Master Thesis

Computational Materials Sciences

Supervisor:
Dr.-Ing.Helal Chowdhury

1st Examiner(Supervisor):
Prof.Dipl.-Ing.Björn Kiefer

2nd Examiner(Reviewer):
Dr.-Ing.Arun Prakash

Mentor:
M. Sc.Alexander Malik

Machine-Learning assisted understanding of RVE-size dependent uncertainties and corresponding hierarchy of properties

Sai Karthikeya Vemuri

DECLARATION

I herewith formally declare that I have written the submitted thesis independently. I did not use any outside support except for the quoted literature and other sources mentioned in the paper.
I clearly marked and separately listed all of the literature and all of the other sources which I employed when producing this academic work, either literally or in content.

Place, Date	Sai Karthikeya Vemuri

ACKNOWLEDGEMENT

I would like to express my sincere and profound gratitude to my advisor, Dr. Helal Chowdhury, whose sincerity, resilience and encouragement I will never forget. Dr. Chowdhury has been an inspiration as I hurdled through the path of this Masters thesis. He is the true definition of a leader and the ultimate role model. This thesis would not have been possible without Prof. Kiefer and Dr. Prakash, whose guidance from the beginning of my masters has shaped my perspective on life. I am thankful for the extraordinary experiences they arranged for me in CMS and for providing opportunities for me to grow professionally. It is an honor to learn from Prof. Kiefer and Dr. Prakash. I would like to thank Mr. Malik for his cheerful, friendly enthusiasm and for being a continuous support throughout the thesis.

PREAMBLE

This work is a part of an ongoing project KILu (KI in der Luftfahrtforschung) that belongs to the institute of Test and Simulation for Gasturbines in Augsburg, an institute of Deutsches Zentrum für Luft- und Raumfahrt e. V. (DLR). The thesis was conceptualized and conducted at DLR in Augsburg as part of KILu activities.

For the completeness of the report, some ideas, pipelines, material properties were taken from KILu activities. For example, original DREAM3D pipeline used to generate Statistical Volume Elements, the idea of creating a sequence from sizes for LSTM network etc. All relevant figures/pipelines that comes from KILu works are marked with *.

Abstract

Materials discovery and development processes are the two forefronts of materials science and engineering. Representative/statistical Volume Elements (RVEs, SVEs) are extensively used to simulate different homogenized properties of an engineered microstructure, facilitating further development or improvement of that materials-system with optimized performance. The reliability of these predicted properties is limited due to various uncertainties which come from various sources like underlying stochastic process, complexity of microstructure etc. These tend to reduce if the size of a volume element (VE) is larger because of incorporation of more information, however, simulations tend to become expensive and cumbersome. Hence, there is a need of finding an acceptable size which balances both higher computational cost and lower uncertainties. A related framework in this regard or an understanding for the estimation of acceptable sizes of a realistic RVE considering a range of properties is necessary. Considering the above background it can be said that the choice of a smaller sized VE could results in pronounced simulation-uncertainties considering a complex material model when used for the same microstructure i.e, a larger size is required for an acceptable RVE if the complexity of the model and the microstructure increases and thus, a hierarchy of acceptable sizes can be mapped for the same microstructure using different material models incorporating complex non-linear behaviours. Such an RVE-Map is supposed to be extremely useful in the context of more reliable decision-making based on homogenization results and multiscale simulations.

In this work, a primary version of an RVE-map has been attempted to be established. Firstly, in order to build statistically equivalent volume-elements (SVE) in terms of necessary microstructural/morphological descriptors, a number of different realizations of increasing size has been generated. Next, a homogenized property is compared by carrying out the similar simulation using all the SVEs and thus, an acceptable size has been selected after convergence study, based on i) the relative error of less than 1% in between two consecutive realizations and ii) a single realization. Thus, the procedure has been repeated by considering a range of properties in order to show the expected hierarchy of selected properties.

The sequence modelling capability of deep learning models has been leveraged

to get the property-predictions of higher sized SVEs effectively, by saving time and computational cost, and thus facilitating faster decision-making. This is done by learning the sequence formed by the property-curves at lower sizes and having predictions at higher sizes until convergence is achieved.

From the established RVE-map, it can be concluded that a cube of side length 60 μm is sufficient for elasticity, while for pure thermal simulation or for thermo-elasticity or for elasto-plasticity under only slip-condition an approximate cube-side of 140 μm is reasonable. On the other hand, if twinning mechanism is added to the plasticity model, then a slightly larger size than that of purely slip-case is expected. When thermal a boundary condition is added to coupled slip-twin based plasticity, then reasonable side-length of the expected RVE reaches to nearly 180 μm.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Understanding of the structure-property relationship is arguably the most important aspect of materials engineering. A large part of the community is dedicated to the study of how tinkering with a structure will affect its property. Many materials (natural or synthetic) may look homogeneous macroscopically, but have heterogeneities once they are sufficiently zoomed in. These heterogeneities may origin from the distribution of different defects in the form of grain boundaries, lamellar boundaries, twin boundaries, anti-phase boundaries, stacking faults, dislocation network, etc. in polycrystals. Similarly it could be due to the presence of inclusions, pores, precipitates in composites. These microscopic heterogeneities play an important role in the overall response of the material, and hence there is a need to analyse how these microscopic heterogeneities actually influence the macroscopic behaviour. A Volume Element, which is periodic in nature containing all these heterogeneities, is assumed to be repeating in all directions to form the macro specimen. Thus, the averaged (homogenized) property of this periodic element is *representative* of the whole material. This is the idea behind the existence of *Representative Volume Elements* or simply RVEs. Hence, broadly, an RVE is a material volume whose effective behaviour is representative of the material as a whole.[12].

There are many not-so-precise definitions of RVE used for different scenarios. A collection of these definitions is provided by a review paper by Gitman et. al [13]. The definitions are as follows:

1. The RVE is a sample that (a) is structurally entirely typical of the whole mixture on average, and (b) contains a sufficient number of inclusions for the apparent overall moduli to be effectively independent of the surface values of traction and displacement, as long as these values are macroscopically uniform.

2. An RVE is the minimal material volume, which contains statistically enough mechanisms of deformation processes. The increasing of this volume should not lead to changes of evolution equations for field-values, describing these mechanisms

3. The RVE must be chosen sufficiently large compared to the microstructural size for the approach to be valid, and it is the smallest material volume element of the composite for which the usual spatially constant overall modulus macroscopic constitutive representation is a sufficiently accurate model to represent the mean constitutive response.

4. The RVE is a model of the material to be used to determine the corresponding effective properties for the homogenised macroscopic model. The RVE should be large enough to contain sufficient information about the microstructure in order to be representative, however it should be much smaller than the macroscopic body (This is known as the Micro–Meso–Macro principle).

5. The RVE is defined as the minimum volume of laboratory scale specimen, such that the results obtained from this specimen can still be regarded as representative for a continuum.

6. The size of the RVE should be large enough with respect to the individual grain size in order to define overall quantities such as stress and strain, but this size should also be small enough in order not to hide macroscopic heterogeneity

The common point among all the definitions is that it should be small enough to be macroscopically homogeneous and large enough to be microscopically heterogeneous. For the development of synthetic microstructures, different stochastic processes are used for the generation of microstructures. These uncertainties might come from various sources like different realizations, sizes, mesh-desntiy, different boundary conditions etc., even the same process of generations of RVE might give different results. It is an inherent factor in various aspects of modelling, processing, design, and development of a materials-system. Uncertainty is not only added from different sources, but also propagated when different physics and numerics are employed to simulate different complex behaviours. One of such prime examples is the scale-bridging problem, where propagation of uncertainty is heavily influenced by individual scales, especially when lower scale information is passed to the higher scale. So, identifying the influential sources from the lower scale can be considered as a very clever step in controlling the propagation of uncertainties, thereby making the results of the simulations reliable and robust.

Towards a precise definition of an RVE

The evolution of the RVE-definition and some relevant literature regarding the RVE size estimation approaches are presented here. Hill et. al [14] in 1963 gave the first and most simple definition of RVE as *a sample that is structurally typical of the whole microstructure for a given material, containing a sufficiently large number of heterogeneities, while being small enough to be considered homogeneous from a continuum viewpoint.* The quantification of RVE size was first done by Drugan and Willis [15] in 1996. They defined RVE in the context of composites as *Volume over which the usual macroscopically homogeneous "effective modulus" constitutive models for composites can be expected to apply.* and they estimated the minimum size of RVEs using this definition and assuming change in constant modulus as 5%. One of the most important aspects of their work was the separation of concept of an RVE in terms of microstructural influence and property perspective. However, they did not discussion the detail of microstructure and property based RVE. Later, Kanit et. al [16] in 2003 elaborated the concept of estimation of size of an RVE. They proposed that an RVE must ensure a given accuracy based on 5 different criteria. These five criteria are the physical property, the contrast of properties, the volume fractions of components, the desired relative precision for the estimation of the effective property and the number of realizations of the microstructure associated with computations that one is ready to carryout. Swaminathan et. al [17] in 2006 proposed the concept of statistically equivalent RVEs, simply SERVES which can be used to evaluate accurate homogenized properties to be used in macroscopic analysis. Equivalence of the homogenized stiffness tensor, independence from loading and location in the microstructure, agreement in the statistical distribution of the microstructural variables with those for the entire microstructure were considered as criteria for determination of SERVE size. Gitman et. al [13] in 2007 have chosen a combined numerical-statistical approach to determine the RVE size and convergence studies have been done usng Chi-square analysis on the property values of different realizations. Echlin et. al [1] in 2014 have shown a method for categorizing and quantifying volume elements based on microstructure, properties, and design(MVE, PVE, and DVE) shown in the figure 1.1. They have shown that for a particular convergence limit, the microstructure based VEs or MVEs are smaller than property based VEs or PVEs, which in turn are smaller than design based VEs or DVEs thus creating a hierarchy as shown in the figure 1.2. Pinz et. al [18] in 2018 have used the concept of *statistically equivalent* RVEs to address the difficulty of quantifying RVE for microstructures with non-uniformly dispersed heterogeneities. They are further catergorized into microstructure based SERVEs (m-SERVES) and property based SERVES(p-SERVES) depending on whether morphological descriptors or properties are used for checking the acceptable size.

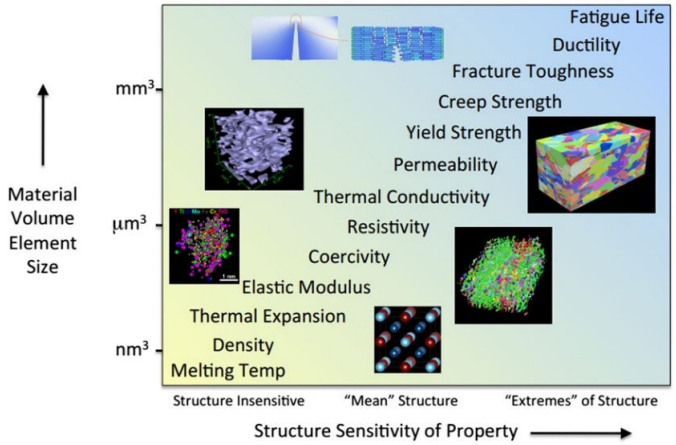


Figure 1.1: Material properties require differently sized VEs to accurately describe them [1].

This thesis takes the concepts of m-SERVES and p-SERVES and uses them together in a framework to have an acceptable size of an RVE and map out a hierarchy when different material models are used. This thesis can also be seen as a detailed quantification of hierarchy given by Echlin et. al[1] in the figure 1.1 with emphasis on microscale.

The uncertainties in an RVE produced by a stochastic process are more when the size is less and they decrease with the increasing size. For larger sizes of RVE the uncertainties might be low, but the computation power required to calculate the response will be more. So naturally the question arises whether there exists an *acceptable size* of an RVE which is large enough to have minimum uncertainties and also small enough to save computation power and speed. The hypothesis is that this size effect is dependent on the complexity of the material model used and there exists a hierarchy of acceptable RVE sizes for all material models. In simpler terms, if the complexity of the material model increases, the uncertainties will be more pronounced and the size of acceptable RVE will be larger.

The idea of this thesis is to validate this hypothesis and map out the hierarchy. This is done first by generating statistical volume elemenrs of different sizes and different realizations for each size. The microstructure which is of interest here is dual phase Gamma Titatnium Aluminide (TiAl). A 2D sample from experiments is used as starting point and it is reconstructed to form samples of SVEs of differ-

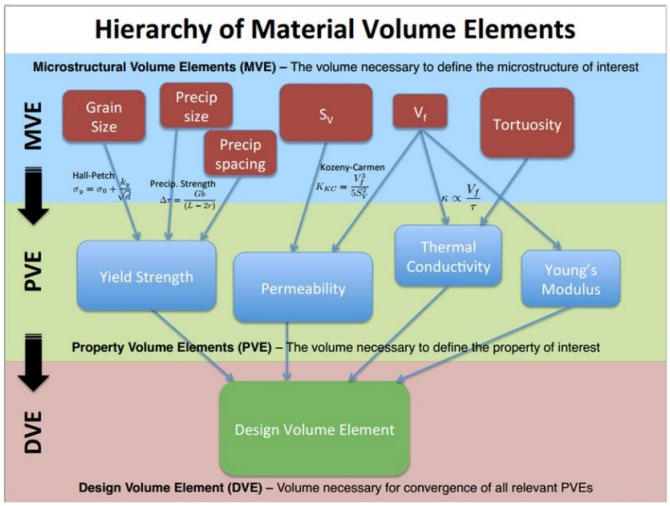


Figure 1.2: Material volume elements can be divided into a hierarchy which is tiered based on their dependence [1].

ent sizes using DREAM3D [2] by matching the crystallography. Microstructural Descriptors are compared of these realizations of all sizes to comply for the fact that these Volume Elements indeed represent same microstructure thus they can be qualified as microstructure based statistically equivalent representative volume elements or simply m-SERVES.

Next, DAMASK [4] is used to measure the stress-strain responses of these SVEs when different material models(pure elastic, elasto-plastic) for the two phases in the microstructure are used. Same tensile test is carried out for all the SVEs of different sizes with same material property. Since these SVEs are already qualified as m-SERVES any uncertainties seen here only attribute to the size effect. The DAMASK results are processed and homogenized stress-strain curves are plotted for all realizations of all sizes. The size is said to be acceptable if the relative error between values of curves of different realizations of same is less than 1 %. This qualifies the SVEs as property based statistically equivalent RVEs or simply p-SERVES.

If the convergence is not seen after five sizes, deep learning is used to save computation time and accelerate the decision making process. A sequence of curves is formed from the existing sizes and the deep learning model is trained to learn this sequence and predict the next occurences until the relative error between succesive occurences is less than 1 %.

A hierarchy of acceptable sizes of RVE is mapped out for different material models(pure elasticity, elasto-plasticity with slip and slip+twin, thermo-elasticity and thermo-elasto-plasticity) by essentially combining the concepts of m-SERVES and p-SERVES to have a reliable acceptable size.

The thesis is structured in the following manner. First an Introduction is provided along wih background which covers the evolution of definitions and concepts of Representative Volume Elements. The modern definitions of microstructure based and property based RVEs are also provided. Next, a little background and theory of tools used within the context of this thesis like DREAM3D, DAMASK, Deep learning along with some information on Titanium Aluminides is provided. In the next section, the comparision of microstructural descriptors of generated SVEs is provided and then, the Deformation Gradient - Piola Kirchoff stress curves are provided for different material models used in these m-SERVES and p-SERVES.

The terms referring the RVEs throughout the thesis is explained here. The generated samples from DREAM3D are called Statistical Volume Elements(SVEs) or simply Volume Elements(VEs) since they are not yet *representative*. After having a comparision of morphological descriptors they are called microstructure based SERVES or m-SERVES and after having convergence of property at a particular size, they are termed as property based or p-SERVES. This combined m- and p-SERVES are the acceptable RVEs.

Chapter 2

Tools, materials and methods

2.1 DREAM.3D for 2D to equivalent 3D generation

All this information in this section is taken from the website and documentation of DREAM.3D which can be found in [2] and [3]. DREAM.3D stands for *Digital Representation Environment for Analyzing Microstructure in 3D*. DREAM.3D is a multidimensional, multimodal data reconstruction, instantiation, quantification, meshing, handling, and visualization software package that is open source, cross-platform, and modular.

Statistical Volume Elements of TiAl need to be generated from the same process at different sizes such that all the VEs indeed represent the same type of material. DREAM.3D is used to generate 3D synthetic SVEs from 2D EBSD image by extracting microstructural descriptors to view them as m-SERVEs. A brief overview of DREAM.3D is provided and the pipeline used to generate VEs is also discussed. Then the microstructural descriptors are compared for different sizes to state the fact that these varied size VEs represent the similar microstructure, thus viewing them as m-SERVES.

DREAM.3D provides tools/filters which can be used to build pipelines. These pipelines are used for microstructure data analysis and manipulation. It provides a flexible data structure in which this microstructure information is stored and this data structure is manipulated by filters according to the task to be done and it is stored in a proprietary format. DREAM.3D is a filter suite developed for materials scientists to use to reconstruct 3D microstructures or synthetically build microstructures. It is built on top of the SIMPL and SIMPLView software projects.

Some of the salient features and functionalities include:

1. Import ASCII, EBSD Vendor Data and standard image files.

2. Alignment, cleaning, reconstruction, segmentation and analysis of imported data Statistics, either synthetically created or from real data, can be used to generate a statistically equivelent material structure

3. Reconstructed and synthetic volumes can be surfaced meshed to allow export into FEM or other simulations.

4. Import/Export of data into/out of ASCII or Binary files

5. Export Surface Mesh as STL Files

6. Over 350 filters to process data

DREAM.3D makes use of an abstract, hierarchical data structure based on combinational topology and shared mesh structure descriptions. The tree structure of the generalized data structure has the following node types:

1. Data Container Array: The root node of the data structure. In most cases, a particular workflow will only have one Data Container Array, but this is not a hard requirement. The Data Container Array has access to create and retrieve all objects that descend from it, not just its immediate Data Container children.

