NTNU
Norwegian University of
Science and Technology

Material Parameter Identification Using Artificial Neural Networks and Genetic Algorithm

Olav Midtgarden

Marine Technology
Submission date: June 2018
Supervisor: Sigmund Kyrre Ås, IMT

Norwegian University of Science and Technology
Department of Marine Technology

Preface

This master's thesis was written as a part of the study programme Marine Technology at Norwegian University of Science and Technology during the spring of 2018. Upon completion, the thesis leads to a master's degree in Marine Structures.

I would like to thank my supervisor Prof. Sigmund Kyrre Ås for his excellent guidance and help throughout the semester.

Trondheim, June 2018

Olav Midtgarden

Abstract

Finite element method (FEM) simulations are widely used to perform structural analysis. To ensure that the results obtained from the simulations are realistic it is crucial to have a material model that is as close to reality as possible. This thesis presents a new method for identifying material parameters based on artificial neural network (ANN), Genetic Algorithm (GA), numerical simulations and experimental tests. The method is based on a classical inverse identification procedure aiming to find the material parameters, minimising the difference between the simulated response by FEM analysis and the measured response from an experiment. However, to save computational time the FEM simulations are substituted by an ANN in the optimisation loop.

In order to test the method, two case studies were carried out with the purpose to identify the parameters in the Chaboche hardening model and in the Gurson–Tvergaard–Needleman (GTN) model. The results from the case studies showed that the method was able to provide relatively accurate results using a short amount of computational time. Furthermore, in both cases it was observed that different set of material parameters could give a nearly identical load-displacement curve. It is therefore recommended to extend the method so that it uses more quantities in the comparison between the simulated and experimental result. However, by making this adjustment it is expected that the method can outperform conventional methods when computational time and accurate results are both emphasised. Therefore this strategy should be considered in the future when the optimal material model is to be determined.

Sammendrag

Simuleringer ved bruk av elementmetoden er bredt anvendt for å utføre styrkeberegninger. For å oppnå realistiske resultater fra disse simuleringene er det avgjørende å bruke en materialmodell som er så nær virkeligheten som mulig. I denne avhandlingen presenteres en ny metode for å identifisere materialparametere basert på kunstig nevralt nettverk, genetisk algoritme, numeriske simuleringer og eksperimentelle forsøk. Metoden er basert på en klassisk invers identifikasjonsprosedyre som tar sikte på å finne materialparametere ved å minimerer forskjellen mellom den simulerte responsen fra FEM analyser og den målte responsen fra et eksperiment. For å redusere beregningstiden er imidlertid FEM-simuleringene erstattet av et kunstig nevralt nettverk i optimaliseringsprosedyren.

For å teste metoden ble to studier utført. Disse studiene hadde som mål å identifisere parameterne i Chaboche-fastningsmodellen modellen og i Gurson-Tvergaard-Needleman-modellen. Resultatene fra studiene viste at den foreslåtte metoden var i stand til å gi nøyaktige resultater samtidig som den brukte kort beregningstid. Videre ble det i begge studiene observert at forskjellige sett av materialparametere kunne gi en nesten identisk last-forskyvningskurve. For å forbedre metoden anbefales det derfor å bruke flere parametere i sammenligningen mellom det simulerte og eksperimentelle resultatet. Ved å foreta denne justeringen forventes det imidlertid at metoden kan overgå konvensjonelle metoder når både beregningstid og nøyaktighet er vektlagt. Denne strategien bør derfor vurderes når den optimale materialemodellen skal bestemmes.

Contents

List of Figures

List of Tables

Nomenclature

α	Backstress
$\bar{\sigma}$	Equivalent stress
ϵ	Total strain
ϵ^{el}	Elastic strain
ϵ^{pl}	Plastic strain
ϵ^{pl}_{kk}	Plastic hydrostatic strain
ϵ_N	Mean effective plastic strain of nucleation
γ	Kinematic hardening parameter Chaboche model
λ	Plastic multiplier
$\sigma\|_0$	Size of the yield surface at zero plastic strain
σ_m	Hydrostatic stress
σ_y	Yield strength in tension/compression
τ_y	Yield strength in shear
a	Output artificial neural network
E_t	Tangent modulus
f	Void volume fraction
f^*	Modified void volume fraction
f_c	Critical void volume fraction
f^*_f	Ultimate value of f
f_N	Nucleation micro-void volume fraction
q_1, q_2, q_3	Fitting parameters GTN-model
Q_∞	Isotropic hardening parameter Chaboche model
S_N	Standard deviation
b	Isotropic hardening parameter Chaboche model
b	Neuron bias

C	Kinematic hardening parameter Chaboche model
E	Elastic modulus
K	Hardening parameter power law
n	Hardening parameter power law
p	Input artificial neural network
t	Target artificial neural network
w	Neuron weight

Chapter 1

Introduction

Numerical simulations using FEM simulations are extensively used to optimise metal forming processes and to reduce the cost of final products. However, to ensure that the result obtained from the simulations are realistic it is crucial to have a material model that is as close to reality as possible. This is especially important when materials are subject to cyclic loads, because the difference in behaviour between actual materials and simulated models widens as the number of cycles increases (Lostado, Martínez-De-Pisón, Fernández, & Fernández, 2010). This thesis will therefore primarily cover the procedure of identifying material parameters.

1.1 Background

To date many different approaches have been proposed in order to identify material parameters. These approaches can in general be classified into theoretical methods, microscopic measurements methods for example image analysis technology, macroscopic measurements methods like digital image correlation measurements and hybrid numerical-experimental methods. Hybrid numerical-experimental methods are also known as inverse techniques, here numerical simulations are combined with experimental results to obtain the material parameters.

The theoretical methods are the simplest and less time consuming methods, however they have a limited extend because no analytic or empirical equation is able to fully model the complex phenoms occurring within the material when it is deformed beyond the proportional limit. As an example of a complex phenom, figure 1.1 shows the strain field for a test specimen loaded in tension just prior to failure.

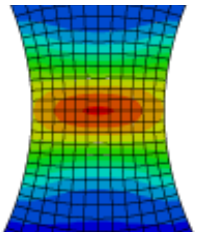


Figure 1.1: Nonuniform strain for material loaded beyond the proportional limit

From the figure it can be seen that the strain field is highly nonlinear. Furthermore, determination of multiple parameters through microscopic and macroscopic measurements are complicated because it requires a large number of experimental tests. Consequently, the hybrid numerical-experimental methods have gained popularity due to its accuracy, simplicity and low experimental cost (Zhong, Xu, Guan, & Zou, 2016).

1.2 Literature review

So far many hybrid numerical-experimental methods have been proposed in the literature and the progress of these are briefly described below. Among the contributions some articles stand out as especially informative and relevant while others introduce new and significant findings.

In general, the aim of the identification strategy is to find a set of material parameters so that the simulated results fits the experimental results as good as possible. In practise this is often executed by minimising an objective function representing the deviation between the simulated and the experimental results. Many researchers have produced work in this field, Mahnken (1999) applied a gradient based algorithm for minimising the objective function in order to find the ductile damage parameters from a tensile test. The objective function was defined as the square difference between the simulated and the experimental results at given displacement points. During the optimisation process the next parameter set was found from the previous parameter set and the improved direction of the parameters. The improved direction was calculated using the Levenberg and Marquardt method (Marquardt, 1963). Faurholdt (2000) applied a similar optimisation procedure for finding the hardening parameters in a linear hardening model and a power law hardening model from a deep-drawing test.

However, the objective function often undergo local minimums in the optimisation process, which is a difficulty for gradient-based algorithms due to their strong dependence on the starting point. To overcome this weakness Lostado et al. (2010) used a genetic algorithm to minimise the objective function. The objective function was in the study defined as the mean absolute error between the simulated and the experimental stress at given strain points. The method was applied to find the hardening parameters in the Chaboche model from a low cycle fatigue test. Muñoz-Rojas, Cardoso, and Vaz (2010) proposed a combined strategy first using a genetic algorithm to find a good initial parameter set near the optimal solution and then using a gradient-based method to minimise the objective function.

A drawback with the genetic algorithm is the computational time. As the number of parameters to be identified increase the number of necessary simulations increase making the optimisation process very time consuming. Many researchers have developed strategies to reduce the computational time. Zhong et al. (2016) proposed a method using a particle swarm optimisation (PSO) algorithm for minimising the objective function. The PSO is known to be effective compared to other optimisation algorithms and it can also avoid local minimums (Kennedy & Eberhart, 1995). The method was applied for finding ductile damage parameters from a tensile test.

Cuesta, Alegre, and Lacalle (2010) developed an alternative method for determining the ductile damage parameters from a small punch test. This method divides the load-displacement curve into six parts and it is assumed that different parts is dominated by different parameters. This allows some parameters to be determined separately from the others, hence reducing the number of necessary simulations. The methodology can also be applied for other material models, however, it requires that the effect of the parameters is uncoupled.

In recent years many scientists have turned to ANN in order to reduce the computational time. Aguir, Belhadjsalah, and Hambli (2011) proposed a method combining FEM simulations, GA and ANN. Here the FEM simulations are substituted by an ANN in the optimisation loop. The FEM simulations are only used to train the network. The input to the ANN is the material parameters and the output is the response at given displacement points. As the output from an ANN is almost instantaneous, the computational time can be significantly reduced using the method if a database of FEM simulations is available. Even though such a database is not

available, Aguir et al. (2011) showed that the method can still reduce the computational time if the parameters used as input for the FEM simulations are chosen wisely. The method was applied for identifying the Karafillis and Boyce criterion and the Voce parameters model of the Stainless Steel AISI 304.

Abbassi et al. (2013) developed an alternative strategy using FEM simulations and ANN. This method utilise the output from the FEM simulation as the input to the ANN while targeting the corresponding material parameters. By providing the results from the experimental test to the trained ANN the material parameters can be predicted directly eliminating the need for an additional optimisation procedure.

1.3 Scope

The scope of this thesis will be to investigate how ANN can be used to reduce the computational time in the procedure for finding material parameters.

Most studies applying ANN for material identification use a direct approach, where the input to the trained ANN is the load-displacement curve and the output is the material parameters. However, using this approach it is difficult to know if the input to the ANN is within the range of the training data. In addition, a theoretical material model will in most cases not be able reproduce the exact behaviour from an experiment and both these factors make it hard to prevent the ANN from extrapolating.

In this thesis a new method for identifying material parameters based on ANN, GA, FEM simulations and experimental tests will be introduced. The ANN will be used within an optimisation algorithm with fixed searching ranges, hence significantly reducing the probability of extrapolation. Two case studies will be carried out in order to illustrate how the method can be applied. The first case study will aim at finding the Chaboche hardening parameters. There exist a few hybrid methods for identifying these parameters, however non of these use ANN. The goal of the second case study is to identify the failure parameters in the GTN-model. Although there are some examples of identification procedures for finding parameters in this model using ANN, non of these use the ANN within an optimisation procedure.

By testing the method on both hardening and fracture models it will be exposed to a wide range of different plastic nonlinear effects.

1.3.1 Problem formulation

The purpose of this thesis is to investigate how artificial neural networks can be used to reduce the computational time in the procedure for finding material parameters. The main contributions can be stated in two points:

- Introduce a new method for identifying material parameters based on ANN, GA, FEM simulations and experimental tests.
- Evaluate the performance of the proposed method by identifying the parameters in the Chaboche hardening model and the GTN-model.

1.4 Thesis overview

The thesis is organised as follows:

Chapter 2 describes relevant theory

Chapter 3 presents the identification method and a case study is carried out to illustrate how the method can be used to identify the material parameters in the Chaboche model.

Chapter 4 presents a case study illustrating how the proposed method can be used to identify the ductile damage parameters in the GTN-model.

Chapter 5 contains the conclusion and recommendations for further work.

Chapter 2

Background

2.1 Elasto-plastic material behaviour

The stress-strain curve of a fictive steel-like material in uniaxial tension is presented in figure 2.1 in order to illustrate the elastic-plastic response of a ductile material.

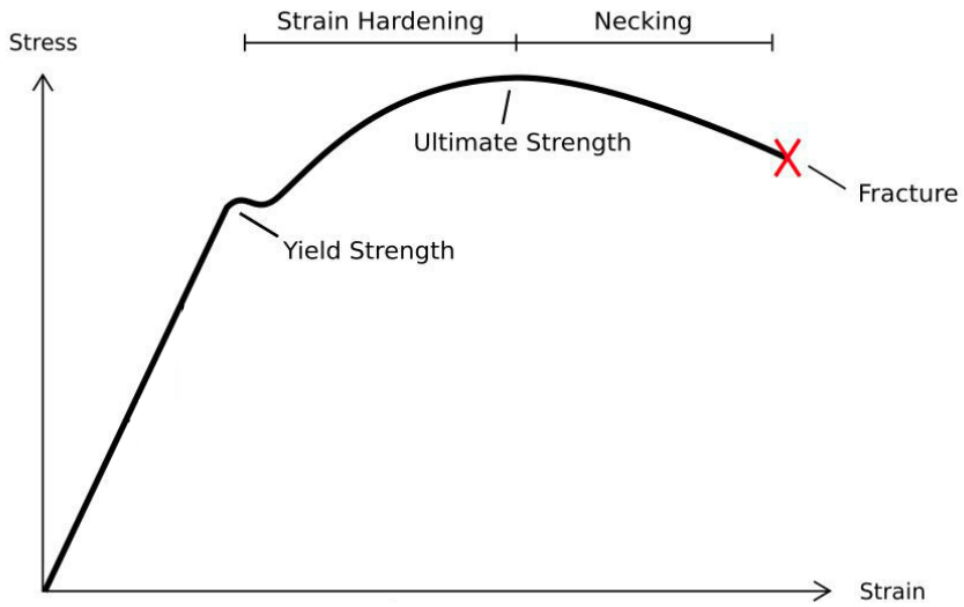


Figure 2.1: Stress-strain curve

The curve can be divided into two main regions, the elastic and the plastic part. The elastic part is characterised by that the stress-stain curve is a straight line given by Hookes law: $\sigma = E\epsilon$. Unloading from a stress level in this region will follow the initial load curve and it will not result in any permanent deformations. If the material is loaded beyond the stress level denoted as the yield strength it will enter the plastic region. Unloading in this region will leave the material with

permanent plastic deformations ϵ^{pl}. From the figure it can be seen that the material's resistance increase when it is deformed from the strain level corresponding to the yield strength point until the strain corresponding to the ultimate strength. This is known as hardening and it is due to slip deformations within the material. If the material is deformed beyond the strain corresponding to the ultimate strength, large localised strains will form at a particular part of the material reducing the cross-sectional area. The material's resistance will then decrease because the effect of the reduction of the cross-section is stronger than the effect of the hardening mechanism. The local reduction of the cross section is called necking and it will eventually lead to failure.

The decoupling of elastic and plastic effects can be justified by studying the deformations on an atomic level. Elastic effects corresponds to variations in the distances between the atoms. On the other hand plastic effects implies slip movements and consequently modification on the interatomic bonds (Lemaitre & Chaboche, 1990).

The total strain ϵ can then be written as:

$$\epsilon = \epsilon^{el} + \epsilon^{pl} \tag{2.1}$$

The constitutive description of a ductile material can in general be represented by three governing principles:

- A yield criterion
- A hardening rule
- A flow rule

2.1.1 Yield criterion

The yield criterion states when yielding is initiated. In the uniaxial case the material starts to yield when $|\sigma| = \sigma_y$, where σ_y denotes the yield strength.

