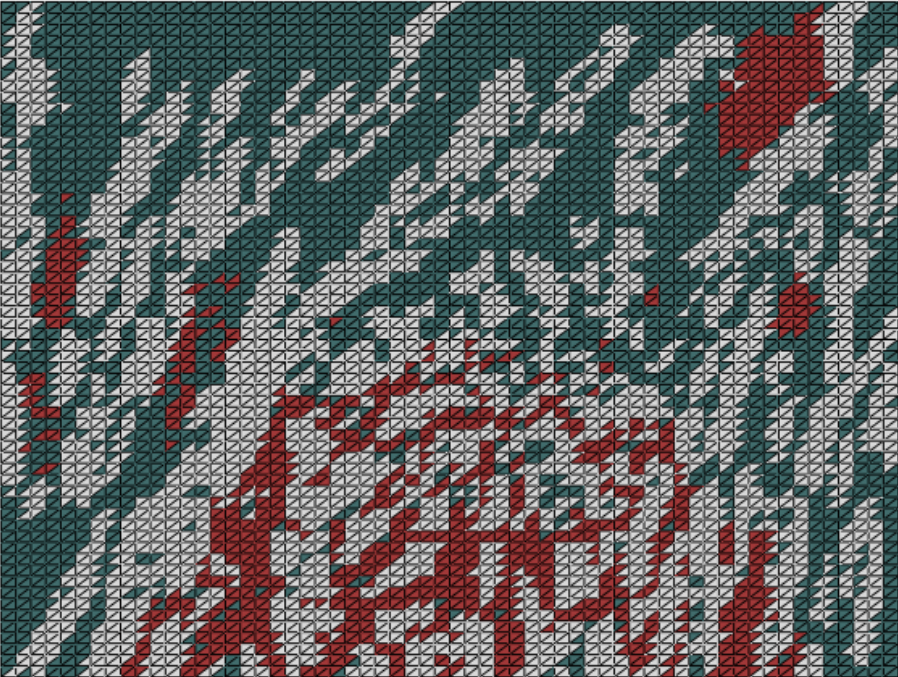


Figure 4.17 Mesh cooling rate of 7.2°Cmin-1



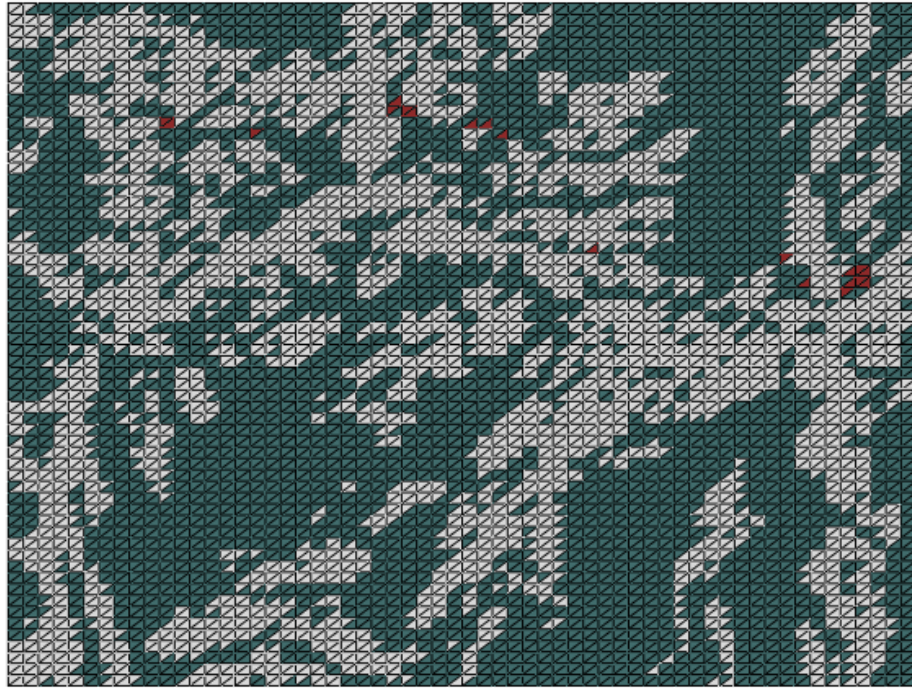


Figure 4.18 Mesh cooling rate of 156°Cmin-1

In Abaqus these meshes, representing REV's, were used, with appropriate boundary condition, as mentioned before, to obtain the elasticity matrix at the macroscale. using a python file to extract the information needed to the MATLAB code previously referred. The results of that calculation those is found in the literature and different from each other.

Micrography	Plane Stress	Plane Strain
<b>Biphasic microstructure (First test)</b>	$\begin{bmatrix} 222090 & 61320 & 40 \\ 61300 & 222010 & 80 \\ 46 & 40 & 80370 \end{bmatrix}$	$\begin{bmatrix} 269230 & 115380 & 0 \\ 115380 & 26923 & 0 \\ 150 & 0 & 76920 \end{bmatrix}$
<b>The cooling rate of 6.6°Cmin<sup>-1</sup></b>	$\begin{bmatrix} 216420 & 64930 & -110 \\ 64920 & 216490 & -100 \\ -220 & -50 & 75770 \end{bmatrix}$	$\begin{bmatrix} 265090 & 113590 & -130 \\ 113580 & 265170 & -110 \\ -240 & -60 & 75780 \end{bmatrix}$
<b>The cooling rate of 7.2°Cmin<sup>-1</sup></b>	$\begin{bmatrix} 222600 & 66790 & -30 \\ 66790 & 222650 & 10 \\ -130 & 30 & 77960 \end{bmatrix}$	$\begin{bmatrix} 272640 & 116810 & -40 \\ 116820 & 272700 & 0 \\ 150 & 40 & 77970 \end{bmatrix}$

$$\text{Cooling rate } 156^{\circ}\text{Cmin}^{-1} \begin{bmatrix} 226600 & 67950 & 90 \\ 67950 & 226460 & 50 \\ 150 & 30 & 79310 \end{bmatrix} \begin{bmatrix} 277530 & 118850 & 90 \\ 118850 & 277370 & 60 \\ 170 & 30 & 79320 \end{bmatrix}$$

Table 4.7 Results of elasticity matrix from different micrographic structures

The same meshes were used to find the thermal expansion coefficient, as previously described.. The results are shown below, table 4.8.

Micrographic	Expansion coefficient ( $\times 10^{-6} \text{ }^{\circ}\text{C}^{-1}$ )	
	X direction	Y-direction
<b>Biphasic microstructure (First test)</b>	14.074	14.091
<b>Cooling rate <math>6.6^{\circ}\text{Cmin}^{-1}</math></b>	15.915	15.930
<b>Cooling rate <math>7.2^{\circ}\text{Cmin}^{-1}</math></b>	14.420	14.420
<b>Cooling rate <math>156^{\circ}\text{Cmin}^{-1}</math></b>	13.206	13.195

Table 4.8 Results of expansion coefficient from different micrographics

Also, below some images obtained during the simulation to find the thermal expansion coefficients, where we can see how different is the deformation behavior of each phase.

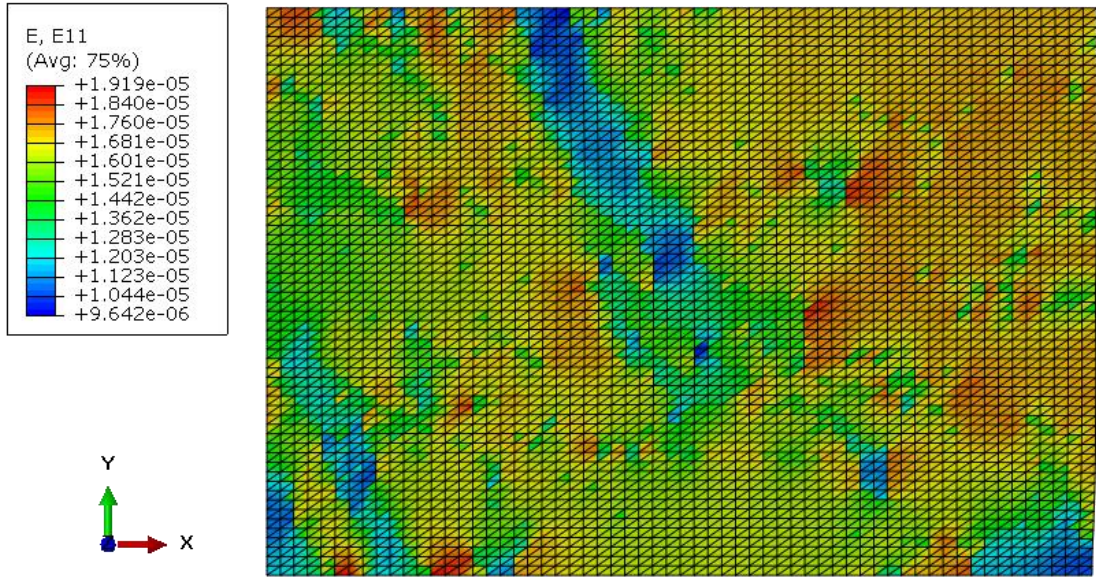


Figure 4.19 cooling rate of 6.6°Cmin-1, deformation in the x-direction

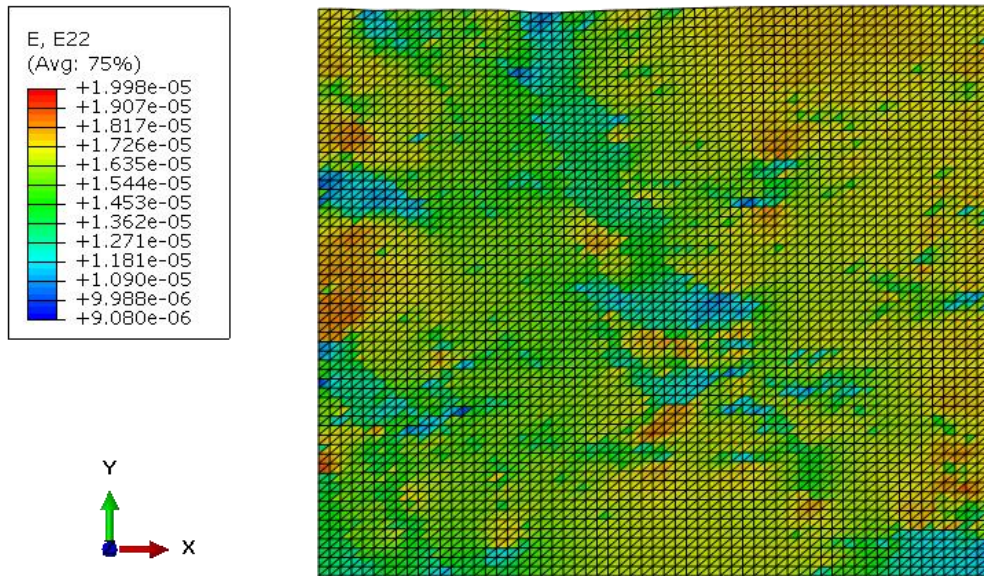


Figure 4.20 cooling rate of 6.6°Cmin-1, deformation in the y-direction



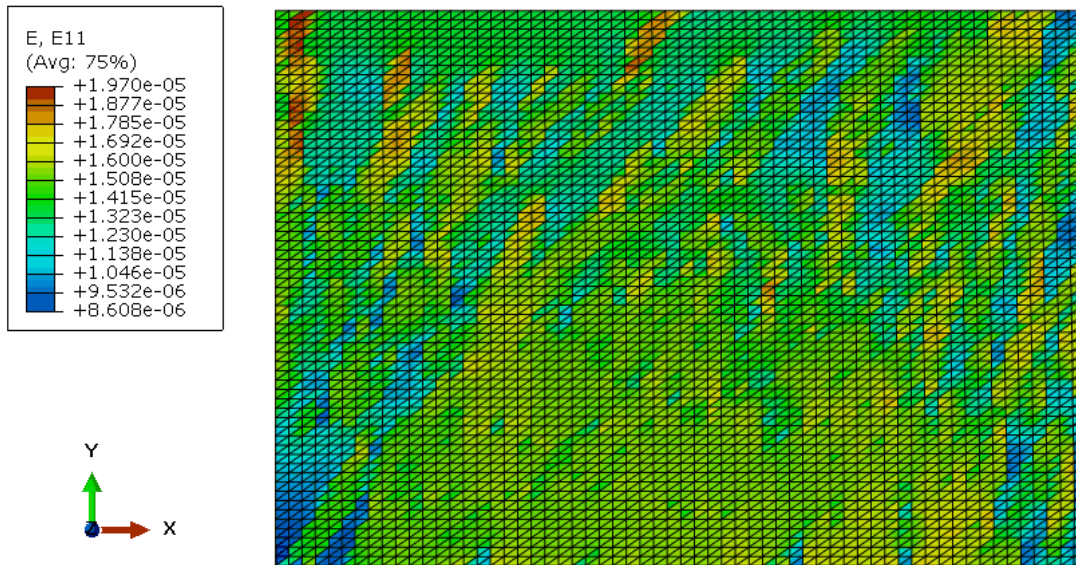


Figure 4.21 cooling rate of  $7.2^{\circ}\text{Cmin}^{-1}$ , deformation in the  $x$ -direction

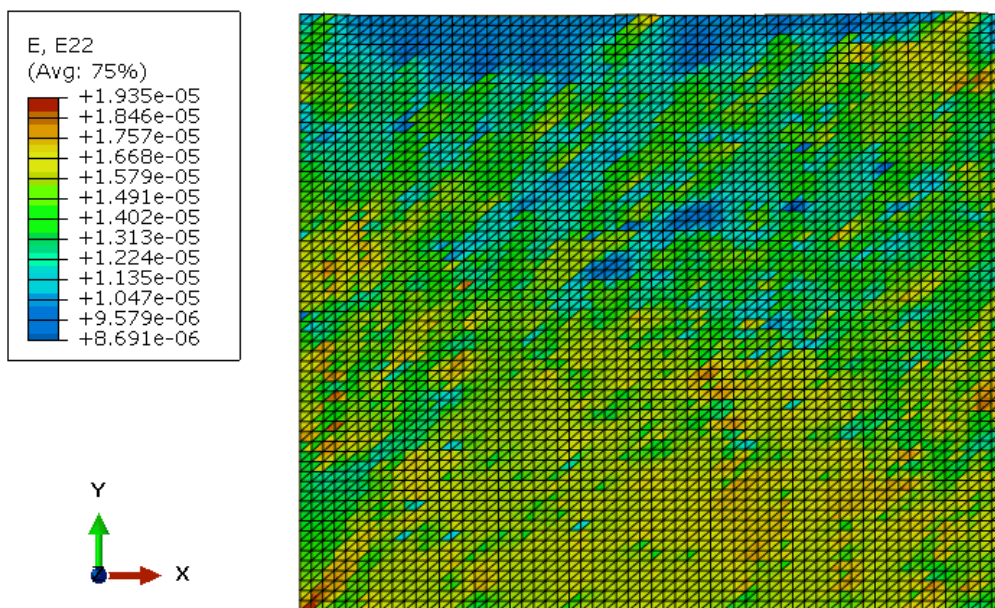


Figure 4.22 cooling rate of  $7.2^{\circ}\text{Cmin}^{-1}$ , deformation in the  $y$ -direction

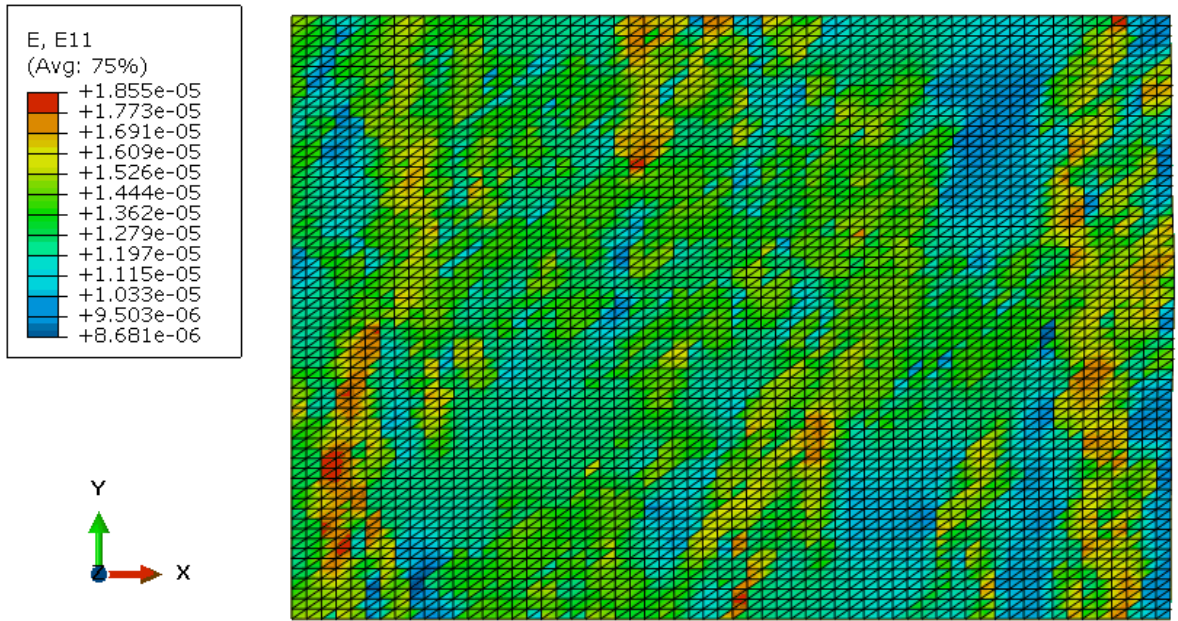


Figure 4.23 cooling rate of 152°C/min-1, deformation in the x-direction

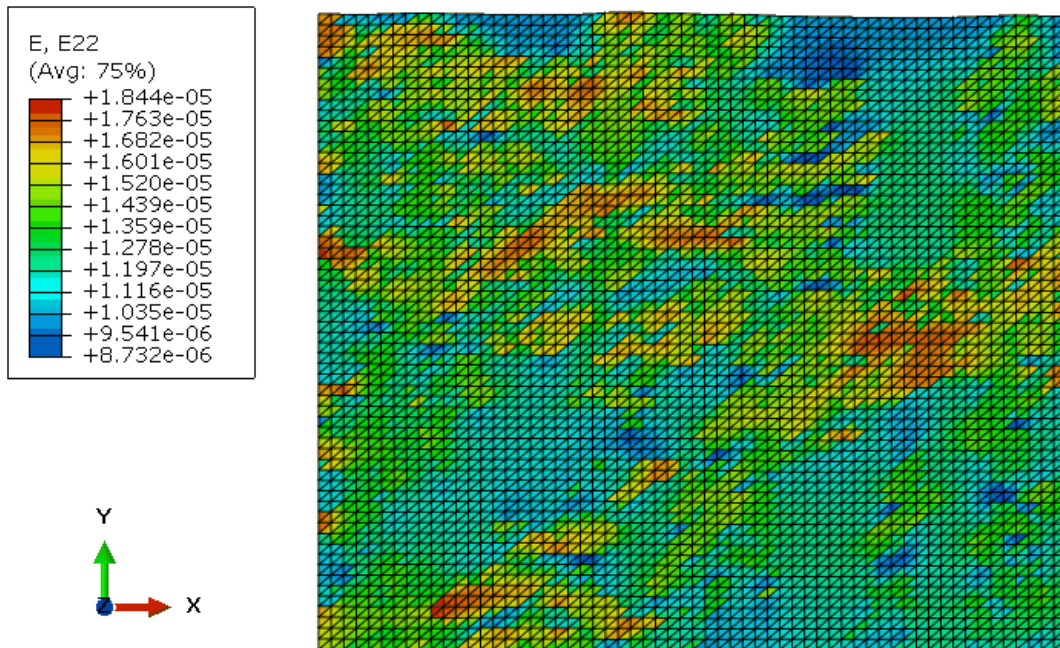


Figure 4.24 cooling rate of 152°C/min-1, deformation in the y-direction

## CHAPTER 5

### Influence of porosity and fractures in the micro and macrostructure

As known, porosity is a common defect in the casting process and it can result in a dramatic deterioration of mechanical properties, in particular, of fatigue and tensile properties. The pores could be formed in solidification with gasification of contaminants at molten-metal temperatures; shrinkage that takes place as molten metal solidifies and unexpected or uncontrolled changes in temperature or humidity.

During the solidification, there is a decrease in temperature. As a consequence, the gas solubility declines in the liquid/solid,  $C_L^{\max}$ , then the gas concentration in a liquid,  $C_L$ , becomes higher and if this concentration reaches a critical value, based on the limit of solubility of the gas in the liquid, then pores start to nucleate and grow. In Figure 5.1 it is possible to see the curves of  $C_L^{\max}$  and  $C_L$  and the intersection at the critical solid fraction for the onset of gas supersaturation. The second curve shows the influence of the local pressure at the growth of gas pores, where  $P_{\text{exp}}$  is the expansion pressure because of



phase transformation and  $P_{shr}$  is the shrinkage resistance to induced flow denoted the negative pressure.