2. Data Container: Holds Attribute Matrices for data that belong to unique geometries. Two Data Containers are distinguised by their associated Geometry.

3. Attribute Matrix: Holds Attribute Arrays, which are the containers for raw data. The type of Attribute Matrix labels the hierarchy level to which its associated Attribute Arrays corresponds.

4. Attribute Array: Holds raw data in memory. DREAM.3D utilizes a flat data storage approach, such that even multidimensional data is allocated into a contiguous section of memory. This approach enables faster and more efficient compute times.

A visualization of the data structure is shown in the figure 2.1

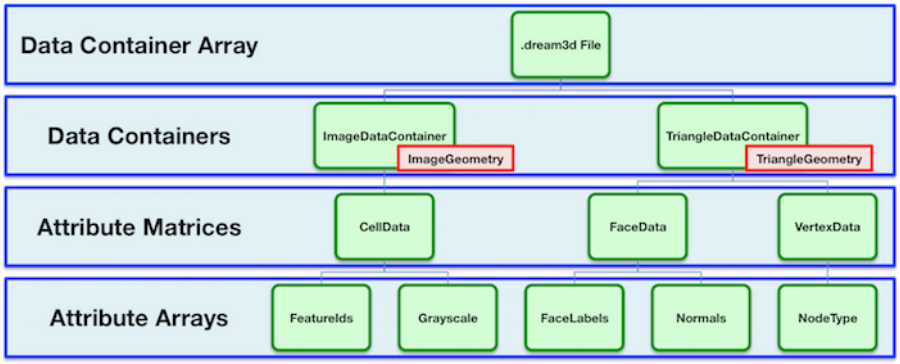


Figure 2.1: Data Structure in DREAM.3D [2] and [3]

Pipeline∗ The processing of Data in DREAM.3D takes place through pipelines. Pipelines are made from series of filters which modify / process the underlying data according to the requirement. The pipeline, just as the name suggests can be imagined as a pipe line built from individual series of pipes, which are here filters. The fluid flowing inside the pipeline is the data which is being processed. This fluid may be compressed, color can be added etc when it enters a pipe similarly the data can be filteres/processed according to the requirement in an individual filter of a pipeline. There are three types of filters to Generate, Modify and Store the Data structure, these are placed sequentially to create a working pipeline as shown in the figure 2.2

The starting point of generation of 3D Volume Elements of different sizes is a 2D EBSD image provided by DLR Institute. This image is read into a data structure named Image Data Container. The basic outline of how 2D to 3D reconstruction is taking place is as follows: The ensemble statistics of 2D image are stored in a container. A synthetic volume of a particular side length with established shape type is created and the crystallography is matched between the Image and created synthetic volume. Iterations over different statistics takes place and they are matched as closely as possible thus giving an equivalent 3D microstructure from a 2D image.

The snippet of the pipeline for 3D and the corresponding Data Structure (Synthetic Data Volume) are shown in the figures 2.3.

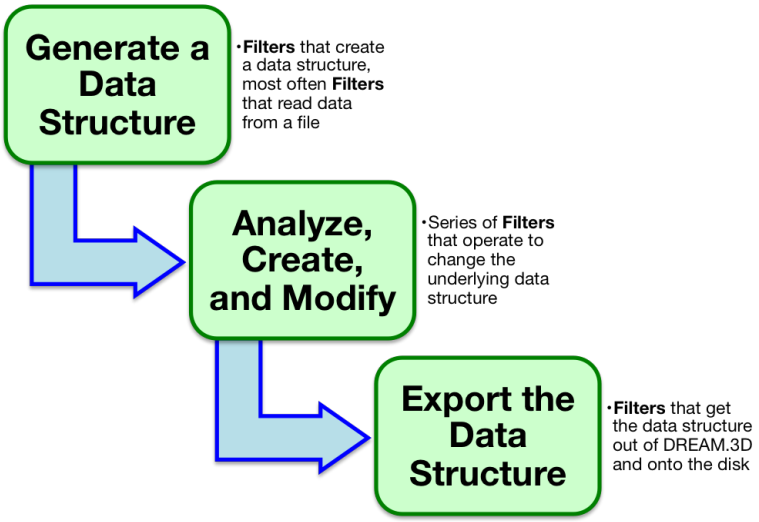


Figure 2.2: Anatomy of a pipeline [2] and [3]

Figure 2.3: Pipeline to generate equivalent 3D volume elements from 2D EBSD Image and the corresponding Data Structure (Synthetic Data Volume)*

2.2 DAMASK – The Düsseldorf Advanced Material Simulation Kit

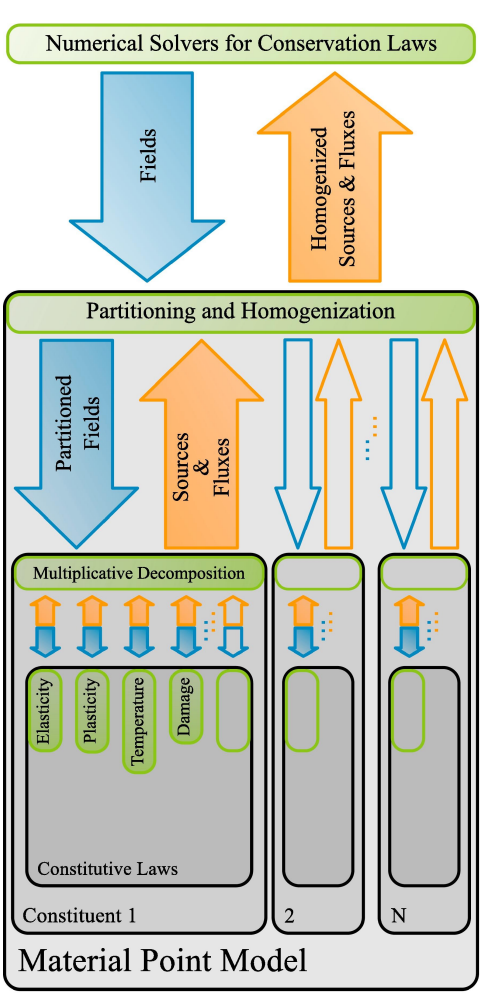
The micromechanical simulations for property analysis are done on the volume elements of all sizes using DAMASK. Hence a small overview of the concept and usage is provided here. This is not a complete description of framework and implementations in DAMASK but only a brief overview which is relevant to the thesis. All the relevenat information and figures are taken from the DAMASK paper [4] and the website *https://damask.mpie.de* contains examples and tutorials.

2.2.1 Concept

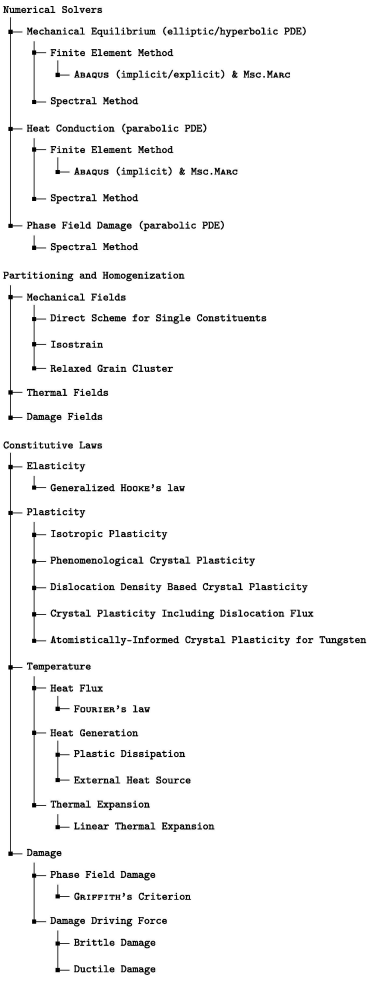
DAMASK provides the scientific community with an open, versatile, and simple-to-use implementation that is highly modular and allows the usage and straightforward implementation of various types of constitutive laws and numerical solvers. DAMASK's internal modular structure is derived directly from the hierarchy inherent in the continuous description used. DAMASK was created to replicate the multi-scale hierarchy and multi-physics structure seen in the underlying material physics of thermo-mechanical loading of complex materials. As a result, template functions that connect numerical solvers, homogenization methods, and constitutive laws are defined. Multiple constitutive laws and homogenization schemes, as well as a specific set of solvers for the associated boundary and/or initial value issues, can be mixed in the same model for more flexibility. To best handle the situation at hand, a choice might be made between model accuracy and numerical effort in this way.

Hierarchichal Structure

The conservation laws require a structured multi-scale description of the fluxes and sources. The hierarchical structure is as follows: At the highest level, a Division and Homogenisation scheme performs the partitioning of prescribed field values on a material point among its underlying microstructural elements and the subsequent homogenization of each constituent's constitutive reaction. The time integration of the underlying constitutive laws for fluxes and sources determines the reaction of each microstructural constituent at the intermediate Constituent Level. To provide this response at the lowest level, constitutive laws based on evolving internal state variables are implemented in DAMASK. The framework is designed in a modular structure to allow for a flexible and expandable implementation. The Material Point Model is made up of abstract modules for



(a) Hierarchical structure at a material point.



(b) Modular structure at the hierarchical levels.

Figure 2.4: Concept of DAMASK. Each material point, which is part of a discretized body on which conservation laws are solved, is made up of multiple constituents that comprise of various constitutive laws [4]

partitioning and homogenization, as well as the Constitutive Level. The provision of these abstract modules is intended to make branching into the various specialized sub-modules present in DAMASK as shown in 2.4.

2.2.2 Spectral Method and FFT Solver

Building a model with a number of material points by using a combination of modules from partitioning, homogenization and constitutive law and applying a conservation law from the continuum formulation gives a PDE which is very difficult to solve analytically. Generally numerical methods like Finite Element Method(FEM), Finite Volume Method(FVM) and Finite Difference Method(FDM) are used to solve these complex PDEs. FEM is a popular and widely used. Galerkin's method is the basis for FEM, which basically involves conversion of a weak formulation of differential equation into a linear and non-linear system of differential equations using ansatz functions. In FEM, these ansatz functions are are usually polynomials of low degree (p $<$ 3) with compact support, meaning they are non-zero only in their domain (i.e. in one element). They equal zero in all other elements. The approximate solution is the result of the assembly of the single elements into a matrix. The matrix links the discrete input values with the discrete output values on the sampling points. Thus, the FEM converts PDEs into system of linear and non-linear equations. The resulting matrix is sparse because only a few ansatz functions are non-zero on each point.
Spectral methods can be viewed as a variant of FEM, where higher degree polynomials are used (p>3) and the anstaz functions are not restricted to elements, i.e taking a global approach. Higher order polynomials give higher accuracy for same number of sampling points and these spectral methods are expected to have exponential convergence, meaning a faster convergence of solution. If these ansatz functions are trigonometric polynomials, the system can be expressed and solved easily in Fourier space. This property is leveraged in DAMASK using FFT solver named FFTW(Fastest Fourier transform in West). The basic concept [19] is outlined here, the function *u(x)* is written as combination of global ansatz functions. If we take the number of sampling points as N+1, then the approximation is given as 2.1

$$u(x) \approx \sum_{n=0}^{N} a_n \phi_n(x) \tag{2.1}$$

This is then used to approximate the required solution as 2.2

$$Lu(x) = f(x) \tag{2.2}$$

here L is the differential operator and *u(x)* is the unknown we are solving for. The residual function 2.3 is the error between the original solution and approximated

Crystal class	Number of C_{ij}	List of elastic constants
Triclinic	21	All possible combinations
Monoclinic	13	$C_{11}; C_{12}; C_{13}; C_{16}; C_{22}; C_{23}; C_{26}; ; C_{33}; C_{36}; C_{44}; C_{45}; C_{55}; C_{66}$
Orthorhombic	9	$C_{11}; C_{12}; C_{13}; C_{22}; C_{23}; C_{33}; C_{44}; ; C_{55}; C_{66}$
Trigonal	6 or 7	$C_{11}; C_{33}; C_{44}; C_{13}; C_{12}; C_{14}; C_{25}$
Tetragonal	6	$C_{11}; C_{33}; C_{44}; C_{13}; C_{12}; C_{66}$
Hexagonal	5	$C_{11}; C_{33}; C_{44}; C_{12}; C_{14}$
Cubic	3	$C_{11}; C_{12}; C_{44}$
Isotropic	2	$C_{11}; C_{44}$

Table 2.1: Independent elastic constants for various crystal symmetries [11].

one and the spectral methods are employed to minimize this residual

$$R\left(x; a_0, a_1, \ldots, a_N\right) = L\left(\sum_{n=0}^{N} a_n \phi_n(x)\right) - f(x) \tag{2.3}$$

2.2.3 Material Models

Some material models and corresponding constitutive laws are discussed. Note that there are more modules available in DAMASK and only the used relevant to the present thesis are discussed. These form the foundational blocks of the hierarchical structure and are characterized by set of internal state variables. Their purpose is to provide the response to the given input fields and fluxes and also to capture the evolution of internal state variables incase of history dependence.

Elasticity Generalized Hookes Law is implemented as the relation between *Second Piola-Kirchoff Stress S* and *Green-Lagrange strain E* by a elastic stiffness matrix $\mathbb{C}$, whose entries are determined by the type of crystal lattice.

$$S = \mathbb{C} : E \tag{2.4}$$

This elastic stiffness matrix $\mathbb{C}$ is a tensor of rank 4 and, due to symmetries there can be a maximum of 21 independent constants. The number of independent constants for different types of crystals is shown in the table 2.1

Plasticity: Phenomenological Crystal Plasticity Plasticity law should provide plastic velocity gradient L_p for a given Mandel Stress M_p, typically involving a set of internal state variables. In polycrystalline materials plasticity occurs on well-defined deformation systems according to the lattice structure. The sum total from these individual systems constitute the total plasticity given by equation

2.5

$$\mathbf{L}_\mathrm{p} = \sum_\alpha \dot{\gamma}^\alpha \underbrace{\left(\mathbf{s}_\mathrm{S}^\alpha \otimes \mathbf{n}_\mathrm{S}^\alpha\right)}_{=:\mathbf{P}_\mathrm{Schmid}^\alpha} \tag{2.5}$$

with *s* and *n* being the directions and normals to the slip planes. The driving force τ^α is generally given by *SCHMID'S LAW* equation 2.6

$$\tau^\alpha = \mathbf{M}_\mathrm{p} \cdot \mathbf{P}_\mathrm{Schmid}^\alpha \tag{2.6}$$

Phenomenological Crystal plasticity is a widely used simple model introduced by Hutchinson for FCC and extended to twinning by Kalidindi. It's modified form is implemented for all crystal structures in DAMASK. The plastic component in internal variables is characterized by resistances on slip and twin systems. This is given by the equation 2.7

$$\begin{aligned}\dot{\xi}^\alpha = {} & h_0^{\mathrm{s-s}} \left(1 + c_1 \left(f_\mathrm{tw}^\mathrm{tot}\right)^{c_2}\right) \left(1 + h_\mathrm{int}^\alpha\right) \times \\ & \sum_{\alpha'=1}^{N_\mathrm{s}} \left|\dot{\gamma}^{\alpha'}\right| \left|1 - \frac{\xi^{\alpha'}}{\xi_\infty^{\alpha'}}\right|^a \mathrm{sgn}\left(1 - \frac{\xi^{\alpha'}}{\xi_\infty^{\alpha'}}\right) h^{\alpha\alpha'} + \sum_{\beta'=1}^{N_\mathrm{tw}} \dot{\gamma}^{\beta'} h^{\alpha\beta'}\end{aligned} \tag{2.7}$$

here $f_\mathrm{tw}^\mathrm{tot}$ is the twin volume fraction and *h* are slip-slip and slip-twin interaction parameters and the rest are model fitting parameters. The resistances on twin systems evolve in a similar way given by equation 2.8

$$\dot{\xi}^\beta = h_0^{\mathrm{tw-s}} \left(\sum_{\alpha=1}^{N_\mathrm{s}} \left|\gamma^\alpha\right|\right)^{c_3} \sum_{\alpha'=1}^{N_\mathrm{s}} \left|\dot{\gamma}^{\alpha'}\right| h^{\beta\alpha'} + h_0^{\mathrm{tw-tw}} \left(f_\mathrm{tw}^\mathrm{tot}\right)^{c_4} \sum_{\beta'=1}^{N_\mathrm{tw}} \dot{\gamma}^{\beta'} h^{\beta\beta'} \tag{2.8}$$

The evolution of shear on each slip system is given by equation 2.9

$$\dot{\gamma}^\alpha = \left(1 - f_\mathrm{tw}^\mathrm{tot}\right) \dot{\gamma}_0^\alpha \left|\frac{\tau^\alpha}{\xi^\alpha}\right|^n \mathrm{sgn}\left(\tau^\alpha\right) \tag{2.9}$$

Thermal The thermal models are implemented to take in account strains due to thermal expansion, heat generation and transport. The transport of heat in solids predominantly takes place due to conduction. *FOURIER'S LAW* of heat conduction given by equation 2.10 uses an anisotropic thermal conduction tensor *K* to relate flux and temperature gradient.

$$f_T = -KGradT \tag{2.10}$$

Heat can be generated as dissipation during plastic deformation, given by the equation 2.11

$$f_T = \kappa S.L_p \tag{2.11}$$

κ is the *TAYLOR-QUINNEY FACTOR* which describes the fraction of plastic work dissipated as heat. A user-defined heat source can also be prescribed as equation 2.12

$$f_T(t_n) = c_n \tag{2.12}$$

The values at time t are interpolated linearly between constants c_n

2.3 Machine Learning and Deep Learning

The previous decade saw an unprecedented growth in the application of Machine and Deep learning techniques in almost all fields of science,technology,finance etc supplemented by the change in ideology from "Think and Store Data" to "Store Data and Think". The basic theme of the whole field of Machine learning is the following : A computer is faced with a task and an associated performance measure, and its goal is to improve its performance in this task with experience which comes in the form of examples and data [20]. Deep learning is a special branch of Machine Learning where Neural networks are used. Some basic overview of neural networks is provided below.[21]

2.3.1 Neural Networks

Neural Networks are a specific type of models which are somewhat inspired by working of human brain. Small blocks of information (called Neurons) fire in a specific manner (according to the activation function) with a specific intensity (called weights) in a specified sequence which will capture the behaviour of complex data by optimizing the weights (by using optimizers) to minimize the loss function constructed using the training data.
Information passes from one set of neurons (called a layer which can be thought of as single unit of computation) to another set of neurons. Based on these interconnections between the layers and type of the individual neuron several architectures of neural networks are developed. Many of such architectures are given in figure 2.5.