For a multi-axial (2D or 3D) stress distribution the yield criterion becomes a function of the individual stress components. Mathematically this can (in the simplest form) be written as:

$$F = F(\boldsymbol{\sigma}) \tag{2.2}$$

where F is the yield function having the following properties (Irgens, 2008):

$$\begin{aligned} F(\boldsymbol{\sigma}) = 0 \quad &\rightarrow \quad \text{plastic material behaviour} \\ F(\boldsymbol{\sigma}) < 0 \quad &\rightarrow \quad \text{elastic material behaviour} \\ F(\boldsymbol{\sigma}) > 0 \quad &\rightarrow \quad \text{invalid} \end{aligned} \tag{2.3}$$

Graphically the yield function can be considered to define a surface in stress space. This surface is denoted as the yield surface. Two of the most applied yield criteria for multidimensional stress distribution are the Von Mises and the Tresca criterion. The initial yield surface given by the two criteria is presented in equation 2.4:

$$F = \bar{\sigma} - \sigma_y = 0 \tag{2.4}$$

where $\bar{\sigma}$ denotes the equivalent stress. The Tresca criterion states that the material yields if the maximum shear stress exceeds the yield stress in shear, τ_y. The equivalent stress in 3D is then given by equation 2.5.

$$\bar{\sigma} = \max(|\sigma_1 - \sigma_2|, |\sigma_2 - \sigma_3|, |\sigma_3 - \sigma_1|) \tag{2.5}$$

where σ_1, σ_2 and σ_3 are the principal stresses.

The equivalent Von Mises stress is given by equation 2.6.

$$\bar{\sigma} = \sqrt{\frac{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2}{2}} \tag{2.6}$$

Figure 2.2 shows a comparison of the two criteria for a plane strain case ($\sigma_3 = 0$). From the figure it can be seen that there are only small differences between the two. For example the Von Mises criterion is continuous while the Tresca criterion is discontinuous. The Tresca criterion is also slightly more conservative.

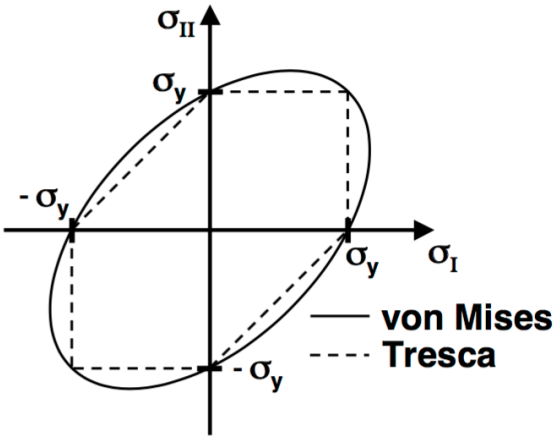


Figure 2.2: Comparison of the Von Mises and Tresca criterion for a plane stress case (Lagace, 2008)

2.1.2 Hardening rule

A hardening rule describes how the yield criterion is modified due to a historical plastic flow. There exist various models for describing the hardening of materials, this thesis will be limited to the following cases:

- Isotropic hardening
- Kinematic hardening
- Chaboche combined hardening

Isotropic hardening
In the isotropic hardening model the yield surface expands symmetrically about the initial centre position as the plastic strains develops. This implies an extension of the elastic area.

Figure 2.3 illustrates isotropic hardening for an uniaxial stress-strain curve. For example, imagine that the material first has been loaded to the maximum point A and then unloaded. The unloading will then follow the blue dashed line until the stress becomes zero. If the material is reloaded the response will be elastic until the $\sigma > \sigma_{max}$. Assume then that the material is loaded in compression, if yielding occurs at $\sigma = -\sigma_{max}$ the hardening is said to isotropic. So if the resistance in tension and compression initially is the same it will remain the same.

Figure 2.4 shows how the yield surface grows when the material is loaded beyond the yield strength in either tension or compression.

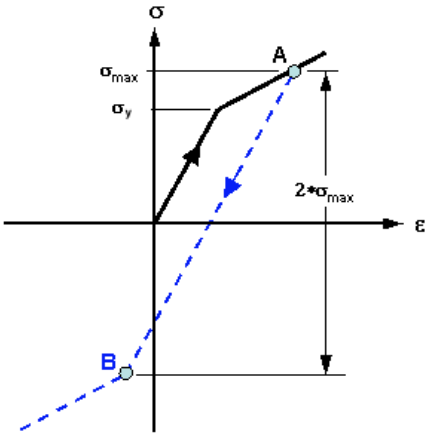


Figure 2.3: Uniaxial stress-strain curve for isotropic hardening (AUTODESK, 2018)

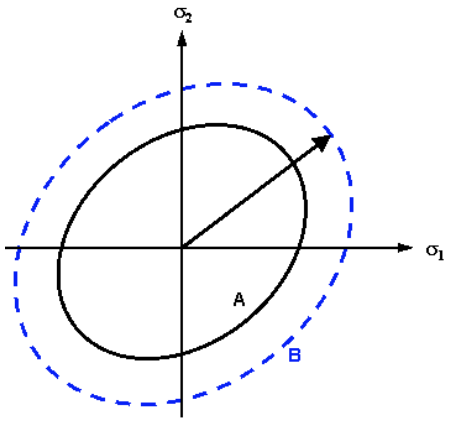


Figure 2.4: Evolution of the yield surface for isotropic hardening (2D) (AUTODESK, 2018)

Mathematically the isotropic hardening rule can be implemented in the yield criterion (equation 2.4) by replacing the constant σ_y with a hardening function, describing how the yield strength increase with increasing plastic strain:

$$F = \bar{\sigma} - H(\bar{\epsilon}^{pl}) = 0 \tag{2.7}$$

where equivalent plastic strain, $\bar{\epsilon}^{pl}$ is (Moan, 2003)

$$\bar{\epsilon}^{pl} = \int_0^{\bar{\epsilon}^{pl}} d\bar{\epsilon}^{pl} \tag{2.8}$$

where $d\bar{\epsilon}^{pl}$ is the equivalent plastic strain increment.

A simple and frequently used isotropic hardening law is the power law. It is given by equation 2.9.

$$\sigma = K{\epsilon^{pl}}^n \tag{2.9}$$

where K and n are hardening parameters.

Kinematic hardening
In the kinematic hardening model the yield surface remains the same shape and size but translates in stress space. This model is closely related to a phenomenon known as the Bauschinger effect. The phenomenon states that hardening in tension leads to a softening in compression. Experiments shows that many materials exhibit this behaviour (Moan, 2003).

Figure 2.6 illustrates kinematic hardening for a uniaxial stress-strain curve. Imagine that the material in the figure is first loaded to the maximum point A and then unloaded. If it is subsequently reloaded in compression it will deform elastically along the dashed line until point B, where yielding occurs. In the kinematic hardening model the difference in stress between point A and B will be twice the yield strength.

Figure 2.4 shows how the yield surface translates in stress space when the material is loaded in tension.

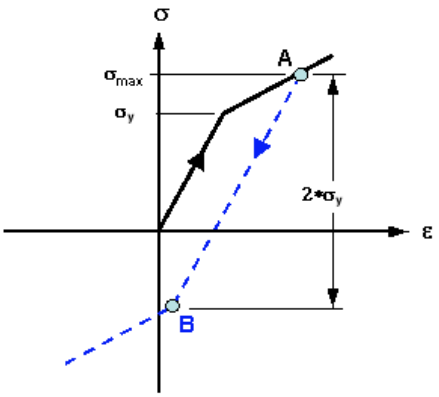


Figure 2.5: Uniaxial stress-strain curve for kinematic hardening (AUTODESK, 2018)

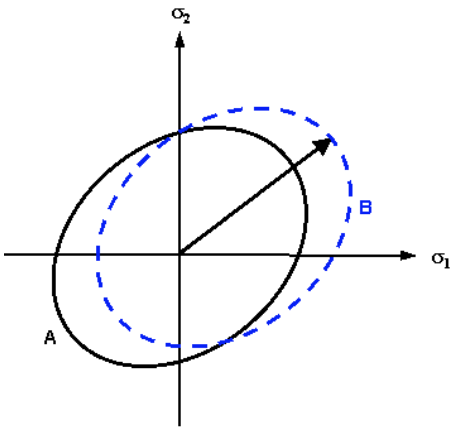


Figure 2.6: Evolution of the yield surface for kinematic hardening (2D) (AUTODESK, 2018)

Kinematic hardening can be implemented in the yield criterion (equation 2.4) by introducing a backstress component in the function for finding the equivalent stress. The backstress describes how the yield stress is shifted in stress space.

Combined hardening
Combined hardening includes features from both the isotropic and kinematic hardening models. Thus the yield surface is able to both expand and translate in stress space.

A widely used combined hardening model is the Chaboche model. The model consists of two distinct components: a nonlinear kinematic component describing the translation of the yield surface via the backstress (α) and an isotropic component describing the size of the yield surface σ^0.

The nonlinear kinematic hardening rule for each backstress component is given by 2.10 when the temperature and field variable dependencies are disregarded.

$$\dot{\boldsymbol{\alpha}}_k = C_k \frac{1}{\sigma^0}(\boldsymbol{\sigma} - \boldsymbol{\alpha})\dot{\bar{\epsilon}}^{pl} - \gamma_k \boldsymbol{\alpha}_k \dot{\bar{\epsilon}}^{pl} \tag{2.10}$$

Here C_k is the initial kinematic hardening modulus and γ_k gives the ratio the kinematic hardening modulus decrease as plastic strain increase (Lostado et al., 2010). If C_k and γ_k are zero there will be no translation of the yield surface and the model reduces to the isotropic hardening model. Equation 2.10 consist of a purely kinematic term and relaxation term introducing the nonlinearity between hardening and the plastic strain.

The total backstress can be found by superimposing the backstress components.

$$\boldsymbol{\alpha} = \sum_{k=1}^{N} \boldsymbol{\alpha}_k \tag{2.11}$$

where N is the number of backstress components.

The isotropic behaviour of the model defines the expansion of the yield surface as a function of the equivalent plastic strain. It is implemented using an exponential rule and it is given by equation 2.12.

$$\sigma^0 = \sigma|_0 + Q_\infty(1 - e^{-b\bar{\epsilon}^{pl}}) \tag{2.12}$$

σ^0 gives the size of yield surface, $\sigma|_0$ defines the size of the yield surface at zero plastic strain. Q_∞ denotes the maximum change of size of the yield surface and b determines the rate the yield surface changes as the equivalent plastic strain develops. When the yield surface is of constant size, i.e. $\sigma^0 = \sigma|_0$, the model reduces to the nonlinear kinematic hardening model.

2.1.3 Flow rule

A flow rule relates the stress and the strain when a material yields. It can in general be written as:

$$d\boldsymbol{\epsilon}^{pl} = \lambda \frac{dQ}{d\boldsymbol{\sigma}} \tag{2.13}$$

where λ is a plastic multiplier determining the amount of plastic strain and Q is a function determining the direction of the plastic strain. If Q is a yield function the plastic strains develops

in a direction perpendicular to the yield surface and the flow rule is termed associative (ANSYS, 2009).

In a uniaxial stress state the relation simplifies to

$$d\sigma = E_t d\epsilon \tag{2.14}$$

where E_t is the tangent modulus.

2.2 GTN ductile damage model

The GTN-model is used to represent plasticity and damage in ductile metals. The model is based on the work proposed by Gurson (1977) and Needleman and Tvergaard (1984).
When plasticity and damage occurs in ductile materials the material goes through a process of void growth, nucleation and coalescence on a microscopic level. These processes are illustrated in figure 2.7. Figure (a) shows how two existing voids grow when the solid matrix is exposed to hydrostatic tension. Figure (b) shows nucleation, new voids are formed due to fractures of inclusions or debounding of the inclusion-matrix. Figure (c) illustrates void coalescence, existing voids form connections causing a more rapid reduction in the load-bearing capacity for increased stress.

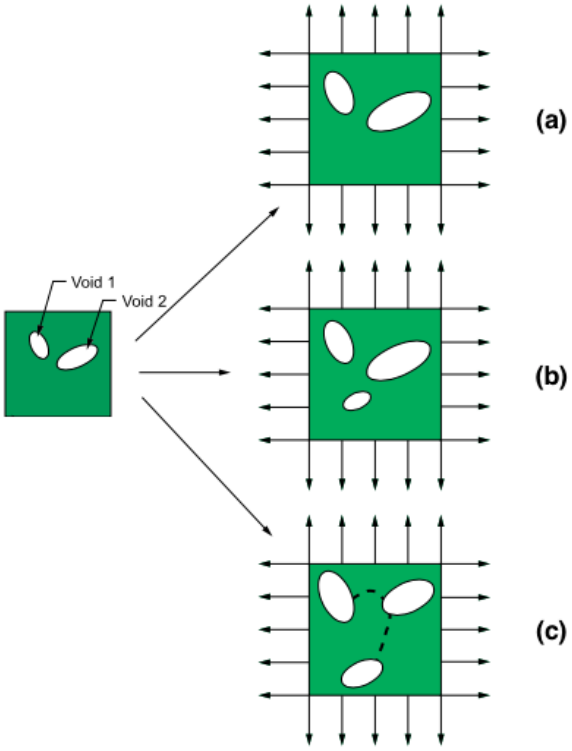


Figure 2.7: Illustration of (a) Growth, (b) nucleation and (c) coalescence of voids in a microscopic scale due to hydrostatic tension (ANSYS, 2009)

In the GTN-model theses three material processes are incorporated into macroscopic plasticity behaviour by changing the void volume fraction and the pressure (ANSYS, 2009).

The yield criteria of the GTN-model is defined as:

$$F = \frac{\bar{\sigma}^2}{\sigma_y^2} + 2f^* q_1 cosh\left(\frac{3}{2} q_2 \frac{\sigma_m}{\sigma_y}\right) - (1 + q_3(f^*)^2) = 0 \tag{2.15}$$

where q_1, q_2 and q_3 are fitting parameters and $\bar{\sigma}$ denotes the Von Mises equivalent stress. σ_m is the hydrostatic stress defined as the average of the normal stress components. f is the void volume fraction given by:

$$f = 1 - \frac{V_M}{V} \tag{2.16}$$

here V and V_M are the volume of the material with and without defects respectively. f^* is the modified void volume fraction used to model the loss of load carrying capacity due to void coalescence and it is given by equation 2.17.

$$f^* = \begin{cases} f & \text{if} \quad f \leq f_c \\ f_c + \frac{f_f^* - f_c}{f_f - f_c}(f - f_c) & \text{if} \quad f_c < f < f_f \\ f_f^* & \text{if} \quad f \geq f_f \end{cases} \tag{2.17}$$

where f_c is the critical volume fraction of the pores at which the voids start to unite and f_f^* gives the ultimate value of f^* at which the stress carrying capacity goes to zero (Abbassi et al., 2013). f_f is the void volume fraction corresponding to fracture.

The rate of void volume fraction depends on both void growth and void nucleation, ie.

$$\dot{f} = \dot{f}_{gr} + \dot{f}_{nucl} \tag{2.18}$$

where the rate of void growth, $\dot{f}_{gr}$, is given by the following relation.

$$\dot{f}_{gr} = (1 - f)\dot{\epsilon}_{kk}^{pl} \tag{2.19}$$

here ϵ_{kk}^{pl} is the plastic hydrostatic strain.