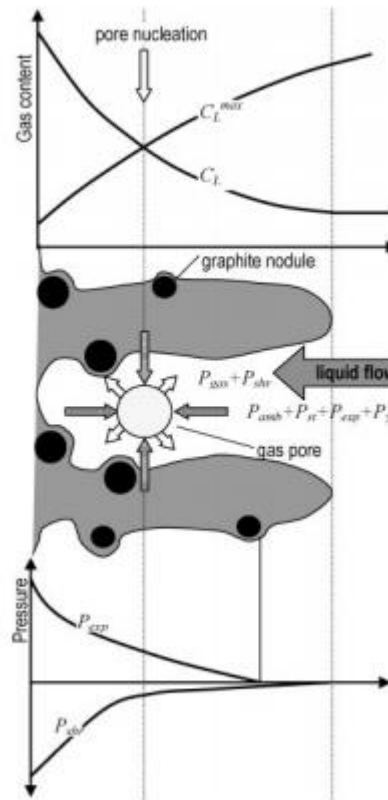


Figure 5.1 Pressure and gas content along the mushy zone of ductile Iron (From D. M. Stefanescu\*1 and A. V. Catalina,20)

There is also a negative volume variation developed during the phase transformation of most alloys and it can also influence the formation of pores resulting from solidification shrinkage. Some actions that must be taken in order to avoid the porosity due to shrinkage include e compensating for by interdendritic liquid flow (i.e., feeding) and for porosity due to gasification include the decrease of pressure in the mushy zone, which should be combined with a decrease in temperature, thus lowering the limit of the solubility of dissolved gases in the liquid.

Fractures or cracks is also a common defect of casting process, and they are formed within a material, usually as a result of residual stress, affecting directly its

resistance to fatigue. Fracture Mechanics addresses the study the cracks and their growth influence on material mechanical behavior. Linear Elastic Fracture Mechanics (LEFM), which will be used here, addresses, mainly, brittle fracture in which the plastic deformation can be disregarded. Ductile fracture in which plastic deformation must be taken into account will not be considered. It also addresses

The basic concepts of LEFM, which is addressed in this chapter can be found at C. Mouro, A. Augusto, Paulo M. S. Tavares de Castro (1986), and one of the numerical methods used to solve it, the XFEM, the extended finite element method, can be found at Duarte A. et Simone A., 2012.

Knowing that the pores and fractures harm the resistance of materials, how they are formed and can be avoided are important issues. The aim of this chapter is to check on how the elastic properties are affected by the existence of pores, inclusions and cracks. Later, at chapter 6, a more general influence on how pores and cracks influence material properties will be addressed.

### **5.1 Elastic properties affected by the existence of pores.**

Three different simulations were made, based on the microstructure of Duplex 2205 steel obtained with cooling rate 156°C/min. With three inclusions of pores in a region, A and B. The region A match a mainly austenitic phase, and the region B is mostly ferritic phase. The microstructure, the phases and the pores are shown in figures 5.2, 5.3, and 5.4 respectively, representing the finite element meshes that were used.

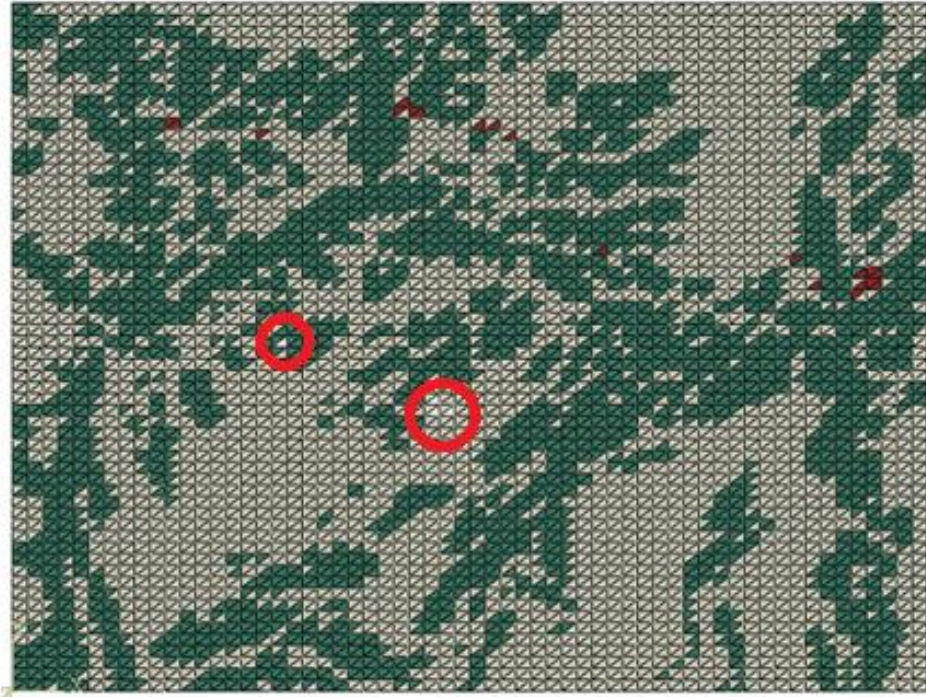


Figure 5.2 Microstructure Mesh cooling rate  $156^{\circ}\text{Cmin}^{-1}$ , four pores inclusion, region A.

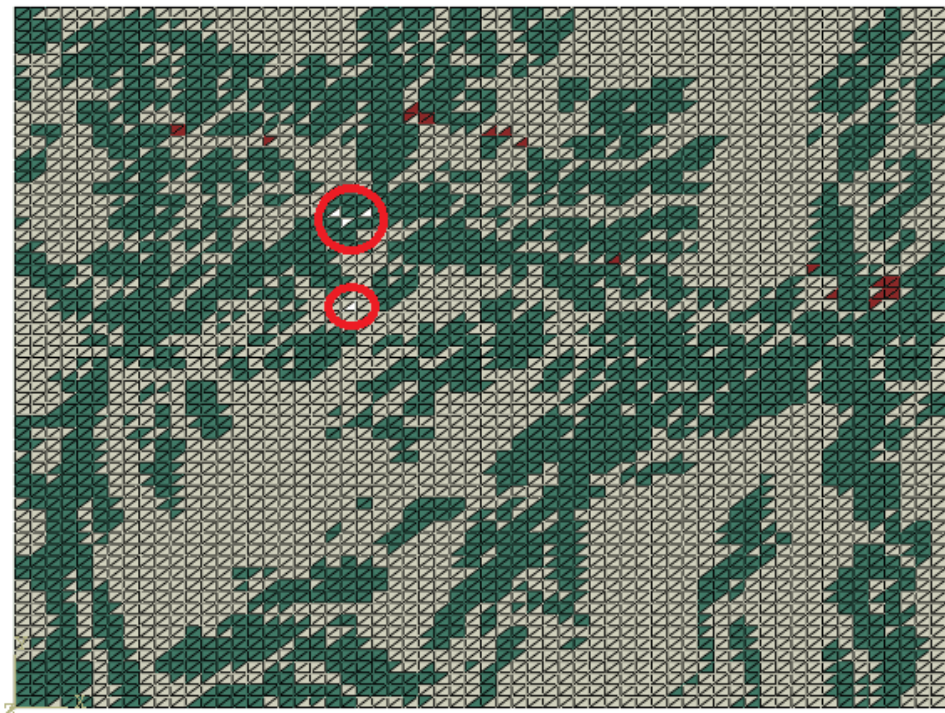


Figure 5.3 Microstructure Mesh cooling rate  $156^{\circ}\text{Cmin}^{-1}$ , four pores inclusion, region B.

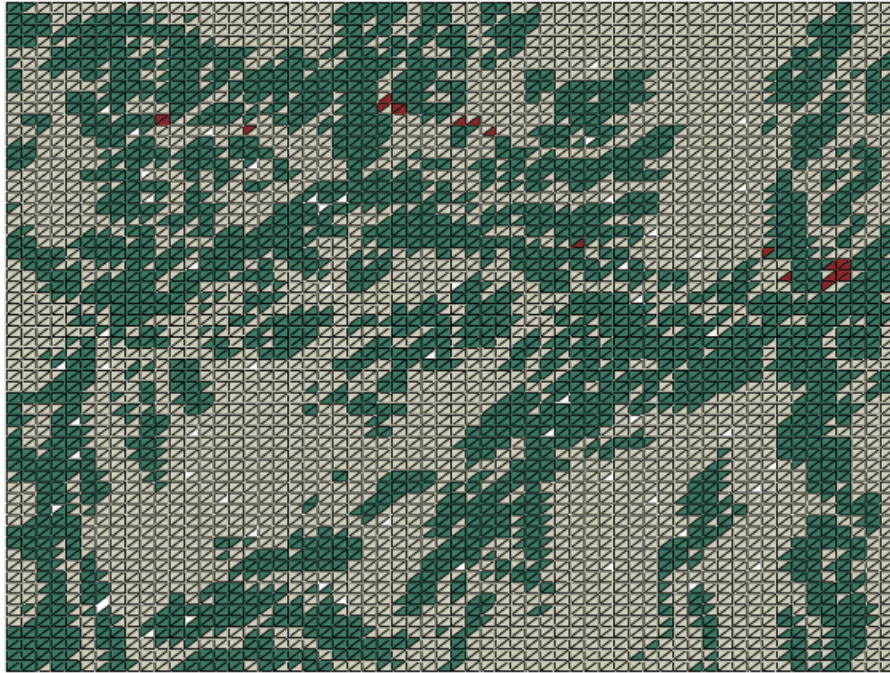


Figure 5.4 Microstructure Mesh cooling rate 156°Cmin-1, High pores inclusion

Table 5.1 presents the results of the elasticity matrices found after the simulation that used the same approach of chapters 3 and 4, with pores and without pores.

Example	Elasticity Matrix found	Elasticity Matrix theory
<b>Low porosity</b> <b>Region A</b>	$\begin{bmatrix} 226460 & 67870 & 60 \\ 67880 & 226260 & 40 \\ 140 & 20 & 79230 \end{bmatrix}$	
<b>Low porosity</b> <b>Region B</b>	$\begin{bmatrix} 226460 & 67880 & 70 \\ 67900 & 226300 & 50 \\ 160 & 20 & 79250 \end{bmatrix}$	$\begin{bmatrix} 226600 & 67950 & 90 \\ 67950 & 226460 & 50 \\ 150 & 30 & 79310 \end{bmatrix}$
<b>High porosity</b>	$\begin{bmatrix} 225700 & 67470 & 170 \\ 67460 & 225060 & 100 \\ 210 & 90 & 78850 \end{bmatrix}$	

Table 5.1 Elasticity Matrix comparison, Low porosity- Region A, B and high porosity, with matrix without pores.

The results showed that the number of pores and the regions of their inclusion is relevant to the elasticity matrix obtained by homogenization.

These results were used in chapter 6 to verify on how the materials could have their resistance affected by the pores.



## 5.2 Elastic properties affected by the existence of fractures.

In figures 5.5, 5.6, 5.7, 5.8 and 5.9 illustrate the location and length of the cracks, was used to find the elasticity matrix four different microstructure with cracks in different positions, lengths and orientation

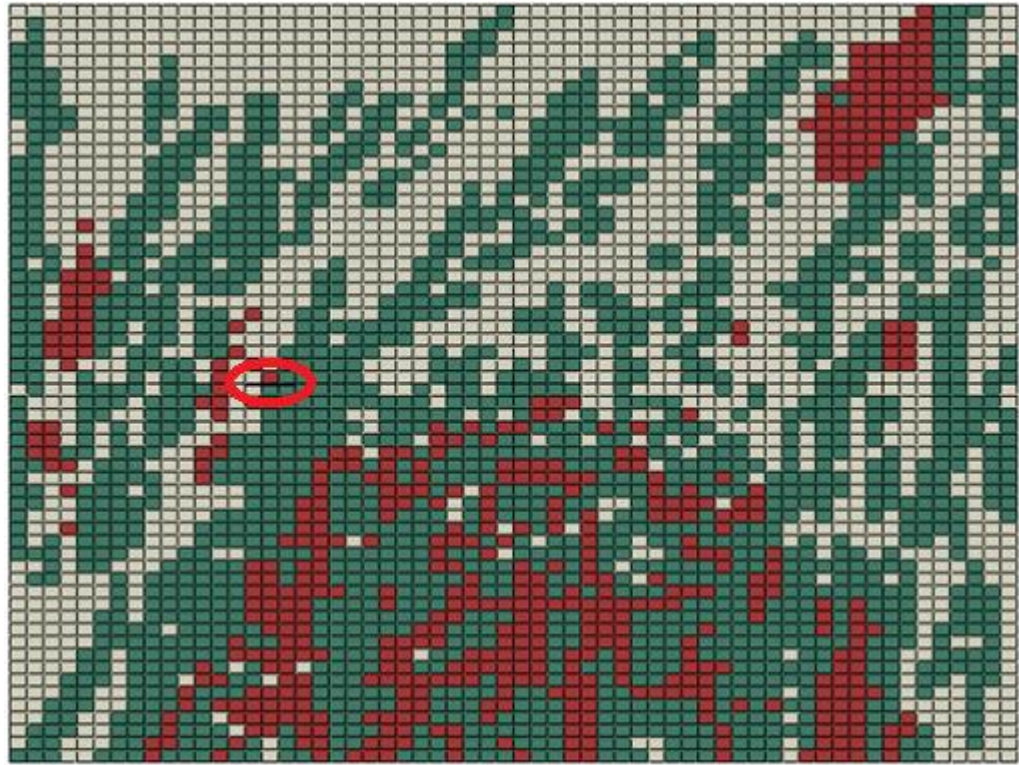


Figure 5.5 Microstructure Mesh cooling rate  $7.2^{\circ}\text{Cmin}^{-1}$ , Horizontal Crack Region A, 30 nm



Figure 5.6 Microstructure Mesh cooling rate 7.2°Cmin-1, Horizontal Crack Region A, 60 nm



Figure 5.7 Microstructure Mesh cooling rate 7.2°Cmin-1, Vertical crack





Figure 5.8 Microstructure Mesh cooling rate  $7.2^{\circ}\text{Cmin}^{-1}$ , Inclined Crack 30 nm

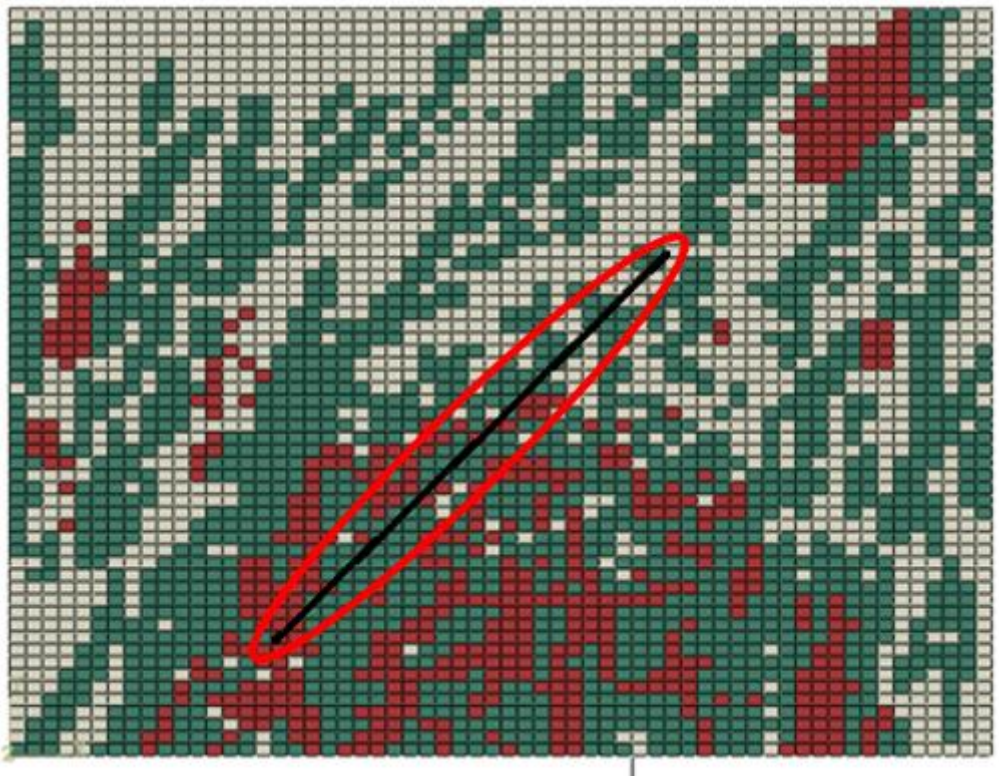


Figure 5.9 Microstructure Mesh cooling rate  $7.2^{\circ}\text{Cmin}^{-1}$ , Inclined Crack 480 nm

Table 5.2 presents the results of the elasticity matrix found after the simulation that used the same approach of chapters 3 and 4 and the comparison with the results of the elasticity matrix without considering any cracks.

Example	Elasticity Matrix found	Elasticity Matrix without cracks
Horizontal Crack Region A 30 nm	$\begin{bmatrix} 221730 & 66200 & -70 \\ 66260 & 220780 & 30 \\ -140 & -50 & 77500 \end{bmatrix}$	
Horizontal Crack Region A 60 nm	$\begin{bmatrix} 221380 & 65040 & -70 \\ 65110 & 216920 & 0 \\ -160 & -80 & 77500 \end{bmatrix}$	$\begin{bmatrix} 222600 & 66790 & -30 \\ 66790 & 222650 & 10 \\ -130 & 30 & 77960 \end{bmatrix}$
Vertical Crack Region B	$\begin{bmatrix} 220980 & 66370 & -150 \\ 66370 & 221780 & -10 \\ -290 & -100 & 77560 \end{bmatrix}$	
Inclined Crack Region C 30 nm	$\begin{bmatrix} 221650 & 66520 & -70 \\ 66520 & 221990 & -30 \\ -200 & -40 & 77670 \end{bmatrix}$	
Inclined Crack Region C 480 nm	$\begin{bmatrix} 161770 & 37450 & 30 \\ 37740 & 167390 & 830 \\ 14310 & 14830 & 77480 \end{bmatrix}$	

Table 5.2 Elasticity Matrix comparison, horizontal crack, 30 and 60 mm, vertical and inclined crack, with matrix without cracks.

The results showed that the region, orientation and length of the cracks are relevant to the homogenised elastic matrix obtained. These results were used in chapter 6 to verify on how the materials could have their resistance affected by the cracks .



## **CHAPTER 6**

### **Macroscale – Thermal/mechanical analysis.**

In order to verify the effects of the different properties at the microscale in the macroscale analysis a study was undertaken in a casting process in a Duplex steel 2205 C-Ring part, figure 6.1. This geometry chosen because it has different cooling rates in different locations, which result in different local microstructures.

The chapter has also a study, using results of a tensile test on stainless steel duplex 2205, on how defects at the microscale may affect the macroscale analysis.

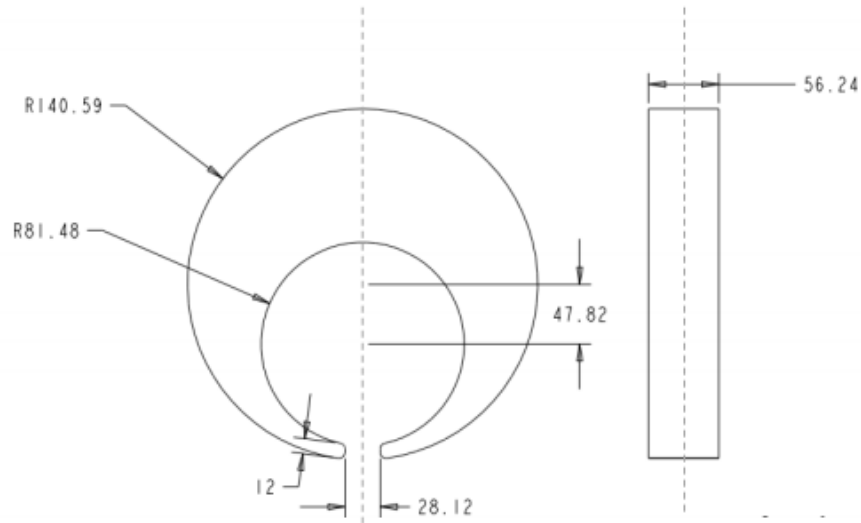


Figure 6.1 Part geometry, C-Ring dimensions

## 6.1 Thermal analysis

To identify different cooling rate zones a simulation was firstly undertaken with a homogenous material, using a static, thermal/mechanical analysis using Abaqus software (SIMULIA, 2010)

This model has various mechanisms of heat transfer that can occur at the same time in the cooling of the part. Initially the cooling inside the mold, with conduction between part and mold, convection between mold and ambient air and radiation between mold and ambient air. Secondly, cooling of the part outside the mold, with convection between part and ambient air and radiation between part and ambient air.

Besides, three parameters should be considered: the contact area, the temperature difference between the part and the mold and the heat transfer coefficient between the part and the mold. The conduction process between the part and the mold plays a critical role in heat transfer mechanism.

The simulation was done in a 2D setting and, therefore, for the third dimension a boundary condition was imposed, i.e., a cooling rate to represent the conduction between

the mold's wall and the part (more information can be found in the appendix). An unstructured f.e. mesh of triangle elements was used.