Activation functions:

Activation function characterizes the behaviour of a neuron. It is a function which manipulates the incoming data into the neuron to produce a output which can be the final output or input to another layer. The activation functions channelize the flow of data leading from input to output via hidden layers. Some typical activation functions used in Neural Networks are given in the figure 2.6 along with corresponding formulae.

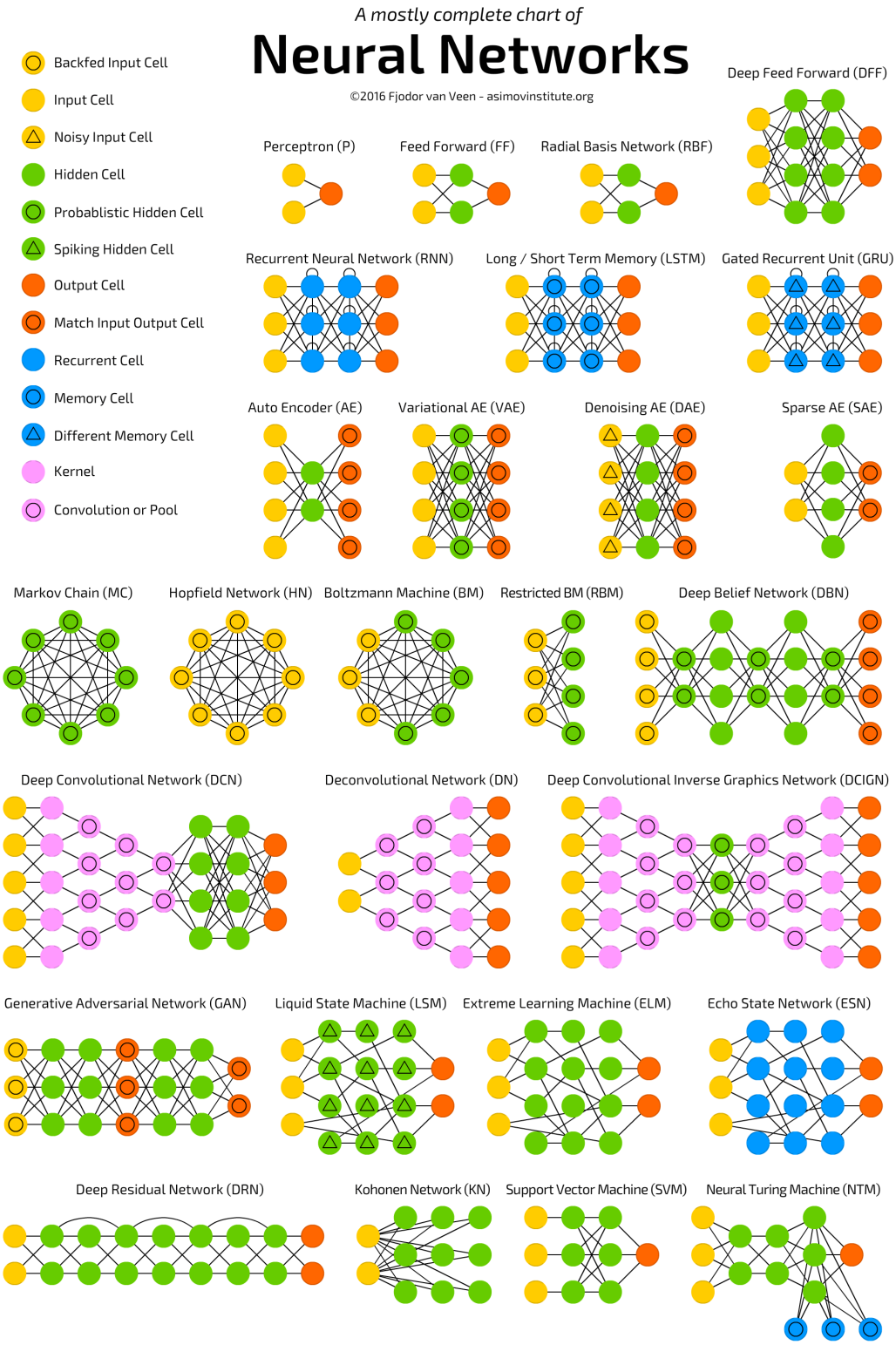


Figure 2.5: Neural Network Architectures [5]

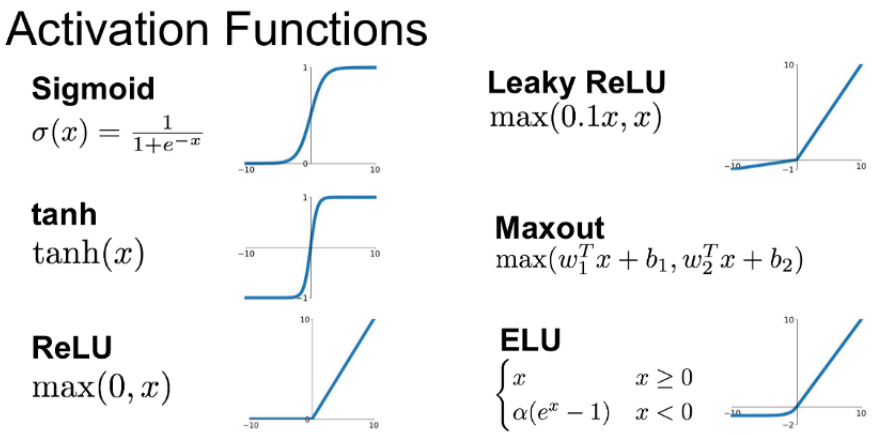


Figure 2.6: Activation functions [6]

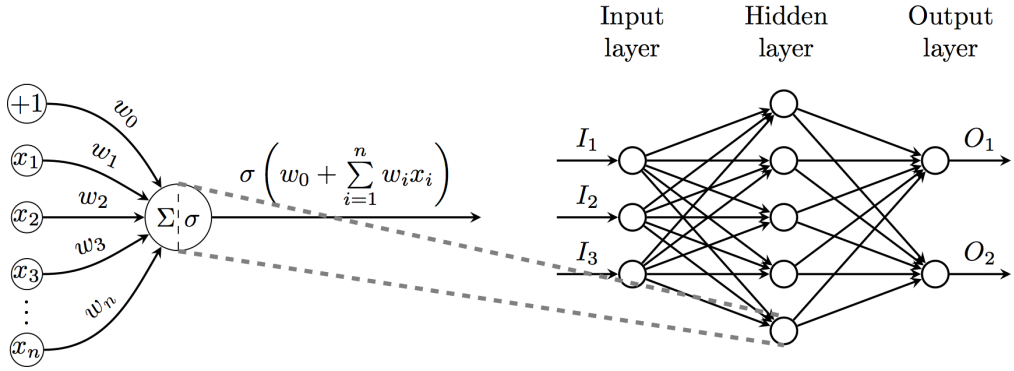


Figure 2.7: A simple Neural Network [7]

A simple Neural Network:

Based on the above definitions, a simple Neural Network, generally called as an Artificial Neural Network(ANN) or a Multi-layer perceptron along with it's mathematical formulation is shown in the figure 2.7.

Loss function:

Loss function is the performance measure in the Neural Networks. Weights are altered by the optimization algorithms to minimize this loss functional. Generally, the loss function penalizes the output of the Neural network against the known training data points.

General mathematical formulation is as follows, Let the training data (known outputs for inputs) be given as input-output pairs as equation 2.13 :

$$\{(x_1, y_1), \ldots, (x_N, y_N)\} \subseteq \mathscr{X} \times \tag{2.13}$$

Where $\mathscr{X}$ is the input sample space and $\mathscr{Y}$ is the output sample space. The goal of a Neural Network (or any ML model for that matter) is to find a function which maps from $\mathscr{X}$ to $\mathscr{Y}$ and gives predictions ($\hat{y}$) according to training data, i.e equation 2.14

$$f : x \mapsto \hat{y} \tag{2.14}$$

This calculation of f is sometimes called "Forward propagation"(as in we are moving forward direction in a Neural Network) and computing the loss and gradients is referred as "Backward propagation"(as in we are moving in the reverse direction in Neural Network calculating the error). As a performance measure(to see whether f is doing a good job) loss function is defined as equation 2.15:

$$L : \mathscr{Y} \times \ \to \mathbb{R} \tag{2.15}$$

Here L measures how close actual y is to the predicted output $\hat{y}$

1. squared loss:
$$L(y, \hat{y}) = |y - \hat{y}|^2 \text{ for } = \mathbb{R}$$
2. zero one loss:
$$L(y, \hat{y}) = \mathbb{1}_y(\hat{y}) \text{ for arbitrary}$$
3. cross-entropy loss:
$$L(y, \hat{y}) = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \text{ for } = [0, 1]$$

Optimizers:

Weights of a Neural Network need to be changed to decrease the loss function. Optimizers govern how these weights are changed to get optimum set of weights which minimize the loss. *Gradient based optimizers* are based on the fact that the gradient of a function gives the direction of it's decrease.
The general theme of all gradient-based optimizers is : Compute the gradients and manipulate these gradients(e.g multiply with negative of learning rate) and add them to present set of parameters. How do we manipulate the gradients to make them more efficient forms the factor for different types of algorithms. This is elucidated in the figure 2.8

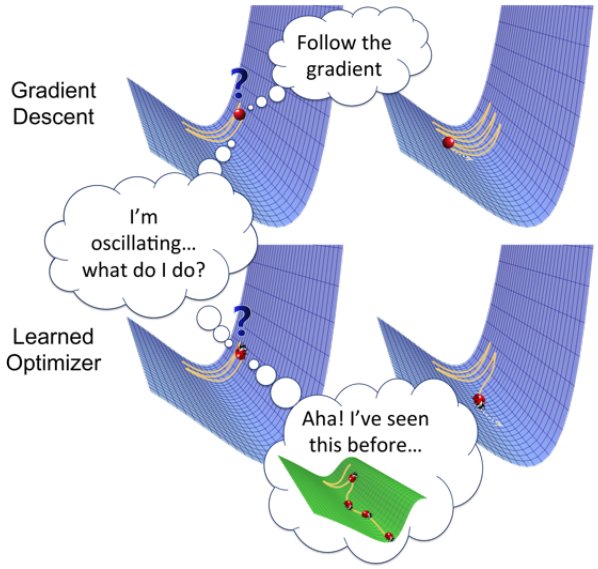


Figure 2.8: General theme of Gradient based Optimizers [8]

2.3.2 Recurrent Neural Networks (RNNs) Long Short Term Memory (LSTM) Networks

Though very versatile and simple, the limitations of ANNs are clearly seen if the complexity of the data at hand is also increased. Sequential Data, i.e data whose output not only depends on current input but also on previous occurences. For example, both sense and meaning and the sentence depends on the order of occurence of individual words, therefore the current output of a speech recognizer, voice detector like Alexa should consider the previous data as well. Another example might be when capturing the stress strain behaviour of non-linear material model, the current stress state not only depends on current strain state but also on the history of occurence of strains. Neural networks need to be equipped with internal variables which capture the history dependence of the sequential data to give meaningful outputs. This is the main reason that led to the development of Recurrent neural networks(RNNs)

Life can only be understood looking backward. It must be lived forward.

-The Curious Case of Benjamin Button

The basic theme of development of RNNs is to acheive this "having a memory of the past" and base it to have accurate/sensible prediction of future. A basic architecture of RNN will look something like 2.9. x is the input state, o is the

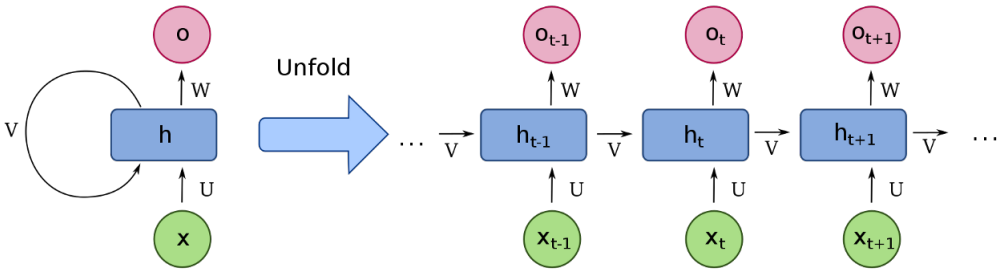


Figure 2.9: Basic RNN architecture from Wikipedia

output and h is the hidden state. U,V,W are the weights applied respectively. The recursive pointing structure forms the crux of RNN. This characterizes few things:

1. Varied sizes of inputs and outputs
2. The hidden state represents the past knowledge, this serves as input to the next step, and is dervied from previous hidden state and current input
3. This hidden state bridges both input and output, past and present

Long Short Term Memiry Networks LSTMs:

The limitations of basic RNNs discussed above will be seen easily when dealing with higher dimensional data. RNN cannot remember too much because of it's less parameters and it will face an overload. For these types of complex situations, RNNs now need to be intelligent enough to output somethings, emphasize somethings and forget somethings. This is the basis of Long Short Term memory networks (simlply LSTMs) which employ an input gate, output gate and forget gate to acheive these functionalities and also sigmoid function to show the amount of information being passed on. This is shown in figure ??

Gated Recurrent Unit(GRU) is seen as a simpler version of LSTMs . Here the three gates of LSTM are replaced with only two gates for reset and update. The intermediate gate for emphasis is removed and both update and output are done by update gate. This is illustrated in the figure

Bi-Directional LSTM:

In LSTM, there is input flow forwards i.e in one direction. When the data

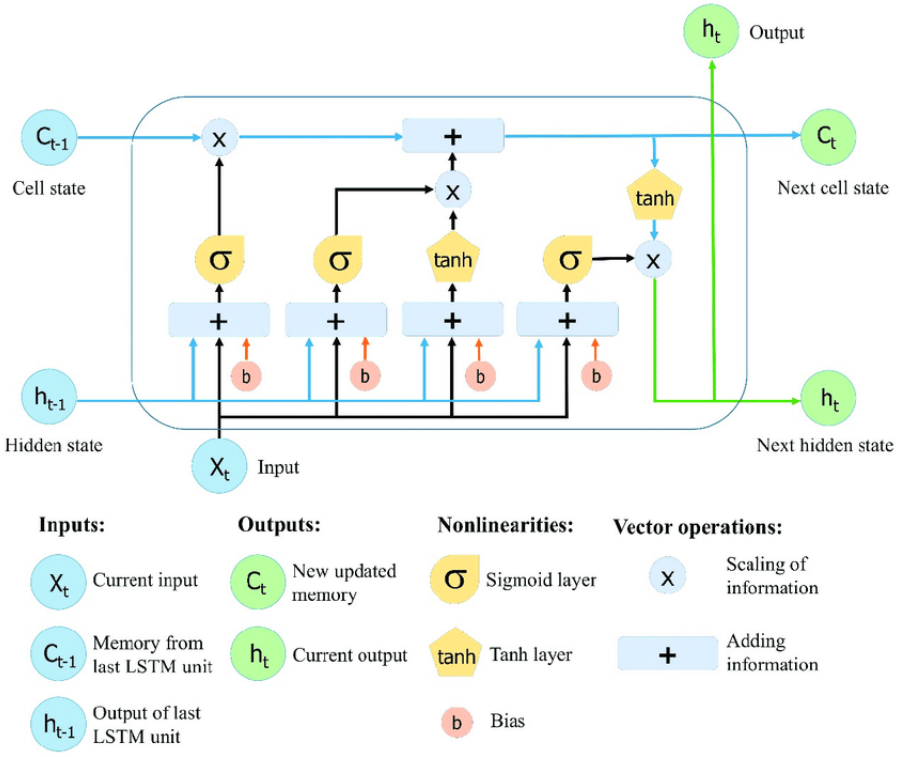


Figure 2.10: Basic LSTM Architecture Source:"A trip down long-short memory lane" by Peter Velickovic (https://www.cl.cam.ac.uk/~pv273/slides/LSTMslides.pdf)

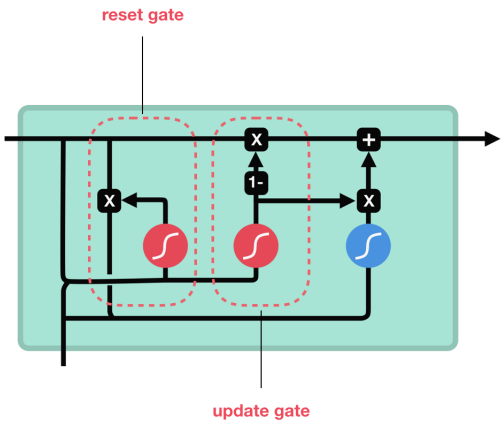


Figure 2.11: Basic GRU unit Source:https://towardsdatascience.com/understanding-rnns-lstms-and-grus-ed62eb584d90

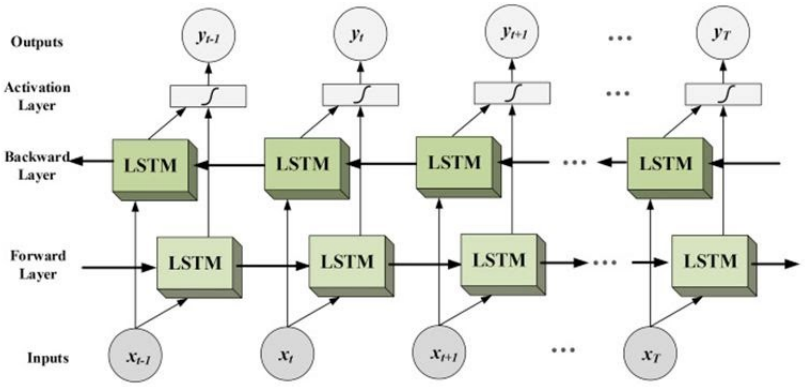


Figure 2.12: Basic Bi-LSTM Architecture https://towardsdatascience.com/understanding-rnns-lstms-and-grus-ed62eb584d90

is sparse, there is a need to capture the underlying nature of occurences both from forwards and backwards for effective usage of sparse data in training.This is facilitated by Bidirectional LSTMs (Bi-LSTMs). With the advent of packages like Tensorflow, PyTorch etc. It has become very simple , elegant to use these architectures right away. In this thesis, *Tensorflow* is used to implement LSTM model neural network.