The rate of void nucleation is given equation 2.20.

$$\dot{f}_{nucl} = \frac{f_N}{S_N\sqrt{2\pi}} exp\left[-\frac{1}{2}\left(\frac{\bar{\epsilon}^{pl} - \epsilon_N}{S_N}\right)^2\right]\dot{\bar{\epsilon}}^{pl} \tag{2.20}$$

where f_N is the nucleation micro-void volume fraction. Furthermore s_N is the standard deviation and ϵ_N is the mean in the Gaussian distribution that is assumed for the void nucleation rate.

2.3 Stress and strain measures

The engineering stress is a widely used stress measure for small displacements. It refers to the initial geometry and it is given as:

$$\sigma_e = \frac{F}{A_0} \tag{2.21}$$

where F is the applied force and A_0 is the initial area.

The corresponding strain measure is the engineering strain. It is given as:

$$\epsilon_e = \frac{L - L_0}{L_0} \tag{2.22}$$

where L and L_0 are the current and the initial length respectively.

In the uniaxial tensile test used to obtain the stress-strain curve, the geometry of the specimen changes significantly in the plastic region. Assuming the load is constant the stress will actually increase and not decrease as the neck develops prior to final fracture. Hence it becomes necessary to use a stress measure accounting for the change in geometry (Moan, 2003). A stress measure having this property is the true stress, σ_t.

$$\sigma_t = \frac{F}{A} \tag{2.23}$$

where A is the actual area.

By using the assumption of incompressibility in the plastic flow of metals (Moan, 2003) the following relation between the initial and the deformed volume is obtained:

$$AL = A_0 L_0 \quad \Rightarrow \quad \frac{A_0}{A} = \frac{L}{L_0} \tag{2.24}$$

Equation 2.24 can then be written in terms of the initial quantities.

$$\sigma_t = \frac{F}{A} = \frac{F}{A_0}\frac{L}{L_0} = \sigma_e(1 + \epsilon_e) \tag{2.25}$$

The true strain can be found by dividing the displacement into infinitesimal parts and adding the strains from all parts, i.e.

$$d\epsilon_l = \frac{L - L_0}{L} = \frac{dL}{L}, \quad \epsilon = \int_{L_0}^{L} \frac{dL}{L} = ln\frac{L}{L_0} = ln(1 + \epsilon_e) \tag{2.26}$$

2.4 Artificial neural networks (ANNs)

This section will serve as a theoretical background for ANNs for function approximation. The concept of ANN is inspired by observations on flow of information between biological neurons in the human brain and it is a very powerful tool for function approximation, in fact Hornik, Stinchcombe, and White (1989) proved that a standard multilayer feedforward network is capable of modelling any measurable function.

2.4.1 Neuron model

An ANN is made up off multiple neurons, a simple neuron model is shown in figure 2.8.

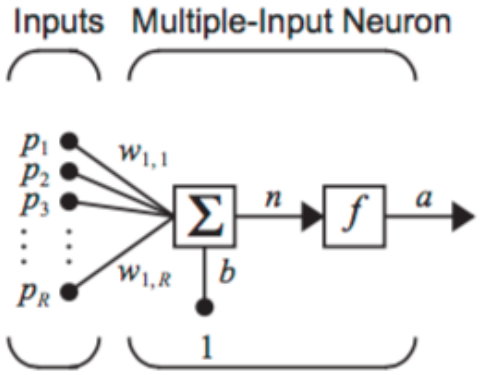


Figure 2.8: Single input neuron (Demuth et al., 2014)

From the figure it can be seen that the individual inputs $p_1, p_2, ..., p_R$ are multiplied by the corresponding weights $w_{1,1}$, $w_{1,2}$, ..., $w_{1,R}$. Here the first index gives the neuron destination for that particular weight and the second index indicates the input number. The weights can be described as numbers expressing the relative importance of the respective inputs to the output.

This product of weights and inputs is then summed with the bias to form the net input n:

$$n = w_{1,1}p_1 + w_{1,2}p_2 + ... + w_{1,R}p_R + b \tag{2.27}$$

in matrix form this can be written as:

$$n = \boldsymbol{wp} + b \tag{2.28}$$

The bias can be thought of as a weight having the constant input 1.

Finally the neuron output can be written as:

$$a = f(\boldsymbol{wp} + b) \tag{2.29}$$

where f is a transfer function.

2.4.2 Transfer function

The main purpose of the transfer function is to introduce nonlinearity into the network. This makes the network capable of approximating nonlinear relations.

The choice of transfer function depends on the type of available data and the assumed distribution of the output variables (Bishop, 2006). Two of the most common transfer functions are plotted in figure 2.9 and 2.10. These are the Log-Sigmoid and the Hyperbolic Tangent Sigmoid function. The Log-Sigmoid function is given by:

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2.30}$$

This function squashes the output of the neuron in to range [0,1].

The Hyperbolic Tangent Sigmoid function is given by:

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2.31}$$

and squashes the output in to the range [-1,1].

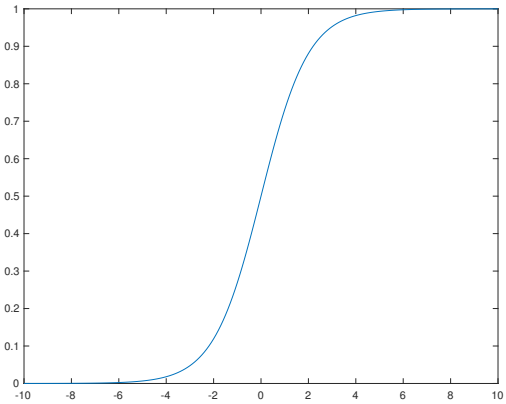


Figure 2.9: Log-Sigmoid function

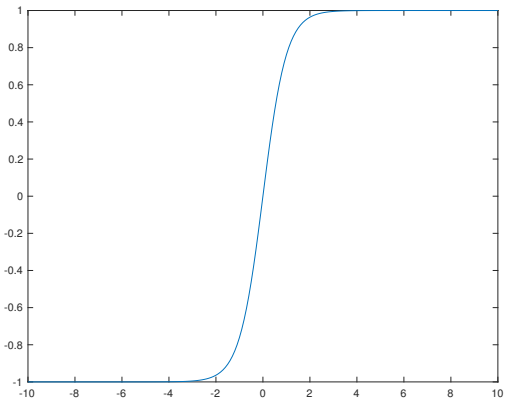


Figure 2.10: Hyperbolic Tangent Sigmoid function

By applying these continuous transfer functions the network will be differentiable and this is an important property with respect to training of neural network.

2.4.3 Network architecture

There exist multiple types of ANNs, only the simplest type, feed-forward networks will be discussed in this thesis.

In a feed-forward networks every neuron is connected to all the neurons in the next layer and all these connections have weights associated with them. An example of a feed-forward network is presented in figure 2.11. The lines in the figure represent the flow of information and in a feed-forward network the information only moves forward.

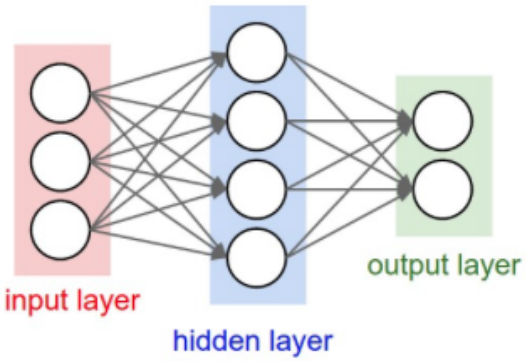


Figure 2.11: Feed-forward network architecture (Karpathy, 2018)

A feed-forward network consist of three types of neurons:

1. Input neurons - These neurons provide the hidden neurons with the input data and they are collectively referred to as the input layer. No computations are executed in the input layer, it simply pass the information to the next layer.
2. Hidden neurons - These neurons performs the computations and transfer the data from the input to the output neurons. They do not interact with the outside world, hence the name "hidden". A set of hidden neurons forms a hidden layer and while a feed-forward network only can have one input and one output layer it can have multiple hidden layers. By increasing the size and the number of the hidden layers the network will be capable of capturing more complex relationships. However it will also increase the computational time and the risk of the network overfitting the data. Overfitting will be explained in detail later.
3. Output neurons - These neurons modifies and combines the data from the hidden neurons to produce the output from the network. The output neurons are collectively called the output layer.

2.4.4 Training of the neural network

The process by which a feed-forward network is trained is called backpropagation. This is a form of supervised training where the network learns from a set of ideal network behaviour:

$$\{\boldsymbol{p}_1, \boldsymbol{t}_1\}, \{\boldsymbol{p}_2, \boldsymbol{t}_2\}, ..., \{\boldsymbol{p}_Q, \boldsymbol{t}_Q\} \tag{2.32}$$

Where p is the input to the network and t is the target. The aim of the training is to optimise the performance of the network by adjusting the network parameters (weights and biases). This is illustrated in figure 2.12.

In the following text that originates from Demuth et al. (2014) the backpropagation algorithm will be described in more detail. The algorithm can be summarised in three steps.

The forward pass
In the first step the input is propagated forward through the network so the output from one

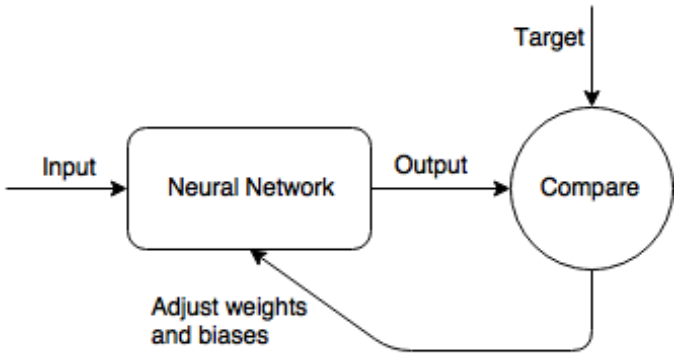


Figure 2.12: Training procedure for neural networks

layer becomes the input for the next layer. This can be described by equation 2.33:

$$\boldsymbol{a}^{m+1} = \boldsymbol{f}^{m+1}(\boldsymbol{W}^{m+1}\boldsymbol{a}^{m} + \boldsymbol{b}^{m+1}) \quad \text{for} \quad m = 0, 1, ..., M-1 \tag{2.33}$$

where $\boldsymbol{a}^{m+1}$ is the output from layer m+1 and M is the total number of layers in the network.

$$\boldsymbol{a}^{0} = \boldsymbol{p} \tag{2.34}$$

Equation 2.34 serves as the staring point for equation 2.33. The output from the last layer can be recognised as the output from the network:

$$\boldsymbol{a} = \boldsymbol{a}^{M} \tag{2.35}$$

Calculating the error
The second step consist of calculating the error between the target and the output from the network. The measure on how well the network performs or how the output is compared to the target in figure 2.12 is called the performance index. In the backpropagation algorithm the mean square error is used:

$$F(\boldsymbol{x}) = E[e^2] = E[(t-a)^2] \tag{2.36}$$

When the network has multiple outputs the equation generalises to:

$$F(\boldsymbol{x}) = E[\boldsymbol{e}^T\boldsymbol{e}] = E[(\boldsymbol{t}-\boldsymbol{a})^t(\boldsymbol{t}-\boldsymbol{a})] \tag{2.37}$$

Widrow and Hoff (1988) showed that by replacing the expectation of the square error by the square error at iteration k, equation 2.37 could be approximated as:

$$\hat{F}(\boldsymbol{x}) = (\boldsymbol{t}(k)-\boldsymbol{a}(k))^T(\boldsymbol{t}(k)-\boldsymbol{a}(k)) = \boldsymbol{e}^T(k)\boldsymbol{e}(k) \tag{2.38}$$

The backward pass
The final step consist of adjusting the network parameters to minimise the function $F(\boldsymbol{x})$, the square error between the target and the output from the network. This is done according to the steepest decent algorithm (for mean square error):

$$w_{i,j}^m(k+1) = w_{i,j}^m(k) - \alpha \frac{\partial \hat{F}}{\partial w_{i,j}^m} \tag{2.39}$$

$$b_i^m(k+1) = b_i^m(k) - \alpha \frac{\partial \hat{F}}{\partial b_i^m} \tag{2.40}$$

The second term in equation 2.39 and equation 2.40 is calculated using the chain rule of calculus. This result in a recurrence relationship in which the weight and bias at layer m depends on the weight and bias at layer m+1.

Theses three steps are then repeated until the error reaches a predetermined limit.

2.4.5 Weight initialisation

Most training algorithms for neural network are based on first initialising the weights to some small random values. Random values are applied to avoid problems related to symmetry in the neural network. The weights are set to small values to avoid saturation of the sigmoid transfer function where the derivative is small, thus leading to slow training. However if the weights are to small, the sigmoid function will be approximately linear which again leads to a slow learning rate (Bishop, 1995).

2.4.6 Generalisation

One of the main advantages of ANNs is their ability to generalise. This means that a trained network can predict accurate output from input it has never seen before. In order to obtain the optimal generalisation the available data is divided into three categories:

- Training samples
- Validation samples
- Testing samples

The training set is used to adjust the weights and biases to minimise the square error between the targets and the output from the neural network. The validation set is applied to evaluate when the network performs the best, hence the training should be stopped. This is done by calculating the validation error for each iteration and then stopping the training when the validation error increase for several iterations. The weights and biases producing the minimum error on the validation set is finally used as the network parameters.

The test set is held aside to after the network has been trained and it is used to measure how well the network will perform on new data. In order for the set to be a valid measure of the network performance, it can not be used for network training and it should also be representative for the cases at which the network will be applied (Demuth et al., 2014).

Two of the most common problem related to training of ANNs are:

1. Over fitting
2. Extrapolation

Overfitting is illustrated in figure 2.13. The figure shows the response of a network with 1 input neuron, 100 hidden neurons and 1 output neuron trained to approximate a sine function. From the figure it can be seen that the network matches the training points, however it clearly fails to reproduce the underlying function and does not generalise well. One method to avoid this problem is to reduce the complexity of the network. By using a small enough network it will not have enough power to overfit the data (MathWorks, 2018). A network with only 6 hidden neurons is showed in figure 2.14. From the figure it can be seen that the network fits the underlying function way better than the complex network.

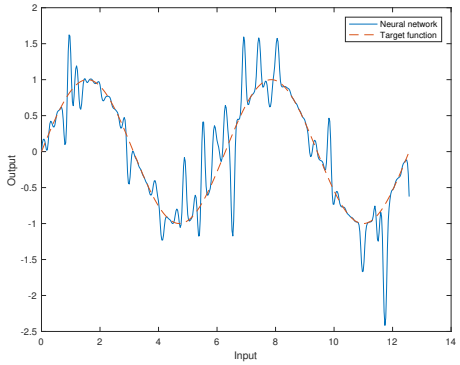


Figure 2.13: Network with hidden layer consisting of 100 neurons

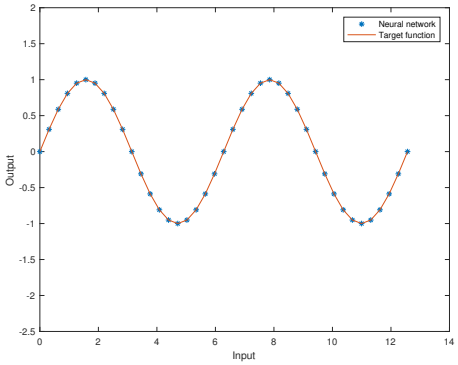


Figure 2.14: Network with hidden layer consisting of 6 neurons

The second type of error, extrapolation, is illustrated in figure 2.15. From the figure it can be seen that the network fails to approximate the sine function for inputs beyond π because there is no training data beyond this point.