In Figure 6.2 is possible to see the different locations with different temperatures right after cooling, that was divided into two steps. In the first step the cooling inside the mold starts with a temperature of 1500°C of the part and 20°C of the mold. The temperature distribution of the part after 14400s is shown in figure 6.2a. In the second step, after 6585s the final temperature distribution is shown in figure 6.2b.

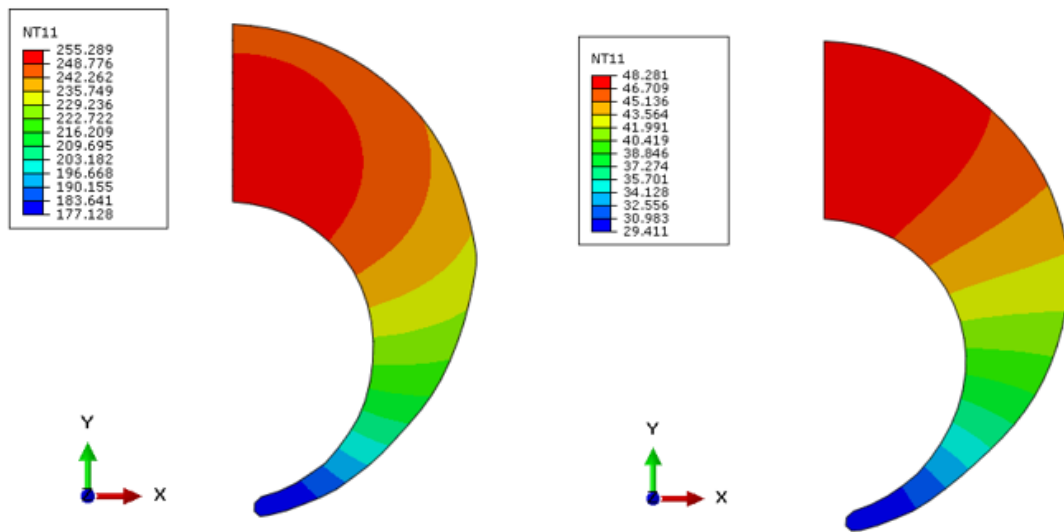


Figure 6.2 Cooling down a) In mould; b) Out of the mould

Some points with different cooling rates were selected along the part, as shown in figure 6.3. The points chosen were nodes 80, 121, 144, 194, 198, 234, 243, 39 of the f.e. mesh.

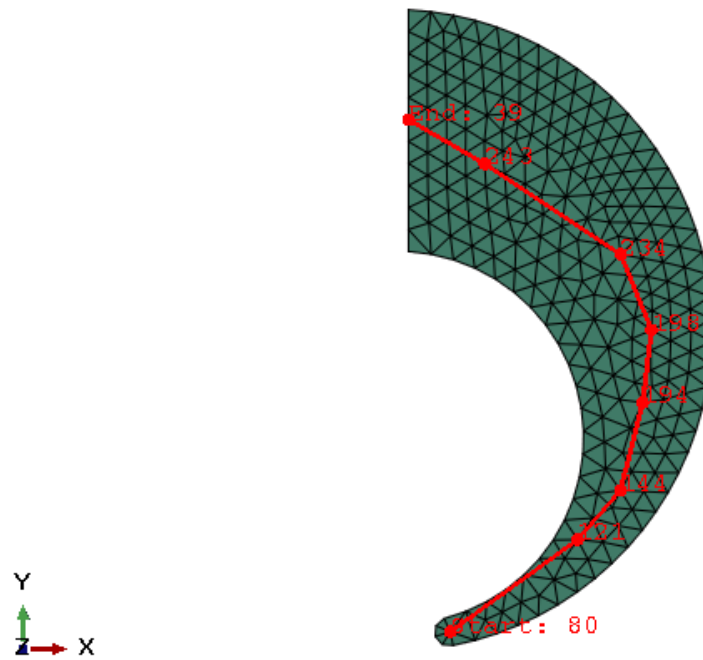


Figure 6.3 Selected point at C- Ring.

Temperature curves of the cooling stage for each node mentioned above are represented in figure 6.4. It is seen, as expected, that the nodes located above have a smaller cooling than the those at the inferior zone.

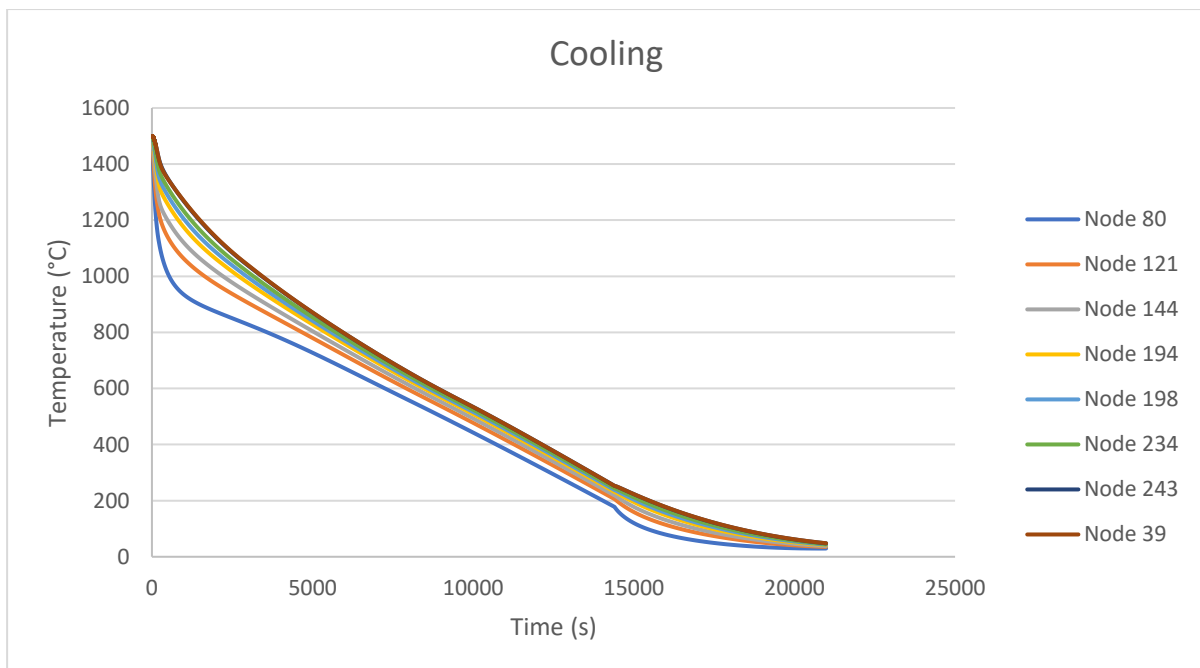


Figure 6.4 Curves of cooling at different locations

The cooling rates for each of those nodes are depicted in figure 6.5.

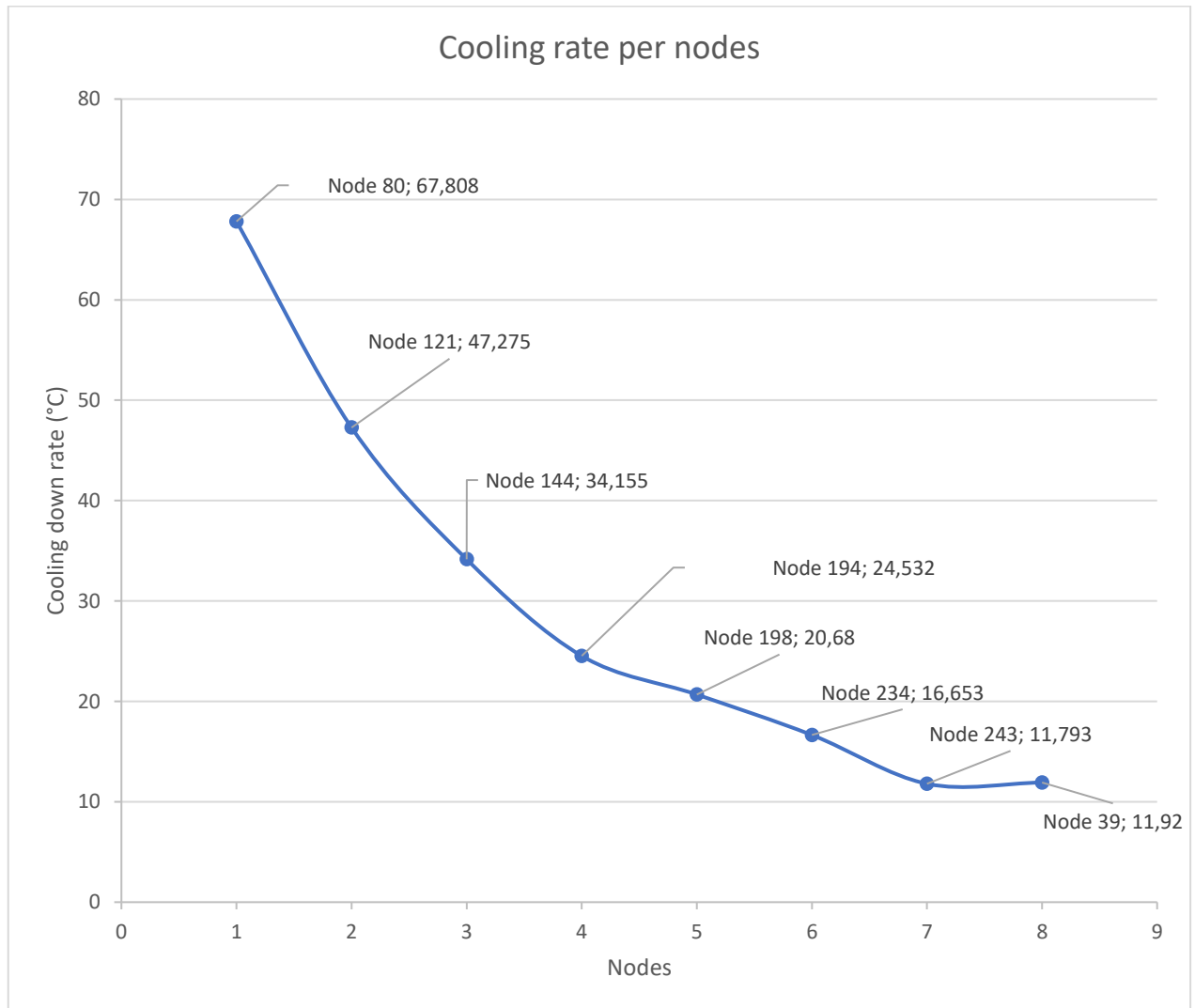


Figure 6.5 Curves of cooling rates at different locations

After the first analysis, it was possible to conclude the location that best fits the micrographics structures studied in chapter 4. The part model was divided in three parts whose cooling rate range approximate those of the known micrographic structures, attributing to each one the respective elasticity matrix as well as the thermal expansion coefficient.

In a second analysis thermal stresses were analysed. With the part divided in those three regions a thermal variation was imposed. Figure 6.6 shows those regions associated with different cooling rates. Only half C-Ring part was considered due to symmetry conditions.

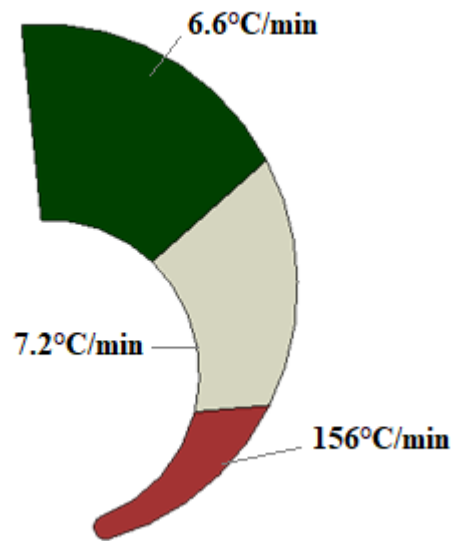


Figure 6.6 C-Ring with the different phases

The initial temperature was set  $900^{\circ}\text{C}$ , according to the curve of chapter 2, figure 2.12. At this temperature the piece has not turned into primary austenite and it was considered there was not a change phase. As in these analysis considering the mould is not essential, apart from the symmetry conditions, vertex 3 was fixed in x and y direction, as shown in figure 6.7. Again an unstructured mesh of triangle elements was considered and the final temperature was set to  $20^{\circ}\text{C}$ .

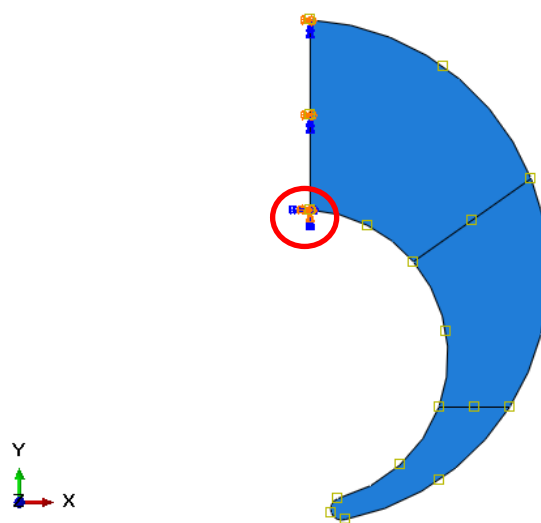


Figure 6.7 Vertex 3 was fixed in X and Y direction.

The resulting von Mises stress is shown in figure 6.8, with stress concentrations in the transition zones regions.

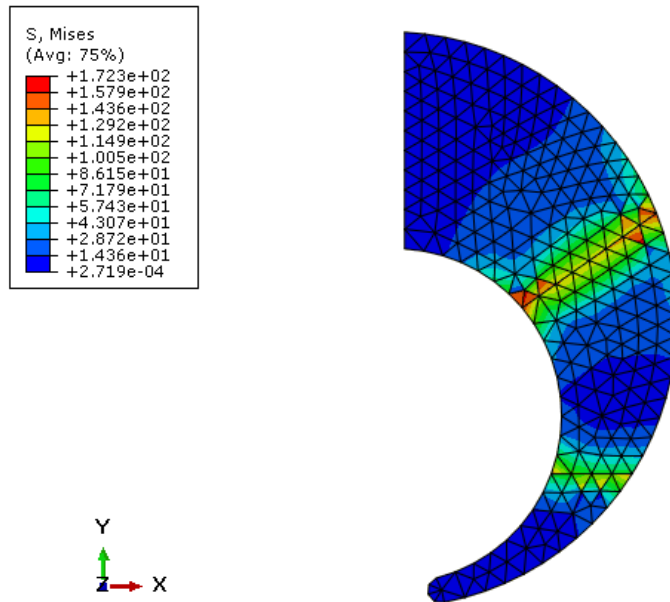


Figure 6.8 Thermal stress analysis at C-Ring.

It is also shown the deformation in real magnitude, figure 6.9. Analyzing the line from the fixed point up to other extreme, before and after cooling, the piece contraction is about 1,4% in y direction for the case of the microstructure obtained with a cooling rate of 6.6°C/min.

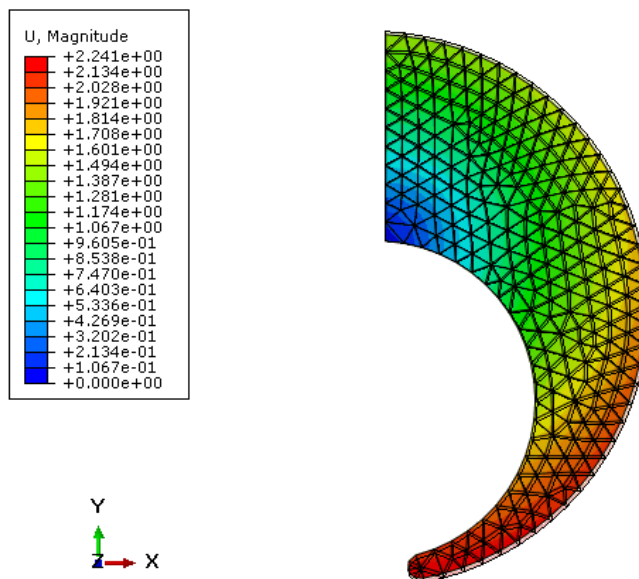


Figure 6.9 Deformation analysis at the C-Ring, before and after cooling.

The next test refers to a half rectangle, with dimensions in appendix B, aiming to represent the part through the thickness near a border, assuming that outside region is cooling faster than the inside, considering the same three different regions with microstructures associated to those different cooling rates, respectively  $6.6^{\circ}\text{C}/\text{min}$  in the middle,  $7.2^{\circ}\text{C}$ , and  $156^{\circ}\text{C}/\text{min}$  at the boundaries, as shown at figure 6.8.

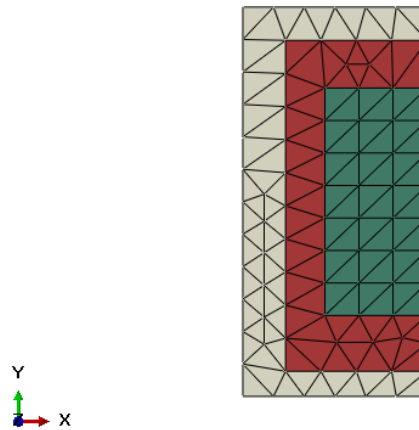


Figure 6.10 Piece with the different phases.

The symmetry condition is considered and the vertex 5 was fixed in x and y direction as shown at figure 6.11. The temperature variation was the same as the previous case, i.e., from  $900$  to  $20^{\circ}\text{C}$ .

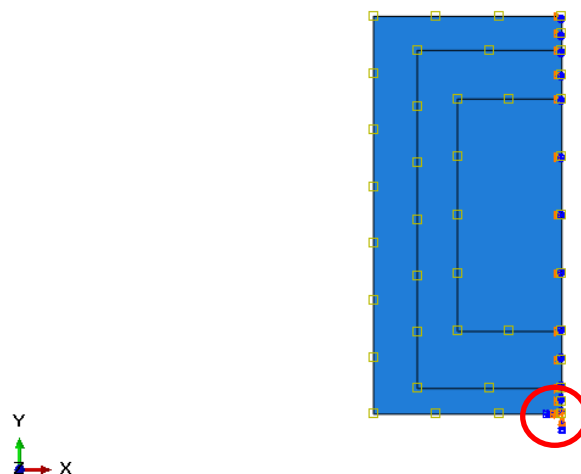


Figure 6.11 Middle point fixed in X and Y direction.



As in the previous example, the stress concentration is in the transition of the regions considered. Figure 6.12 shows the von Mises stress distribution resulting from the different thermal expansion/contraction coefficient of the zones and consequent stress concentrations that may be responsible for cracks and other damage defects.

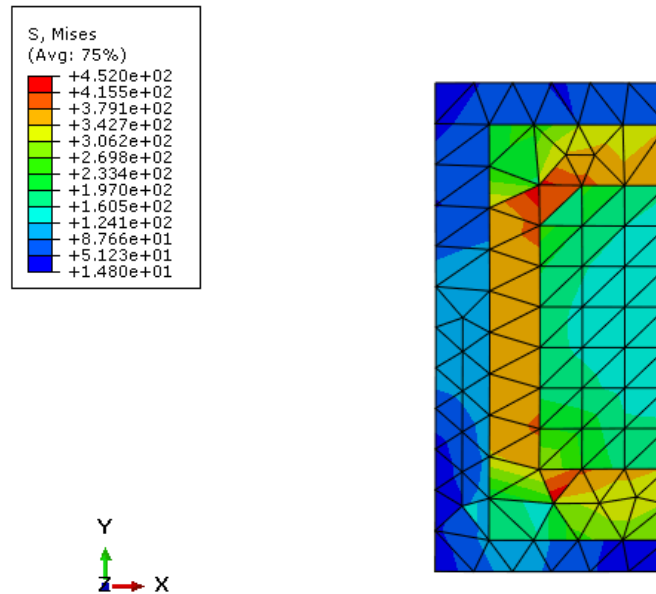


Figure 6.12 Thermal stress analysis at the piece,

The deformation in magnitude is shown, figure 6.13. Analyzing each phase, before, and after cooling the “156°C/min” zone has a contraction of about 1.25%, the “7.2°C/min” zone has a contraction of about 1.08% and the “6.6°C/min” zone has a contraction of about 1.38% in y direction. (The methodology to obtain the dimensions of each phase was the same as in previous example.)

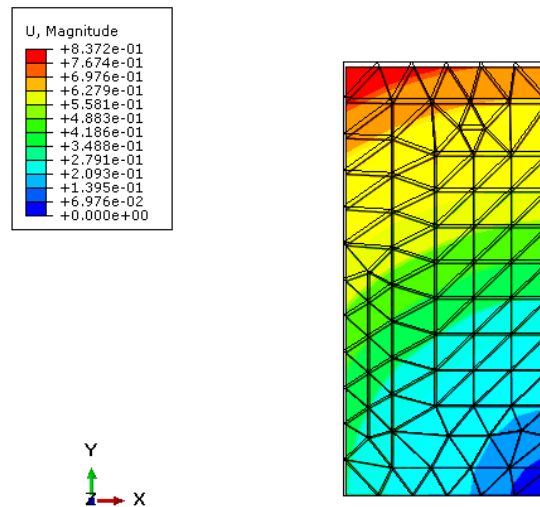


Figure 6.13 Deformation analysis at the piece, before and after cooling.