2.3.3 Implementation in current problem:

Micromechanical simulations for larger sizes of Volume Elements(VEs) becomes expensive and time consuming. An Elasto-Plastic simulation of VE with 140 units (1 Voxel per unit) takes around 10GB of data and arouud 8-10 hours to complete the simulation with 10 cores used parallelly. In this thesis, deep learning models for sequence learning and prediction are used to save both time and computational power by implementing a light sequence predictor which trains for lower sizes which are relatively faster and inexpensive to obtain data and used to predict the property for higher sizes. The idea is to leverage the fact that *for lower sizes of VEs the property has more uncertainity (more spread about mean) and this spread decreases with increasing size of the VEs* and this almost disappears or is within the acceptable limits for larger sizes. This can be used to generate a *sequence* of property data for increasing size of VEs. The difference between each consecutive size of VE will be less than the previous ones due to the decrease in uncertainity. This data when trained with a proper sequential learning neural network (here, LSTM), with each prediction at higher size, the difference

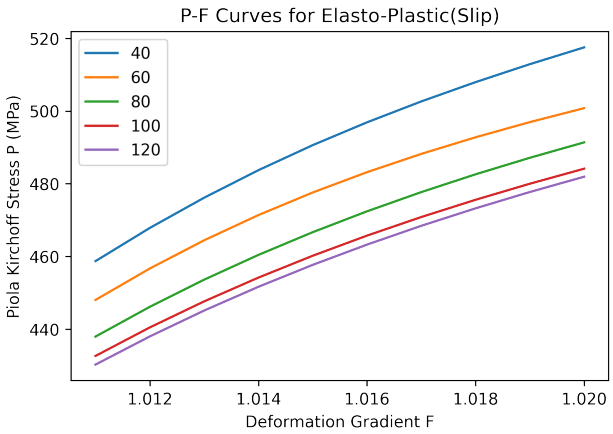


Figure 2.13: F-P curves for Elasto-Plastic simulations to illustrate the convergence behaviour at increasing sizes. It can be seen that the distance between subsequent curves is decreasing with increasing size

of proprty between consecutive size will be decreasing until it is minimized within acceptable limits. This means that our solution is converged at a partculat size and this size will be the acceptable size of our VE for that particulat material model and against that particular property.

This is shown in the figure 2.13, it can be observed that the curves have distance between them decreasing and the predictions will be made easily until required convergence is met. The training data is prepared from this pattern as shown in the table 2.2. The data is arranged as pairs which output the occurence at next size. This gives the deep learning model a good idea of the underlying correlation. Once the training is done, prediction at higher size can be made by giving the data at previous two sizes as input and this prediction is used as input to the next prediction and this is repeated until required convergence is acheived.

Here in the example provided for Elasto-Plastic model, VEs upto size 120units are used for training . After training , a prediction for 140 units is made by giving data at 100 and 120 units as input. The next prediction of 160 units can be made by using the current prediction of 140units and this is continued until the convergence is reached.

The LSTM neural network architecture used to do this task is given in the figure 2.14 . The data is trained against this model. The measure of error is *Mean Squared Error*. Optimizer used is *Adam* with learning rate as 0.5e-3 and rest of the parameters are default values.

The performance and predictions of neural network will be discussed in the

Input	Output
Training	
[P_40,P_60]	[P_80]
[P_60,P_80]	[P_100]
[P_80,P_100]	[P_120]
Prediction	
[P_100,P_120]	[P_140]
[P_120,P_140]	[P_160]

Table 2.2: Strategy of sequence generation for training at lower sizes and prediction at higher sizes, the term *P_(side_length_VE)* is the Property which the neural network is trained at that particular size

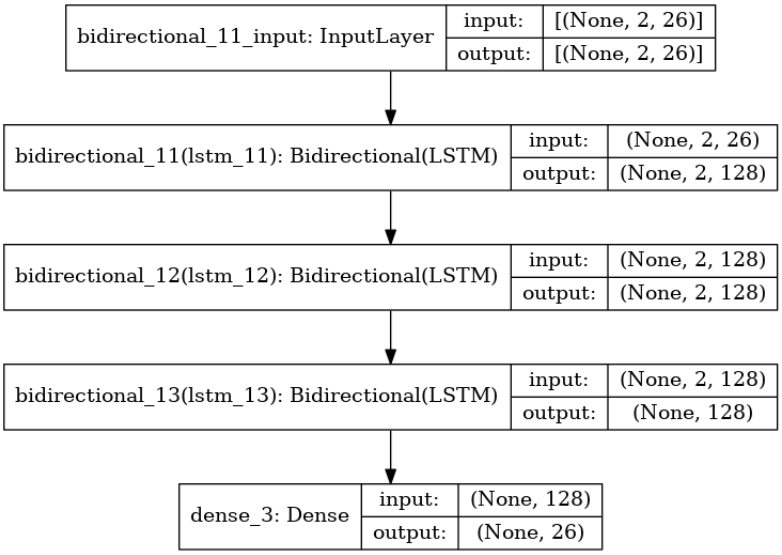


Figure 2.14: Architecture of the neural network*

Phase	γ-TiAl	$\alpha_2 - \mathrm{Ti_3Al}$
System	fct	hcp
Lattice parameters		
a	0.400 nm	0.577 nm
c	0.407 nm	0.465 nm

Table 2.3: Some characteristics of the most important phases in γbased alloys (from Pearson's Handbook of Crystallographic Data for Intermetallic Phases, Villars and Calvert 1991

subsequent chapters

2.4 Titanium Aluminide

The material which is considered in the context of thesis is Titanium Aluminide. There are a spectrum of Titanium Aluminides based on the amount of phases present. These phases in the Ti-Al system have been widely recognized as a viable basis for the creation of innovative lightweight alloys for high-temperature structural applications since the 1970s. Since then, the fundamental driving reason behind extensive research has been to partially replace considerably denser superalloys or less temperature resistant titanium alloys in gas turbines. TiAl based LPT blades are already in use since 2001. Ti-based alloy (minor Al) has been used in the engine parts in other military application for nearly 50 years. In the field of aeronautics, Intermetallic TiAl are being considered for turbine blades. In blade applications, not only can direct weight savings be achieved by using lighter components, but indirect weight savings can also be achieved by reducing the centrifugal force acting on the disks upon which the turbine blades are mounted.[10]

The phase diagram of Titanium Aluminide is shown in the figure 2.15. The type which is considered here has two phases : γ-TiAl and $\alpha_2 - \mathrm{Ti_3Al}$, often mentioned as Gamma and Alpha2. The percentage of Alpha2 phase is around 35%. The important characteristics of these phases are given in the table 2.3 and the crystal structures are given in the figure 2.16

The properties of these materials, like elastic constants and inelastic parameters are mentioned in the property based RVE section wherever a particular type of constitutive law is used. There are these two phases in the polycrystalline RVEs throughout this work. The grains are oriented randomly.

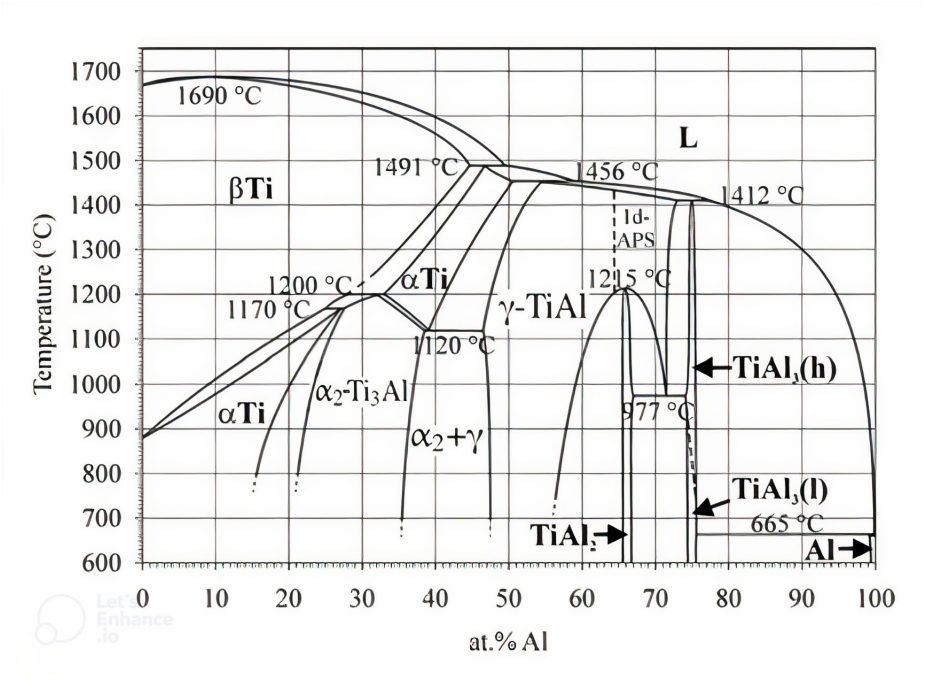


Figure 2.15: Binary Ti-Al phase diagram [9]

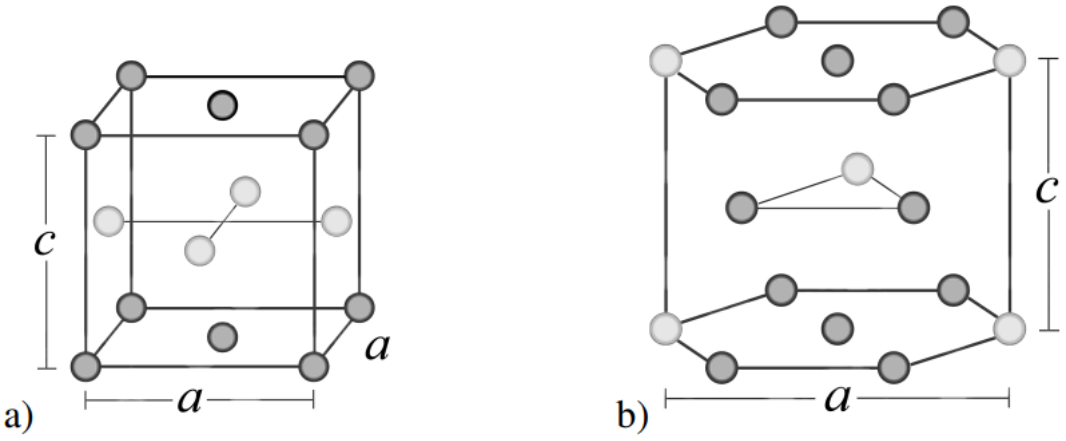


Figure 2.16: Crystal structures of a) γ-TiAl ; b) $\alpha_2 - \mathrm{Ti_3Al}$ dark spheres depict Ti atom positions. [10]

Chapter 3

Results and Discussions

3.1 Comparision of Microstructural Descriptors

The generated Statistically Equivalent VEs are compared using different microstructural descriptors. By comparing the morphological descriptors at different sizes and realizations, we can get a good overview of how similar or how different the microstructures are. Any uncertainties which will be seen in property based simulations will then be attributed only to the size and therefore the concept of acceptable sized RVE will be more meaningful. Any realization which shows a large deviation from the expected trend is omitted and thus the remaining statistical volume elements can be called as microstructure based statistically equivalent representative volume elements , simply m-SERVES. All the images, entries in graphs and plots have the indication of side length of the SVE in μm.

Visualization of Volume Elements:
Different sizes of statistical volume elements are produced from the pipeline.They are visualized usíng *ParaView[22]* . The IPF magnitudes are shown in the figure 3.1 and the phases are shown in the figure 3.2

Volume Fraction The ratio of volume of a phase in a material to the total volume of all phases is called *Volume Fraction* of that particular phase. Since this is a two phase material, volume fraction of one phase will be enough to quantify the distribution of phases. Volume fraction of the Gamma phase in VE is shown for all realizations of all sizes as a boxplot to have a statistical view of the distribution of the values in the figure 3.3. It is clearly illustrated that the deviation about mean is less than 1% for all the sizes hence representing the similar microstructure.

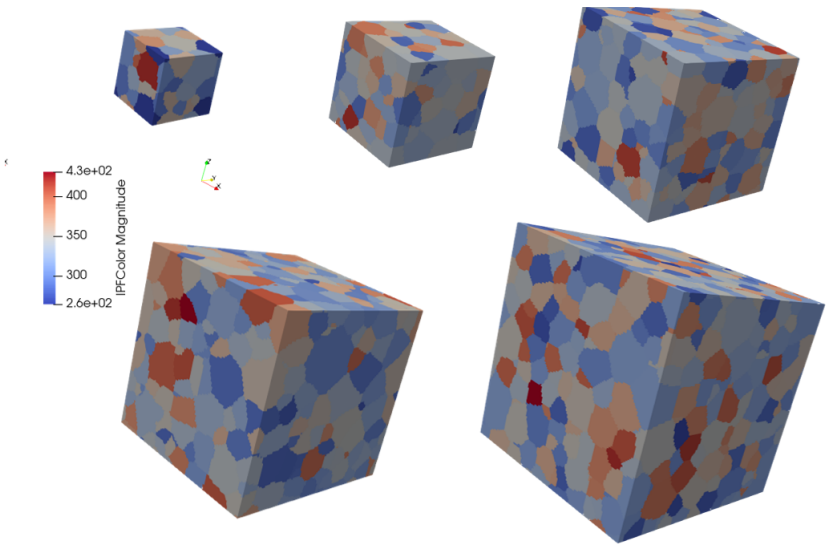


Figure 3.1: IPF Color Magnitudes of Volume Elements of all sizes in the order of side length 40, 60, 80, 100, 120μm.

Mean ESD (equivalent sphere diameter) The grains of the two phases are cosidered to be ellipsoids in the DREAM3D pipeline, therefore the descriptor quantifying the grain size will be equivalent sphere diameter of the grain. The mean of equivalent sphere diameter of all grains in a m-SERVE is also a morphological statistical descriptor. The value at different sizes is shown in the figure 3.4. The MeanESD is almost same for all the m-SERVES.

The difference in the values of reference 2D and generated 3D samples is due to the fact that, ESD is calculated on the basis of area in reference 2D and it is calculated on the basis of volume in generated 3D SVEs. Hence, it is natural to find that mean ESD of reference 2D sample is slightly less than the generated 3D SVEs. This similar behaviour is also seen in dustribution of diameters also.

Distribution of Diameters Different sizes of grains are distributed inside the m-SERVES and their distribution is also morphological descriptor. The figure 3.5 shows that for all sizes the distribution curves look similar. The inference can be made by virtue of this descriptor that these-SERVES represent the same microstructure .

Orientation Distribution The atoms in a grain are arranged in a specific manner and this is quantified by *Orientation* of that particular grain. There are many representations of orientation (Euler Angles, Quaternions, Rotation

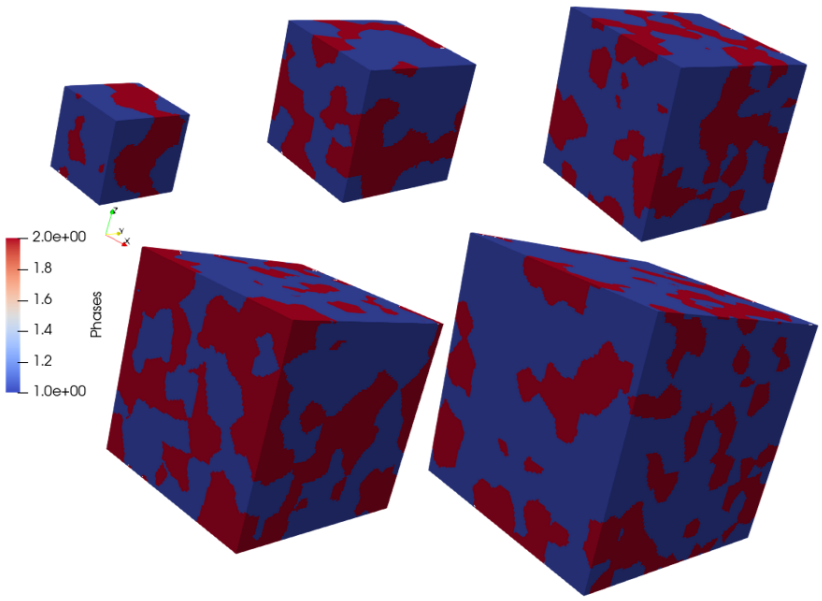


Figure 3.2: Distribution of Phases Volume Elements of all sizes in the order of 40, 60, 80, 100, 120.