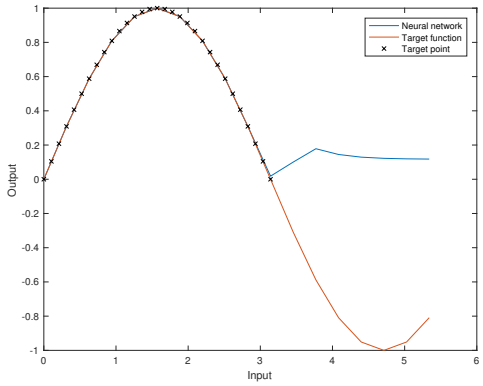


Figure 2.15: Extrapolation example

2.5 Genetic algorithm (GA)

Genetic algorithms are probabilistic optimisation algorithms based on natural selection. These algorithms are applicable for problems that are not well suited for standard optimisation algorithms, including problems in which the objective function is noisy, discontinuous or highly nonlinear (Goldberg, 1989). A flowchart showing the main steps of a genetic algorithm is presented in figure 2.16.

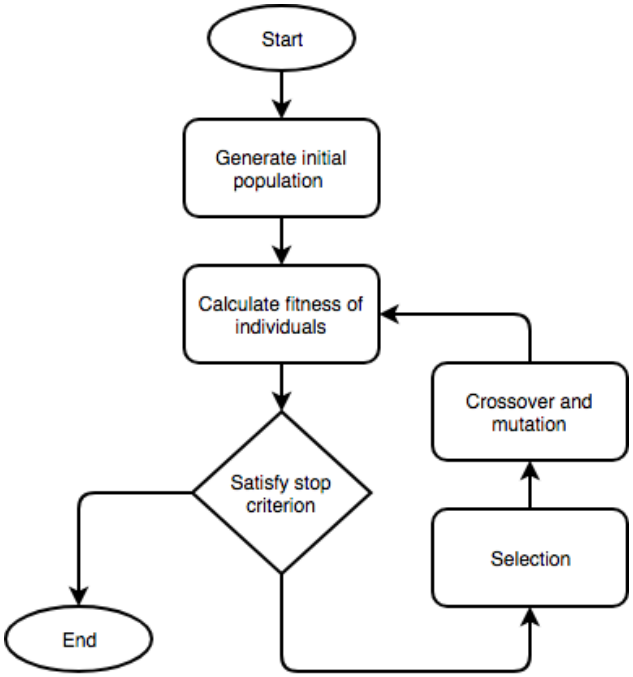


Figure 2.16: Flowchart illustrating the main steps of a genetic algorithm

The first step of the algorithm is to generate a random population of individuals where each individual represent a solution to the given problem. The second step is to score each member of the population. This is done using a predefined fitness function (objective function) measuring the goodness of the solution. The next step consist of selecting some of the individuals to become parents. This process is guided by the score of the members. Finally children are created by either making a random change to one of the parents or by combining the genes of the parents. These operations are called mutation and crossover respectively. Then the children replace the current population and the process is repeated until the solution satisfies a predefined stop criteria.

In order to apply the genetic algorithm the individuals must be represented in way so that genetic operations can be performed on them. Examples of representation, selection and genetic operations are given below.

Representation
In the study of genes, genotype is an organism's hereditary information and phenotype is the observed properties of the organism. For example a persons eye colour (phenotype) depends on the hereditary information (genotype) in the persons genes. The genetic algorithm performs operation on genotypes represented as strings. Goldberg (1989) illustrates this with an example

where he shows how the function $f(x) = x^2$ can be maximised in the interval $[0, 31]$ by first representing the value of x as a string of binary unsigned integers and then performing genetic operations on the string. By applying strings of length 5 he is able to represent all numbers between 0 (00000) and 31 (11111). For example the string 10011 decodes to the number 19:

$$1 * 2^4 + 0 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0 = 16 + 2 + 1 = 19$$

In this example the phenotypes (values of x) are simply the numerical value of the genotypes, however phenotypes can be everything from eye colour to material parameters.

Selection

The process of choosing the members of the current population to become parents for the next population is called selection. The selection strategy is based on letting the fittest individuals having a higher chance of being parents. Generally crossover and mutation explore the search area while selection reduce the search space by discarding the poorest individuals. However the worst individuals should still have some chance of being chosen because they may contain some useful genetics. The search technique should therefore have some trade-off between exploration and exploitation in order to determine the global optimum (Mohd & Geraghty, 2011).

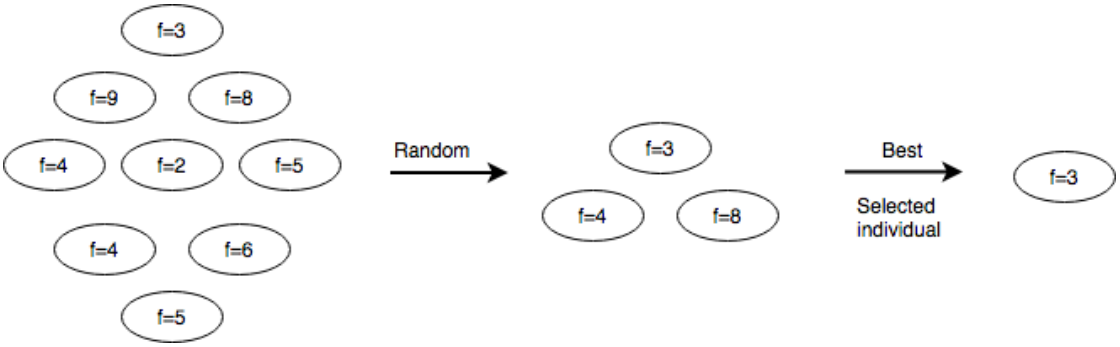


Figure 2.17: Selection using the tournament strategy

A popular selection strategy providing this trade-off is the tournament selection. The method is illustrated in figure 2.17. In the method n individuals are drawn randomly from a bigger population and the one with the best fitness score is selected to reproduce. In the example above the tournament size is set to three meaning that three individuals are chosen to compete against each other.

Another widely used selection/replacement method that can be combined with the tournament strategy is elitism. In this case the best individuals in the current population automatically survive to the next generation. This forces the algorithm to retain the best individuals at each generation and it has been shown that this significantly improves the performance of the algorithm (Purshouse & Fleming, 2002).

Crossover

Crossover is the process of combining the genes of the individual parent solutions to produce a child solution. One of the most popular crossover operators, uniform crossover is illustrated in figure 2.18. In this method each of the genes in the child solution is treated separately and for each gene the child has the same probability of inheriting from parent 1 and parent 2.

In the example in figure 2.18, child 1 inherit the first gene from parent 2, the second from parent 1 and the third and fourth from parent 2 and so on.

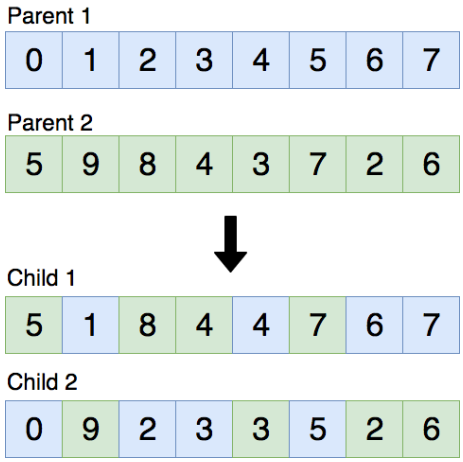


Figure 2.18: Uniform crossover: the genotypes of the two parents are combined to create the genotypes of two new children

Mutation

Mutation can be defined as a small random alteration in the genes of an individual to obtain a new individual. Two of the most common mutation operators are called bit flip and random resetting. In the bit flip mutation one or more genes are randomly chosen and then flipped. This method is used for binary encoded genotypes and it is shown in figure 2.19.

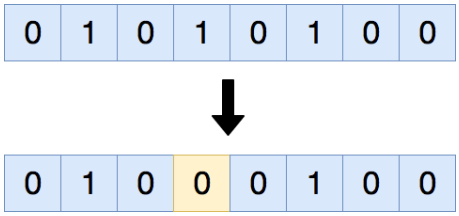


Figure 2.19: Bit flip mutation

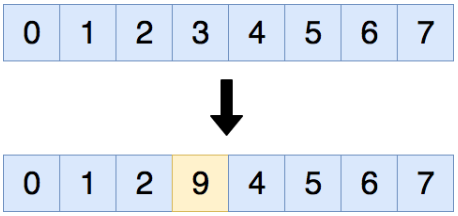


Figure 2.20: Random resetting

Random resetting is an extension of the bit flip for integer encoded genotypes. The method is illustrated in figure 2.20. Here a random value from a set of predefined permissible values is assigned to a randomly chosen position in an individual's genotype.

Mutation is used to maintain and introduce diversity in the population. This is crucial because without mutation a population can evolve to a population where all the individual's genotypes have the same value at a given position. Reproduction by crossover will then only carry this value on to the next generation, however with mutation there is a possibility that the value is being changed.

2.6 Finite Element Method (FEM)

The Finite Element Method is a numerical method for solving problems within engineering fields like structural analysis, heat transfer, fluid flow and mass transport. The method is applicable for solving problems that can be expressed by a set of partial differential equations and associated boundary conditions. The method models a structure as an assemblage of small simple elements connected by nodes. This is illustrated in figure 2.21 (a) and (b) showing a ship frame and a possible FEM model of a part of the same frame.

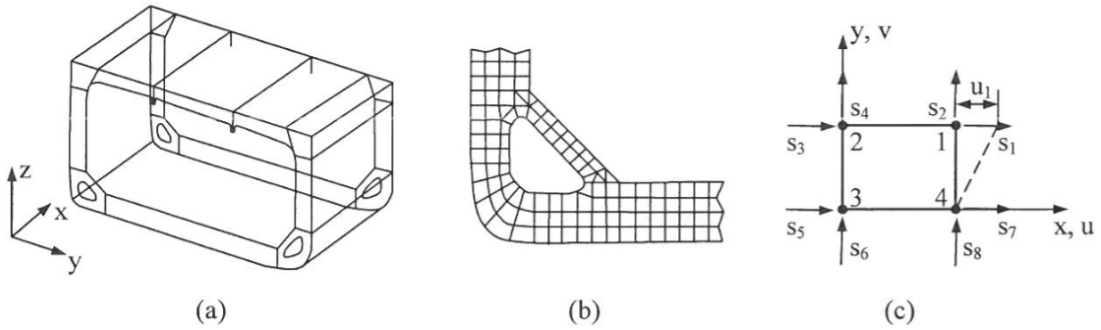


Figure 2.21: (a) Transverse frame in a ship, (b) Possible FEM model of the frame corner, (c) FEM element showing nodal forces (Moan, 2003)

The discretization of the structure makes it possible to determine the constitutive and kinematic relations for each element in a simple manner. Theses relations can further be assembled to a system valid over the whole structure making it possible to solve complex problems that are not solvable by analytic solutions. This is one of the great advantages with FEM.

From figure 2.21 (c) it can also be seen that displacements are only defined at the nodes so shape functions are used to interpolate the displacements within the element. The displacement field describing the displacements are obtained by multiplying the element shape function matrix, $\boldsymbol{N}$, by the nodal displacements vector $\boldsymbol{v}$.

$$\boldsymbol{u} = \boldsymbol{N}\boldsymbol{v} \tag{2.41}$$

The strains are then found as:

$$\boldsymbol{\epsilon} = \Delta\boldsymbol{u} = \Delta\boldsymbol{N}\boldsymbol{v} = \boldsymbol{B}\boldsymbol{v} \tag{2.42}$$

where

$$\boldsymbol{B} = \Delta\boldsymbol{N} \tag{2.43}$$

is the strain-displacement matrix.

The finite element method will in the following be formulated using the principle of virtual displacement (PVD). This principle states that virtual work performed by the real external forces over the virtual displacements is equal to the virtual work performed by the real internal stresses over the virtual strains (Bell, 2013).

In the following all symbols marked by (~) are virtual quantities. The PVD can be formulated mathematically as:

$$\tilde{W}_e = \tilde{W}_i \tag{2.44}$$

where $\tilde{W}_e$ is the external virtual work and $\tilde{W}_i$ is the internal virtual work. For an arbitrary finite element this equation reads:

$$\tilde{\boldsymbol{v}}^T \boldsymbol{S} + \int_{V_e} \tilde{\boldsymbol{u}}^T \boldsymbol{F} dV + \int_{S_T} \tilde{\boldsymbol{u}}^T \boldsymbol{\Phi} dS = \int_{V_e} \tilde{\boldsymbol{\epsilon}}^T \boldsymbol{\sigma} dV \tag{2.45}$$

where the external forces are the nodal forces, $\boldsymbol{S}$, the volume forces, $\boldsymbol{F}$, and the traction forces, $\boldsymbol{\Phi}$. The internal forces are the stresses $\boldsymbol{\sigma}$. Furthermore V_e is the volume and S_T is the surface of the element.

By rewriting equation 2.45 and substituting for $\tilde{\boldsymbol{u}}$ and $\tilde{\boldsymbol{\epsilon}}$ the following equation is obtained:

$$\tilde{\boldsymbol{v}}^T \left[\boldsymbol{S} + \int_{V_e} \boldsymbol{N}^T \boldsymbol{F} dV + \int_{S_T} \boldsymbol{N}^T \boldsymbol{\Phi} dS = \int_{V_e} \boldsymbol{B}^T \boldsymbol{\sigma} dV \right] \tag{2.46}$$

Then by using a stress-strain relationship on the form $\boldsymbol{\sigma} = \boldsymbol{C}(\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_0)$ where ϵ_0 denotes the initial strain and the fact that equation 2.46 must be valid for any $\tilde{\boldsymbol{v}}$ the well known element stiffness relation is obtained.

$$\boldsymbol{S} = \int_{V_e} \boldsymbol{B}^T \boldsymbol{C} \boldsymbol{B} dV \boldsymbol{v} - \int_{V_e} \boldsymbol{B}^T \boldsymbol{C} \boldsymbol{\epsilon}_0 dV - \int_{V_e} \boldsymbol{N}^T \boldsymbol{F} dV + \int_{S_T} \boldsymbol{N}^T \boldsymbol{\Phi} dS = \boldsymbol{k}\boldsymbol{v} + \boldsymbol{S}^0 \tag{2.47}$$

where

$$\boldsymbol{k} = \int_{V_e} \boldsymbol{B}^T \boldsymbol{C} \boldsymbol{B} dV \tag{2.48}$$

is the element stiffness matrix and

$$\boldsymbol{S}^0 = \boldsymbol{S}^0_{\epsilon 0} + \boldsymbol{S}^0_F + \boldsymbol{S}^0_\Phi \tag{2.49}$$

is the load vector with the following contributions:

$$\boldsymbol{S}^0_{\epsilon 0} = -\int_{V_e} \boldsymbol{B}^T \boldsymbol{C} \boldsymbol{\epsilon}_0 dV \tag{2.50a}$$

$$\boldsymbol{S}^0_{F} = -\int_{V_e} \boldsymbol{N}^T \boldsymbol{F} dV \tag{2.50b}$$

$$\boldsymbol{S}^0_{\Phi} = -\int_{S_T} \boldsymbol{N}^T \boldsymbol{\Phi} dS \tag{2.50c}$$

Finally the system stiffness relation can be found using equation 2.51 and 2.52

$$\boldsymbol{R}^c = \left(\sum_{i=1}^{m} (\boldsymbol{a}^i)^T \boldsymbol{k}^i \boldsymbol{a}^i \right) \boldsymbol{r} + \sum_{i=1}^{m} (\boldsymbol{a}^i)^T \boldsymbol{k}^{0i} = \boldsymbol{K}\boldsymbol{r} + \boldsymbol{R}^0 \quad \Rightarrow \quad \boldsymbol{K}\boldsymbol{r} = \boldsymbol{R} \tag{2.51}$$

where

$$\boldsymbol{R} = \boldsymbol{R}^c - \boldsymbol{R}^0 \tag{2.52}$$

and $\boldsymbol{a}$ is a matrix relating the global and nodal displacements.