## 6.2 Mechanical analysis

To verify changes in the mechanical behaviour resulting from admitting different phase regions, pores or cracks when compared to the homogenous material a tensile test simulation was performed consisting on 13 cases.

The elasticity matrices for a plane stress state were obtained from microstructure homogenization . The dimensions of the tensile specimens are specified on the appendix B. The specimens were fixed at the two extremities and a gradual force from 0 to 2777 N was imposed. That force was chosen so that specimen remained in the elastic zone. Figure 6.14 shows the specimen and the boundary conditions imposed.

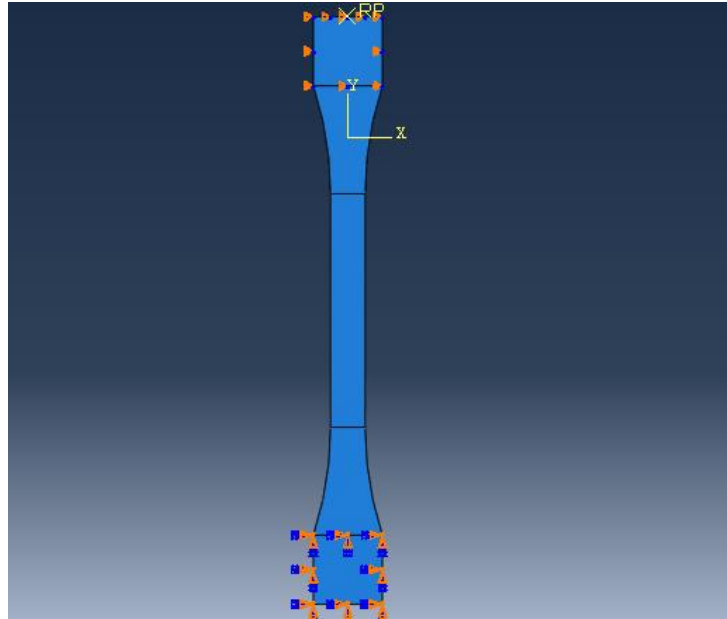


Figure 6.14- Specimen and boundary conditions.

A point in the middle of the specimen was chosen to analyse strain and stress evolution. In the first curves it is possible to see the tensile test for duplex microstructure specimen, figure 6.15 and for specimens with the phase resulting from the cooling rate of 6.6°C/min, figure 6.16 of 7.2°C/min, figure 6.17 and of 156°C/min, figure 6.18, and the comparison with the curve of stainless steel 2205.

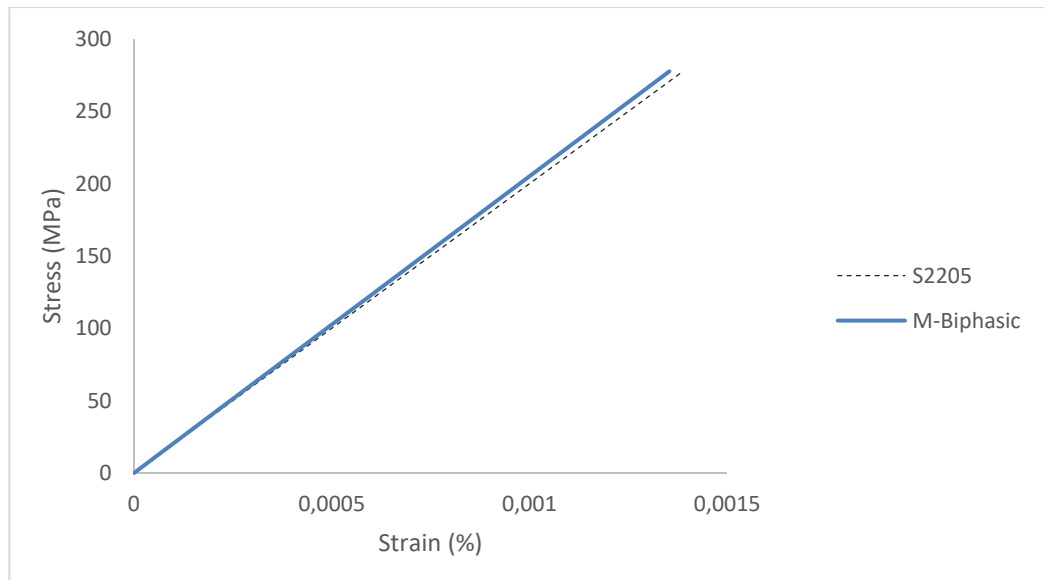


Figure 6.15 Curve of tensile test for Duplex microstructure specimen and the curve of stainless steel 2205.

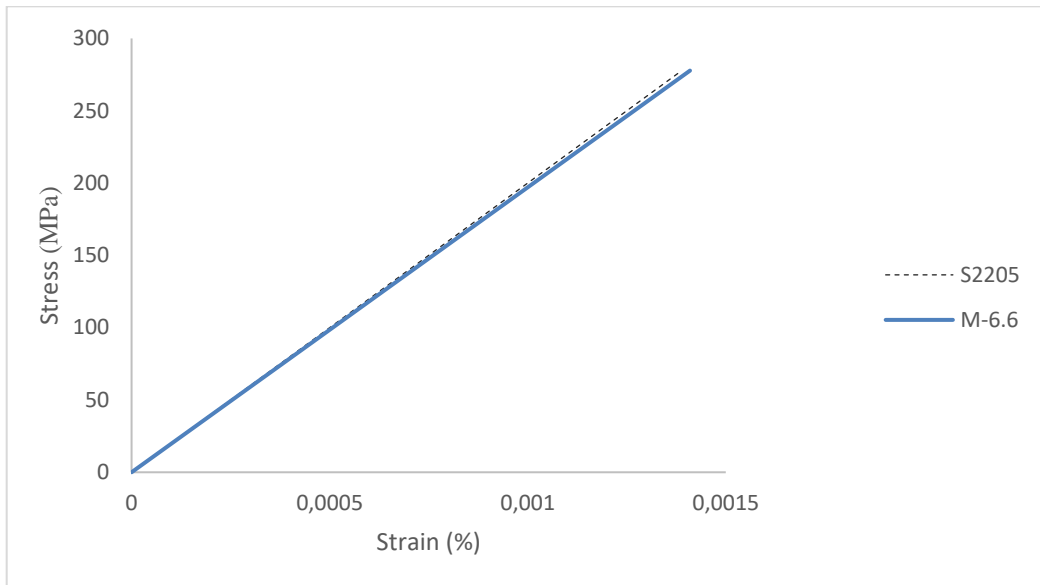


Figure 6.16 Curve of tensile test for 6.6°C/min microstructure specimen and the curve of stainless steel 2205.

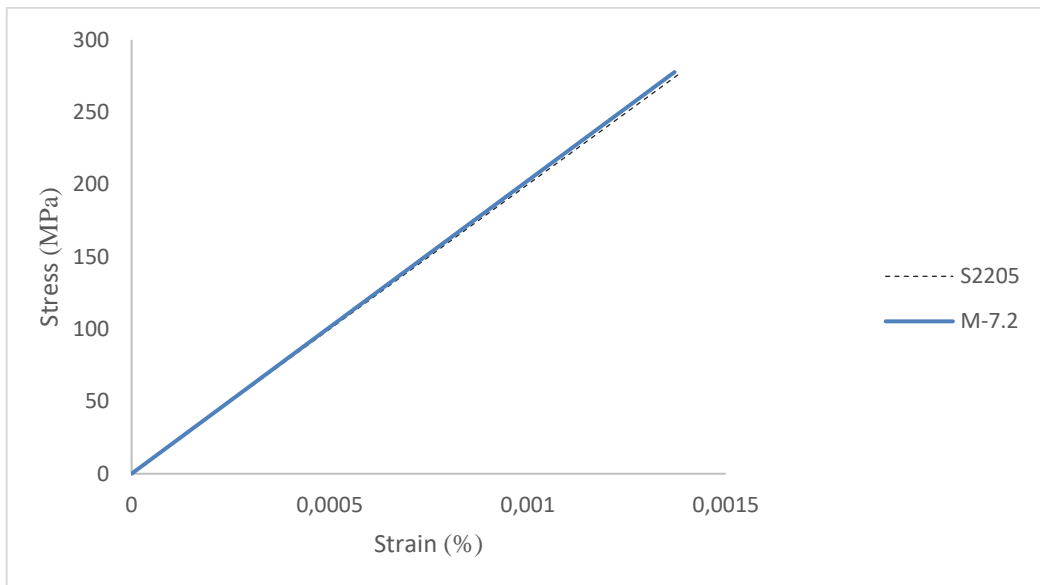


Figure 6.17 Curve of tensile test for 7.2°C/min microstructure specimen and the curve of stainless steel 2205.

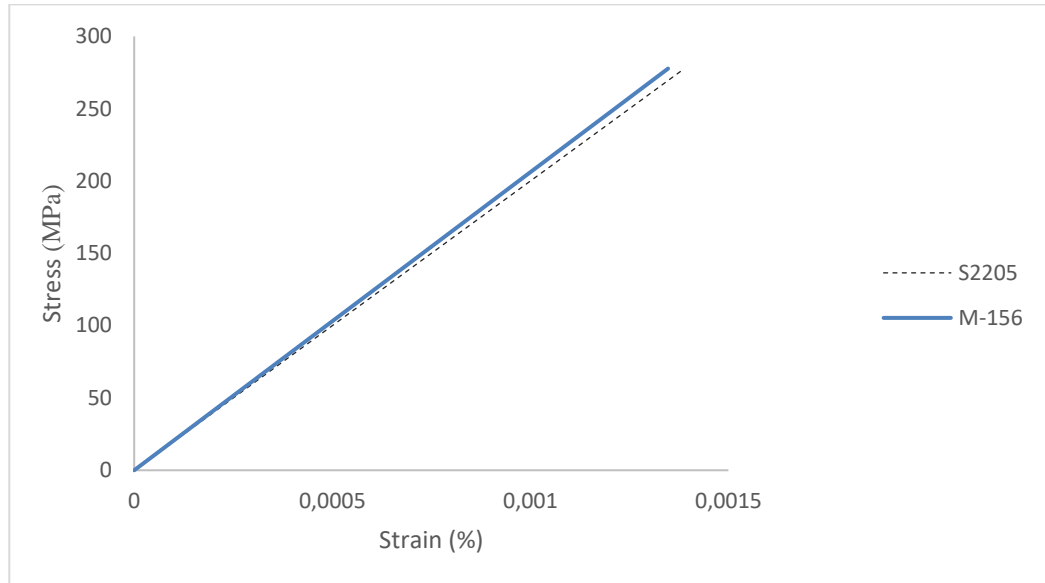


Figure 6.18 Curve of tensile test for 156°C/min microstructure specimen and the curve of stainless steel 2205.

The Young Modules of which curve is in the table below.

	Young Modules (GPa)	Steel 2205 (GPa)
<b>M-Duplex</b>	205,10	
<b>M-6.6°C/min</b>	197,02	
<b>M-7.2°C/min</b>	202,62	200
<b>M-156°C/min</b>	206,09	

Table 6.1 Young Modules of tensile test microstructure specimen and the stainless steel 2205.

After analysing of the curves and the Young modules it was possible to conclude that they are similar. The slope represents the stiffness or resistance of the material to elastic deformation, thus the larger module, the more rigid the material is or the smaller is the elastic deformation that results from the application of a given tension. The higher Young modulus comes from specimen with the microstructure resulting with a cooling rate of 156°C/min and the smaller one from the specimen with the microstructure resulting with a cooling of 6.6°C/min.

In a second set of tests pores were considered included in the microstructure of the specimens. The following figures show the curves obtained. Figure 6.19 shows the result with a specimen with low porosity, region A and region B, compared with a specimen with the the microstructure without pores, obtained with a cooling rate of  $156^{\circ}\text{C}/\text{min}$ . The second figure, 6.20, shows the curve of a case of high porosity compared with the specimen with no porosity.

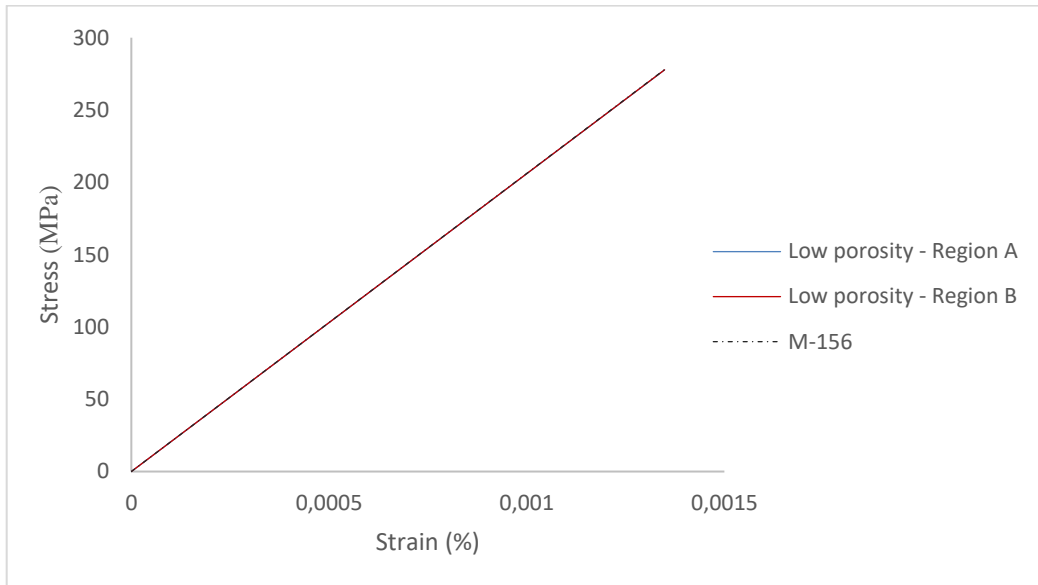


Figure 6.19 Curve of tensile test for low porosity  $156^{\circ}\text{C}/\text{min}$  microstructure specimen, region A, B and  $156^{\circ}\text{C}/\text{min}$  microstructure with no porosity.

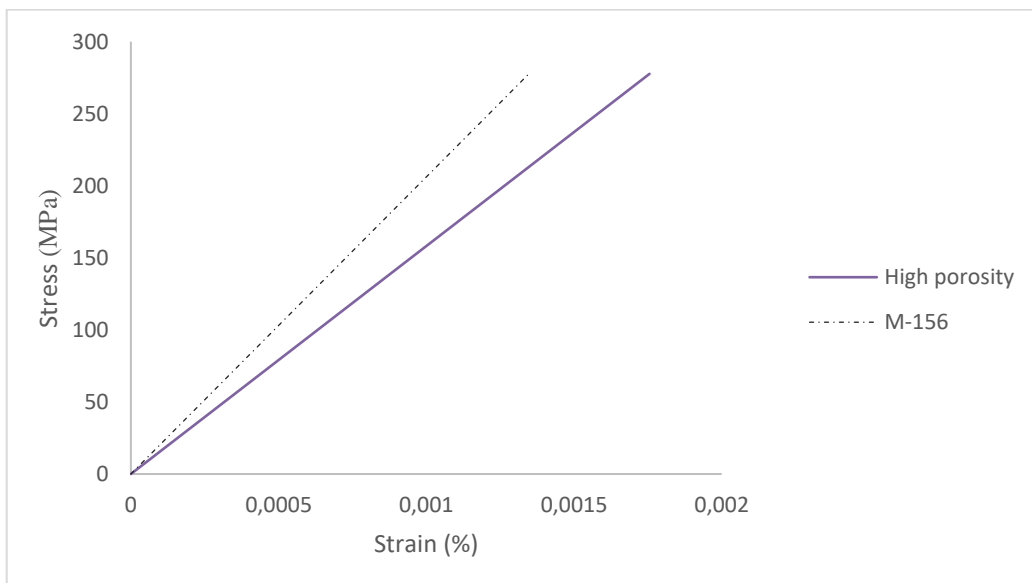


Figure 6.20 Curve of tensile test for High porosity  $156^{\circ}\text{C}/\text{min}$  microstructure specimen, region A, B and  $156^{\circ}\text{C}/\text{min}$  microstructure with no porosity.

The Young Modules of figure 7.16 and 7.17 are in the table below.

	Young Modules (GPa)	M-156°C/min (GPa)
<b>Low porosity Region A</b>	205,93	
<b>Low porosity Region B</b>	205,96	206,09
<b>High porosity</b>	158,15	

Table 6.2 Young Modules of tensile test for porosity microstructure specimen and the stainless steel 2205.

It is possible to see that inclusion of pores makes the material lose rigidity and it could depend on area affected and number of pores found.

In the third and final stage specimens that included cracks at the microstructure were tested. Figure 6.21 shows the comparison of the existence of a horizontal crack with 30 nm and with 60 nm in the microstructure of a specimen with a material obtained with a 7.2 °C/min cooling rate, chosen to simulate the presence of a crack. Figure 6.22 shows the comparison of the case of the existence of a vertical crack and also the original material without cracks. The final figure, 6.23 shows the inclined crack case also compared with the previous one.

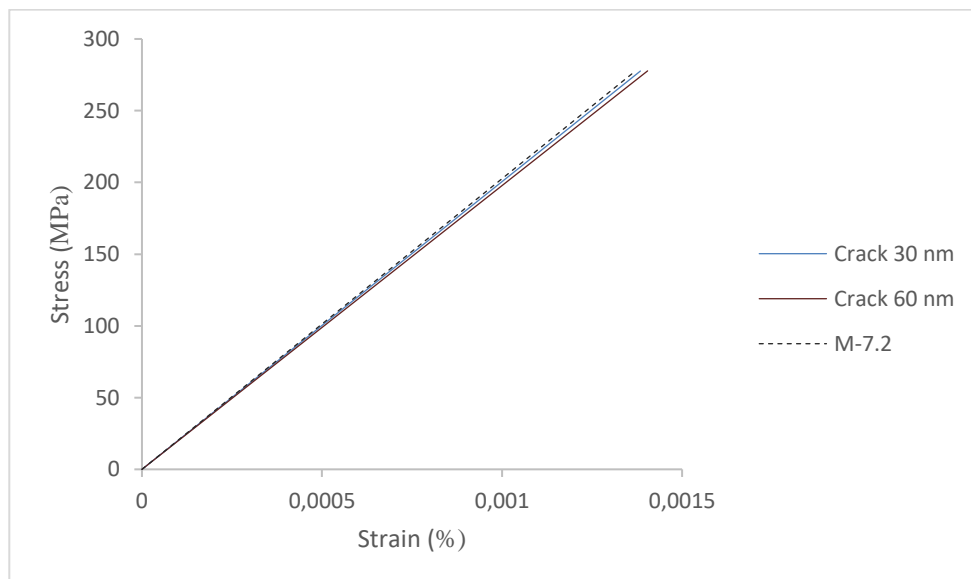


Figure 6.21 Curve of tensile test for crack 30nm and 60 nm in 7.2°C/min microstructure specimen and 7.2°C/min microstructure with no cracks.