Matrices, Axis-Angle pair etc.). The distribution of orientations is given by a *Pole Figure*, which is a collection of stereographic projection of poles representing the orientation of grains. This characterizes the texture of the material, i.e it shows whether the grains are favourably oriented towards a specific direction or randomly oriented. The Orientation distribution of all the grains in the m-SERVES against the original 2D sample will show that with increasing size of m-SERVE the orientation looks more like the original 2D sample as shown in the figure 3.6

Misorientation Distribution Misorientation is the measure of difference in the crystallographic orientation between two adjacent grains or cells. The average angle of misorientation of a grain with all the other grains it is in contact with is stored for all grains of all realizations and it is plotted as a distribution curve to get an idea of misorientation distribution. It is observed that for this particular descriptor, for the sizes from and above 60µm the curves look similar.

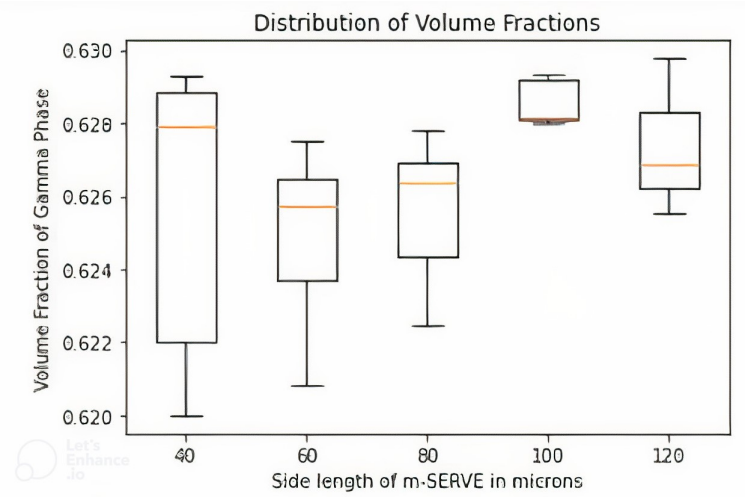


Figure 3.3: Volume Fraction of Gamma phase of all realizations

3.2 Mesh convergence study

The uncertainties due to meshing are also minimized by having a mesh convergennce analysis. The same simulation is made with different mesh sizes to see how mesh is actually affecting the homogenized F-P curve. The curves for different mesh sizes are shown in the figure 3.8. It is clear that the values converge after a mesh size of half of the the side length of the SERVE. To be on a safer side, the meshes considered in this whole thesis are 1 fourier point per voxel. (A SERVE of side length 120µm has a mesh of 120x120x120).

3.3 Property based analysis on different material models

Different realizations of m-SERVES will have different behaviour owing to the uncertainties which are now mainly due to the different spatial distribution of descriptors with increasing size or even within the same size. In other words, the response of different realizations of m-SERVES of the same size will be different for same simulation performed with them. The uncertainties tend to reduce with the increasing size and ideally they would disappear at a particular size and that can be regarded as an acceptable size. In this thesis we are searching for a size where consecutive realizations will have the difference in property within acceptale limit (relative error is less than 1 %). Thus we will get the acceptable size of RVE for which one realization (checked for morphological descriptors, thus

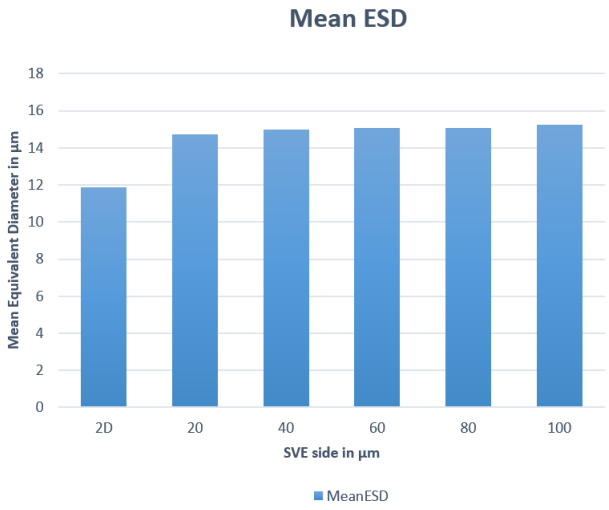


Figure 3.4: Mean equivalent grain diameter of all realizations

m-SERVE) and has 1% of relative error of property (thus p-SERVE). The simulations are carried out in DAMASK. The response or property which is used for investigation is the Deformation Gradient (F) vs Piola Kirchoff Stress(P) curves. To carry out the simulations in DAMASK, the DREAM3D files are converted into DAMASK material and geometry files using the preprocessing tools of DAMASK. The loading is same for all the simulations, tension in 11 direction.

The main aim of this work is also to prove that this size effect becomes more pronounced if the material model employed in the simulation is complex, in other words, for a more complex constitutive model (complex meaning more types of phenomenon are involved in the material behaviour), larger size of SERVE is needed to get minimum uncertainity and map out the corresponding hierarchy of the sizes for different material models. For this, the same simulation is done for different types of material (constitutive) models for the two phases (Gamma and Alpha2) and are probed for a size where the relative error of property is less than 1%.

The DAMASK simulations become more and more computationally expensive and time taking with the increasing size and also the increased complexity of the material model. For example, For size of 120µm, carrying out a tensile simulation for 25 seconds with the elasto-plastic constitutive model and using 10 cores of CPU for parallellization, it takes around 12 hours to complete the simulation and the result file will be around 15GB. This can be evaded by using

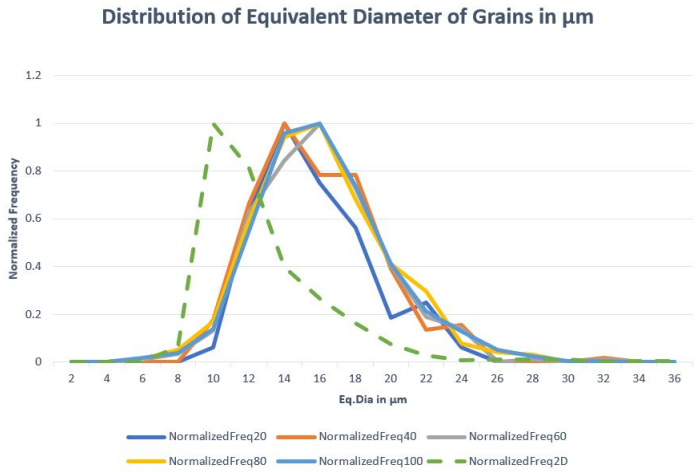


Figure 3.5: Mean equivalent grain diameter distribution of all realizations

the property data at lower sizes (which are computationally less expensive) and use this data to get the predictions and look for convergence using the sequential learning deep learning models. The convergence criteria here will be: If the property between consecutive predictions will be less than 1 % this will be considered the acceptable size of the RVE for that particular material model.

It became very simple and convenient setting up the LSTM model for sequence prediction, thanks to well developed commercial packages like Tensorflow. The dataset being dealt with is sparse, therefore the model has fewer layers and it is faster (takes no more than 6 minutes to train). Once the model is trained, it can be used to make predictions of the larger sizes instantaneously with minimal computational effort until the convergence. Thus it can be considered a clever and efficient usage of deep learning in this scenario for faster and precise decision making. The startegy for training and predictions is already discussed in detail in the previous section. Here, the results are presented.

In the following sections, each type of material model and the corresponding values of parameters used to describe the model are presented for the two phases which are present in the present material of interest, Gamma-Titanium Aluminide which are Gamma and Alpha2 phases.Then the F-P curves are shown (the whole and/or zoomed views for better visualization). For complex models, deep learning model and sequence used to train the model are given.The predictions made and the accuracy of the predictions are tested by comparing them by an actual simulation carried out for that size. Then, relative error graph for the acceptable

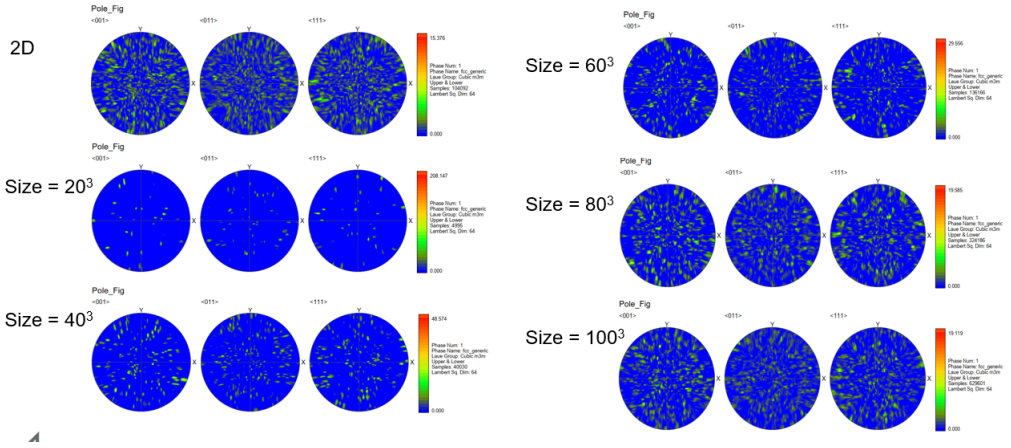


Figure 3.6: Pole figures of all the realizations

Property	Gamma	Alpha2
Lattice	FCC	HCP
C11	165e9	150e9
C13	-	42e9
C33	-	182e9
C44	56e9	28e9
C12	52e9	85e9

Table 3.1: Elastic Properties of two phases of TiAl

size is given and at last, a hierarchy is presented.

3.3.1 Pure Elastic

This is the simplest constitutive model that can be employed for two phases of TiAl. Generalizd Hooke's law is used and crystal structure and the resulting values of elastic constants due to symmetry are provided by DLR. The values are provided in the table 3.1.

All the values are given in SI units unless and unil any other units are specified. With these values used in the material file, the tensile test has been carried out for all the realizations. The F-P curves are drawn for each size with and they are shown in the figure 3.9. Zooming in the last part of the curves will give a good view of the uncertainties given in the figure 3.10. The relative error graph for the SERVES of size 60µm are shown in the figure 3.11 (only the relative error between the realizations which are visually farthest from each other). From this an inference is made that acceptable size of an RVE for of this particular

Figure 3.7: Average Misorientation distribution

Property	Gamma	Alpha2
Number of Slip systems per family	[12] <110>{111} system	[3,3,0,6]:<-1-1.0>{00.1}, <-1-1.0>{1-1.0}, <-1-1.0>{-11.1}, <11.3>{-10.1}
Initial CRSS	112e6	[25e7,50e7,250e6,1250e6]
CRSS at Infinity	300e6	[30e7,60e7,300e6,1600e6]
Reference hardening for slip	1e9	1e9
Stress Exponent	35	30
Reference shear strain slip	0.001	0.001

Table 3.2: Parameters used in Phenomenological Crystal Plastic Law to describe plasticity due to Slip

microstructure with pure elasticity constitutive model is 60µm.

3.3.2 Elasto-Plastic (Slip)

Phenomenological power law is used for description of plasticity due to slip for the both phases. The values are provided in the table 3.2. The F-P curves for the same simulation conducted are given in the whole and zoomed in view in the figures.3.12 and 3.13 respectively.

Here more pronounced uncertainties are seen and the convergence is not seen at 120µm. In this case deep learning can be used to get the predictions for higher sizes. The sequence used for training is shown in the figure 3.14. Using the trained model the predictions for the next 2 sizes (140 and 160 µm) are made. To test the accuracy of the prediction, DAMASK simulations are also made for SERVES of those 2 sizes and they are compared in the figure 3.15. This shows the accuracy of the predictions made by the nerural network. The relative error curve between the last two sizes of SERVES is drawn which satisfies the criteria

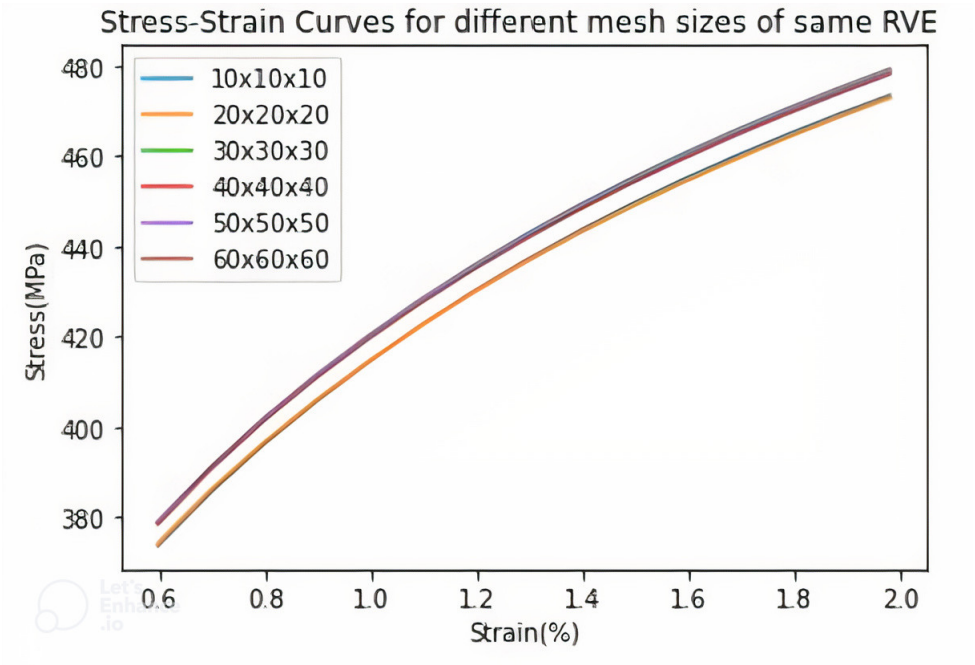


Figure 3.8: Curves for same RVE with different mesh sizes

Property	Gamma
Number of Twin planes	4
stress exponent for twin	15
Reference shear strain twin	0.001
Initia CRSS Twin	40e6
push-up factor for slip saturation	10

Table 3.3: Twin parameters

of less than 1%, shown in the figure 3.16. Hence an inference is made that for this microstructure using this particular constitutive law, the acceptable size of SERVE is 140µm.

3.3.3 Elasto-Plastic (Slip+Twin)

In this case, more complexity is added by introducing twinning in the Gamma phase to already existing elasto-plastic parameters. The twin parameters are shown in the table 3.3. The F-P curves (zoomed view) is shown in the figure 3.17.

Convergence is not seen at 120µm and therefore the deep learning model is

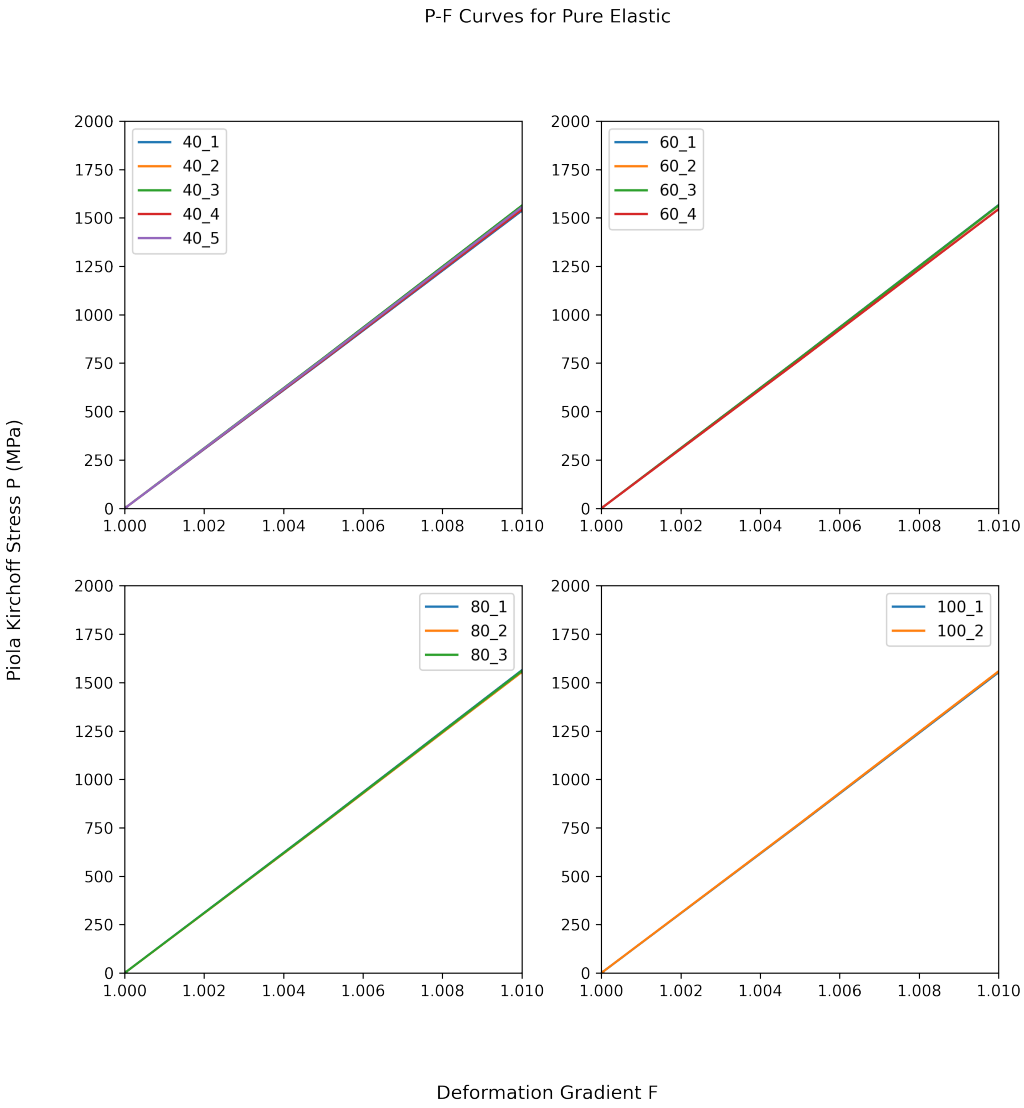


Figure 3.9: Deformation Gradient vs Piola Kirchoff Stress Curves for different sizes of SERVES