2.6.1 Linear finite element analysis

A linear static analysis is an analysis where there is a linear relationship between the applied force and the displacement. In practice, this is applicable to structural problems where small deformations and linear elastic material behaviour can be assumed. If these assumptions holds, the stresses will remain in the elastic range and no irreversible deformations will occur. Moreover the equilibrium equations can be established with reference to the initial geometry of the structure (Moan, 2003) and the stiffness matrix $\boldsymbol{K}$ will be constant. Consequently equation 2.51 will form a system of linear equations and the solution process will be short compared to a nonlinear analysis of the same problem.

2.6.2 Nonlinear finite element analysis

In a nonlinear analysis there is a nonlinear relation between the applied force and the displacement. Nonlinear effects can in general be divided in to three groups:

- Material nonlinearities
- Geometric nonlinearities
- Boundary nonlinearities

Material
Material nonlinearities deals with the nonlinear relation between the stress and strain above the proportional limit. This is covered in section 2.1.

Geometric
When structures exhibit large deformations the applied load and resistance (of the structure) can change due to change in the geometry of the problem. This is known as geometric nonlinearities and to handle this type of problems it is important to use the actual geometry at the current time step and not the initial geometry. The Snap-Through problem is an example of a geometric nonlinearity, this is illustrated in figure 2.22.

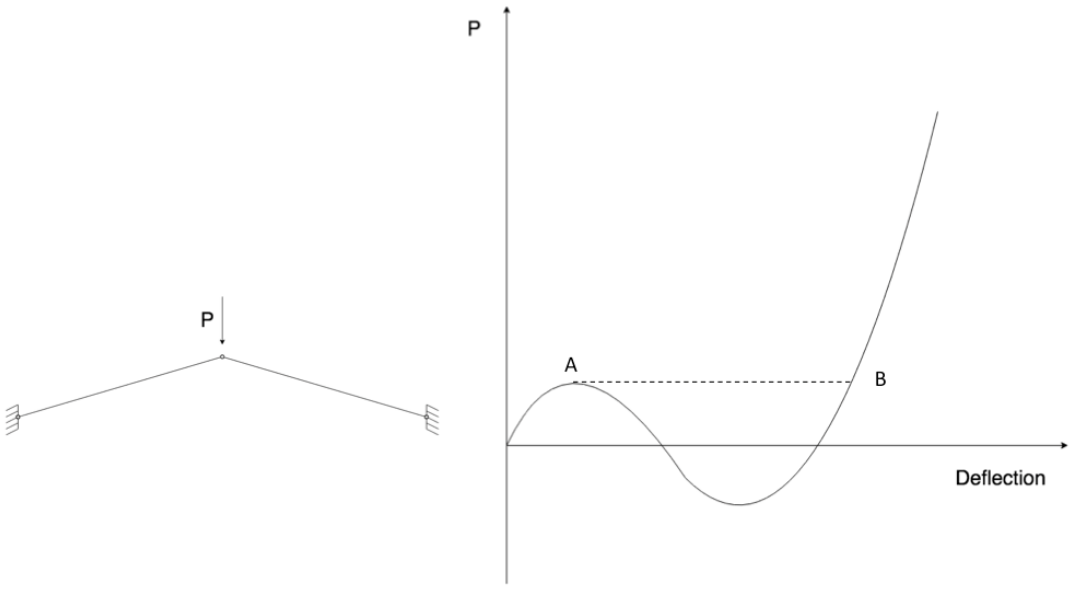


Figure 2.22: Snap-Through, example of a problem exhibiting geometric nonlinearities

When the load P is increased the structure will reach the value at point A. If the load is increased beyond this value the structure will invert and jump to point B along the dotted line. Numerically the stiffness matrix will become singular between the two points.

Boundary
Boundary nonlinearities are characterised by a change in the deformation state of a system during the increase of a prescribed load (Wriggers, 2008). This occurs in most contact problems even when a linear material model is applied and small displacements are exhibited due to nonlinear dependence between the size of the contact area and the applied load. A nonlinear contact problem is illustrated in figure 2.23.

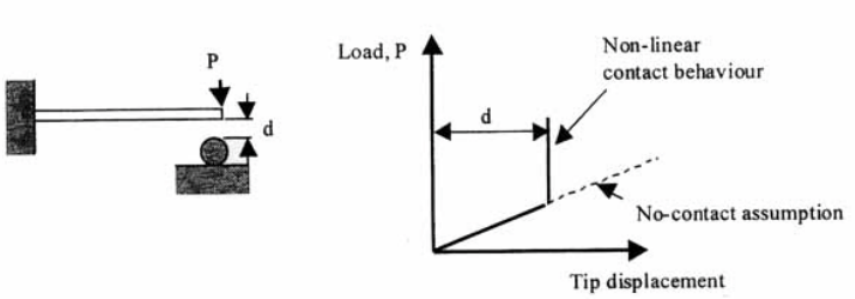


Figure 2.23: Example of a problem involving boundary nonlinearities, when the displacement reaches the value d, there will be a nonlinear relation between the displacement and the applied load (Moan, 2003)

2.7 The identification problem

The response of a structure exposed to mechanical impacts can in general be written as:

$$\boldsymbol{R}(\boldsymbol{x},t) = f_{\tau=0}^{t} t\{\mathfrak{G}, \boldsymbol{c}, \boldsymbol{F}(\boldsymbol{x},\boldsymbol{\tau})\} \tag{2.53}$$

where f is a function depending on the geometry, $\mathfrak{G}$, the material parameters, $\boldsymbol{c}$ and the loading history $\boldsymbol{F}(\boldsymbol{x},t)$. The function f can be given by an experiment or by an analytic or numerical solution to an initial boundary value problem (Brocks & Steglich, 2007). Equation 2.53 is known as a direct problem where $\mathfrak{G}, \boldsymbol{c}$ and $\boldsymbol{F}$ are the inputs and $\boldsymbol{R}(\boldsymbol{x},t)$ is the response describing for example a stress, strain or displacement field. $\boldsymbol{x}$ and t denotes the coordinates and the time respectively.

Given that the geometry and the loading history is known, the function f reduces to a mapping:

$$\boldsymbol{c} \underset{\mathfrak{G},\boldsymbol{F}}{\mapsto} \boldsymbol{R}: \quad \boldsymbol{R} = \mathcal{R}(\boldsymbol{c}) \tag{2.54}$$

with $\boldsymbol{c} \in \mathbb{C}$ and $\boldsymbol{R} \in \mathbb{R}$, $\mathbb{C}$ and $\mathbb{R}$ are the space of possible material parameters and responses respectively (Brocks & Steglich, 2007).

A widely used identification technique is the inverse technique. It consist of finding the material parameters, c, corresponding to an experimentally measured response, $\boldsymbol{R}$. Using the same notation the inverse problem can be formulated mathematically as:

$$\boldsymbol{c} \mapsto \boldsymbol{R}: \quad \boldsymbol{c} = \mathcal{R}^{-1}(\boldsymbol{R})|_{\mathfrak{G},\boldsymbol{F}(\boldsymbol{x},t)} \tag{2.55}$$

Chapter 3

Case study - Chaboche model

In this chapter a method for for identifying material parameters is presented and a case study is carried out in order to illustrate how the method can applied to find the material parameters in the Chaboche model (described in section 2.1.2).

The Chaboche model consist of two distinct components, a nonlinear kinematic component describing the translation of the yield surface in stress space and an isotropic component describing the size of the yield surface. In the experimental test performed in the laboratory, the specimen was only loaded in tension. Thus the Chaboche model reduces to a nonlinear kinematic hardening model. Furthermore, initial FEM simulations showed that two backstress components were sufficient to reproduce the measured response from the experiment. The material parameters to be identified were then C_1, γ_1, C_2 and γ_2. Where C_1 and C_2 gives the initial kinematic hardening modulus for backstress component 1 and 2 respectively. Similarly, γ_1 and γ_2 determines the ratio the kinematic hardening modulus decrease as the plastic strain increase for the given component.

3.1 Method

A flowchart illustrating the method used for identifying the material parameters in the Chaboche model is given in figure 3.1. The method is based on a classical inverse identification procedure aiming to find the material parameters minimising the difference between the simulated response by FEM analysis and the measured response from an experiment. However, to save computational time the FEM simulations are substituted by an ANN in the optimisation loop. The FEM simulations of the tensile test are only used to train the ANN.

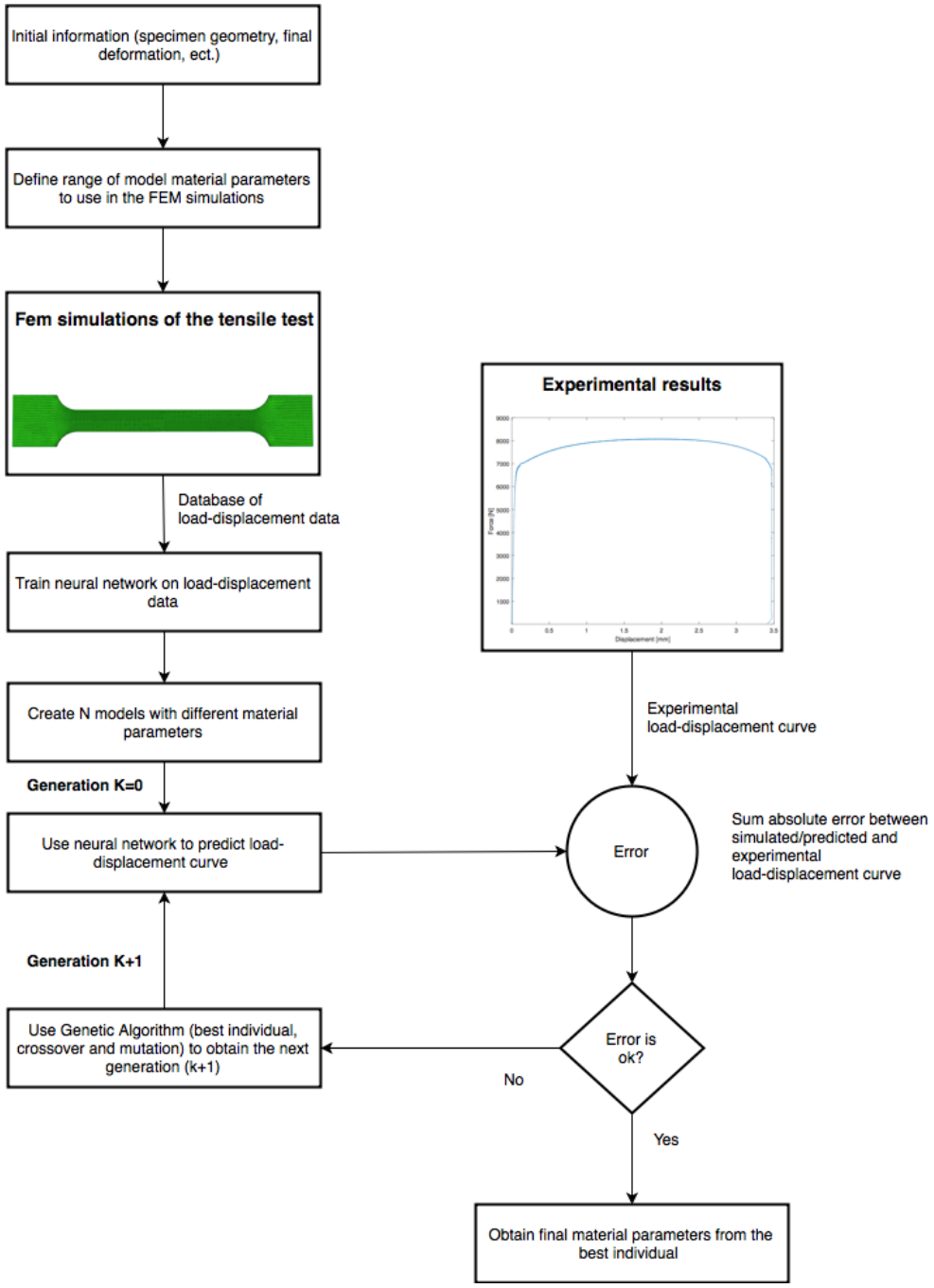


Figure 3.1: Flowchart of the method for optimising the material parameters for the Chaboche model

3.1.1 Experimental data

In the first part of the investigation a low rate tensile test was carried out in the laboratory. In the test the strain was increased at a given rate and an extensometer recorded the specimen's elongation until it ruptured completely.

The specimen used for the testing was a smooth tensile specimen similar to the one shown in figure 3.2. The specimen was initially cut from a circular pipe using a band saw. It was then flattened in a rolling process and finally cut into the shape shown in the figure.

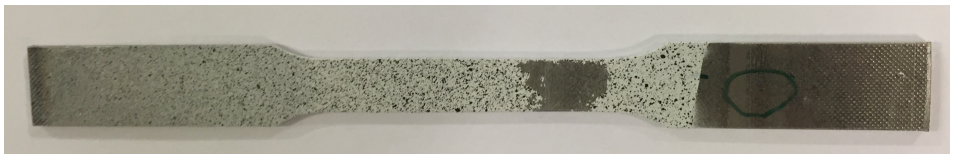


Figure 3.2: Plane test specimen similar to the one used for the tensile test

The geometry of the specimen is given in table 3.1 and the quantities are defined in figure 3.1:

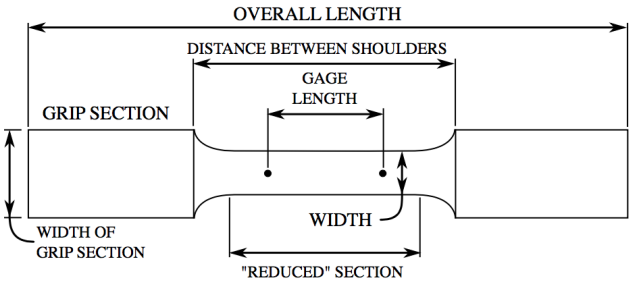


Figure 3.3: Nomenclature of test specimen

Table 3.1: Dimensions of the test specimen

Description	Value [mm]
Overall length	139.1
Width of grip section	24.0
Distance between shoulders	99.1
Reduced section	80.0
Gage length	50.0
Width	10.0
Thickness	1.81

The tests specimen was made out of a steel type classified as: E220+CR2+S2. Where E220 describes that the minimum requirement for the yield strength is 220 MPa, CR means that it is cold rolled and S means that it is treated to improve shearability.

The chemical composition of this steel type is given in table 3.2

From the recorded data, both the load-displacement curve and the engineering stress-strain curve can be calculated. However, the load-displacement curve was used in the identification process to avoid errors related to conversion between different stress-strain measures.

Table 3.2: Chemical composition of steel applied steel type in percent

C	Si	Mn	P	S
0.14	0.35	0.7	0.045	0.045

The smoothed load-displacement curve obtained from the experiment is presented in figure 3.4. The curve was smoothed using Savitzky-Golay filter. This was done to remove noise and to make it possible to interpolate using standard functions.