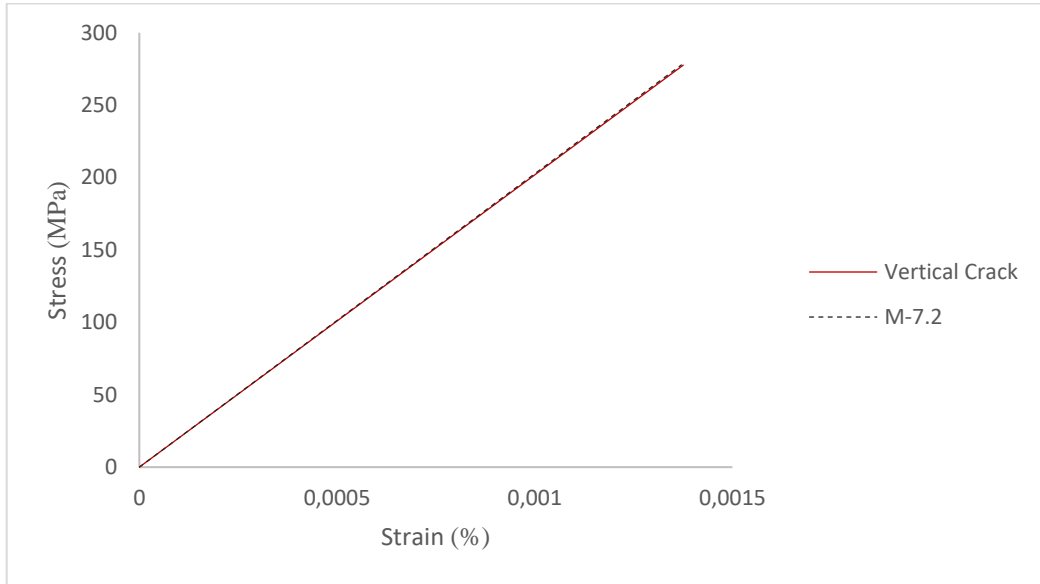


Figure 6.22 Curve of tensile test for vertical crack in 7.2°C/min microstructure specimen and 7.2°C/min microstructure with no cracks

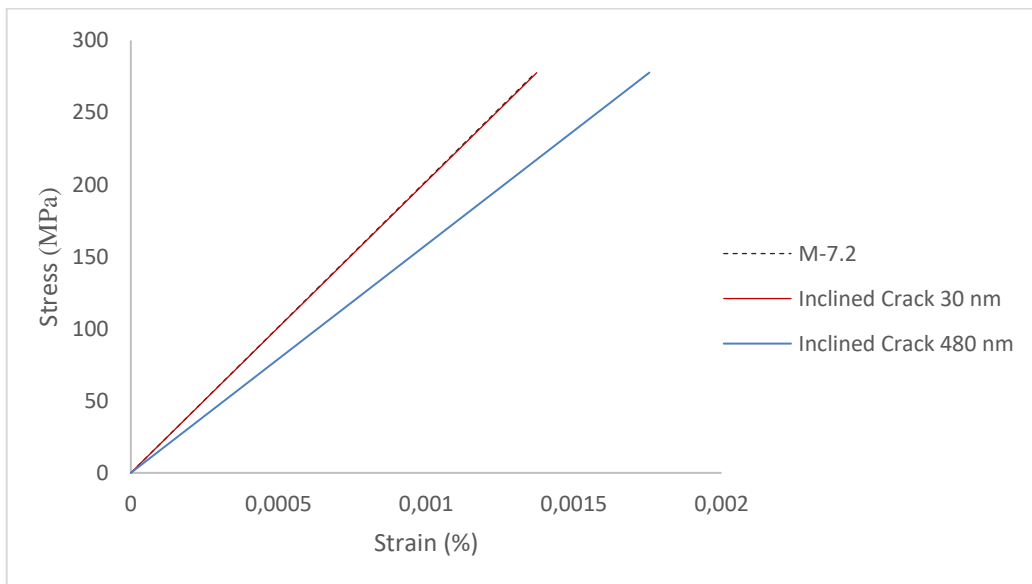


Figure 6.23 Curve of tensile test for inclined crack 30nm and 480 nm in 7.2°C/min microstructure specimen and 7.2°C/min microstructure with no cracks

The Young Modules of figure 6.18, 6.19 and 6.20 are in the table below.

**Young Modules(GPa)      M-156°C/min (GPa)**



<b>Horizontal Crack 30 mm</b>	200,58	
<b>Horizontal Crack 60 mm</b>	197,80	202,63
<b>Vertical Crack 30 mm</b>	201,83	
<b>Inclined Crack 30 mm</b>	202,01	
<b>Inclined Crack 480 mm</b>	158,17	

Table 6.3 Young Modules of tensile test microstructure specimen and the stainless steel 2205.

Analysing the data it is possible to conclude that, as expected, the cracks at the microstructure makes the rigidity smaller and the effect depends on the size and orientation of the cracks.

## **CHAPTER 7**

### **Conclusion and future work**

Multiscale modelling together with the recent technological advances in the computational area brings an advantage to design, process and evaluation of the materials' behaviour. This method is a powerful tool which makes it possible to predict the material behaviour at the macroscale, by analyzing it at smaller scales. In this thesis several simulations were carried out at the microscale scale, in manufactured part by a casting process, in order to predict the elasticity homogenised properties at the macroscale and evaluate possible resulting residual stresses due to the inhomogeneity at the microscale.

In the microstructural scale analysis, the OOF2 software was employed allowing to generate finite element meshes based on real images of microstructures. Those meshes were then exported to the Abaqus software, to evaluate traction/compression in x, y

directions and shear in x and y directions responses of a RVE. The results were extracted by a python file and used in a MATLAB code. The first stage was to use a simple square with homogenous material to test the script. The outcomes from that experience showed a very small error for 2D and 3D analysis, which proves the efficiency of the code developed. Unstructured and structured meshes, were also tested, showing an expected smaller error for the structured meshes.

Real microstructures with a heterogeneous material were used in a second stage, showing different homogenised elastic properties when compared to a homogenous material. Microstructures with pores included were used in a third stage. The examples simulated consisted in high and low porosity situations. In both cases different homogenised elasticity matrices were obtained, with, as expected, a sharper decay of the elasticity coefficients for higher porosities. Inclusion of cracks tested at a fourth stage, including horizontal, vertical and inclined cracks with different lengths. The elasticity properties obtained depend on orientation, length and position of the cracks. In a particular example, the inclusion of an inclined crack of 480 nm at the microstructure almost halved the homogenised elasticity coefficients.

Finally, the results obtained at the second stage were used in a C-Ring part obtained in a casting process and a rectangular part. Thermal analyses were initially performed for the parts, with a homogenous material, in order to evaluate different cooling rates in different localizations. Different properties were attributed to different locations within certain ranges of the cooling rates that were found, including different thermal expansion coefficients. Results showed the formation of residual stress in the zones of transition of phases.

A tensile test was the last simulation made to analyse the influence of the change of elasticity properties due to different microstructures obtained with different cooling rates in the casting process. The inclusion of pores and cracks at the microstructured level was also analysed and their influence on the final properties observed.

Some recommendations for future work may be proposed:.

- 1- To perform a multiscale analysis including an elastoplastic constitutive model at the microscale to allow a more predictive analysis of the material properties at the macroscale.

- 2- To perform the multiscale analysis with the high number of microstructures associated with different cooling rates, and consequently with more different phases or even considering with a continuous gradual change, in order to gain a more realistic information on possible stress concentration locations.
- 3- To obtain 3D micrographic results to perform this type of analysis in a three-dimension setting.

# References

Abaqus (2010). Abaqus 6.0 Analysis Manual. SIMULIA

Abaqus MIT, (2020) available on <<https://abaqus-docs.mit.edu/2017/English/SIMACAECAERefMap/simacae-c-engconcxmfemoverview.htm>> Accessed in May 2020

Abass H. et al. (1994). Oriented Perforations - A Rock Mechanics View. New Orleans, Louisiana: Society of Petroleum Engineers.

Aprodinox (2020)– Propriedades e aplicações aços inoxidáveis mais utilizados available on <<http://www.aprodinox.org.br/site/propriedades-e-aplicacoes-acos-util-2205.php>> accessed in April 2020

Atom molecules and ions (2020) – chemistry classroom, available on <<https://sites.google.com/site/comerschistryclassroom/useful-links/atom-molecules-and-ions>> accessed in April 2020

Britannica (2020) – Body centred cubic structure, available on <<https://www.britannica.com/science/body-centred-cubic-structure>> accessed in April 2020

BRITANNICA (2020) available on <<https://www.britannica.com/technology/steel/Application>>accessed in June 2020

Bruschitech (2020) – Shrinkage and porosity causes and remedies available on <<https://www.bruschitech.com/blog/shrinkage-porosity-causes-and-remedies>> accessed in April 2020

C. Mouro, A. Augusto, Paulo M. S. Tavares de Castro (1986) – Fadiga de estruturas soldadas –2 ed. Fundação Calouste Gulbenkian, Lisboa

Callister, W.D. (2010) Materials Science and Engineering: An Introduction. 8th Edition, John Wiley & Sons, New York.

Carvalho M. (2016) Computational Framework for Multi-Scale Analysis of Heterogeneous Materials using Digital Image Recognition

De, Sabrina & Mariotto, Fátima & Guido, Vanessa & Liu, Yao & Cho, Liu & Pacheco Soares, Cristina & Kátia, Regina & Cardoso, Katia. (2011). Porous Stainless Steel for Biomedical Applications. *Materials Research*. 14. 10.1590/S1516-14392011005000021.

Degarmo, E., Black, J. and Kohser, R. (2003) *Materials and Processes in Manufacturing*. 9th Edition, John Wiley & Sons, New York, 31 p.

Dieter, George Ellwood, and D. J. Bacon. *Mechanical Metallurgy*. London: McGraw-Hill, 1988.

Duarte, A. and Simone A. (2012). An Introduction to Partition of Unity and Generalized Finite Element Methods. Department of Civil Engineering, UNICAMP, Campinas, Brazil.

F. Incropera, David P. DeWitt, Theodore L. Bergman, Adrienne S. Lavine. (2006). *Fundamentals of Heat and Mass Transfer*. John Wiley & Sons.

Fish and T. Belytschko (2007) *A First Course in Finite Elements* - John Wiley & Sons, Ltd

Fish, Jacob. (2014). *Practical multiscale*. John Wiley & Sons

FRACTORY (2020) available on <<https://fractory.com/iron-carbon-phase-diagram/>> accessed in June 2020

Hill, R. (1963). Elastic properties of reinforced solids: Some theoretical principles. *Journal of the Mechanics and Physics of Solids*, 11(5):357-372

Horstemeyer, Mark. (2009). *Multiscale Modelling: A Review*. 10.1007/978-90-481-2687-3\_4

IMOA. (2014). *Practical Guidelines for the Fabrication of Duplex Stainless Steel*. Munich: International Molybdenum Association (IMOA).

Irgens, Fridtjov. (2008). *Continuum Mechanics*. 10.1007/978-3-540-74298-2.

Jacob Aboudi, Steven M. Arnold, Brett A. Bednarczyk, (2013) Chapter 3 - Fundamentals of the Mechanics of Multiphase Materials, Micromechanics of Composite Materials, Butterworth-Heinemann, Pages 87-145, ISBN 9780123970350,

Lai, M., Krempl, E., and Ruben, D. (2010). Introduction to Continuum Mechanics. ButterworthHeinemann, fourth edition.

Logan, D., A First Course in the Finite Element Method, Second Edition, PWS Publishing Company, ITP, 1993

Medinas M. (2015) An Extended Finite Element Method (XFEM) approach to hydraulic fractures: Modelling of oriented perforations

MIT (2020) Lectures-notes -unit 9 available on <<https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-20-structural-mechanics-fall-2002/lecture-notes/unit9.pdf>> Accessed in May 2020

MIT (2020) Module 3 Constitutive Equations available on <[http://web.mit.edu/16.20/homepage/3\\_Constitutive/Constitutive\\_files/module\\_3\\_with\\_solutions.pdf](http://web.mit.edu/16.20/homepage/3_Constitutive/Constitutive_files/module_3_with_solutions.pdf)> accessed May 2020

Neto, S., Peric, D., and Owen, D. R. J. (2008). Computational Methods for Plasticity, volume 55

O. C. Zienkiewicz, The Finite Element Method in Structural and Continuum Mechanics, McGraw-Hill, London (1971)

Petrini, M. (2019) Multiscale methodology for material characterization of advanced materials and processes.

Reis, F. J. P. (2014). Multi-Scale Modelling and Analysis of Heterogeneous Solids at Finite Strains.

SIMULIA. (2010). Abaqus 6.0 Analysis Manual. SIMULIA

Stefanescu, Doru & Catalina, Adrian. (2011). Physics of microporosity formation in casting alloys – sensitivity analysis for Al-Si alloys. International Journal of Cast Metals Research. 24. 144-150. 10.1179/136404611X13001912813780.

Technology, N. (2020). OOF2. <http://www.ctcms.nist.gov/oof/oof2/>, Accessed in April 2020.





**Table C2 - Duplex stainless-steel grade 2205 true stress-strain table, for different temperatures (Reduced version)**

	25 °C	400 °C	700 °C	1100 °C	700 °C
Plastic Strain	Stress (GPa)	Stress (GPa)	Stress (GPa)	Stress (Gpa)	Stress (Gpa)
0.00	549.3809	356.6855	331.9727	18.1361	331.9727
0.04	770.6174	542.7272	501.4245	26.6841	501.4245
0.08	835.7273	595.3232	525.0923	26.6755	525.0923
0.12	877.1831	628.9428	524.4298	26.639	524.4298
0.16	908.089	654.0954	522.8231	26.5508	522.8231
0.20	932.9191	674.3631	519.9471	26.3966	519.9471
0.24	953.7679	691.4239	515.8724	26.1929	515.8724
0.28	971.793	706.2061	510.9685	25.969	510.9685
0.32	987.7044	719.28	505.6098	25.6512	505.6098
0.38	1008.5926	736.4788	497.3664	25.2274	497.3664
0.42	1020.9729	746.6917	491.978	25.0065	491.978
0.46	1032.3649	756.1019	486.774	24.6827	486.774
0.50	1042.9234	764.8345	481.7902	24.4088	481.7902
0.54	1052.7691	772.9868	477.0409	24.1881	477.0409
0.58	1061.9975	780.6362	472.5176	24.0136	472.5176
0.62	1070.6859	787.8452	468.2227	23.7404	468.2227
0.66	1078.8979	794.6653	464.1393	23.5024	464.1393
0.70	1086.6859	801.1391	460.2528	23.3073	460.2528
0.74	1094.0943	807.3024	456.5506	23.1483	456.5506
0.78	1101.1606	813.1858	453.0292	23.0192	453.0292
0.82	1107.9169	818.8154	449.659	22.8049	449.659
0.86	1114.391	824.2137	446.4412	22.6039	446.4412
0.90	1120.6068	829.4003	443.3627	22.4343	443.3627
0.94	1126.5855	834.3923	440.4125	22.2915	440.4125
0.98	1132.3457	839.2049	437.5823	22.1711	437.5823
1.02	1137.9038	843.8515	434.8668	22.0695	434.8668
1.06	1143.2743	848.3439	432.2501	21.9604	432.2501
1.10	1148.4703	852.6928	429.7333	21.7824	429.7333
1.14	1153.5036	856.9077	427.3082	21.6278	427.3082
1.18	1158.3845	860.9973	424.9726	21.4934	424.9726
1.22	1163.1228	864.9693	422.7091	21.3765	422.7091
1.26	1167.7269	868.8309	420.5239	21.2747	420.5239
1.30	1172.2048	872.5883	418.4111	21.1859	418.4111
1.34	1176.5637	876.2476	416.3655	21.1083	416.3655
1.38	1180.8102	879.8141	414.3829	21.0404	414.3829
1.42	1184.9502	883.2927	412.4601	20.9504	412.4601
1.46	1188.9893	886.688	410.5943	20.8139	410.5943
1.50	1192.9326	890.0042	408.7838	20.6921	408.7838
1.54	1196.785	893.2451	407.026	20.5832	407.026
1.58	1200.5506	896.4144	405.3086	20.4858	405.3086
1.62	1204.2337	899.5153	403.6411	20.3986	403.6411
1.66	1207.838	902.5511	402.0238	20.3204	402.0238
1.70	1211.367	905.5245	400.4371	20.2501	400.4371
1.74	1214.824	908.4383	398.8999	20.1869	398.8999
1.78	1218.2121	911.295	397.393	20.13	397.393
1.82	1221.5341	914.097	395.9307	20.0787	395.9307
1.86	1224.7928	916.8465	394.4936	20.0323	394.4936
1.90	1227.9907	919.5455	393.0996	19.9734	393.0996
1.94	1231.1301	922.1961	391.7302	19.8682	391.7302
1.98	1234.2133	924.8	390.393	19.7719	390.393
2.00	1235.7345	926.085	389.7373	19.7269	389.7373

## Furan Sand properties

**Table C3 - Furan sand thermo-physical properties: (a) Thermal conductivity; (b) Density; (c) Specific heat**

Thermal Conductivity (W/m.K)	Temp (°C)	Density (Kg/m3)	Temp (°C)	Specific heat (J/Kg.K)	Temp (°C)
0.87	50	1500	1	700	1
0.82	100	1495	100	741	25
0.79	150	1493	200	850	99
0.77	200	1490	300	850	100
0.76	250	1487	400	850	101
0.74	300	1478	500	886	127
0.7	400	1476	600	991	227
0.7	450	1470	2000	1082	326
0.7	500			1082	327
0.7	550			1167	427
0.7	600			1248	526
0.7	650			1248	527
0.71	700			1285	573
0.76	750			1160	574
0.79	800			1162	627
0.83	850			1167	727
0.88	900			1172	827
0.93	950			1177	927
0.99	1000			1181	1027
1.05	1050			1186	1127
1.12	1100			1190	1227
1.3	1150			1194	1327
1.45	2000			1250	3000

## Heat transfer properties

**Table C4 - Heat transfer properties: (a) Heat transfer coefficient Mold-Part; (b) Film condition Part-Ambient**

Temp (°C)	k (W/M <sup>2</sup> *K)	Temp (°C)	h (W/M <sup>2</sup> *K)
1	104	1	1.883
1000	104	30	3.19
1200	380	80	5.444
1600	940	180	7.094
2000	940	380	8.589
		580	9.445
		780	10.033
		980	10.439
		1180	10.696
		1380	10.866
		1580	10.886
		1780	11.091
		1980	11.191
		2000	11.2

## Properties for k (T,g)

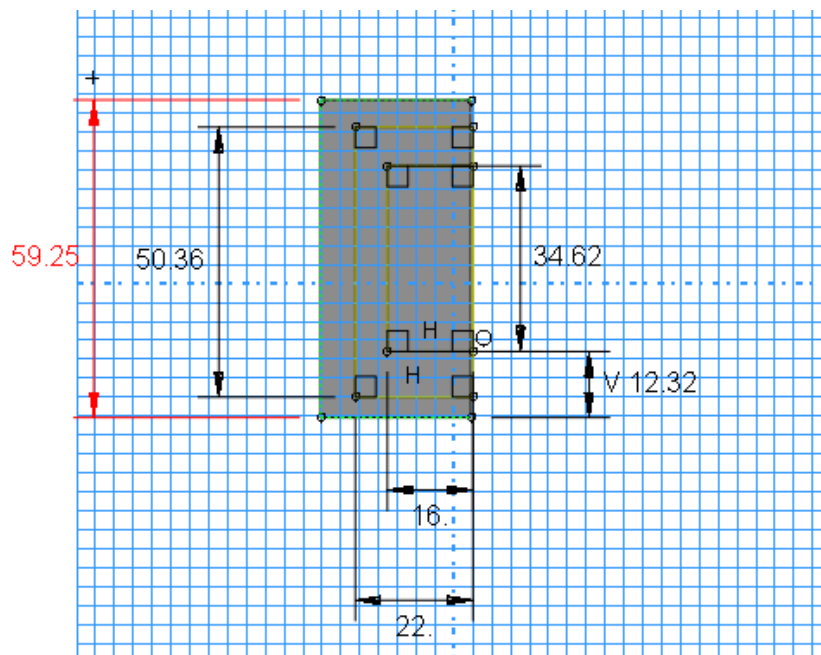
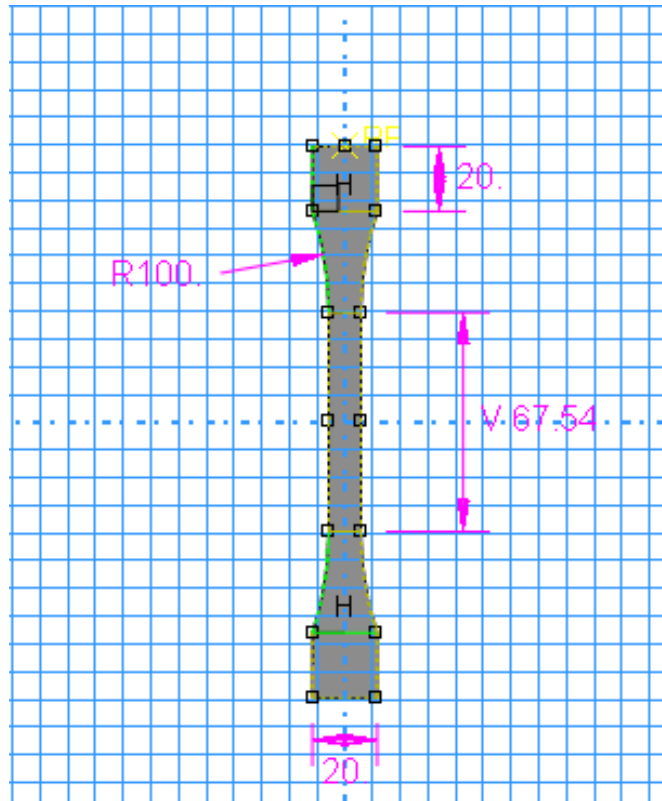
Table C5 - Parameters for the estimation of k (T, g)

Temp (°C)	1	1000	1200	1600	2000
hrad (W/m <sup>2</sup> *K)	4.77E-04	5.79E+01	9.96E+01	2.35E+02	4.58E+02
Air Conductivity (W/m*K)	24	24	24	24	24
Emissivity	0.76	0.76	0.76	0.76	0.76
H(T) (W/m <sup>2</sup> *K)	104	104	380	940	940

Table C6 - Estimation of k (T, g) (Reduced)

	1 °C	1000 °C	1200 °C	1600 °C	2000 °C
g (mm)	H(T,g) (W/m <sup>2</sup> *K)				
0	104.00	104.00	380.00	940.00	940.00
0.2	55.71	65.63	139.19	257.78	357.99
0.4	38.05	55.25	112.41	224.64	334.04
0.6	28.89	50.42	102.11	212.87	325.61
0.8	23.28	47.62	96.66	206.83	321.31
1	19.50	45.80	93.29	203.16	318.70
1.2	16.77	44.52	90.99	200.70	316.95
1.4	14.72	43.58	89.33	198.93	315.69
1.6	13.11	42.84	88.07	197.59	314.74
1.8	11.82	42.26	87.08	196.55	314.01
2	10.76	41.79	86.29	195.72	313.41
2.2	9.87	41.40	85.63	195.03	312.93
2.4	9.12	41.06	85.09	194.46	312.52
2.6	8.48	40.78	84.62	193.98	312.18
2.8	7.92	40.54	84.23	193.56	311.89
3	7.43	40.32	83.88	193.20	311.63
3.2	7.00	40.13	83.57	192.89	311.41
3.4	6.61	39.97	83.31	192.61	311.21
3.6	6.27	39.82	83.07	192.36	311.04
3.8	5.95	39.69	82.85	192.14	310.88
4	5.67	39.56	82.66	191.94	310.74
4.2	5.42	39.45	82.48	191.76	310.61
4.4	5.18	39.35	82.32	191.59	310.49
4.6	4.97	39.26	82.18	191.44	310.39
4.8	4.77	39.18	82.05	191.30	310.29
5	4.59	39.10	81.92	191.18	310.20
5.2	4.42	39.03	81.81	191.06	310.12
5.4	4.26	38.96	81.70	190.95	310.04
5.6	4.12	38.90	81.61	190.85	309.97
5.8	3.98	38.84	81.51	190.76	309.90
6	3.85	38.79	81.43	190.67	309.84

## Appendix B: Rectangle and Tensile specimens dimensions



# Appendix C: MATLAB Scrip

The MATLAB Scrip developed in section 4.3 and used in section 5.2 and 6.3 is shown, in this section.