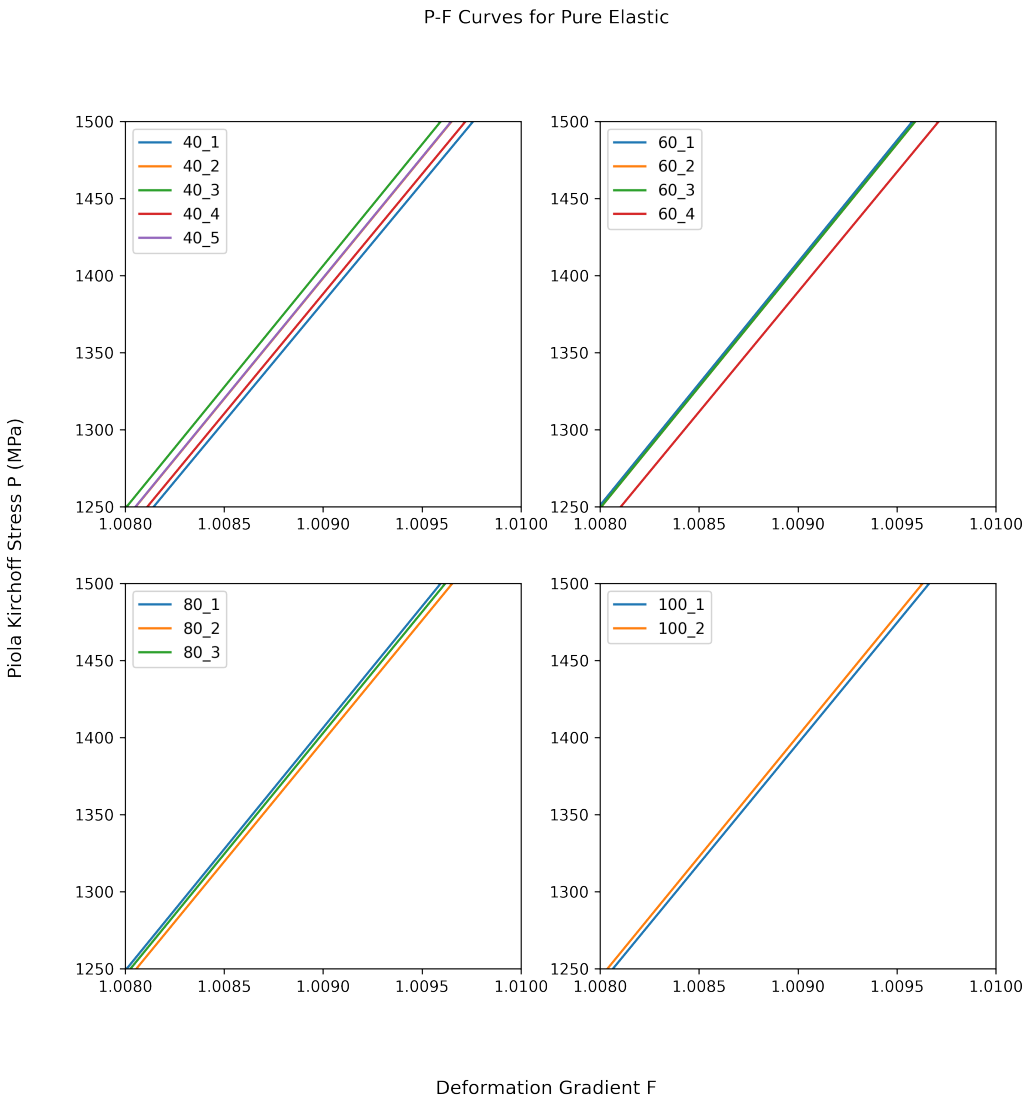


Figure 3.10: Zoomed in viewDeformation Gradient vs Piola Kirchoff Stress Curves for different sizes of SERVES

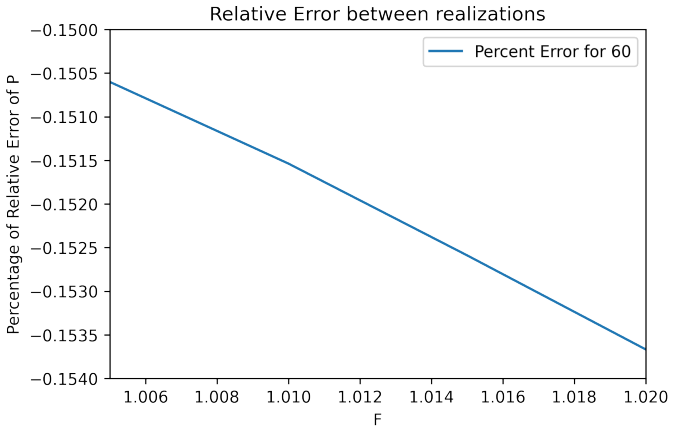


Figure 3.11: Relative error between realizations of size 60µm

trained with the sequence as shown in the figure 3.19. Predictions for the sizes 140 and 160 µm are made and the accuracy of the predictions are compared to the curves obtained from simulations.The comparision is shown in the figure 3.18. The predictions made almost match with the original simulation values proving the reliability of the neural network model in this specific context. The realtive error is plotted between the last two realizations shown in the figure 3.20. The relative error is slightly larger than that for the plasticity only due to slip case (which is an expected behaviour) and it is also slightly greater than 1%. Thus it is safe to infer that *150µm* is the acceptable size of SERVE for this particular microstructure and material model.

3.3.4 Thermo-Elastic

Now in this case, thermal behaviour is modelled to the already existing elastic model by using the constitutive equations of flux, conductivity and thermal expansion, the respective parameters are provided in table 3.4. It should be noted that these values are exaggerated to get appreciable thermal stresses and therefore only be seen from a simulation perspective but not from experimental/original values. In DAMASK , thermal boundary conditions can be given as amount of flux flowing in and out of the system and initial temperature at the start of the simulation. The thermal boundary conditions for this particular problem are an initial temperature of 573K and a constant amount of flux throughout the simulation. The mechanical loading for the first 10 seconds is defined to be zero deformation gradient and undefined stress to observe pure thermal stresses that will develop due to the free stress boundary condition. The next 10 seconds

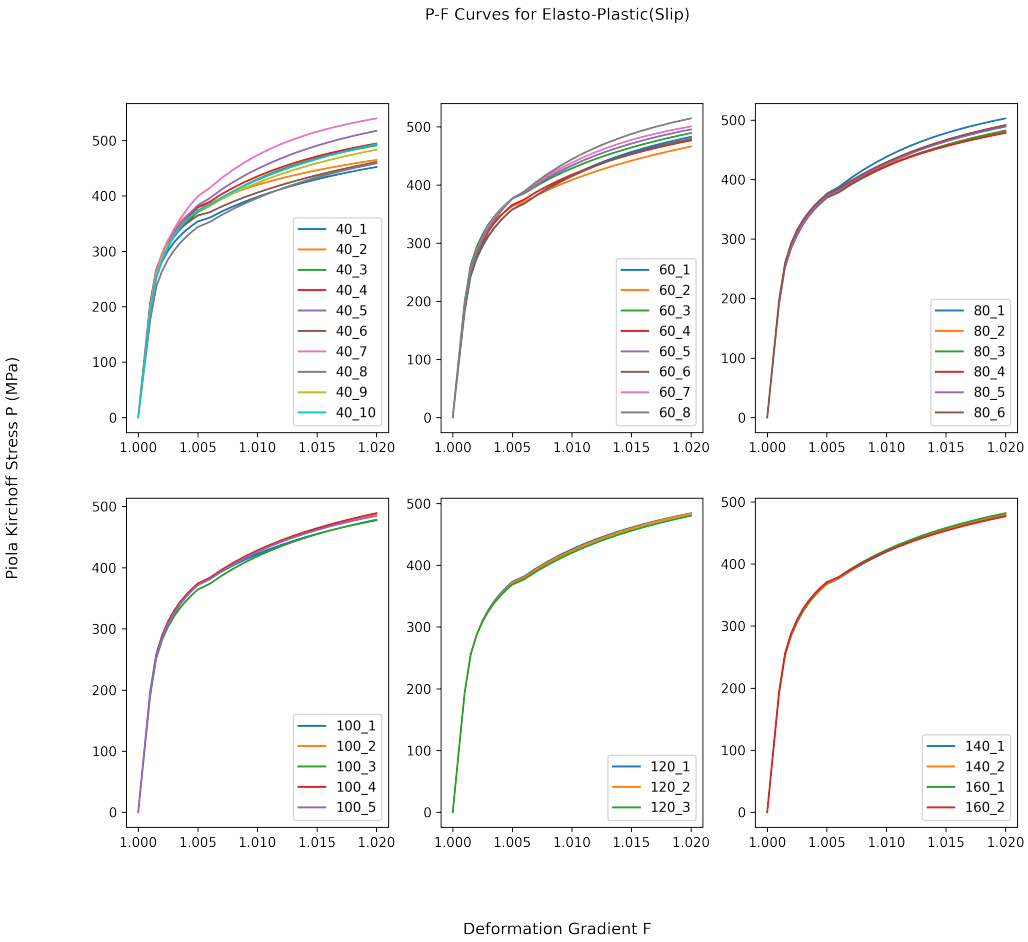


Figure 3.12: Deformation Gradient vs Piola Kirchoff Stress Curves for different sizes of SERVES

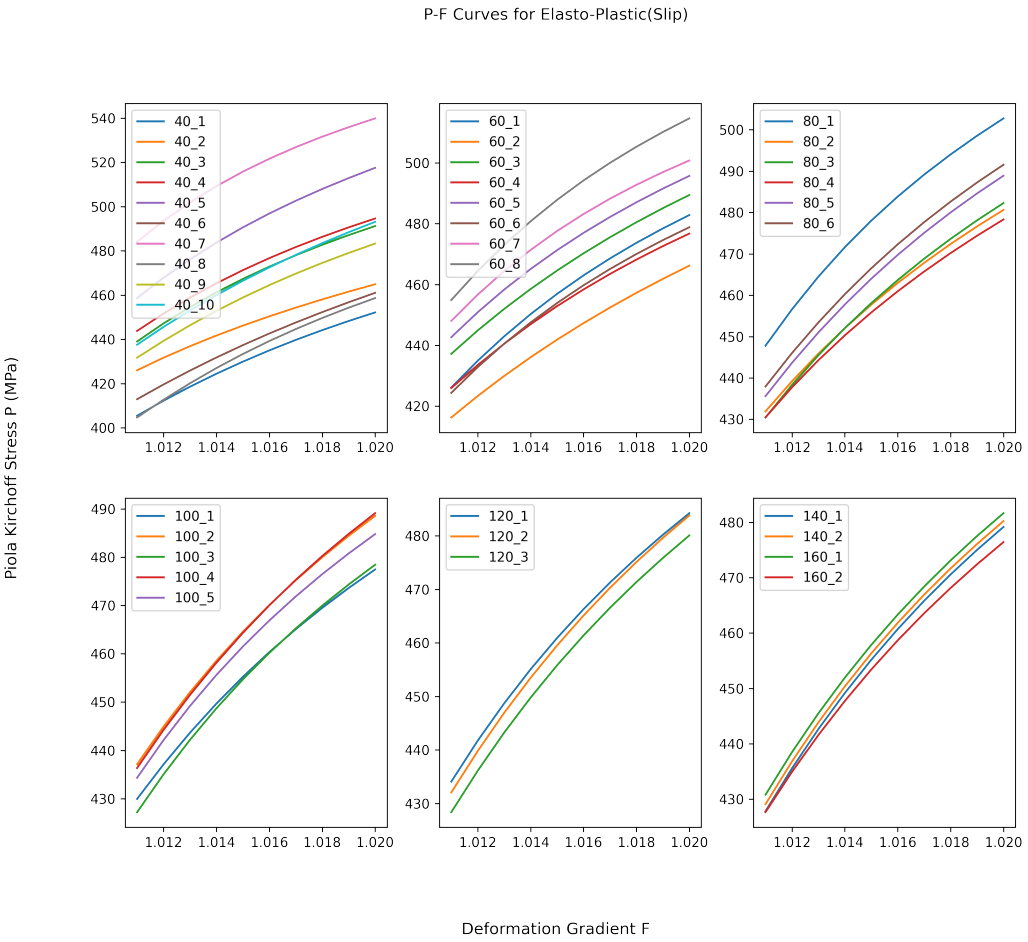


Figure 3.13: Zoomed in viewDeformation Gradient vs Piola Kirchoff Stress Curves for different sizes of SERVES

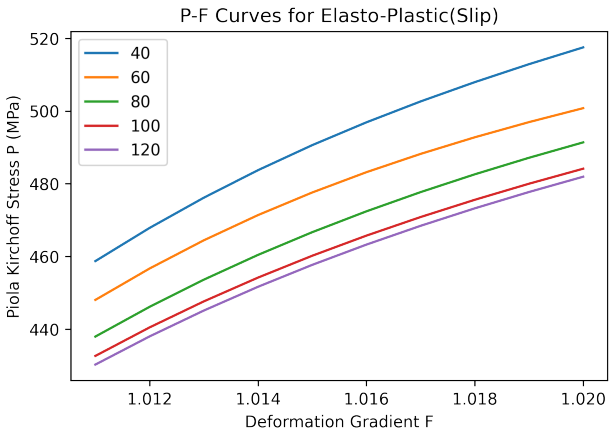


Figure 3.14: The sequence of elasto-plastic curves used for training

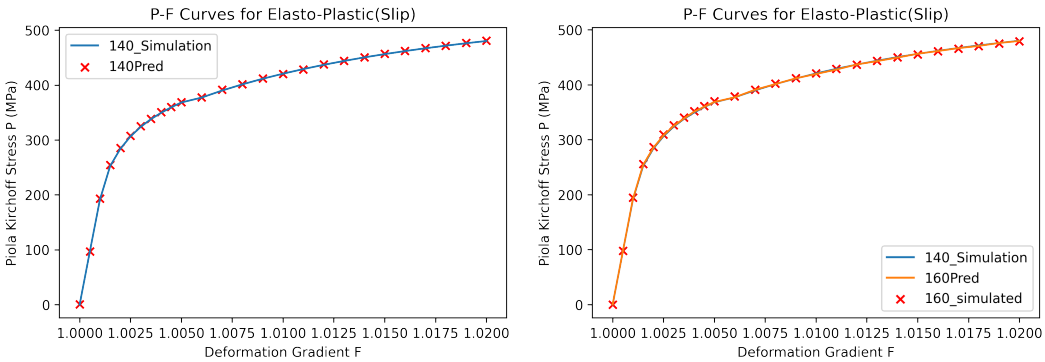


Figure 3.15: Predictions made by neural network compared with simulation results

the load is given as deformation gradient of 0.02. The P-t curves look like 3.21. More deviations are observed at the junction where mechanical loading starts, hence a zoomed in view at this junction is shown in the figure 3.22. The relative error between realizations of size 140µm are shown in the figure 3.23 shows that it is within the acceptable limit. Therefore the acceptable size of SERVE for this material model and microstructure is *140µm*.