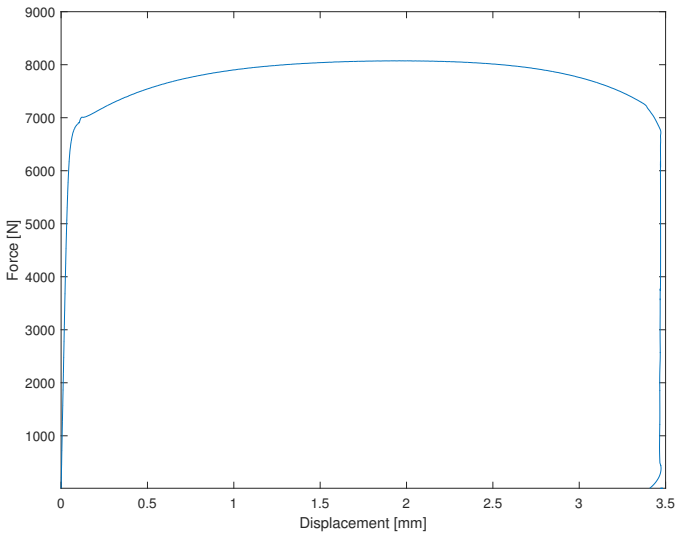
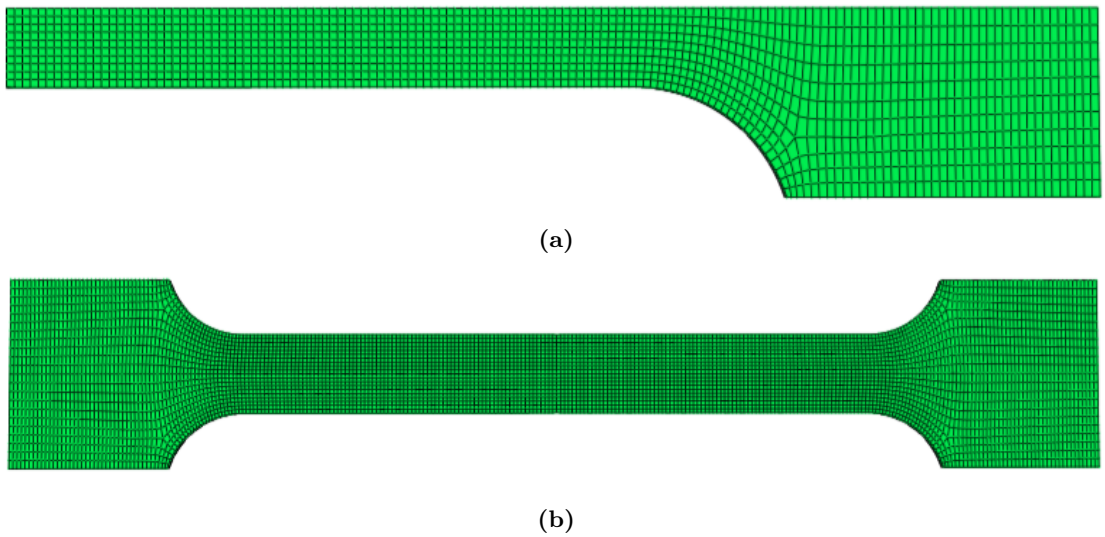


Figure 3.4: Load-displacement curve from experiment

From figure 3.4 it can be seen that the curve does not have a Lüders plateau, this is because the material undergoes work hardening in the shaping process.

3.1.2 Abaqus FEM simulation model

The FEM simulation model was based on script developed by the supervisor of this thesis, Sigmund Kyrre Ås. Due to symmetry about the three planes only 1/8 of the test specimen was modelled in the FEM software. This part is shown in figure 3.5a. A FEM model of the entire test specimen is shown in figure 3.5b.



(a)

(b)

Figure 3.5: (a) The part of the test specimen modelled in Abaqus FEA (b) The full test specimen displayed using mirroring about the three symmetry planes

In order to get the model shown in figure 3.5a to behave as the model shown in figure 3.5b, symmetry boundary conditions was applied along the three symmetry planes. This was implemented by setting the displacement vector perpendicular to the given plane and the rotations about the two axis parallel to the given plane to zero. In figure 3.6, the symmetry plane perpendicular to the x-axis is highlighted. To obtain the intended response the displacement of the surface in the x-direction and the rotations about the y-axis and z-axis is set to zero.

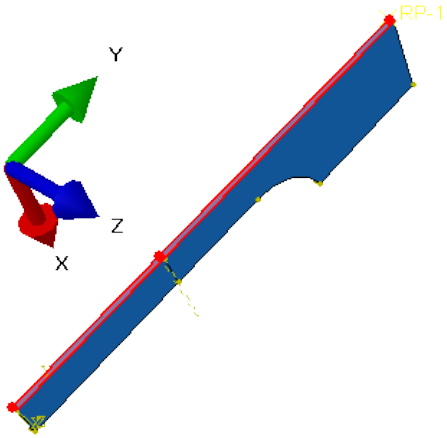


Figure 3.6: Symmetry plane perpendicular to the x-axis

The model was meshed using C3D8R elements, 8-node linear hexahedral solid element with reduced integration as shown in figure 3.7. This element was used because it is effective with respect to computational time. It is effective due to that it applies reduced integration and because it consist of relatively few nodes, thus leading to a relatively small set of equations to be solved. Generally, if two elements have similar shape and properties, the element with highest number of nodes will give the most accurate results. So, in order to check the accuracy of the model, a test was carried out to compare the model using the C3D8R elements with a model using more complex elements.

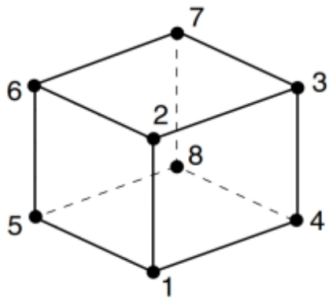


Figure 3.7: 8-noded linear hexahedral solid element with reduced integration (Fanaie et al., 2015)

In FEM modelling, a finer mesh generally results in a more accurate solution. However, the computational time increase as the mesh is made finer. In order to find a mesh size providing both accurate results and acceptable computational time a mesh convergence study was carried out. This was done by increasing the mesh density until the results converged satisfactorily. The convergence study was performed using a FEM model with a random generated material model.

Besides the global mesh size found from the convergence study, a command giving a refined mesh was applied to increase the mesh density in the reduced section of the specimen. This command is especially useful when the loads applied to one area are higher than another area.

This is clearly the case for a tensile test specimen due to the large ratio between the width in the reduced section and in the grip section.

A common problem when simulating tensile tests is that the neck forms outside the range of the extensometer. To avoid this problem the mesh was adjusted by moving two of the nodes on the surface of the specimen to create an initial geometrical non-uniformity. This is illustrated in figure 3.8.

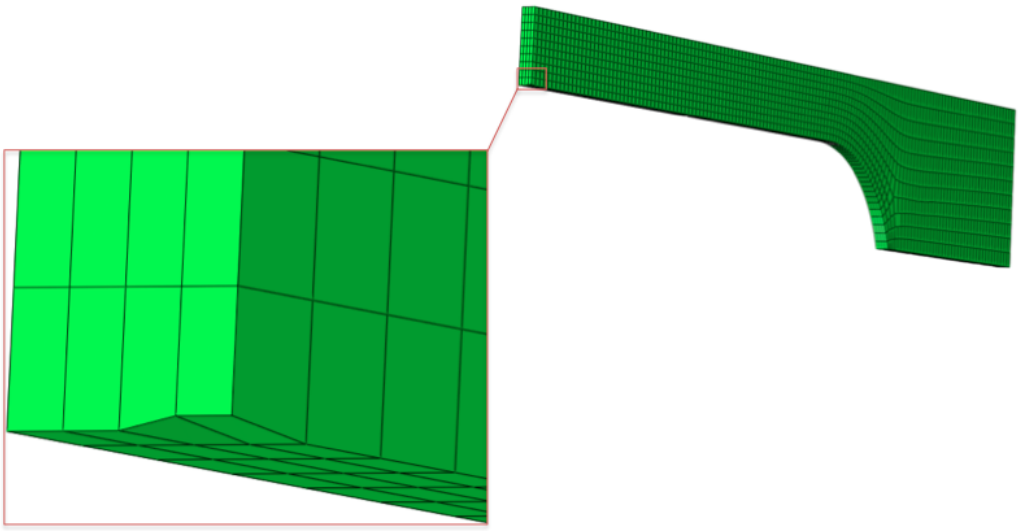


Figure 3.8: Adjusted mesh to ensure the neck forms within the range of the extensometer

The material constants used in the simulations are given in table 3.3. These were determined based on the given steel type.

Table 3.3: Constants used in Abaqus FEA model

	Value	Unit	Description
E	210	GPa	Youngs modulus
ν	0.3	-	Poissons number
σ_y	320	MPa	Yield strength

The material was further assumed to be homogeneous and elasto-plastic.

The FEM model was solved using an implicit solution scheme in which quantities in the current time step is calculated from the previous time step. The implicit method is unconditional stable, meaning that even if large time steps are applied the solution remains stable.

Database

In order to develop a database for training the ANN, FEM simulations were carried out for different set of material parameters. A parameter set fitted to minimise the deviations between the one-dimensional representation of the Chaboche model and the experimental data was used as a starting point for defining the ranges of the values used to develop the database. This set was found using a method published by Jiang and Kurath (1996) and applied to the nonlinear kinematic Chaboche model by Kalnins, Rudolph, and Willuweit (2015). The method consist of two main steps. In the first step the stress-plastic strain curve found from experiment is divided into M segments, where M is one more than the number of desired Chaboche components. This is illustrated in figure 3.9.

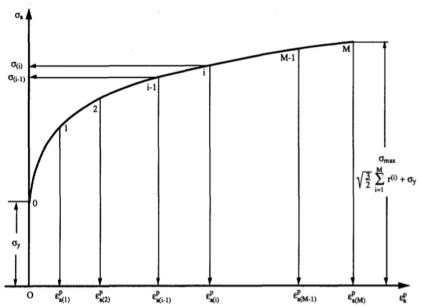


Figure 3.9: Stress-plastic strain curve divided in to M segments (Jiang & Kurath, 1996)

Then a first approximation of the Chaboche components $\{C_k,\gamma_k\}$ can be found from equation 3.1 and 3.2.

$$C_K = \left(\frac{\sigma_{(K)} - \sigma_{(K-1)}}{\epsilon_{p(K)} - \epsilon_{p(K-1)}} - \frac{\sigma_{(K+1)} - \sigma_{(K)}}{\epsilon_{p(K+1)} - \epsilon_{p(K)}} \right) \tag{3.1}$$

$$\gamma_k = \frac{1}{\epsilon_{p(k)}} \tag{3.2}$$

In the material model implemented in Abaqus FEA two Chaboche components were applied so the stress-plastic strain curve was divided into 3 segments. This is shown in figure 3.10. The corresponding stresses and plastic strains to the points marked in the figure are given in table 3.4.

Table 3.4: Stress-strain data used for calibration of the one-dimensional nonlinear kinematic Chaboche model

Point	Stress [MPa]	Plastic strain [%]
0	320.15	0.0
1	371.66	0.148
2	419.46	2.339
3	436.51	4.698

This resulted in the following initial parameter set:

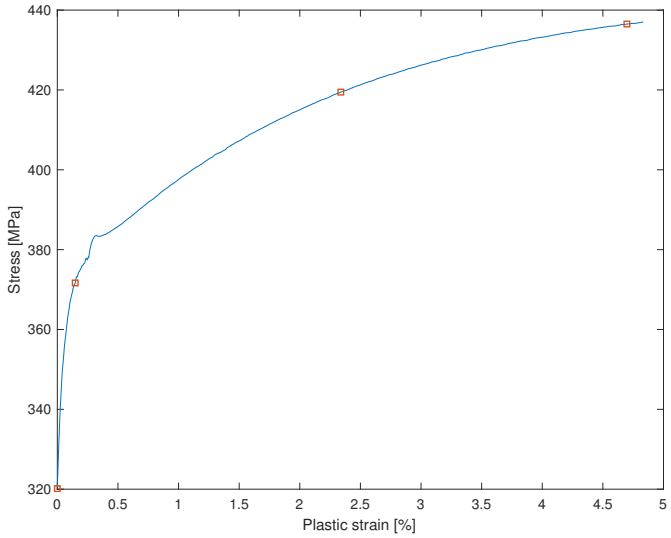


Figure 3.10: Stress-plastic strain curve divided into three segments

Table 3.5: Initial parameter set

Backstress component	C_k [MPa]	γ_k [-]
1	32682	676,80
2	1459	42,76

The one-dimensional nonlinear kinematic backstress curve (equation 3.3) can then be compared to the the backstress found from the experiment, where the one-dimensional representation of the backstress is given by equation 3.4.

$$\alpha_{NLK} = \sum_{K+1}^{N} \left(\frac{C_K}{\gamma_K} \right) [1 - exp(-\gamma_K \epsilon_p)] \tag{3.3}$$

$$\alpha = \sigma - \sigma_y \tag{3.4}$$

Where σ is the stress and σ_y is the initial yield stress.

From figure 3.11 it can be seen that the calculated backstress curve lies below the given curve so the parameters have to be further calibrated. This is the second step of the algorithm and it is done by changing the parameters until the deviation between the two curves is acceptable. In this thesis it was carried out using a build in optimisation algorithm in MATLAB.

The optimisation process resulted in the backstress curve presented in figure 3.12 and the parameters given in table 3.6.

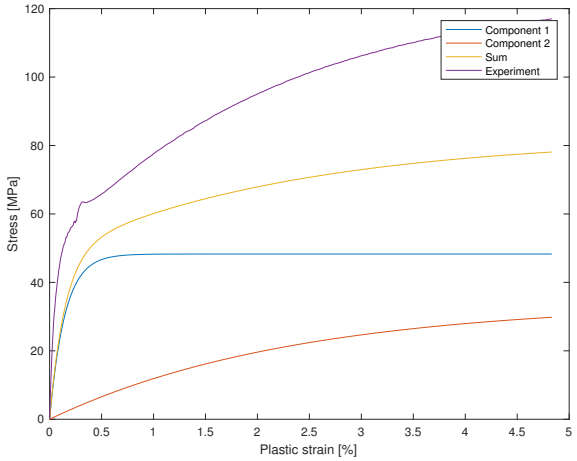


Figure 3.11: One-dimensional backstress curve, initial parameter set

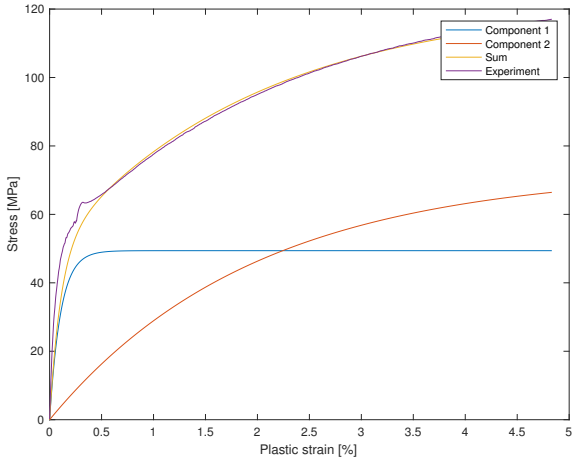


Figure 3.12: One-dimensional backstress curve, final parameter set

Table 3.6: Final parameter set

Backstress component	C_k [MPa]	γ_k [-]
1	46213	935,60
2	3685	50,67

After doing some initial FEM simulations and comparing the result to the experimental curve, the following parameter values were chosen to be used to develop the database.