## 2D Tringle mesh

```
clear; clc; close all;
R = readtable('ReportX.csv', 'HeaderLines',7);
T = readtable('ReportY.csv', 'HeaderLines',7);
U = readtable('ReportZ1.csv', 'HeaderLines',7);
Z = readtable('ReportZ2.csv', 'HeaderLines',7);
V = readtable('node.csv');
% shape fuction
xi=1/3;
et=1/3;
n(1)=1.0-xi-et;
n(2)=xi;
n(3)=et;
h = height(V); % nodes number
w=1/2;
R=table2array(R);
T=table2array(T);
U=table2array(U);
Z=table2array(Z);
D=zeros(3,3);
A=648*864; %area
for i=1:h
    a=table2array(V(i,2));
    b=table2array(V(i,3));
    c=table2array(V(i,4));
    %primeira coluna
    j=abs((R(b,12)-R(a,12))*(R(c,13)-R(a,13)))-((R(c,12)-
R(a,12))*(R(b,13)-R(a,13))); %Jacobiana
    D(1,1)=D(1,1)+((R(a,2)*n(1)+R(b,2)*n(2)+R(c,2)*n(3))*w*j);
    D(2,1)=D(2,1)+((R(a,3)*n(1)+R(b,3)*n(2)+R(c,3)*n(3))*w*j);
    D(3,1)=D(3,1)+((R(a,5)*n(1)+R(b,5)*n(2)+R(c,5)*n(3))*w*j);
    % % %segunda coluna
    D(1,2)=D(1,2)+((T(a,2)*n(1)+T(b,2)*n(2)+T(c,2)*n(3))*w*j);
    D(2,2)=D(2,2)+((T(a,3)*n(1)+T(b,3)*n(2)+T(c,3)*n(3))*w*j);
    D(3,2)=D(3,2)+((T(a,5)*n(1)+T(b,5)*n(2)+T(c,5)*n(3))*w*j);
    % % %terceira coluna
    D(1,3)=D(1,3)+(((U(a,2)*n(1)+U(b,2)*n(2)+U(c,2)*n(3))*w*j)+((Z(a,2)*n(
1)+Z(b,2)*n(2)+Z(c,2)*n(3))*w*j))/2;
    D(2,3)=D(2,3)+(((U(a,3)*n(1)+U(b,3)*n(2)+U(c,3)*n(3))*w*j)+((Z(a,3)*n(
1)+Z(b,3)*n(2)+Z(c,3)*n(3))*w*j))/2;
    D(3,3)=D(3,3)+(((U(a,5)*n(1)+U(b,5)*n(2)+U(c,5)*n(3))*w*j)+((Z(a,5)*n(
1)+Z(b,5)*n(2)+Z(c,5)*n(3))*w*j))/2;
end
M=D/A;
```

## 2D Quadrangular mesh

```
clear; clc; close all;
R = readtable('Reportx.csv', 'HeaderLines',7);
```

```

T = readtable('Reporty.csv', 'HeaderLines',7);
U = readtable('Reportz1.csv', 'HeaderLines',7);
Z = readtable('Reportz2.csv', 'HeaderLines',7);
V = readtable('node.csv');
% shape fuction
xi=0;
et=0;
n(1)=1/4*(1-xi)*(1-et);
n(2)=1/4*(1-xi)*(1+et);
n(3)=1/4*(1+xi)*(1+et);
n(4)=1/4*(1+xi)*(1-et);
h = height(V); % nodes number
w=2*2;
R=table2array(R);
T=table2array(T);
U=table2array(U);
Z=table2array(Z);
D=zeros(3,3);
A=648*864; %area
for i=1:h
    a=table2array(V(i,2));
    b=table2array(V(i,3));
    c=table2array(V(i,4));
    d=table2array(V(i,5));
    %primeira coluna
    J=1/4*[-R(a,12)+R(b,12)+R(c,12)-R(d,12) -R(a,13)+R(b,13)+R(c,13)-
R(d,13);-R(a,12)-R(b,12)+R(c,12)+R(d,12) -R(a,13)-
R(b,13)+R(c,13)+R(d,13)];
    % J=1/4*[-R(a,12)+R(b,12)+R(c,12)-R(d,12) -R(a,12)-
R(b,12)+R(c,12)+R(d,12);-R(a,13)+R(b,13)+R(c,13)-R(d,13) -R(a,13)-
R(b,13)+R(c,13)+R(d,13)];
    j=abs(det(J)); %Jacobiana

D(1,1)=D(1,1)+((R(a,2)*n(1)+R(b,2)*n(2)+R(c,2)*n(3)+R(d,2)*n(4))*w*j);

D(2,1)=D(2,1)+((R(a,3)*n(1)+R(b,3)*n(2)+R(c,3)*n(3)+R(d,3)*n(4))*w*j);

D(3,1)=D(3,1)+((R(a,5)*n(1)+R(b,5)*n(2)+R(c,5)*n(3)+R(d,5)*n(4))*w*j);
% % % %segunda coluna

D(1,2)=D(1,2)+((T(a,2)*n(1)+T(b,2)*n(2)+T(c,2)*n(3)+T(d,2)*n(4))*w*j);

D(2,2)=D(2,2)+((T(a,3)*n(1)+T(b,3)*n(2)+T(c,3)*n(3)+T(d,3)*n(4))*w*j);

D(3,2)=D(3,2)+((T(a,5)*n(1)+T(b,5)*n(2)+T(c,5)*n(3)+R(d,5)*n(4))*w*j);
% % % %terceira coluna

D(1,3)=D(1,3)+(((U(a,2)*n(1)+U(b,2)*n(2)+U(c,2)*n(3)+U(d,2)*n(4))*w*j)
+(Z(a,2)*n(1)+Z(b,2)*n(2)+Z(c,2)*n(3)+Z(d,2)*n(4))*w*j)/2;

D(2,3)=D(2,3)+(((U(a,3)*n(1)+U(b,3)*n(2)+U(c,3)*n(3)+U(d,3)*n(4))*w*j)
+(Z(a,3)*n(1)+Z(b,3)*n(2)+Z(c,3)*n(3)+Z(d,3)*n(4))*w*j)/2;

D(3,3)=D(3,3)+(((U(a,5)*n(1)+U(b,5)*n(2)+U(c,5)*n(3)+U(d,5)*n(4))*w*j)
+(Z(a,5)*n(1)+Z(b,5)*n(2)+Z(c,5)*n(3)+Z(d,5)*n(4))*w*j)/2;
end
M=D/A;

```

### 3D Tetraedron mesh

```

clear; clc; close all;
R = readtable('Reportx.csv', 'HeaderLines',7);
T = readtable('Reporty.csv', 'HeaderLines',7);
U = readtable('Reportz.csv', 'HeaderLines',7);
Z = readtable('Reportxy1.csv', 'HeaderLines',7);
E = readtable('Reportyz1.csv', 'HeaderLines',7);
F = readtable('Reportxz1.csv', 'HeaderLines',7);
G = readtable('Reportxz2.csv', 'HeaderLines',7);
H = readtable('Reportxz2.csv', 'HeaderLines',7);
J = readtable('Reportxz2.csv', 'HeaderLines',7);
V = readtable('node.csv');
shape fuction
xi1=0.25;
xi2=0.25;
xi3=0.25;
%função de forma elemento tetraedral
n(1)=xi1;
n(2)=xi2;
n(3)=xi3;
n(4)=1-xi1-xi2-xi3;
h = height(V); % nodes number
w=1*1/6;
R=table2array(R);
T=table2array(T);
U=table2array(U);
Z=table2array(Z);
E=table2array(E);
F=table2array(F);
D=zeros(6,6);
for i=1:h
    a=table2array(V(i,2));
    b=table2array(V(i,3));
    c=table2array(V(i,4));
    d=table2array(V(i,5));
V=648*864; %volume

    primeira coluna
    %determinante da jacobiana
    %x1*y2*z3 - x1*y3*z2 - x2*y1*z3 + x2*y3*z1 + x3*y1*z2 - x3*y2*z1 -
    %x1*y2*z4 + x1*y4*z2 + x2*y1*z4
    %- x2*y4*z1 - x4*y1*z2 + x4*y2*z1 + x1*y3*z4 - x1*y4*z3 - x3*y1*z4 +
    %x3*y4*z1 + x4*y1*z3 - x4*y3*z1
    %- x2*y3*z4 + x2*y4*z3 +
    %x3*y2*z4 - x3*y4*z2 - x4*y2*z3 + x4*y3*z2

    %Jacobiana
    J=[R(a,16)-R(d,16) R(a,17)-R(d,17) R(a,18)-R(d,18);R(b,16)-R(d,16)
R(b,17)-R(d,17) R(b,18)-R(d,18);R(c,16)-R(d,16) R(c,17)-R(d,17)
R(c,18)-R(d,18)]
J=[R(b,16)-R(a,16) R(b,17)-R(a,17) R(b,18)-R(a,18);R(c,16)-R(a,16)
R(c,17)-R(a,17) R(c,18)-R(a,18);R(d,16)-R(a,16) R(d,17)-R(a,17)
R(d,18)-R(a,18)];
j(i)=abs(det(J));
    j(i)=abs(R(a,16)*R(b,17)*R(c,18)-R(a,16)*R(c,17)*R(b,18)-
R(b,16)*R(a,17)*R(c,18)+R(b,16)*R(c,17)*R(a,18)+...
R(c,16)*R(a,17)*R(b,18)-R(c,16)*R(b,17)*R(a,18)-
R(a,16)*R(b,17)*R(d,18)+R(a,16)*R(d,17)*R(b,18)+...
R(b,16)*R(a,17)*R(d,18)-R(b,16)*R(d,17)*R(a,18)-
R(d,16)*R(a,17)*R(b,18)+R(d,16)*R(b,17)*R(a,18)+...

```



```

R(a,16)*R(c,17)*R(d,18)-R(a,16)*R(d,17)*R(c,18)-
R(c,16)*R(a,17)*R(d,18)+R(c,16)*R(d,17)*R(a,18)+...
R(d,16)*R(a,17)*R(c,18)-R(d,16)*R(c,17)*R(a,18)-
R(b,16)*R(c,17)*R(d,18)+R(b,16)*R(d,17)*R(c,18)+...
R(c,16)*R(b,17)*R(d,18)-R(c,16)*R(d,17)*R(b,18)-
R(d,16)*R(b,17)*R(c,18)+R(d,16)*R(c,17)*R(b,18));

%primeira coluna

D(1,1)=D(1,1)+((R(a,2)*n(1)+R(b,2)*n(2)+R(c,2)*n(3)+R(d,2)*n(4))*w*j(i));
D(2,1)=D(2,1)+((R(a,3)*n(1)+R(b,3)*n(2)+R(c,3)*n(3)+R(d,3)*n(4))*w*j(i));
D(3,1)=D(3,1)+((R(a,4)*n(1)+R(b,4)*n(2)+R(c,4)*n(3)+R(d,4)*n(4))*w*j(i));
D(4,1)=D(4,1)+((R(a,5)*n(1)+R(b,5)*n(2)+R(c,5)*n(3)+R(d,5)*n(4))*w*j(i));
D(5,1)=D(5,1)+((R(a,6)*n(1)+R(b,6)*n(2)+R(c,6)*n(3)+R(d,6)*n(4))*w*j(i));
D(6,1)=D(6,1)+((R(a,7)*n(1)+R(b,7)*n(2)+R(c,7)*n(3)+R(d,7)*n(4))*w*j(i));

%segunda coluna

D(1,2)=D(1,2)+((T(a,2)*n(1)+T(b,2)*n(2)+T(c,2)*n(3)+T(d,2)*n(4))*w*j(i));
D(2,2)=D(2,2)+((T(a,3)*n(1)+T(b,3)*n(2)+T(c,3)*n(3)+T(d,3)*n(4))*w*j(i));
D(3,2)=D(3,2)+((T(a,4)*n(1)+T(b,4)*n(2)+T(c,4)*n(3)+T(d,4)*n(4))*w*j(i));
D(4,2)=D(4,2)+((T(a,5)*n(1)+T(b,5)*n(2)+T(c,5)*n(3)+T(d,5)*n(4))*w*j(i));
D(5,2)=D(5,2)+((T(a,6)*n(1)+T(b,6)*n(2)+T(c,6)*n(3)+T(d,6)*n(4))*w*j(i));
D(6,2)=D(6,2)+((T(a,7)*n(1)+T(b,7)*n(2)+T(c,7)*n(3)+T(d,7)*n(4))*w*j(i));

%terceira coluna

D(1,3)=D(1,3)+((U(a,2)*n(1)+U(b,2)*n(2)+U(c,2)*n(3)+U(d,2)*n(4))*w*j(i));
D(2,3)=D(2,3)+((U(a,3)*n(1)+U(b,3)*n(2)+U(c,3)*n(3)+U(d,3)*n(4))*w*j(i));
D(3,3)=D(3,3)+((U(a,4)*n(1)+U(b,4)*n(2)+U(c,4)*n(3)+U(d,4)*n(4))*w*j(i));
D(4,3)=D(4,3)+((U(a,5)*n(1)+U(b,5)*n(2)+U(c,5)*n(3)+U(d,5)*n(4))*w*j(i));
D(5,3)=D(5,3)+((U(a,6)*n(1)+U(b,6)*n(2)+U(c,6)*n(3)+U(d,6)*n(4))*w*j(i));
D(6,3)=D(6,3)+((U(a,7)*n(1)+U(b,7)*n(2)+U(c,7)*n(3)+U(d,7)*n(4))*w*j(i));

%Quarta coluna

D(1,4)=D(1,4)+((Z(a,2)*n(1)+Z(b,2)*n(2)+Z(c,2)*n(3)+Z(d,2)*n(4))*w*j(i))+((G(a,2)*n(1)+G(b,2)*n(2)+G(c,2)*n(3)+G(d,2)*n(4))*w*j(i));
D(2,4)=D(2,4)+((Z(a,3)*n(1)+Z(b,3)*n(2)+Z(c,3)*n(3)+Z(d,3)*n(4))*w*j(i))+((G(a,3)*n(1)+G(b,3)*n(2)+G(c,3)*n(3)+G(d,3)*n(4))*w*j(i));
D(3,4)=D(3,4)+((Z(a,4)*n(1)+Z(b,4)*n(2)+Z(c,4)*n(3)+Z(d,4)*n(4))*w*j(i))+((G(a,4)*n(1)+G(b,4)*n(2)+G(c,4)*n(3)+G(d,4)*n(4))*w*j(i));
D(4,4)=D(4,4)+((Z(a,5)*n(1)+Z(b,5)*n(2)+Z(c,5)*n(3)+Z(d,5)*n(4))*w*j(i))+((G(a,5)*n(1)+G(b,5)*n(2)+G(c,5)*n(3)+G(d,5)*n(4))*w*j(i));
D(5,4)=D(5,4)+((Z(a,6)*n(1)+Z(b,6)*n(2)+Z(c,6)*n(3)+Z(d,6)*n(4))*w*j(i))+((G(a,6)*n(1)+G(b,6)*n(2)+G(c,6)*n(3)+G(d,6)*n(4))*w*j(i));

```

```

D(6,4)=D(6,4)+(Z(a,7)*n(1)+Z(b,7)*n(2)+Z(c,7)*n(3)+Z(d,7)*n(4))*w*j(i)
)+(G(a,7)*n(1)+G(b,7)*n(2)+G(c,7)*n(3)+G(d,7)*n(4))*w*j(i));
% Quinta columna

D(1,5)=D(1,5)+(E(a,2)*n(1)+E(b,2)*n(2)+E(c,2)*n(3)+E(d,2)*n(4))*w*j(i)
)+(H(a,2)*n(1)+H(b,2)*n(2)+H(c,2)*n(3)+H(d,2)*n(4))*w*j(i));
D(2,5)=D(2,5)+(E(a,3)*n(1)+E(b,3)*n(2)+E(c,3)*n(3)+E(d,3)*n(4))*w*j(i)
)+(H(a,3)*n(1)+H(b,3)*n(2)+H(c,3)*n(3)+H(d,2)*n(4))*w*j(i));
D(3,5)=D(3,5)+(E(a,4)*n(1)+E(b,4)*n(2)+E(c,4)*n(3)+E(d,4)*n(4))*w*j(i)
)+(H(a,4)*n(1)+H(b,4)*n(2)+H(c,4)*n(3)+H(d,2)*n(4))*w*j(i));
D(4,5)=D(4,5)+(E(a,5)*n(1)+E(b,5)*n(2)+E(c,5)*n(3)+E(d,5)*n(4))*w*j(i)
)+(H(a,5)*n(1)+H(b,5)*n(2)+H(c,5)*n(3)+H(d,2)*n(4))*w*j(i));
D(5,5)=D(5,5)+(E(a,6)*n(1)+E(b,6)*n(2)+E(c,6)*n(3)+E(d,6)*n(4))*w*j(i)
)+(H(a,6)*n(1)+H(b,6)*n(2)+H(c,6)*n(3)+H(d,2)*n(4))*w*j(i));
D(6,5)=D(6,5)+(E(a,7)*n(1)+E(b,7)*n(2)+E(c,7)*n(3)+E(d,7)*n(4))*w*j(i)
)+(H(a,7)*n(1)+H(b,7)*n(2)+H(c,7)*n(3)+H(d,2)*n(4))*w*j(i));
%Sexta columna

D(1,6)=D(1,6)+(F(a,2)*n(1)+F(b,2)*n(2)+F(c,2)*n(3)+F(d,2)*n(4))*w*j(i)
)+(J(a,2)*n(1)+J(b,2)*n(2)+J(c,2)*n(3)+J(d,2)*n(4))*w*j(i));
D(2,6)=D(2,6)+(F(a,3)*n(1)+F(b,3)*n(2)+F(c,3)*n(3)+F(d,3)*n(4))*w*j(i)
)+(J(a,3)*n(1)+J(b,3)*n(2)+J(c,3)*n(3)+J(d,3)*n(4))*w*j(i));
D(3,6)=D(3,6)+(F(a,4)*n(1)+F(b,4)*n(2)+F(c,4)*n(3)+F(d,4)*n(4))*w*j(i)
)+(J(a,4)*n(1)+J(b,4)*n(2)+J(c,4)*n(3)+J(d,4)*n(4))*w*j(i));
D(4,6)=D(4,6)+(F(a,5)*n(1)+F(b,5)*n(2)+F(c,5)*n(3)+F(d,5)*n(4))*w*j(i)
)+(J(a,5)*n(1)+J(b,5)*n(2)+J(c,5)*n(3)+J(d,5)*n(4))*w*j(i));
D(5,6)=D(5,6)+(F(a,6)*n(1)+F(b,6)*n(2)+F(c,6)*n(3)+F(d,6)*n(4))*w*j(i)
)+(J(a,6)*n(1)+J(b,6)*n(2)+J(c,6)*n(3)+J(d,6)*n(4))*w*j(i));
D(6,6)=D(6,6)+(F(a,7)*n(1)+F(b,7)*n(2)+F(c,7)*n(3)+F(d,7)*n(4))*w*j(i)
)+(J(a,7)*n(1)+J(b,7)*n(2)+J(c,7)*n(3)+J(d,7)*n(4))*w*j(i));
end
M=D/V;