3.3.5 Thermo-Elasto-Plastic

In this case, all the models explained above are together implemented. Within the context of the thesis, this is the most complex model implemented. The initial temperature of 273K and external flux is applied gradually (i.e flux is 0 at time 0 and it increases linearly with time upto $1.6Wm^{-}2$). Since the neural

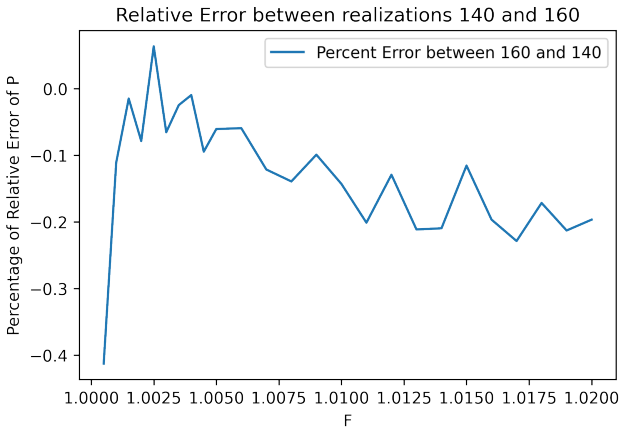


Figure 3.16: Relative error between curves of sizes 140 and 160µm

Property	Gamma	Alpha2
Thermal Expansion Coefficient	9.8e-5/K	3.2e-5/K
Thermal Conductivity	7e5 W/mK	22e5 W/mK
Specific Heat	6.2e8 J/kg.K	6.2e8 J/kg.K
Reference Temperature	273.15K	273.15K
Temperature Dependency of Elastic Constants	Yes	Yes
Temperature Dependency of Conductivity coefficients	Yes	Yes
External Flux	1e13 W/m^2	1e13 W/m^2

Table 3.4: Thermal Parameters

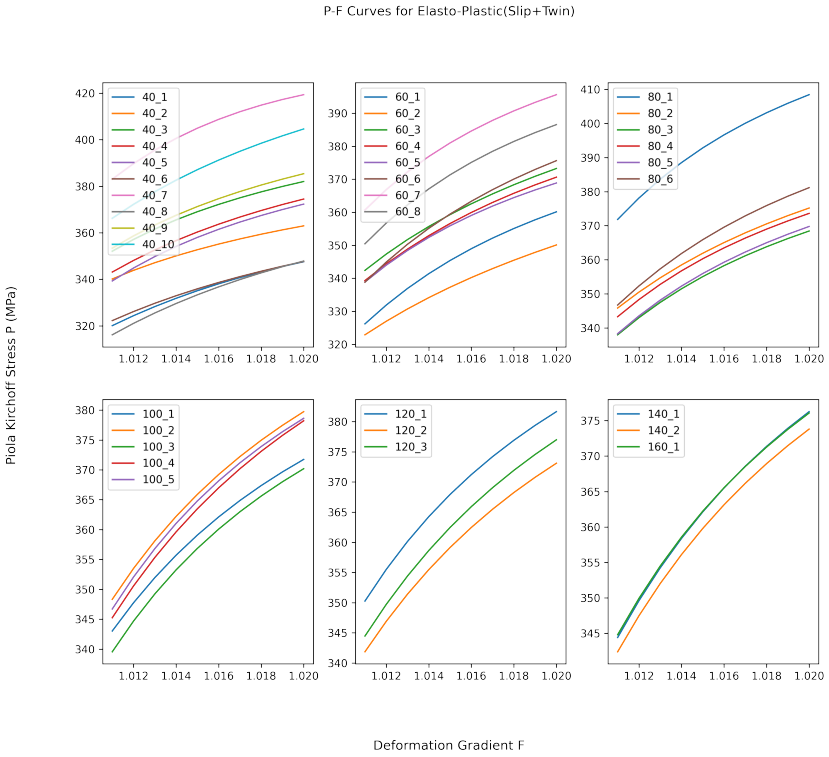


Figure 3.17: Deformation Gradient vs Piola Kirchoff Stress Curves for different sizes of SERVES

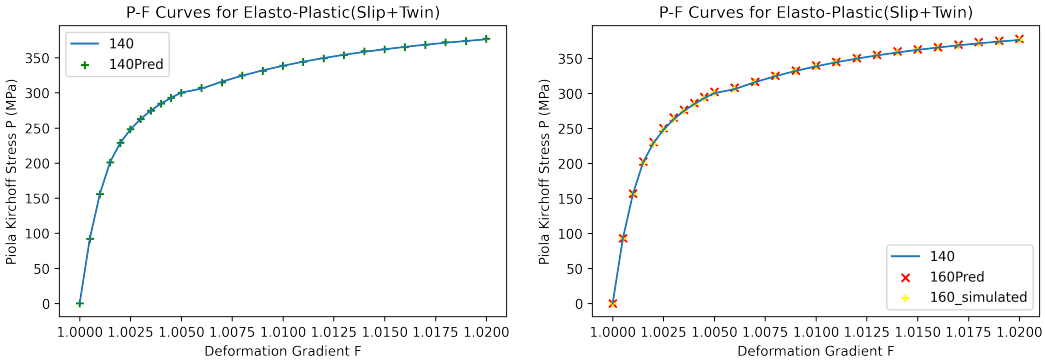


Figure 3.18: Predictions made by neural network compared with simulation results

network is tested with actual simulation results to establish the accuracy and dependability of predictions for the previous cases, here it is used for predicting even more higher sizes (upto 200µm) which will take around days to get results

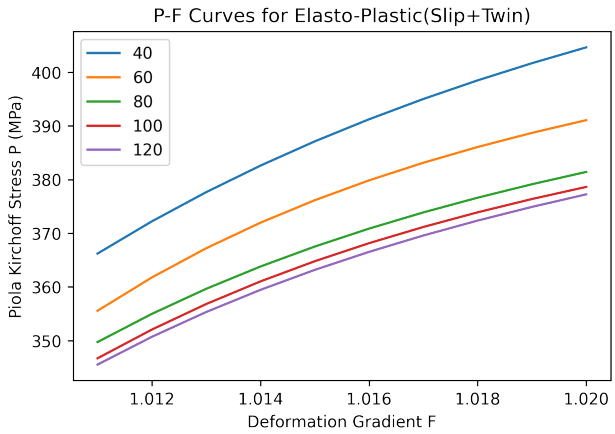


Figure 3.19: The sequence of elasto-plastic curves used for training

from a simulation. The F-P curves are shown in the figure 3.24. Here thermal effect is zero at the start of the curve and it gradually increases towards the end of the simulation.

The pattern of sequence used for training the neural network is given in the figure 3.25. The model is trained with this sequence and predictions are made until a convergence is seen. Here the predictions are made until 200µm. The sequence is plotted with the predictions to have a visualization of the convergence shown in the figure 3.26. It is observed that the distance between successive realizations keeps on decreasing until a point where no gap is seen (convergence). The relative error between last two sizes is given in the figure 3.27. Since it is less than 1 % (our tolerance limit), the acceptable size of RVE for the most complicated model is 180µm.

3.4 Convergence study on local fields

The property considered for p-SERVES in this thesis is only the homogenized P values but not the local field values. Even at the acceptable size the local field values may differ because of the underlying generation process but the overall homogenized values will be same. Thus acceptability condition does not ensure that the local values will also be same but it only ensures that the overall homogenized constitutive response is within the tolerable limits of uncertainty. The violin plots of local field values are given in the figure 3.28. A better visualization is given by hexbins which are shown in the figure 3.29 which show differences in local fields but the homogenized response is same.

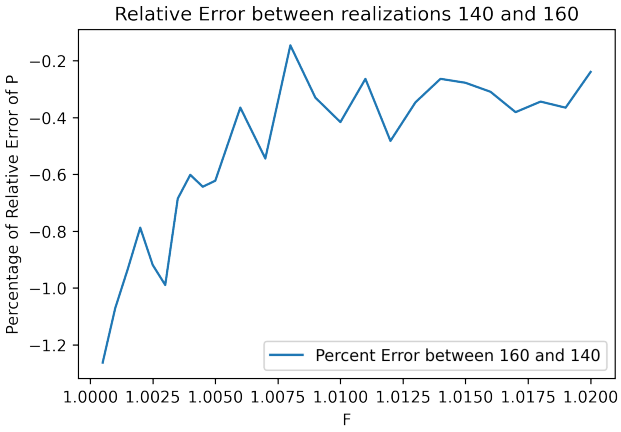


Figure 3.20: Relative error between curves of sizes 140 and 160 µm.

3.5 RVE-map and Hierarchy

With the different material models implemented and having a look at the acceptable sizes for each of them, visualized in the figure 3.30. It shows that, for the same microstructure, if the material model is purely elastic the acceptable SERVE (which was already tested for microstructural descriptors) size for which the uncertainties are minimum is 60µm. This acceptable size increases with added complexity to the material model. For plasticicty due to slip the acceptable size is 140µm and if twinning is also added then the acceptable size also increases to 150 µm. Introducing thermal behaviour for pure elasticity increased the acceptable size to 140µm .Lastly combining elasticity, plasticity and thermal altogether gave an acceptable size of 180µm. Looking at the map, an inference can be made that multi-physics behaviour of material has pronnounced effect on the uncertainties. This can be seen as a strong case to prove the hypothesis that a more complex material model needs a larger size of RVE to have lesser uncertainties.

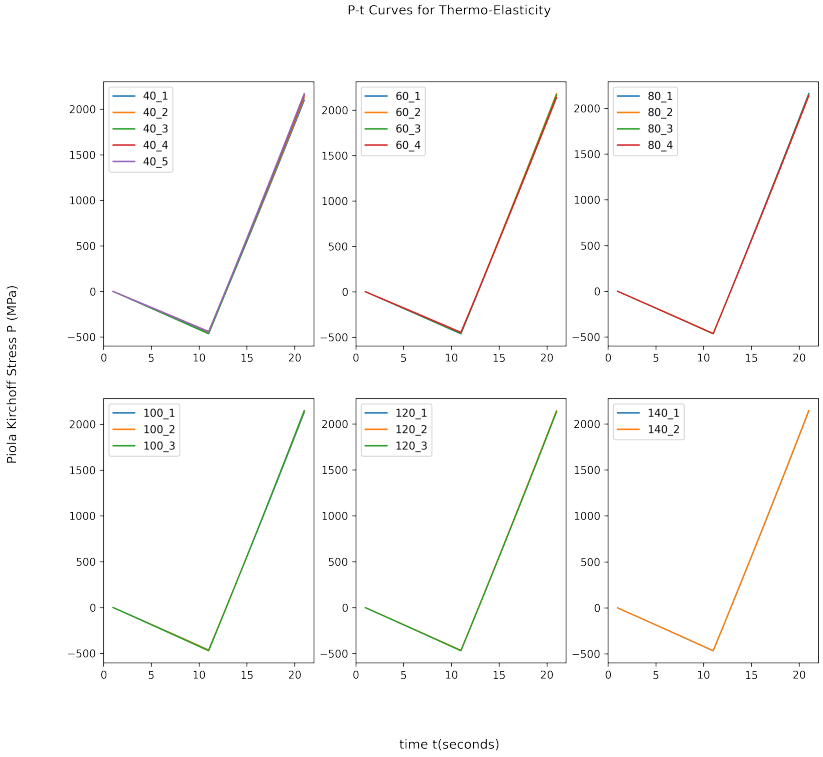


Figure 3.21: Deformation Gradient vs Piola Kirchoff Stress Curves for different sizes of SERVES

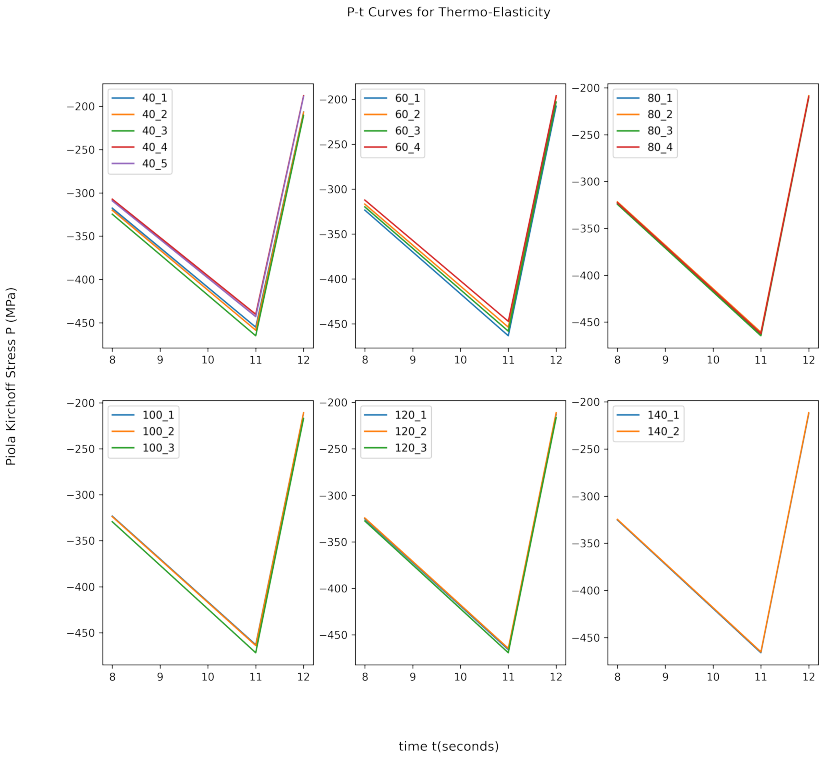


Figure 3.22: Deformation Gradient vs Piola Kirchoff Stress Curves for different sizes of SERVES

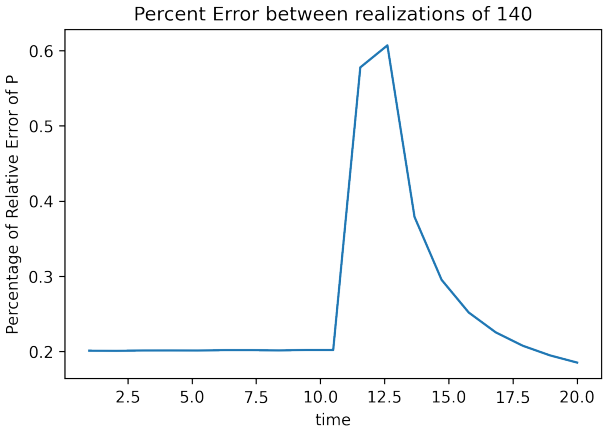


Figure 3.23: Relative error between curves of size 140µm.

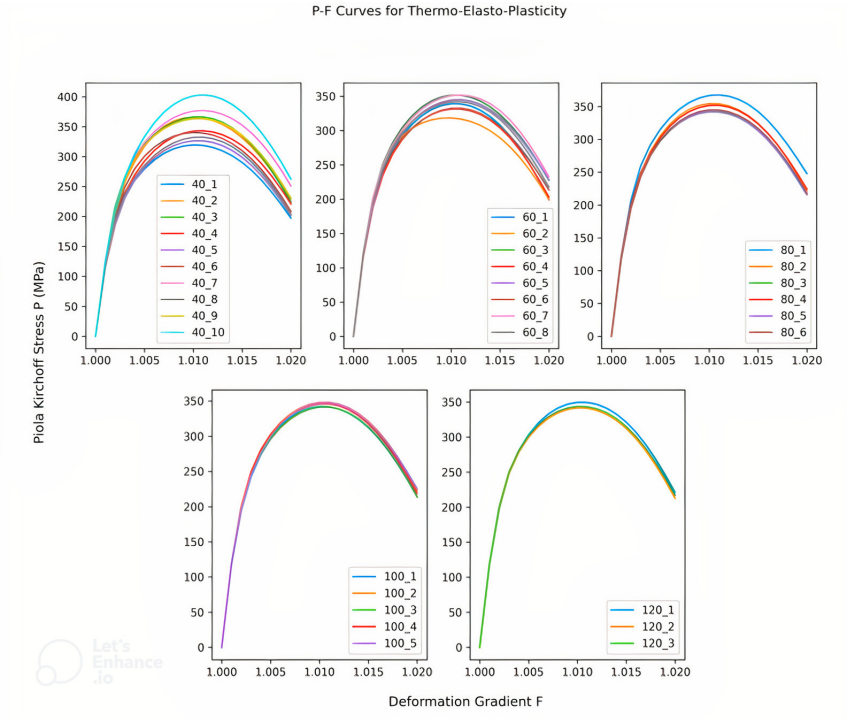


Figure 3.24: Deformation Gradient vs Piola Kirchoff Stress Curves for different sizes of SERVES

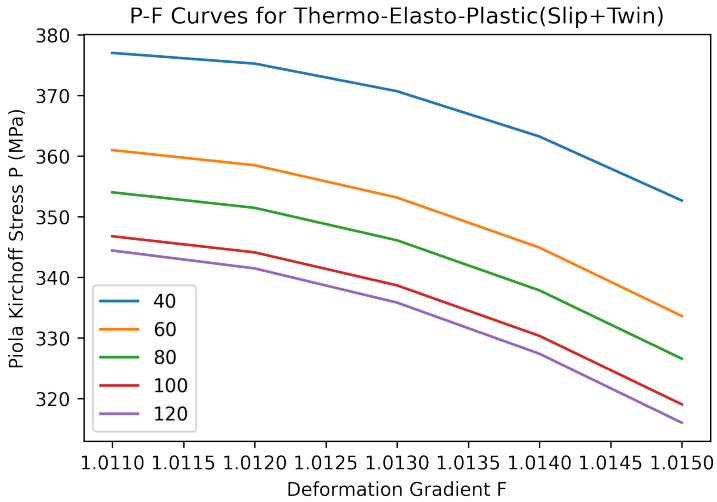


Figure 3.25: The sequence of thermo-elasto-plastic curves used for training

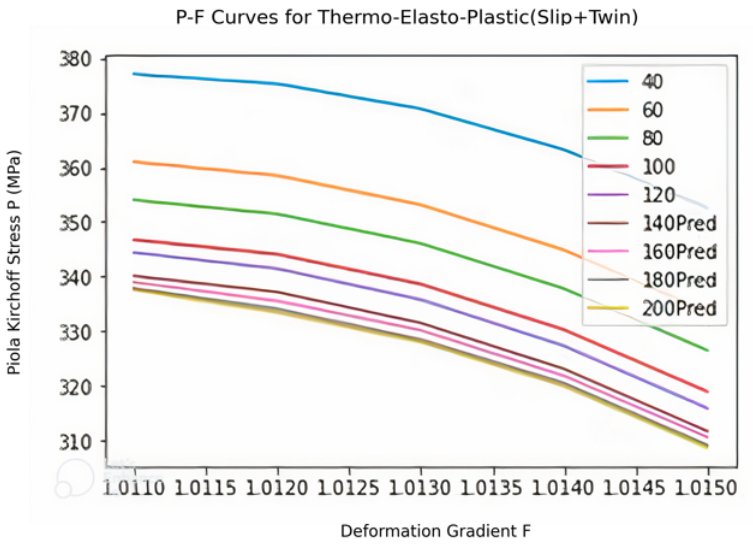


Figure 3.26: Predictions made by the neural network

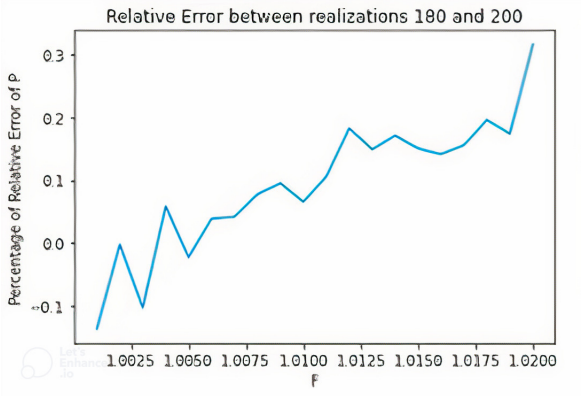


Figure 3.27: Relative error between realizations

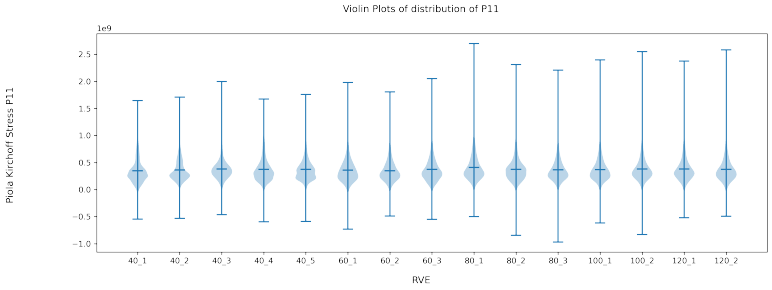


Figure 3.28: Violin plots showing local field values of P, (x axis shows size and number of realization as size_realization)