Table 3.7: Parameter values used to develop the database of FEM simulations

Parameter	1	2	3	4	5	Unit
C_1	48000	51750	55500	59250	63000	MPa
γ_1	850	900	950	1000	1050	-
C_2	2900	3225	3550	3875	4200	MPa
γ_2	32.00	37.25	42.50	47.75	53.00	-

Combining all these parameter results in a total of $5^4 = 625$ different simulation models.

In order to compare the simulated and the experimental load-displacement curve, six pairs of load-displacement data were chosen to represent the curve. The displacement values are given in table 3.8 and the corresponding load values were found using linear interpolation.

Table 3.8: Displacement values used for comparing the simulated and the experimental load-displacement curve

Number	Value [mm]
1	0.25
2	0.50
3	0.75
4	1.00
5	1.25
6	1.48

3.1.3 Artificial neural network

In order to obtain an optimal generalisation, the available data was first split randomly into the three sample sets:

- Training samples 70 % (437 samples)
- Validation samples 15 % (94 samples)
- Testing samples 15 % (94 samples)

The training samples were used to train the ANN, the validation samples were used evaluate when the training should be stopped and the testing samples were held aside to after the ANN had been trained in order to analyse the performance.

The structure of the ANN is a very important factor when it comes to obtaining good results and the numbers of hidden neurons, training algorithm and transfer function all directly affect the network performance (Abbassi et al., 2013). In order to determine a suitable number of hidden neurons, a trial and error approach was applied. For different numbers of hidden neurons the ANN was trained ten times and an average score on the test samples were calculated. Then both the score and the number of hidden neurons were compared for the different ANNs and the lowest number of hidden neurons that still provided satisfying results was chosen to be implemented in the final ANN. The procedure showed that a hidden layer consisting of ten neurons was a reasonable choice.

The final structure of the ANN is shown in figure 3.13.

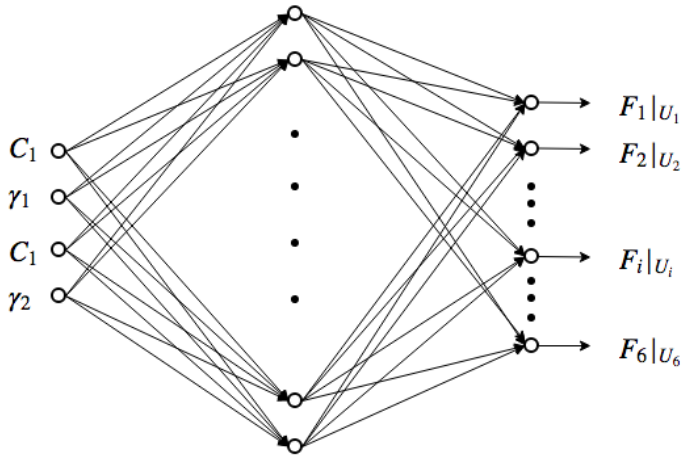


Figure 3.13: Illustration of ANN structure, showing inputs to and outputs from the network

The ANN has a (4-10-6) structure, consisting of 4 neurons in the input layer, 10 neurons in the hidden layer and 6 neurons in the output layer. The neurons of the input layer corresponds to the material parameters to be identified $\{C_1,\gamma_1,C_2,\gamma_2\}$. The neurons of the output layer represent the force corresponding to the six displacements given in table 3.8.

Furthermore a Levenberg-Marquardt algorithm was used to train the ANN, this is a variation of the backpropagation algorithm presented in section 2.4.4. The Hyperbolic Tangent Sigmoid function was used as the transfer function.

The ANN was developed using the Neural Network Toolbox in the commercial software MATLAB.

3.1.4 Genetic algorithm

The function to be optimised, the objective function, was defined as the sum absolute error between the experimental and the simulated load corresponding to the displacement points given in table 3.8, i.e.

$$F_{objective} = \sum_{i=1}^{6} \left| y_{ANN}(i) - y_{EXP}(i) \right| \tag{3.5}$$

Where $y_{ANN}(i)$ is the load predicted by the ANN and $y_{EXP}(i)$ is the load obtained from the experiment at point i.

The objective function was minimised as follows. First 50 individuals with random generated material parameters were created. This set of individuals was known as generation 0. Then the trained ANN was used to predict the load at the given displacement points for each individual and the error was calculated according to equation 3.5. The individuals with lowest error were then selected and used to obtain the next generation (generation 1).

The new generation was made up as follows:

1. 2 individuals comprises of the best individuals from the previous generation.
2. 38 individuals comprises of individuals created by crossover from selected parents. The process consist of combining the strings of a pair of parents. In this work the multi point crossover operator was used and then both the position and the number of digits to be changed were selected randomly. Figure 3.14 shows an example of how a new individual was created by combining three parts of the chromosomes of the parents. The figure also shows how the genotype, the continuous string (chromosome), is related to the phenotype, the material parameters.
3. The remaining 10 individuals were created using mutation. This was done by making a random change to one or more of the digits in an individuals chromosome in order to explore a wider solution area.

Individual	C_1	γ_1	C_2	γ_2
Parent 1	52522	937.2	3174.2	44.943
Parent 2	58638	998.6	3411.7	48.650

Parent 1: {52522, 937.2, 3174.2, 44.943}
Chromosome: {5252293723174244943}

Parent 2: {58638, 998.6, 3411.7, 48.650}
Chromosome: {5863899863411748650}

Parent 1: {5 $\overbrace{2522}^{4}$ 9 $\overbrace{372}^{3}$ 31 $\overbrace{74}^{2}$ 244943}
Parent 2: {5 86389 986 34 11 748650}
Son: {5 25229 372 34 74 748650}
= {52522, 937.2, 3474.7, 48.650}

Figure 3.14: Crossover example: shifts in position 2, 7 and 12 changing 4, 3 and 2 digits respectively

The objective function was optimised within the intervals given in table 3.9. These are the same ranges that was applied for developing the database used for training of the ANN, hence avoiding problems related to extrapolation of the ANN.

Table 3.9: Parameter ranges used for optimisation of material parameters

Parameter	Minimum	Maximum	Unit
C_1	48000	63000	MPa
γ_1	850	1050	-
C_2	2900	4200	MPa
γ_2	32.00	53.00	-

3.2 Results

3.2.1 Abaqus FEM simulations

Convergence study
The result from the convergence test is presented in table 3.10 and in figure 3.15. The load referred to in the table and in the plot is the load corresponding to a (extensometer) recorded deformation of 1 [mm].

Table 3.10: Convergence study

Mesh size [mm]	Load [N]
2.00	7807.36
1.50	7807.24
1.20	7807.53
1.00	7807.37
0.80	7807.14
0.70	7807.24
0.60	7807.29
0.55	7807.35
0.50	7807.38
0.45	7807.41
0.40	7807.15
0.30	7807.38
0.20	7807.41

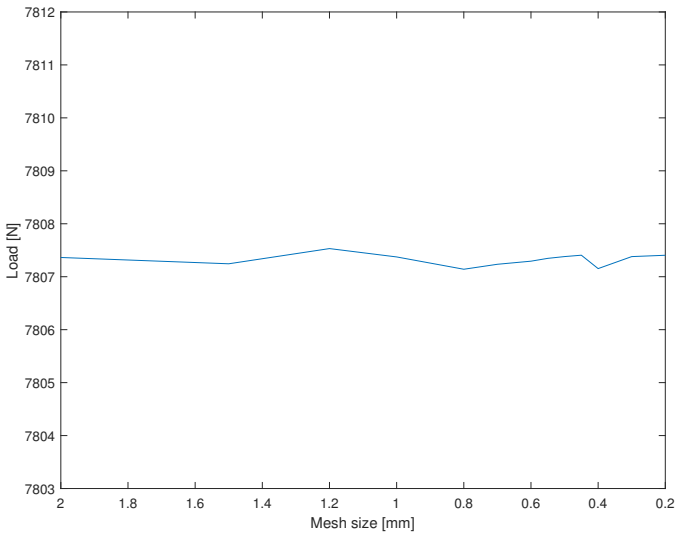


Figure 3.15: Convergence plot showing the load corresponding to a deformation of 1 [mm] for different mesh sizes

From the figure it can be seen that there are only small deviations between the results obtained using the different mesh sizes. Consequently, it seems like the results have converged on a relatively high mesh size. This could be because a refined mesh was used in the reduced section of the specimen and because the number of elements through the thickness was kept constant.

In the remaining simulations of the case study a mesh size of 0.5 [mm] was used even though the convergence test showed that the results converged at a much higher mesh size. This choice was motivated by the fact that a large number of simulations with different material models was going to be carried out and some of these simulations could have shown a different behaviour than the one used for the convergence study.

Figure 3.16 shows a comparison of the load-displacement curve obtained when the mesh was made up of elements having 8 nodes (C3D8R) and elements having 20 nodes (C3D20R). Generally if two elements have similar shape and properties the element with highest number of nodes will give the most accurate results.

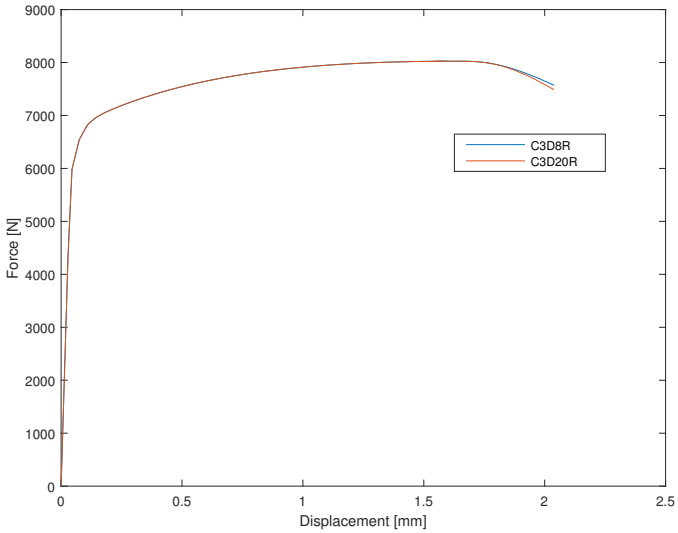


Figure 3.16: Comparison of the results obtained when using C3D8R and C3D20R elements

From figure 3.16 it can be seen that the two curves coincide in the hardening region. The curves differs slightly in the necking region, however, the aim of this case study was to identify the hardening parameters so the C3D8R element seems to be a reasonable choice because simulations using this element requires a lot less computational time.

Effect of adjusted mesh

The intention behind adjusting the mesh was to ensure that the neck formed within the range of the extensometer. However it was not intended to affect the shape of the load-displacement curve. Figure 3.17 shows an comparison of the results obtained with the original and the adjusted mesh.

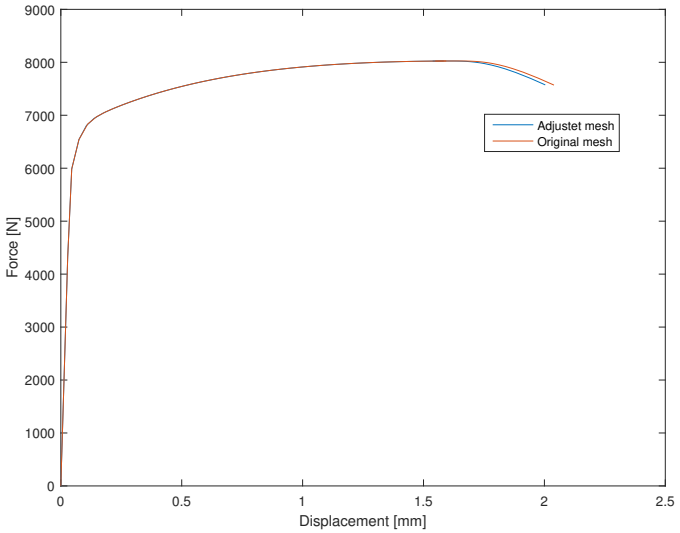


Figure 3.17: Figure showing the effect of adjusted mesh

From the figure it can be seen that the two curves coincide in the region of consideration.

3.2.2 Artificial neural network

The mean square errors (MSE) between the target and the output from the ANN for the different samples are given in table 3.11.

Table 3.11: Mean square error on sample sets

Sample	MSE [-]
Training	0.0073
Validation	0.0070
Testing	0.0078

From the table it can be seen that the final MSE on the three samples are small. This indicates that the network is capable of fitting the underlying trends.

In figure 3.18 the MSE for the three samples are plotted against the number of epochs. One epoch is defined as one forward pass and one backward pass of all the training data.

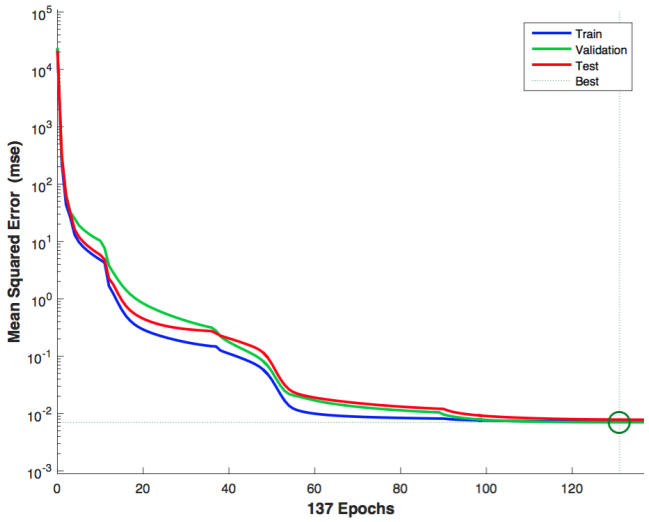


Figure 3.18: Plot of the performance of the ANN versus the number of epochs

From the figure it can be seen that the best validation performance was obtained after 131 epochs. By stopping the training at this point the network will perform well on new data and it will have less probability of over fitting. Furthermore, the curves do not indicate any notable problems with the training. The validation curve and the test curve have similar characteristics. If the curves on the other hand were significantly different this could be an indication of either over fitting or extrapolation.

Figure 3.19 illustrates the distribution of the network errors.

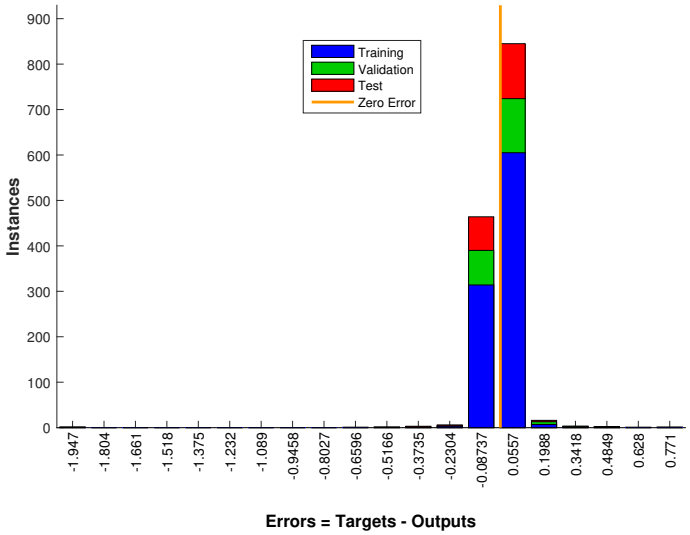


Figure 3.19: Plot of the distribution of the network errors

In the histogram the blue bars represents the training data, the green bars represents the validation data and the red bars represents the test data. The yellow vertical line marks the point of zero error. The histogram give an indication of the outliers, the data points having a significantly poorer fit than the main part of the set. From the figure it can be seen that most errors are small. However, there are a few outliers, these could be a result of problems with the simulations for example necking outside the extensometer.