```

# Appendix D: Python Scrip

## Script python to get a strain and stress 3D

```
# -*- coding: utf-8 -*-
#
# Cria ficheiro REPORT dos valores do Tensor
# (e.g. Tensoes ou Extensoes)
# a partir de UM .odb ESCOLHIDO
#
# ADS - Maio 2013 - FEUP/INEGI
# RLA - Junho 2017 - FEUP/INEGI
# APC - Dezembro 2019 - FEUP/INEGI
#
# usage: > abaqus python filename.py
# Requirements: - odb file
#             - nodes to output information
#
#####
#####
from abaqusConstants import *
from odbAccess import *
import odbAccess
import os, glob

def
getNodeResults(nnode,odbFile,instanceName,if
r,outFile):
    assembly = odbFile.rootAssembly
    ##### OBTER DADOS DE NODE
    # Considerar node, retirar os dados to tensor
    (stress/strain) e numero de elementos
    # ligados ao node para fazer media
    # nodeLabel = [1]
    nodeLabel=nnode
    #nodeSet =
assembly.NodeSetFromNodeLabels(name='my
NSet',
nodeLabels=((instanceName,nodeLabel),))
    nodeSet =
assembly.instances[instanceName].getNodeFro
mLabel(nodeLabel)
    # Considerar dados de stress e strain
    stressField = odbFile.steps.values()[-
1].frames[ifr].fieldOutputs['S'] # S PE LE
PEEQ
    strainField = odbFile.steps.values()[-
1].frames[ifr].fieldOutputs['E']
    peeqField = odbFile.steps.values()[-
1].frames[ifr].fieldOutputs['PEEQ']
    #
    stressSubField =
stressField.getSubset(region=nodeSet,position=
ELEMENT_NODAL)
    strainSubField =
strainField.getSubset(region=nodeSet,position=
ELEMENT_NODAL)
    peeqSubField =
peeqField.getSubset(region=nodeSet,position=E
LEMENT_NODAL)
    str1,str2,str3,str12,strMises,e1,e2,e3,e12,peeq =
0, 0, 0, 0, 0, 0, 0, 0,0
    # print stressSubField.__members__, '\n
__members__'
    # print stressSubField.componentLabels, '\n
componentLabels'
    for val in stressSubField.values:
        str1=str1+val.data[0]
        str2=str2+val.data[1]
        str3=str3+val.data[2]
        str12=str12+val.data[3]
        strMises=strMises+val.mises
    for val in strainSubField.values:
        e1=e1+val.data[0]
        e2=e2+val.data[1]
        e3=e3+val.data[2]
        e12=e12+val.data[3]
    for val in peeqSubField.values:
        peeq=peeq+val.data
    numElements=len(stressSubField.values)
    str1=str1/numElements
    str2=str2/numElements
```

```

str3=str3/numElements
str12=str12/numElements
strMises=strMises/numElements
e1=e1/numElements
e2=e2/numElements
e3=e3/numElements
e12=e12/numElements
peeq=peeq/numElements

outFile.write("\n %5s, %15s,%15s, %15s,
%15s, %15s, %20s,%20s, %20s, %20s, %20s,
    %(val.nodeLabel, str1,str2,str3,str12,
strMises, e1,e2,e3,e12,peeq))

# Considerar dados de Coordenadas
coordEnd = odbFile.steps.values()[-
1].frames[ifr].fieldOutputs['COORD']
coordStart = odbFile.steps.values()[-
1].frames[0].fieldOutputs['COORD']
coordEndSub=coordEnd.getSubset(region=nod
eSet)
coordStartSub=coordStart.getSubset(region=no
deSet)

for val in coordStartSub.values:
#   print 'Coords
inicio:',val.data[0],val.data[1]

    outFile.write('%20s, %20s,'
%(val.data[0],val.data[1]))

for val in coordEndSub.values:
#   print 'Coords
finais:',val.data[0],val.data[1]

    outFile.write('%20s, %20s'
%(val.data[0],val.data[1]))

#
disp1=val.data[0]
return disp1;
#
#####
#####
# INICIO DO SCRIPT
#####
#####
#
# Adaptar path e filename em conformidade
tDr='C:\Abaqusm\mnt1'
os.chdir(tDr)

```

```

#abaqus python GetStressAllNodesv8.py
#fileName='\MSmatV342t30P10s.odb'
NewOdbFiles=glob.glob('*.*odb') # this will list
all the odb files in the folder
print NewOdbFiles
for file in NewOdbFiles:
    #print 'INPUT FILE = %s' %file
#   fileNameinp=NewInputFiles[0]
        fileNameodb=file
        lastcharacter=len(fileNameodb)
        fileName=fileNameodb[0:lastcharacter]
        print '\n FICHEIRO', fileName
#fileName='\MSmatV230t50P30.odb'
        odbPath=tDr+'/' +fileName
        odbFile=odbAccess.openOdb(path=od
bPath,readOnly=True)
        outFilename='Report'+fileName[0:-
4]+'*.csv'
        outFile = open(outFilename, "w")
        stepname=odbFile.steps.keys()[-1]
# step 'Step-Velocity'
        nframes=len(odbFile.steps[stepname].f
rames)
        instanceName = 'PART-1-1'
        assembly = odbFile.rootAssembly
        outFile.write('* RESULTADOS DE
TENSOES,EXTENSOES, COORDENADAS
DE NODESET ,,,,,,,,,,')
        outFile.write("\n Caso em estudo
,,,,,,,,,')
        outFile.write("\n %5s ,,,,,,,,,,'
%(fileName[1:-5]))
        outFile.write("\n N.Frames ,,,,,,,,,,')
        outFile.write("\n %5s ,,,,,,,,,,'
%(nframes))
        for ifr in range(1,nframes):
            outFile.write("\n*
Frame='+str(ifr)+' ,,,,,,,,,,')
            outFile.write("\n %5s,
%15s,%15s, %15s, %15s,
%15s,%20s,%20s,%20s,%20s,%20s,%20s,%20s,%20s,%20s '
            %('Node','S1','S2','S3','S12','Mises','e1',
'e2','e3','e12','Peeq','XXini','YYini','XXfin','YYfi
n'))

```

```

        print ' n.frame:',ifr,
'/total:',nframes
    #
    nTotNodes=len(assembly.instances[instanceName].nodes)
        for inode in
range(nTotNodes):
            numno=assembly.instances[instanceName].nodes[inode].label
                #if numno > 10:
                    # break
                disp=getNodeResults(numno,odbFile,instanceName,ifr,outFile)
                odbFile.close()
exit()

#
# DEFINIR COORDENADAS LOCAIS DAS
# TENSOES OU EXTENSOES
#
# ver Abaqus Scripting Reference
# 33.14.2 DatumCsysByThreeNodes
# 33.14.1 DatumCsysByThreePoints - ver
http://www.english-tips.com/viewthread.cfm?qid=341824

```

### Script python to get a strain and stress 3D

```

# -*- coding: utf-8 -*-
#
# Cria ficheiro REPORT dos valores do Tensor
# (e.g. Tensoes ou Extensoes) Para arquivos em
# 3D
#
# a partir de UM .odb ESCOLHIDO
#
# ADS - Maio 2013 - FEUP/INEGI
# RLA - Junho 2017 - FEUP/INEGI
# APC - Dezembro 2019 - FEUP/INEGI
#
# usage: > abaqus python filename.py
# Requirements: - odb file
#
# - nodes to output information
#
#####
#####

```

```

#
coordSys =
odb.rootAssembly.DatumCsysByThreePoints(n
ame='CSYS',
    coordSysType=CARTESIAN, origin=(0,0,0),
    point1=(-1.0, 1.0, 0), point2=(0.0, 1.0, 0.0) )
#    point1=(1.0, 0.0, 0), point2=(0.0, 1.0, 0.0)
)
#    coordenadas Globais
stressField = framep.fieldOutputs['S'] # S PE
LE PEEQ
stressSubField =
stressField.getSubset(region=nodeSet,position=
ELEMENT_NODAL)
stressField = framep.fieldOutputs['S'] # S PE
LE PEEQ
stressSubField =
stressField.getSubset(region=nodeSet,position=
ELEMENT_NODAL)
#    coordenadas TRANSFORMADAS
stressSubFieldT =
stressField.getSubset(region=nodeSet,
position=ELEMENT_NODAL).getTransforme
dField(datumCsys=coordSys)

```

```

from abaqusConstants import *
from odbAccess import *
import odbAccess
import os, glob
def
getNodeResults(nnode,odbFile,instanceName,if
r,outFile):
    assembly = odbFile.rootAssembly
    ##### OBTER DADOS DE NODE
    # Considerar node, retirar os dados to tensor
(stress/strain) e numero de elementos
    # ligados ao node para fazer media
    # nodeLabel = [1]
    nodeLabel=nnode
    #nodeSet =
assembly.NodeSetFromNodeLabels(name='my
NSet',
nodeLabels=((instanceName,nodeLabel),))

```

```

nodeSet =
assembly.instances[instanceName].getNodeFromLabel(nodeLabel)

# Considerar dados de stress e strain
stressField = odbFile.steps.values()[-1].frames[ifr].fieldOutputs['S'] # S PE LE PEEQ
strainField = odbFile.steps.values()[-1].frames[ifr].fieldOutputs['E']
peeqField = odbFile.steps.values()[-1].frames[ifr].fieldOutputs['PEEQ']
#
stressSubField =
stressField.getSubset(region=nodeSet,position=ELEMENT_NODAL)
strainSubField =
strainField.getSubset(region=nodeSet,position=ELEMENT_NODAL)
peeqSubField =
peeqField.getSubset(region=nodeSet,position=ELEMENT_NODAL)
str1,str2,str3,str12,str13,str23,strMises,e1,e2,e3,
e12,e13,e23,peeq = 0, 0, 0, 0, 0, 0, 0,
0,0,0,0,0,0
# print stressSubField.__members__, '\n __members__'
# print stressSubField.componentLabels, '\n componentLabels'
for val in stressSubField.values:
str1=str1+val.data[0]
str2=str2+val.data[1]
str3=str3+val.data[2]
str12=str12+val.data[3]
str13=str13+val.data[4]
str23=str23+val.data[5]
strMises=strMises+val.mises
for val in strainSubField.values:
e1=e1+val.data[0]
e2=e2+val.data[1]
e3=e3+val.data[2]
e12=e12+val.data[3]
e13=e13+val.data[4]
e23=e23+val.data[5]
for val in peeqSubField.values:
peeq=peeq+val.data

numElements=len(stressSubField.values)
str1=str1/numElements
str2=str2/numElements
str3=str3/numElements
str12=str12/numElements
str13=str13/numElements
str23=str23/numElements
strMises=strMises/numElements
e1=e1/numElements
e2=e2/numElements
e3=e3/numElements
e12=e12/numElements
e23=e23/numElements

peeq=peeq/numElements
outFile.write("\n %6s, %15s,%15s, %15s,
%15s, %15s, %15s, %15s, %20s,%20s, %20s,
%20s, %20s,%20s,%20s,'
%(val.nodeLabel,
str1,str2,str3,str12,str13,str23,strMises,
e1,e2,e3,e12,e13,e23,peeq))
# Considerar dados de Coordenadas
coordEnd = odbFile.steps.values()[-1].frames[ifr].fieldOutputs['COORD']
coordStart = odbFile.steps.values()[-1].frames[0].fieldOutputs['COORD']
coordEndSub=coordEnd.getSubset(region=nodeSet)
coordStartSub=coordStart.getSubset(region=nodeSet)
for val in coordStartSub.values:
# print 'Coords
inicio:',val.data[0],val.data[1],val.data[2]
outFile.write('%20s, %20s, %20s,'
%(val.data[0],val.data[1],val.data[2]))
for val in coordEndSub.values:
# print 'Coords
finais:',val.data[0],val.data[1],val.data[2]
outFile.write('%20s, %20s, %20s,'
%(val.data[0],val.data[1],val.data[2]))
#
disp1=val.data[0]
return disp1;
#

```

```

#####
#####
# INICIO DO SCRIPT
#####
#####
#
# Adaptar path e filename em conformidade
tDr='C:\Abaqusm2\d3'
os.chdir(tDr)
#abaqus python GetStressAllNodesv8.py
#fileName='\MSmatV342t30P10s.odb'
NewOdbFiles=glob.glob('*.*odb') # this will list
all the odb files in the folder
print NewOdbFiles
for file in NewOdbFiles:
    #print 'INPUT FILE = %s' % file
# fileNameinp=NewInputFiles[0]
    fileNameodb=file
    lastcharacter=len(fileNameodb)
    fileName=fileNameodb[0:lastcharacter]
    print '\n FICHEIRO', fileName
#fileName='\MSmatV230t50P30.odb'

    odbPath=tDr+'/'+fileName
    odbFile=odbAccess.openOdb(path=odbPath,readOnly=True)
    outFilename='Report'+fileName[0:-4]+'*.csv'
    outFile = open(outFilename, "w")
    stepname=odbFile.steps.keys()[-1]
# step 'Step-Velocity'
    nframes=len(odbFile.steps[stepname].frames)
    instanceName = 'PART-1-1'
    assembly = odbFile.rootAssembly

    outFile.write('* RESULTADOS DE
TENSOES,EXTENSOES, COORDENADAS
DE NODESET ,,,,,,,,,,')
    outFile.write('\n Caso em estudo
,,,,,,,,,')
    outFile.write('\n %5s ,,,,,,,,,,'
%(fileName[1:-5]))
    outFile.write('\n N.Frames ,,,,,,,,,,')

    outFile.write('\n %5s ,,,,,,,,,,'
%(nframes))
    for ifr in range(1,nframes):
        outFile.write('\n*
Frame='+str(ifr)+' ,,,,,,,,,,')
        outFile.write('\n %5s,
% 15s,% 15s, % 15s, % 15s, % 15s, % 15s,
% 15s,% 20s,% 20s,% 20s,% 20s,% 20s,% 20s,% 20s,% 20s,% 20s'
        %('Node','S1','S2','S3','S12','s13','s23','
Mises','e1','e2','e3','e12','e13','e23','Peeq','XXini',
'YYini','ZZini','XXfin','YYfin','ZZfin'))
        print ' n.frame:',ifr,
'/total:',nframes
        #
        nTotNodes=len(assembly.instances[instanceName].nodes)
        for inode in
range(nTotNodes):
            numno=assembly.instances[instanceName].nodes[inode].label
            #if numno > 10:
            # break
            disp=getNodeResults(numno,odbFile,instanceName,ifr,outFile)
            odbFile.close()

    exit()
#
# DEFINIR COORDENADAS LOCAIS DAS
TENSOES OU EXTENSOES
#
# ver Abaqus Scripting Reference
# 33.14.2 DatumCsysByThreeNodes
# 33.14.1 DatumCsysByThreePoints - ver
http://www.eng-tips.com/viewthread.cfm?qid=341824
#
coordSys =
odb.rootAssembly.DatumCsysByThreePoints(name='CSYS',
    coordSysType=CARTESIAN, origin=(0,0,0),
    point1=(-1.0, 1.0, 0), point2=(0.0, 1.0, 0.0) )
# point1=(1.0, 0.0, 0), point2=(0.0, 1.0, 0.0)
# coordenadas Globais
stressField = framep.fieldOutputs['S'] # S PE
LE PEEQ

```

```
stressSubField =  
stressField.getSubset(region=nodeSet,position=  
ELEMENT_NODAL)  
  
stressField = framep.fieldOutputs['S'] # S PE  
LE PEEQ  
  
stressSubField =  
stressField.getSubset(region=nodeSet,position=  
ELEMENT_NODAL)  
  
#   coordenadas TRANSFORMADAS  
  
stressSubFieldT =  
stressField.getSubset(region=nodeSet,  
  
position=ELEMENT_NODAL).getTransforme  
dField(datumCsys=coordSys)  
  
*Output, history, variable=PRESELECT  
*End Step
```



# Appendix E: Abaqus file

In this next section, the input files generated by the Abaqus for the analysis developed in section 4.7.4 is shown. Due to the number of nodes contained in the mesh of the model and because there are temperature-dependent properties with a large amount of data, it was decided not to show the matrices generated by the script. Hence, each time the input files present a matrix, "[Matrix not shown]" is displayed.

## The cooling rate of 6.6 Cmin.

```
*HEADING
ABAQUS-style file created by OOF2 on
2019-12-23 12:50:04.689182 from a mesh
of the microstructure Cooling rate 6.6
Cmin.tif.
** Materials defined by OOF2:
** Ferrite:
** austenite:
** sigma:
** Master elements used in OOF2:
** 2: D2_2, Isoparametric 2-noded edge
element
** 3: T3_3, Isoparametric 3 noded triangle
with linear interpolation for both fields and
positions.
** 4: Q4_4, Isoparametric 4 noded
quadrilateral with bilinear interpolation for
both positions and fields
** Boundary Conditions:
** Notes:
** The set of nodes and elements may be
different from the set
** created from a skeleton depending on
the element type and if the
** mesh was refined.
** The materials and boundary conditions
provided by OOF2 may be
** translated into ABAQUS by the user.
** The element type provided below should
be verified and modified
** accordingly.
** Only elements (and nodes of such
elements) that have an associated
** material are included in this file.
*NODE
[matrix not shown]
** The OOF2 element type is T3_3. The
type provided for ABAQUS is only a guess
** and may have to be modified by the user
to be meaningful.
*ELEMENT, TYPE=CPS3
[matrix not

** Point boundaries in OOF2
*NSET, NSET=topleft
3547
*NSET, NSET=bottomleft
1
*NSET, NSET=topright
3717

*NSET, NSET=bottomright
244
** Edge boundaries in OOF2
*NSET, NSET=top
[matrix not shown]
*NSET, NSET=right
[matrix not shown]
*NSET, NSET=left
[matrix not shown]
*NSET, NSET=bottom
[matrix not shown]
*ELSET, ELSET=Ferrite
[matrix not shown]
*SOLID SECTION, ELSET=Ferrite,
MATERIAL=Ferrite
*ELSET, ELSET=austenite
[matrix not shown]
*SOLID SECTION, ELSET=austenite,
MATERIAL=austenite
*ELSET, ELSET=sigma
[matrix not shown]
*SOLID SECTION, ELSET=sigma,
MATERIAL=sigma
*MATERIAL, NAME=Ferrite
** Use the information in the header to
complete these fields under MATERIAL
*MATERIAL, NAME=austenite
** Use the information in the header to
complete these fields under MATERIAL
*MATERIAL, NAME=sigma
** Use the information in the header to
complete these fields under MATERIAL
```

**2D thermal model input**

```

*Heading
** Job name: t-6 Model name: 1ThermalModel
** Generated by: Abaqus/CAE 6.13-1
** Preprint, echo=NO, model=NO, history=NO,
contact=NO
**
** PARTS
**
*Part, name=Mold
*Node
[matrix not shown]
*Element, type=CPE3T
[matrix not shown]
*Nset, nset=Set-1, generate
1, 176, 1
*Elset, elset=Set-1, generate
1, 283, 1
** Section: Section-1-SET-4
*Solid Section, elset=Set-1, material=ARENA
,
*End Part
**
*Part, name=Part
*Node
[matrix not shown]
*Element, type=CPE3T
[matrix not shown]
*Nset, nset=Set-1, generate
1, 272, 1
*Elset, elset=Set-1, generate
1, 463, 1
*Nset, nset=Set-9, generate
1, 272, 1
*Elset, elset=Set-9, generate
1, 463, 1
** Section: Section-2
*Solid Section, elset=Set-9, material=STEEL
,
*End Part
**
** ASSEMBLY
**
*Assembly, name=Assembly
**
*Instance, name=Part-1, part=Part
[matrix not shown]
*End Instance
**
*Instance, name=Mold-1, part=Mold
[matrix not shown]
*End Instance
**
*Nset, nset=Set-16, instance=Mold-1
3, 4, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
51, 52, 53
*Elset, elset=Set-16, instance=Mold-1
35, 49, 122, 150, 156, 157, 158, 160, 161,
163, 164, 165, 166, 168, 175
*Nset, nset=Set-17, instance=Mold-1
4, 5, 54, 55, 56, 57, 58, 59, 60
*Elset, elset=Set-17, instance=Mold-1
12, 49, 50, 56, 125, 126, 128, 130
*Nset, nset=Set-18, instance=Mold-1
2, 3, 33, 34, 35, 36, 37, 38, 39
*Elset, elset=Set-18, instance=Mold-1
28, 35, 108, 118, 120, 132, 134, 186
*Nset, nset=Set-19, instance=Mold-1
1, 2, 5, 6, 26, 27, 28, 29, 30, 31, 32, 61, 62,
63
*Elset, elset=Set-19, instance=Mold-1
18, 28, 56, 59, 87, 91, 111, 114, 115, 139,
180, 183
*Nset, nset=Set-20, instance=Mold-1, generate
1, 176, 1
*Elset, elset=Set-20, instance=Mold-1, generate
1, 283, 1
*Nset, nset=Set-21, instance=Part-1, generate
1, 272, 1
*Elset, elset=Set-21, instance=Part-1, generate
1, 463, 1
*Nset, nset=Set-22, instance=Part-1, generate
1, 272, 1
*Nset, nset=Set-22, instance=Mold-1, generate
1, 176, 1
*Elset, elset=Set-22, instance=Part-1, generate
1, 463, 1
*Elset, elset=Set-22, instance=Mold-1, generate
1, 283, 1
*Nset, nset=Set-23, instance=Part-1
1,
*Nset, nset=Set-24, instance=Part-1
1, 2, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43
*Elset, elset=Set-24, instance=Part-1
1, 3, 4, 32, 53, 80, 82, 128, 130, 133, 138
*Nset, nset=Set-25, instance=Mold-1
2, 3, 4, 5, 33, 34, 35, 36, 37, 38, 39, 54, 55,
56, 57, 58
59, 60
*Elset, elset=Set-25, instance=Mold-1
12, 28, 35, 49, 50, 56, 108, 118, 120, 125,
126, 128, 130, 132, 134, 186
*Nset, nset=Set-26, instance=Part-1
1, 2, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43
*Elset, elset=Set-26, instance=Part-1
1, 3, 4, 32, 53, 80, 82, 128, 130, 133, 138
*Nset, nset=Set-27, instance=Part-1, generate
1, 272, 1
*Elset, elset=Set-27, instance=Part-1, generate
1, 463, 1
*Nset, nset=Set-28, instance=Part-1
2,
*Nset, nset=Set-29, instance=Part-1, generate
1, 272, 1
*Elset, elset=Set-29, instance=Part-1, generate
1, 463, 1