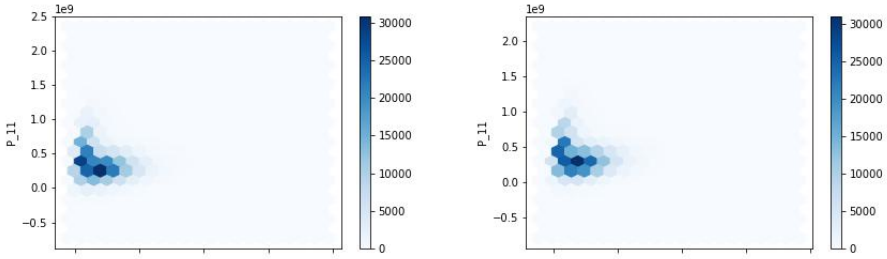


Figure 3.29: Hexbins of realizations of size 140 µm (F_11 vs P_11 Hexbins)

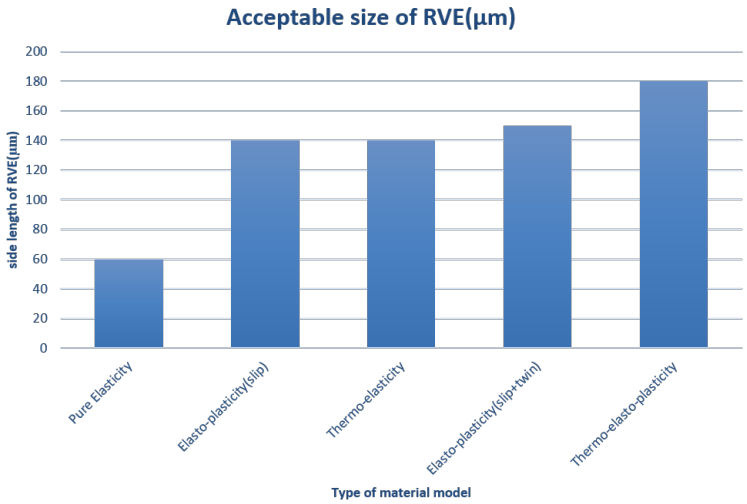


Figure 3.30: Hierarchy of acceptable sizes of SERVES of different material models

Chapter 4

Summary and Outlook

In this thesis, an attempt is made to provide a framework for selecting acceptable size of a Representative Volume Element which can be used for homogenization and/or multi-scale analysis. This is done by combining the evolved concepts of m-SERVE and p-SERVE to have an acceptable size which is both microstructure and property based. Microstructure based analysis provided a comparision of morphological descriptors to ensure the 2D to 3D reconstruction always leads to similar morphology and from this the property analysis provided a study of how the material model impemented makes the size effect on uncertainties pronounced. This can be presented as a strong case to the hypothesis stated earlier that *As complexity of the material model increases, the acceptable size to have minimum uncertainties also increases*, elucidated in the figure 3.30.

Machine Learning and deep learning techniques are used to accelerate the decision making by using their sequnece learning and predicting power, effectively reducing both time and computational power. This is an attempt to make the results of the simulations more reliable by removing size related uncertainity and dependence on complexity of the material model.

The connection between size and uncertainties can be understood by looking from an information perspective. A smaller SVE has less amount of information about the microstructure. A simple model (relatively less complex) of the material will need relatively less amount of information to *represent* the microstructure. For example, if the microstructure is single phase and the material model is elastic isotropic, then only one material point is enough to fully represent the microstructure. Each added property, both in terms of microstructure (phases, grains, orientations, underlying process of generating SVEs) and also material model will need more information to represent the VE. This can be taken as intuition to understand why there is a size dependence.

This thesis stops at thermal-elasto-plastic model, but this can be easily ex-

tended to any type of material model like damage models, dislocation based plasticity models, atomistically informed plasticity models etc. The acceptable size for damage models will be greater than what has been covered in the thesis and it will be useful for actually looking at a simulation closer to real-life scenario.

The use of word *framework* is to expound the fact that it is not limited to cases described here both in terms of morphological descriptors, material models, property under consideration. This thesis is limited to morphological descriptors like Volume Fraction, orientation and misorientation, equivalent diameter. When dealing with other materials like composites etc, there might be different descriptors which are more meaningful to quantify the morphology of that particular type of material. This is one interesting prospect of finding a conformable set of morphological descriptors for each type of microstructure.

The property is the response of the RVE for a particular boundary condition. Only F-P curve is covered in this thesis but there are many other responses to actually compare the convegence behaviour like stiffness constants etc. It is another interesting prospect to compare different properties and their convergence behaviours and then to choose the property which gives highest acceptable size to have a greater reliability of results. It can be assumed that stress-strain curves are more reliable since they already encapsulate the uncertainties coming from properties like stiffness, local fields etc but this assumption also needs to explored in depth.

It is once more restated that the uncertainties here are only size related and it is discussed how the uncertainties related to microstructure and meshing are minimized. The proposed RVE-map is for 1 realization with 1% relative error in the chosen property. This can be extended not only in terms of properties/models, but also in terms of relaxed condition like 5% relative error, 10% relative error etc. This will give a more clear picture in terms of practical applications because every application does not need similar stringent condition.

The use of LSTM neural network for the sequential model learning and for faster decision making is proven to be very fast and reliable. This can be further more optimized by exploring different types of architectures and finetuning hyperparameters. There are many more types of models for sequence learning and prediction like Gated Recurrent Units(GRUs), Tranform nets, Convolutional neural networks(CNNs) etc .

Appendix

Usage of DAMASK

To run a DAMASK simulation, three files are needed :

1. Geometry file (.vti)
2. Material file (material.yaml)
3. Loadcase file (.yaml)

A number of pre-processing tools are available to create and modify these files, some will be discussed here.

The Geometry File Geometry file is a file with ".vti"(VTK Image data format) which contains the information about the geometries present and material IDs. There are two ways to generate a Geometry file for Damask Simulation.

1. From scratch using Voronoi/Laguerre Tessellation
2. Loading from a Dream3d file.

A geometry file can be generated using Voronoi tessellation as.

```
import damask
import numpy as np
size = np.ones(3)*1e-5
cells = [16,16,16]
N_grains = 200
seeds = damask.seeds.from_random(size,N_grains,cells)
grid = damask.Grid.from_Voronoi_tessellation(cells,size,seeds)
grid.save(f'Polycystal_{N_grains}_{cells[0]}x{cells[1]}x{cells[2]}')
```

To get the geometry from a Dream3d file (a file with ".dream3d" extension) use command

```
damask.Grid.load_DREAM3D(file name)
```

To get the corresponding material.yaml , use a similar command given below.

```
damask.MaterialConfig.load_DREAM3D(file name)
```

Homogenization and phase entries are empty and need to be defined separately.(more on these in material.yaml file explanation) NOTE: There is a DAMASK filter in Dream3d, this is deprecated. This filter is no longer valid for DAMASK3. The output ".dream3d" file from pipeline is enough to generate the files required for DAMASK using the preprocessing tools as shown above.

A short note on yaml and hdf5 and their implementation in DAMASK: From DAMASK 3, the whole interface and the way to deal with configuration files has been changed to a large extent. The main change is the implementation of YAML style input configuration files and HDF5 style output files.

YAML is a human-readable data-serialization language. It is commonly used for configuration files and in applications where data is being stored or transmitted. YAML targets many of the same communications applications as Extensible Markup Language but has a minimal syntax.(Wiki)

HDF5 is a data model, library, and file format for storing and managing data. It supports an unlimited variety of datatypes, and is designed for flexible and efficient I/O and for high volume and complex data. HDF5 is portable and is extensible, allowing applications to evolve in their use of HDF5. The HDF5 Technology suite includes tools and applications for managing, manipulating, viewing, and analyzing data in the HDF5 format.

The material.yaml file: This file contains information about Homogenization, Phase and Material. The file can be directly loaded from Dream3d or can be configured from scratch using damask.ConfigMaterial class.
Homogenization:
Homogenization is a dictionary that contains and any number of arbitrary labeled keys. Each entry contains a key *N_constituents* which specifies the number of homogenized constituents. At least the type of the employed homogenization scheme for each active field (mechanical, thermal, or damage) is given; the further configuration details depend on the selected homogenization scheme. Example:

```
homogenization:
  SX:
    N_constituents: 1
    mechanical: {type: pass}
```

SX is the name of the homogenization, Number of constituents is set to 1 and the employed scheme is mechanical of type pass.(No homogenization is taking place) Some other types of schemes which are available(modules given in the source code, no explicit documentation is given):

1. Mechanical - Isostrain
2. Mechanical – RGC
3. Damage-Pass
4. Thermal -Pass
5. Thermal -Isotemperature

Phase:
phase is a dictionary that contains and any number of arbitrary labeled keys. Each entry contains a key lattice which specifies the lattice structure in Pearson notation. At least the type of the employed constitutive model for each active field (mechanical/elastic, mechanical/plastic, mechanical/eigen, thermal/source(s), or damage) is given; the further configuration details depend on the selected constitutive model.

Example:

```
phase:
  Aluminum:
    lattice: cF
    mechanical:
      output: [F, P, F_e, F_p, L_p, O]
      elastic: {type: Hooke, C_11: 106.75e9, C_12: 60.41e9, C_44: 28.34e9}
      plastic:
        type: phenopowerlaw
        N_sl: [12]
        a_sl: 2.25
        atol_xi: 1.0
        dot_gamma_0_sl: 0.001
        h_0_sl-sl: 75e6
        h_sl-sl: [1, 1, 1.4, 1.4, 1.4, 1.4, 1.4]
```

```
n_sl: 20
output: [xi_sl]
xi_0_sl: [31e6]
xi_inf_sl: [63e6]
```

There are two entries in the phase named aluminum , the type of lattice – cF (meaning FCC) , the mechanical properties in which there are two more entries named elastic and plastic. Elastic has the type Hooke and corresponding independent elastic constants. Plastic is of type phenopower law and their corresponding terms like number of slip systems, hardening parameters(slip-slip). Resistance to Slipping (initial and infinity) are given.(The parameter list is given in the phase.f90 files in src) Available lattice models (list taken from lattice.f90):

- aP: Isotropic
- cF: FCC
- cI: BCC
- oP: Orthorhombic primitive
- hP: Hexagonal
- tI: Body centered Tetragonal

The Available plasticity models(the name is the keyword, which are easily understandable):

- Dislocation based on tungsten
- Dislocation based on twin
- Isotropic
- Kinematic hardening
- None
- Nonlocal
- Phenomenological Power law

The Available damage models:

- Anisobrittle
- Isobrittle

- Isoductile

The Available Eigen models:

- Cleavage opening
- Slip plane opening
- Thermal expansion

The Available Thermal models:

- Dissipation
- External Heat

Material:
Material is a list. The number of entries is at least as long as the maximum material ID reference in the employed geometry. Each entry contains the specification of the employed homogenization referenced by its label and a list of constituents whose length matches N_constituents. Each constituent entry contains the specification of the employed phase referenced by its label, v, the volume fraction, and O the crystallographic orientation as a unit quaternion. Example:

```
- homogenization: SX
    constituents:
      - phase: Aluminum
        v: 1.0
        O: [1.0, 0.0, 0.0, 0.0]
  - homogenization: SX
    constituents:
      - phase: Aluminum
        v: 1.0
        O: [0.7936696712125002, -0.28765777461664166,
            -0.3436487135089419, 0.4113964260949434]
```

For every material ID given in geometry file, one entry should be specified containing the reference to homogenization name , phase name(these should already present in the file), volume fraction and orientation as unit quaternion.

The Loadcase file (.yaml): Though the file extension is .yaml, the format of load file is not actually YAML but an undocumented proprietary format. The loading can be given as deformation gradient(F), velocity gradient(L), Second Piola Kirchoff stress(P) These can be used individually or in combination. When used in combination, care must be taken such that the load case follows the rules given below: Mixed boundary conditions need to fulfill the following requirements:

- Stress and Deformation BCs are mutually exclusive.
- The stress boundary conditions must not allow for rotation, e.g. the opposite off-diagonal elements cannot have stress components at the same time.
- If a velocity gradient is prescribed, each row of the tensors must either contain stress or velocity gradient.

Why these rules ? Except for special cases (simple shear, rotation, etc.), a load case prescribing all components of F will lead to a non-volume preserving load. Therefore, the deformation should be undefined at least one component of the 3x3 tensor F and a stress must be prescribed at those components to get a unique solution. To leave a component of the deformation undefined, use an 'x' at the corresponding position.

Example of a load case:

```
loadstep:
  - boundary_conditions:
      mechanical:
        dot_F: [[1.0e-3, 0, 0],
                [0,      x, 0],
                [0,      0, x]]
        P: [[x, x, x],
            [x, 0, x],
            [x, x, 0]]
    discretization:
      t: 10
      N: 40
    f_out: 4
```

This load case is applying deformation gradient in XX direction(tension). Some values in tensor are left undefined and corresponding Piola-Kirchoff stress is defined to get a unique solution (prescribed rules). t=10 meaning the load is applied for 10 seconds , N=40 meaning there are 40 load increments and f_out=4 shows that the solution is saved for every four increments.

Bibliography

[1] W.C. Lenthe M.P. Echlin. Three-dimensional sampling of material structure for property modeling and design. *Integrating Materials and Manufacturing Innovation*, 2014.

[2] http://dream3d.bluequartz.net/. Dream3d documentation.

[3] Michael Groeber and Michael Jackson. Dream.3d: A digital representation environment for the analysis of microstructure in 3d. *Integrating Materials and Manufacturing Innovation*, 3:5, 02 2014.

[4] F. Roters, M. Diehl, P. Shanthraj, P. Eisenlohr, C. Reuber, S.L. Wong, T. Maiti, A. Ebrahimi, T. Hochrainer, H.-O. Fabritius, S. Nikolov, M. Friák, N. Fujita, N. Grilli, K.G.F. Janssens, N. Jia, P.J.J. Kok, D. Ma, F. Meier, E. Werner, M. Stricker, D. Weygand, and D. Raabe. Damask – the düsseldorf advanced material simulation kit for modeling multi-physics crystal plasticity, thermal, and damage phenomena from the single crystal up to the component scale. *Computational Materials Science*, 158:420–478, 2019.

[5] https://paulvanderlaken.com/2017/10/16/neural-networks-101/.

[6] Shruti Jadon. Introduction to different activation functions for deep learning. *Medium, Augmenting Humanity*, 16, 2018.

[7] Cambridge Coding Academy. Deep learing for complete beginners: Recognising handwritten digits.

[8] https://towardsdatascience.com/optimizers-for-training-neural-network 59450d71caf6. Various optimization algorithms for training neural network.

[9] Julius Schuster and Martin Palm. Reassessment of the binary aluminum–titanium phase diagram. *Journal of Phase Equilibria and Diffusion*, 27:255–277, 06 2006.

[10] Claudio Renato Zambaldi. Micromechanical modeling of γ-tial based alloys. Master's thesis, Rheinisch-Westfälischen Technischen Hochschule Aachen, 2014.

[11] Guangyan Li, Gary Lamberton, and J. Gladden. High temperature resonant ultrasound spectroscopy methods. *International Journal of Spectroscopy*, 2010, 01 2010.

[12] Swantje Bargmann, Benjamin Klusemann, Jürgen Markmann, Jan Eike Schnabel, Konrad Schneider, Celal Soyarslan, and Jana Wilmers. Generation of 3d representative volume elements for heterogeneous materials: A review. *Progress in Materials Science*, 96:322–384, 2018.

[13] I.M. Gitman, H. Askes, and L.J. Sluys. Representative volume: Existence and size determination. *Engineering Fracture Mechanics*, 74(16):2518–2534, 2007.

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

[15] W.J. Drugan and J.R. Willis. A micromechanics-based nonlocal constitutive equation and estimates of representative volume element size for elastic composites. *Journal of the Mechanics and Physics of Solids*, 44(4):497–524, 1996.

[16] T. Kanit, S. Forest, I. Galliet, V. Mounoury, and D. Jeulin. Determination of the size of the representative volume element for random composites: statistical and numerical approach. *International Journal of Solids and Structures*, 40(13):3647–3679, 2003.

[17] Shriram Swaminathan, Somnath Ghosh, and N. J. Pagano. Statistically equivalent representative volume elements for unidirectional composite microstructures: Part i - without damage. *Journal of Composite Materials*, 40(7):583–604, 2006.

[18] M. Pinz, G. Weber, W.C. Lenthe, M.D. Uchic, T.M. Pollock, and S. Ghosh. Microstructure and property based statistically equivalent rves for intragranular γ-γ' microstructures of ni-based superalloys. *Acta Materialia*, 157:245–258, 2018.

[19] Martin Diehl. A spectral method using fast fourier transformto solve elastoviscoplastic mechanical boundaryvalue problems. Master's thesis, Technische Universität München, 2010.

[20] A. Al-Aradi, Adolfo Correia, D. Naiff, G. Jardim, and Yuri F. Saporito. Solving nonlinear and high-dimensional partial differential equations via deep learning. *arXiv: Computational Finance*, 2018.

[21] Sai Karthikeya Vemuri. Personal programming project- solving differential equations using neural networks.

[22] A.Henderson. *ParaView Guide, A Parallel Visualization Application.* Kitware Inc, 2007.