```

```

*Nset, nset=Set-30, instance=Mold-1, generate
  1, 176, 1
*Elset, elset=Set-30, instance=Mold-1, generate
  1, 283, 1
*Elset, elset=_Surf-5_S2, internal,
instance=Mold-1
  12, 28, 35, 49, 50, 108, 118, 120, 122, 125,
126, 128, 130, 132, 134, 150
156, 157, 158, 160, 161, 163, 164, 165, 166,
168, 175, 186
*Elset, elset=_Surf-5_S3, internal,
instance=Mold-1
  35,
*Elset, elset=_Surf-5_S1, internal,
instance=Mold-1
  49, 56
*Surface, type=ELEMENT, name=Surf-5
_Surf-5_S2, S2
_Surf-5_S3, S3
_Surf-5_S1, S1
*Elset, elset=_Surf-6_S2, internal,
instance=Mold-1
[matrix not shown]
*Surface, type=ELEMENT, name=Surf-6
_Surf-6_S2, S2
*Elset, elset=_Surf-7_S2, internal,
instance=Part-1
  1, 3, 4, 53, 80, 82, 128, 130, 133, 138
*Elset, elset=_Surf-7_S2, internal,
instance=Mold-1
[matrix not shown]
*Elset, elset=_Surf-7_S1, internal,
instance=Part-1
  32,
*Elset, elset=_Surf-7_S1, internal,
instance=Mold-1
  28, 49, 56
*Elset, elset=_Surf-7_S3, internal,
instance=Mold-1
  18, 35
*Surface, type=ELEMENT, name=Surf-7
_Surf-7_S2, S2
_Surf-7_S1, S1
_Surf-7_S3, S3
*Elset, elset=_Surf-8_S2, internal,
instance=Part-1
  1, 3, 4, 53, 80, 82, 128, 130, 133, 138
*Elset, elset=_Surf-8_S2, internal,
instance=Mold-1
[matrix not shown]
*Elset, elset=_Surf-8_S1, internal,
instance=Part-1
  32,
*Surface, type=ELEMENT, name=Surf-8
_Surf-8_S2, S2
_Surf-8_S1, S1
*Elset, elset=_Surf-11_S2, internal,
instance=Part-1
[matrix not shown]
*Elset, elset=_Surf-11_S2, internal,
instance=Mold-1
[matrix not shown]
*Elset, elset=_Surf-11_S1, internal,
instance=Part-1
  53,
*Elset, elset=_Surf-11_S1, internal,
instance=Mold-1
  49, 56
*Elset, elset=_Surf-11_S3, internal,
instance=Mold-1
  35,
*Surface, type=ELEMENT, name=Surf-11
_Surf-11_S2, S2
_Surf-11_S1, S1
_Surf-11_S3, S3
*Elset, elset=_Surf-12_S2, internal,
instance=Mold-1
  12, 28, 35, 49, 50, 108, 118, 120, 122, 125,
126, 128, 130, 132, 134, 150
156, 157, 158, 160, 161, 163, 164, 165, 166,
168, 175, 186
*Elset, elset=_Surf-12_S3, internal,
instance=Mold-1
  35,
*Elset, elset=_Surf-12_S1, internal,
instance=Mold-1
  49, 56
*Surface, type=ELEMENT, name=Surf-12
_Surf-12_S2, S2
_Surf-12_S3, S3
_Surf-12_S1, S1
*Elset, elset=_Surf-13_S2, internal,
instance=Part-1
[matrix not shown]
*Elset, elset=_Surf-13_S1, internal,
instance=Part-1
  53,
*Surface, type=ELEMENT, name=Surf-13
_Surf-13_S2, S2
_Surf-13_S1, S1
*Elset, elset=_Surf-14_S2, internal,
instance=Part-1
[matrix not shown]
*Elset, elset=_Surf-14_S1, internal,
instance=Part-1
  53,
*Surface, type=ELEMENT, name=Surf-14
_Surf-14_S2, S2
_Surf-14_S1, S1
*Elset, elset=_Surf-15_S2, internal,
instance=Part-1
[matrix not shown]
*Elset, elset=_Surf-15_S1, internal,
instance=Part-1
  53,
*Surface, type=ELEMENT, name=Surf-15
_Surf-15_S2, S2
_Surf-15_S1, S1

```

```

*Elset, elset=_Surf-16_S2, internal,
instance=Part-1
[matrix not shown]
*Elset, elset=_Surf-16_S1, internal,
instance=Part-1
53,
*Surface, type=ELEMENT, name=Surf-16
_Surf-16_S2, S2
_Surf-16_S1, S1
*Elset, elset=_m_Surf-3_S2, internal,
instance=Mold-1
[matrix not shown]
*Surface, type=ELEMENT, name=m_Surf-3
_m_Surf-3_S2, S2
*Elset, elset=_m_Surf-9_S2, internal,
instance=Mold-1
[matrix not shown]
*Elset, elset=_m_Surf-9_S1, internal,
instance=Mold-1
24,
*Surface, type=ELEMENT, name=m_Surf-9
_m_Surf-9_S2, S2
_m_Surf-9_S1, S1
*Elset, elset=_s_Surf-3_S2, internal,
instance=Part-1
[matrix not shown]
*Elset, elset=_s_Surf-3_S1, internal,
instance=Part-1
53,
*Surface, type=ELEMENT, name=s_Surf-3
_s_Surf-3_S2, S2
_s_Surf-3_S1, S1
*Elset, elset=_s_Surf-9_S2, internal,
instance=Part-1
[matrix not shown]
*Elset, elset=_s_Surf-9_S1, internal,
instance=Part-1
53,
*Surface, type=ELEMENT, name=s_Surf-9
_s_Surf-9_S2, S2
_s_Surf-9_S1, S1
*End Assembly
**
** MATERIALS
**
*Material, name=ARENA
*Conductivity
[matrix not shown]
*Elastic
2.1e+08, 0.3
*Specific Heat
[matrix not shown]
*Density
[matrix not shown]
*Elastic
[matrix not shown]
*Plastic
[matrix not shown]
**
** INTERACTION PROPERTIES
**
**
** Surface Interaction, name=Cond
1.,
** Gap Conductance
[matrix not shown]
** Film Property, name=Conv
[matrix not shown]
** Surface Interaction, name=INTPROP-1
1.,
** Friction, slip tolerance=0.005
0.,
**
** PHYSICAL CONSTANTS
**
** Physical Constants, absolute zero=-273.15,
stefan boltzmann=5.67e-11
**
** BOUNDARY CONDITIONS
**
** Name: BC-1 Type:
Symmetry/Antisymmetry/Encastre
*Boundary
Set-16, XSYMM
** Name: BC-5 Type:
Symmetry/Antisymmetry/Encastre
*Boundary
Set-22, XSYMM
** Name: BC-6 Type:
Symmetry/Antisymmetry/Encastre
*Boundary
Set-25, YSYMM
**
** PREDEFINED FIELDS
**
** Name: Predefined Field-3 Type:
Temperature
*Initial Conditions, type=TEMPERATURE
Set-20, 20.
** Name: Predefined Field-4 Type:
Temperature
*Initial Conditions, type=TEMPERATURE
Set-21, 1500.
**
** INTERACTIONS
**
** Interaction: Cond
*Contact Pair, interaction=Cond,
type=SURFACE TO SURFACE
s_Surf-3, m_Surf-3
** -----
**
** STEP: InMold
**
** Step, name=InMold, nlgeom=YES, inc=14400
** Coupled Temperature-displacement,
creep=none, deltmx=5., stabilize=0.0002,
allsdtol=0.05, continue=NO
10., 14400., 1e-20, 14400.
**

```

```

** LOADS
**
** Name: Load-1 Type: Body heat flux
*Dflux
Set-29, BF, -0.1744
** Name: Load-2 Type: Body heat flux
*Dflux
Set-30, BF, -0.039
**
** INTERACTIONS
**
** Interaction: Rad
*Sradiate
Surf-12, R, 20., 0.76
** Interaction: superfilm-1
*Sfilm
Surf-5, F, 20., 0.02
**
** OUTPUT REQUESTS
**
*Restart, write, frequency=0
**
** FIELD OUTPUT: F-Output-1
**
*Output, field
*Node Output
NT, U
**
** HISTORY OUTPUT: H-Output-1
**
*Output, history, variable=PRESELECT
*End Step
** -----
**
** STEP: OutMold
**
*Step, name=OutMold, nlgeom=YES,
inc=6585
*Coupled Temperature-displacement,
creep=none, deltmx=5.
10., 6585., 1e-12, 6585.
**
** BOUNDARY CONDITIONS
**
** Name: BC-1 Type:
Symmetry/Antisymmetry/Encastre
*Boundary, op=NEW
Set-16, XSYMM
** Name: BC-3 Type:
Symmetry/Antisymmetry/Encastre
*Boundary, op=NEW
Set-28, ENCASTRE
** Name: BC-4 Type: Displacement/Rotation
*Boundary, op=NEW
Set-24, 3, 3
Set-24, 4, 4
Set-24, 5, 5
** Name: BC-5 Type:
Symmetry/Antisymmetry/Encastre

```

```

*Boundary, op=NEW
** Name: BC-6 Type:
Symmetry/Antisymmetry/Encastre
*Boundary, op=NEW
** Name: BC-7 Type: Displacement/Rotation
*Boundary, op=NEW
Set-27, 1, 1
Set-27, 5, 5
Set-27, 6, 6
**
** LOADS
**
** Name: Load-1 Type: Body heat flux
*Dflux, op=NEW
Set-29, BF, -0.028
** Name: Load-2 Type: Body heat flux
*Dflux, op=NEW
**
** INTERACTIONS
**
** Interaction: Cond
*Model Change, type=CONTACT PAIR,
remove
s_Surf-3, m_Surf-3
** Interaction: Rad
*Sradiate, op=NEW
** Interaction: Rad-2
*Sradiate, op=NEW
Surf-16, R, 20., 0.76
** Interaction: superfilm-1
*Sfilm, op=NEW
** Interaction: superfilm-2
*Sfilm, op=NEW
Surf-15, F, 20., Conv
**
** OUTPUT REQUESTS
**
*Restart, write, frequency=0
**
** FIELD OUTPUT: F-Output-3
**
*Output, field
*Node Output
NT, U
**
** HISTORY OUTPUT: H-Output-2
**

```

## 2D Thermal- Stress analysis with different phases

```

*Heading
** Job name: MT2 Model name: Model-1
** Generated by: Abaqus/CAE 6.13-1
*Preprint, echo=NO, model=NO, history=NO,
contact=NO
**
** PARTS
**
*Part, name=Part-1

```

```

*Node
[matrix not shown]
*Element, type=CPE3
[matrix not shown]
*Nset, nset=Set-2, generate
  1, 88, 1
*Elset, elset=Set-2, generate
  1, 140, 1
*Nset, nset=_PickedSet6, internal, generate
  1, 88, 1
*Elset, elset=_PickedSet6, internal, generate
  1, 140, 1
*Nset, nset=Set-5
[matrix not shown]
*Elset, elset=Set-5, generate
  71, 140, 1
*Orientation, name=Ori-1
  1., 0., 0., 0., 1., 0.
  3, 0.
** Region: (Section-1:Set-2), (Material
Orientation:Picked)
*Elset, elset=_I1, internal, generate
  1, 70, 1
** Section: Section-1
*Solid Section, elset=_I1, orientation=Ori-1,
material=Material-1
,
** Section: Section-2
*Solid Section, elset=Set-5, orientation=Ori-1,
material=Material-2
,
*End Part
**
**
** ASSEMBLY
**
*Assembly, name=Assembly
**
*Instance, name=Part-1-1, part=Part-1
*End Instance
**
*Nset, nset=Set-1, instance=Part-1-1
  3, 4, 17, 18, 19, 20, 21, 22
*Elset, elset=Set-1, instance=Part-1-1, generate
  58, 70, 2
*Nset, nset=Set-2, instance=Part-1-1
  5, 6, 31, 32, 33, 34, 35, 36
*Elset, elset=Set-2, instance=Part-1-1, generate
  128, 140, 2
*Nset, nset=Set-3, instance=Part-1-1
[matrix not shown]
*Elset, elset=Set-3, instance=Part-1-1, generate
  1, 70, 1
*Nset, nset=Set-4, instance=Part-1-1, generate
  1, 88, 1
*Elset, elset=Set-4, instance=Part-1-1, generate
  1, 140, 1
*Nset, nset=Set-5, instance=Part-1-1, generate
  1, 88, 1
*Elset, elset=Set-5, instance=Part-1-1, generate
  1, 140, 1
*End Assembly
**
** MATERIALS
**
*Material, name=Material-1
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Expansion, type=ANISO
  1.2e-05, 1.3e-05, 0., 0., 0., 0.
*Material, name=Material-2
*Elastic
  70000., 0.3
*Expansion
  1e-05,
**
** PREDEFINED FIELDS
**
*Initial Conditions, type=TEMPERATURE
Set-5, 1500.
** -----
-----
**
** STEP: Step-1
**
*Step, name=Step-1, nlgeom=YES
*Static
  1., 1., 1e-05, 1.
**
** BOUNDARY CONDITIONS
**
** Name: BC-1 Type:
Symmetry/Antisymmetry/Encastre
*Boundary
Set-1, ENCASTRE
** Name: BC-2 Type: Displacement/Rotation
*Boundary
Set-2, 1, 1
Set-2, 2, 2
Set-2, 6, 6
**
** PREDEFINED FIELDS
**
** Name: Predefined Field-2 Type:
Temperature
*Temperature
Set-4, 20.
**
** OUTPUT REQUESTS
**
*Restart, write, frequency=0
**
** FIELD OUTPUT: F-Output-1
**
*Output, field, variable=PRESELECT
**
** HISTORY OUTPUT: H-Output-1
**
*Output, history, variable=PRESELECT
*End Step

```

## Tensile Test

```

*Heading
** Job name: F1 Model name: Model-1
** Generated by: Abaqus/CAE 6.13-1
*Preprint, echo=NO, model=NO, history=NO,
contact=NO
**
** PARTS
**
*Part, name=Part-1
*Node
[matrix not shown]
*Element, type=CPS4R
[matrix not shown]
*Node
    268,    0.,    35.,    0.
*Nset, nset=Part-1-RefPt_, internal
268,
*Nset, nset=Set-1, generate
    1, 267, 1
*Elset, elset=Set-1, generate
    1, 212, 1
*Nset, nset=_PickedSet5, internal, generate
    1, 267, 1
*Elset, elset=_PickedSet5, internal, generate
    1, 212, 1
*Nset, nset=_PickedSet12, internal, generate
    1, 267, 1
*Elset, elset=_PickedSet12, internal, generate
    1, 212, 1
*Orientation, name=Ori-2
1., 0., 0., 0., 1., 0.
3, 0.
** Section: Section-1
*Solid Section, elset=Set-1, orientation=Ori-2,
material=F1-Mat
1.,
*End Part
**
**
** ASSEMBLY
**
*Assembly, name=Assembly
**
*Instance, name=Part-1-1, part=Part-1
*End Instance
**
**
*Nset, nset=Set-7, instance=Part-1-1
4,
*Nset, nset=Set-8, instance=Part-1-1
268,
*Nset, nset=Set-11, instance=Part-1-1
268,
*Nset, nset=Set-12, instance=Part-1-1
[matrix not shown]
*Elset, elset=Set-12, instance=Part-1-1, generate
37, 72, 1
*Nset, nset=Set-13, instance=Part-1-1
[matrix not shown]
*Elset, elset=Set-13, instance=Part-1-1, generate
    1, 36, 1
*Nset, nset=Set-16, instance=Part-1-1
268,
*Nset, nset=Set-22, instance=Part-1-1, generate
    3, 5, 1
*Nset, nset=Set-25, instance=Part-1-1
[matrix not shown]
*Elset, elset=Set-25, instance=Part-1-1, generate
37, 72, 1
*Nset, nset=Set-29, instance=Part-1-1, generate
    3, 5, 1
*Nset, nset=Set-31, instance=Part-1-1
268,
*Nset, nset=Set-34, instance=Part-1-1
268,
*Nset, nset=b_Set-19, instance=Part-1-1
[matrix not shown]
*Elset, elset=b_Set-19, instance=Part-1-1,
generate
    1, 36, 1
*Nset, nset=m_Set-9, instance=Part-1-1
268,
*Nset, nset=m_Set-14, instance=Part-1-1
268,
*Nset, nset=m_Set-17, instance=Part-1-1
268,
*Nset, nset=m_Set-30, instance=Part-1-1
268,
*Nset, nset=m_Set-33, instance=Part-1-1
268,
*Nset, nset=s_Set-9, instance=Part-1-1
[matrix not shown]
*Elset, elset=s_Set-9, instance=Part-1-1,
generate
    1, 36, 1
*Nset, nset=s_Set-14, instance=Part-1-1
    3, 4, 5, 26, 27, 28, 29
*Elset, elset=s_Set-14, instance=Part-1-1,
generate
31, 36, 1
*Nset, nset=s_Set-17, instance=Part-1-1
    3, 4, 5, 26, 27, 28, 29
*Elset, elset=s_Set-17, instance=Part-1-1,
generate
31, 36, 1
*Nset, nset=s_Set-22, instance=Part-1-1
    3, 4, 5, 26, 27, 28, 29
*Elset, elset=s_Set-22, instance=Part-1-1,
generate
31, 36, 1
*Nset, nset=s_Set-32, instance=Part-1-1
[matrix not shown]
*Elset, elset=s_Set-32, instance=Part-1-1,
generate
    1, 36, 1
*Elset, elset=_Surf-1_S3, instance=Part-1-1, generate

```

```

34, 36, 1
*Surface, type=ELEMENT, name=Surf-1
_Surf-1_S3, S3
*Elset,      elset=_Surf-2_S3,      internal,
instance=Part-1-1, generate
31, 36, 1
*Surface, type=ELEMENT, name=Surf-2
_Surf-2_S3, S3
*Elset,      elset=_s_Surf-3_S3,      internal,
instance=Part-1-1, generate
31, 36, 1
*Surface, type=ELEMENT, name=s_Surf-3
_s_Surf-3_S3, S3
*Surface, type=NODE, name=s_Set-32_CNS_,
internal
s_Set-32, 1.
** Constraint: Constraint-1
** Coupling, constraint name=Constraint-1, ref
node=m_Set-33, surface=s_Set-32_CNS_
** Kinematic
** End Assembly
** Amplitude, name=Amp-1
[matrix not shown]
**
** MATERIALS
**
*Material, name=F1-Mat
*Elastic, type=ANISOTROPIC
217300., 66230., 220780., 0., 0., 1., -105.,
-10.
0., 77500., 0., 0., 0., 0., 1., 0.
0., 0., 0., 0., 1.
*Material, name=F2
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=F3
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=F4
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=M-66
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=M-72
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=M-156
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=M1
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=P1
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=P2
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=P3
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=S-2
*Elastic, type=ANISOTROPIC
[matrix not shown]
*Material, name=S-2205
*Elastic
200000., 0.32
**
** BOUNDARY CONDITIONS
**
** Name: BC-2 Type: Displacement/Rotation
*Boundary
Set-13, 1, 1
Set-13, 6, 6
**      Name:      BC-3      Type:
Symmetry/Antisymmetry/Encastre
*Boundary
Set-25, ENCASTRE
** -----
-----
**
** STEP: Step-1
**
*Step, name=Step-1, nlgeom=NO
*Static
0.1, 1., 1e-05, 1.
**
** LOADS
**
** Name: Load-3 Type: Concentrated force
*Load, amplitude=Amp-1
Set-34, 2, 2777.
**
** OUTPUT REQUESTS
**
*Restart, write, frequency=0
**
** FIELD OUTPUT: F-Output-1
**
*Output, field
*Node Output
CF, RF, U
*Element Output, directions=YES
E, S
*Contact Output
CDISP, CSTRESS
**
** HISTORY OUTPUT: H-Output-1
**
*Output, history, variable=PRESELECT
*End Step

```