In order to evaluate the reliability of the ANN, the load-displacement curve from a FEM simulation and from the trained ANN were compared using the same material model. The material model was generated randomly to avoid using the same inputs that were used for training because the network is expected to perform better close to these values. Figure 3.20 shows the results obtained from the FEM simulation (blue line) and the ANN (orange circles).

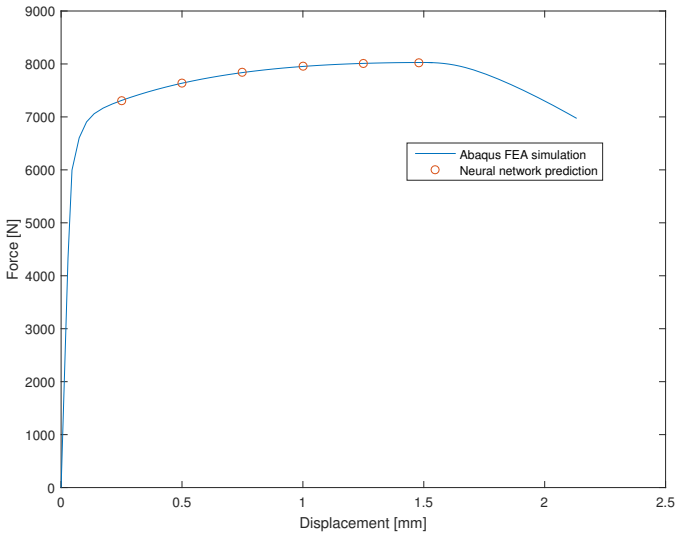


Figure 3.20: Plot for testing the reliability of the ANN

From the figure it can seen that the ANN is able to predict the results from the FEM simulation with a high accuracy.

3.2.3 Parameter identification

Figure 3.21 shows the evolution of the sum absolute error for the best individuals in each generation.

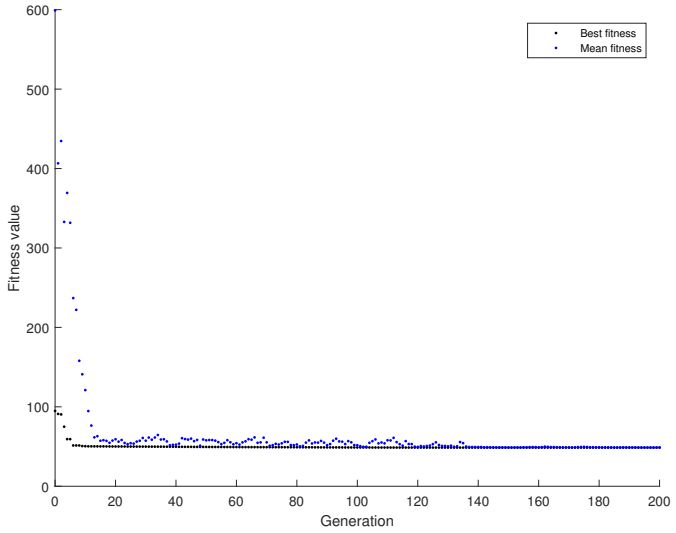


Figure 3.21: Plot showing the evolution of the sum absolute error for the best individuals in each generation

From the figure it can be seen that the minimum error was found after 137 generations. The minimum error correspond to the parameters given in table 3.12.

Table 3.12: Identified material parameters

Parameter	Value	Unit
C_1	50506.52	MPa
γ_1	935.31	-
C_2	3811.25	MPa
γ_2	34.04	-

Figure 3.22 shows a comparison of the load-displacement curve corresponding to the optimal parameters and the experimental load-displacement curve. The discrepancy between the two curves is minimised within the region delimited by the two black vertical lines. The figure shows that the two curves matches well within the region of consideration. However, for large displacements there is significant deviation between the two curves. From the figure it can be observed that the localised necking occurs at a much lower displacement for the simulated test than for the experimental test.

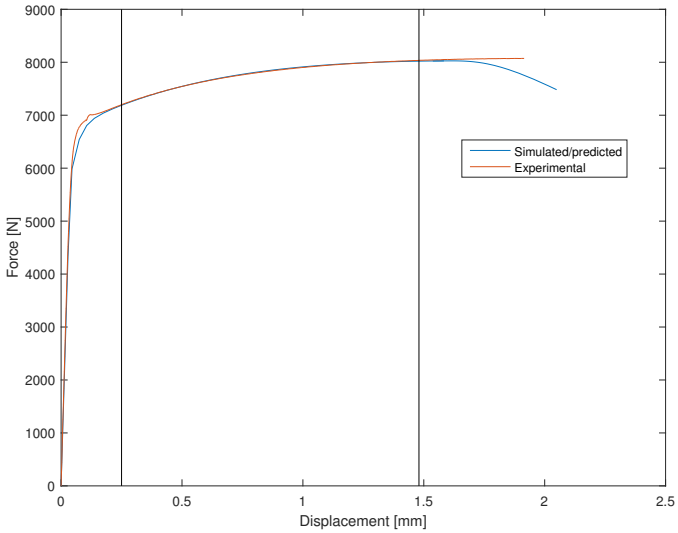


Figure 3.22: Comparison of the experimental load-displacement curve and the curve corresponding to the identified parameters

Figure 3.23 shows a closer look of the two curves within the region of consideration.

The discrepancy between the experimental and the simulated/predicted response was minimised by comparing six displacement points spaced uniformly along the region of consideration. By inspecting the deviations between the curves in figure 3.23 it can be seen that six points seems to be sufficient and it is not expected that the fit would improve if the number of points were increased.

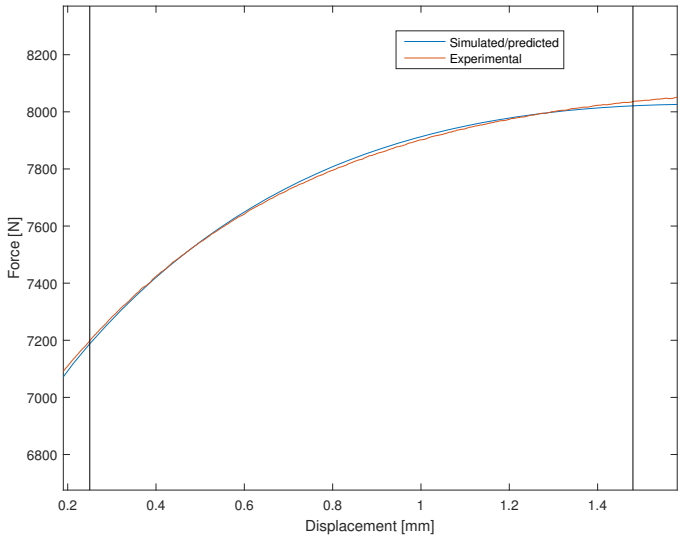


Figure 3.23: Comparison of the experimental load-displacement curve and the curve corresponding to the identified parameters within the region of consideration

By running the optimisation algorithm multiple times it was observed that different set of material parameters could lead to a nearly identical load-displacement curve.

In table 3.13 an alternative parameter set giving a similar sum absolute error is given.

Table 3.13: Alternative parameter set

Parameter	Value	Relative difference [%]
C_1	56134.71 [MPa]	10.56
γ_1	1027.02 [-]	9.35
C_2	3744.08 [MPa]	1.78
γ_2	33.46 [-]	1.72

The last column in the table gives the relative difference between the identified material parameters given in table 3.12 and the alternative parameter set.

Figure 3.24 presents a comparison of the load-displacement curve corresponding to the alternative parameter set and the experimental load-displacement curve.

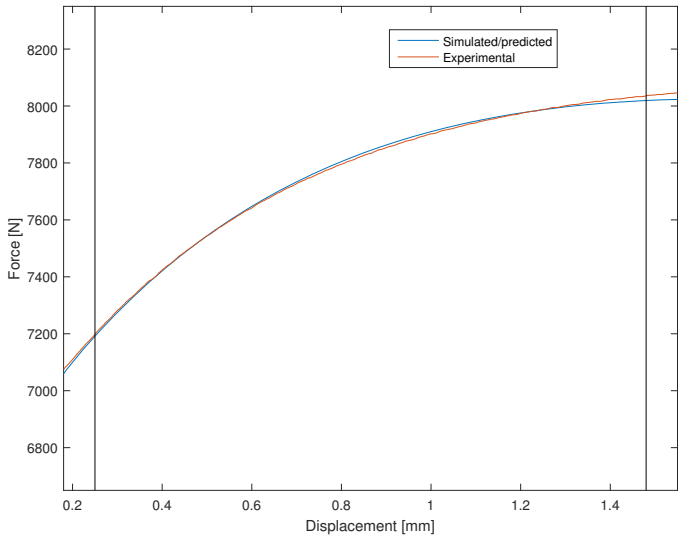


Figure 3.24: Comparison of the experimental load-displacement curve and the curve corresponding to the alternative parameter set

The figure shows that also the load-displacement curve corresponding to the alternative parameter set is able to match the experimental load-displacement curve well.

3.2.4 Comparison of the CPU time

The table below shows a comparison of the CPU time for the proposed method and a classical inverse identification method. The time estimates for the FEM simulations are based on the observed computing time for the simulations in this work. It is further assumed that the number of necessary generations in the optimisation process (for the results to stabilise) is four times larger using the proposed method than the classical inverse identification method due to the errors introduced by the ANN. This estimate is assumed to be conservative.

Table 3.14: Comparison of the CPU time

Training time for 500 simulations (min)	≈ 0.5	-
Population size	50	50
Number of generations	100	25
CPU time for one generation (min)	$50 * 0.0001$	$50 * 3$
CPU time for all generations (min)	$100 * 50 * 0.0001$	$25 * 50 * 3$
Total time (h)	≈ 0.01	≈ 63

From the table it can be seen that the total computing time using the proposed method is very low compared to the classical method where FEM simulations are an integrated part of the optimisation procedure. In the comparison it is assumed that there exists a database of FEM simulations which can be used to train the ANN. If only one material model is to be determined and a such database first has to be developed, there won't be a large difference in the computational time between the two methods. However, even if a database of FEM simulations is not available, the method will still be effective with respect to computational time if several similar material models are to be determined because the same database can be used for all the identification processes.

3.3 Concluding remarks

In this chapter a method for identifying the material parameters was presented and a case study was carried out to illustrate how the method can be used to find the parameters in the Chaboche hardening model. The main advantage with the proposed method is the reduction in computational time compared to the classical inverse identification methods if a database of FEM simulations is available or if the method is to be applied to determine several similar material models. In addition, the method works less as a black box. By inspecting the distribution of the errors between the target and the output from the ANN it is easy to identify and remove simulations showing an undesirable behaviour. These simulations could possibly lead to a slow convergence rate or even cause errors in a classical identification method.

The results from the case study showed that the load-displacement curve corresponding to the identified parameters matched the experimental curve well within the area of consideration. However, the material model was insufficient to model the behaviour of the material for large strains. During the study it was also observed that different set of material parameters could give a nearly identical load-displacement curve. So it is therefore recommended to verify the identified parameters with another validation.

Chapter 4

Case study - GTN-model

The GTN-model described in section 2.2 includes nine parameters. Three parameters (f_0,f_c,f_f) describe the evolution of damage up to coalescence and final failure, three (ϵ_n,f_n,s_n) are used to model void nucleation and the remaining three (q_1,q_2,q_3) characterise the yield behaviour of the material.

A combined strategy using both numerical simulations and experimental data is necessary in order to determine some of the parameters. The remaining parameters can be determined from suggestions in the literature. Table 4.1 gives some values of the parameters for steel found in literature.

Table 4.1: GTN parameters for steel found in literature

Reference	q_1	q_2	ϵ_N	s_N	f_0	f_f	f_N	f_c
Skallerud and Zhang (1997)	1.25	1	0.3	0.1	0.0003	0.15	0.006	0.026
Schmitt, Sun, and Blauel (1997)	1.5	1	0.3	0.1	0	0.212	0.002	0.06
Betegon, Rodriguez, and Belzunce (1997)	1.5	1	0.3	0.1	0.0015	0.15	0.00085	0.035
Benseddiq and Imad (2008)	1.5	1	0.3	0.1	~ 0	~ 0.2	0.0002-0.002	0.004-0.06

From the table it can be observed that $q_1 = 1.5$ is the standard value for steel, this is also the value suggested by Tvergaard and Needleman (1984). Furthermore $q_2 = 1$ and $q_3 = (q_1)^2$ in the studies. In this case study f_0 was also fixed to zero. This is also done in research by Cuesta et al. (2010) and Zhong et al. (2016) with good results. The parameters to be identified were then ϵ_N, s_N, f_f, f_N and f_c.

4.1 Method

The method is similar to the one applied for identifying the Chaboche hardening parameters. However, some minor changes are done to optimise the method for the given material model. These changes will be specified in the following subsections.

4.1.1 Experimental data

The tensile test was done as a part of a demonstration performed by the company LaVision. In the test, the displacement field was measured using a technique called Digital Image Correlation (DIC). The basic principle of the technique is that the displacement field can be found by tracking the movement of applied surface patterns. In practise, this is done by taking a large number of pictures of the specimen during the experiment and then use a software for searching for and matching the patterns.

The principle is illustrated in figure 4.1.

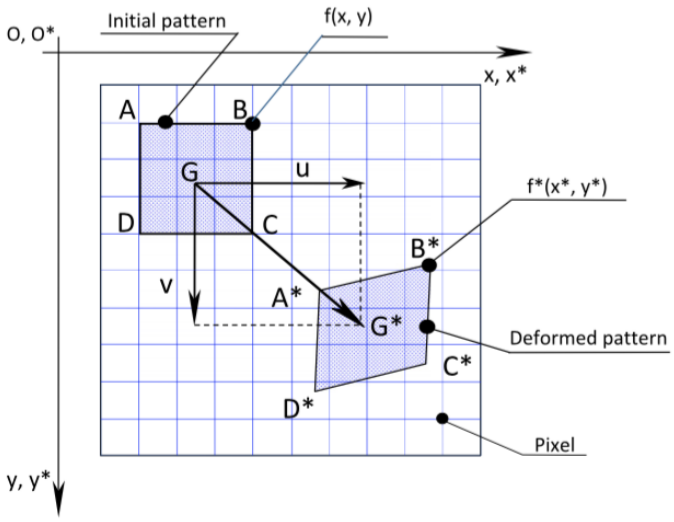


Figure 4.1: Initial and deformed pattern (Abbassi et al., 2013)

The figure shows the initial pattern (A,B,C,D) and deformed pattern (A*,B*,C*,D*). By comparing the two patterns as in equation 4.1,

$$\begin{aligned} x^* &= x + u(x, y) \\ y^* &= y + v(x, y) \end{aligned} \tag{4.1}$$

it is possible to find the displacement fields $u(x, y)$ and $v(x, y)$.

The applied surface pattern that was used in the experiment is shown in figure 4.2.

A notched test specimen was applied in the experiment. This geometry was used because it ensures that the neck forms exactly in the middle of specimen. Hence providing robustness and repeatability of both the experiment and the simulations. The dimensions of the specimen is given in table 4.2.

The notch width in the table denotes the width in the middle of the notch.

The test specimen was made out of the same steel type as the one used for the Chaboche case study.

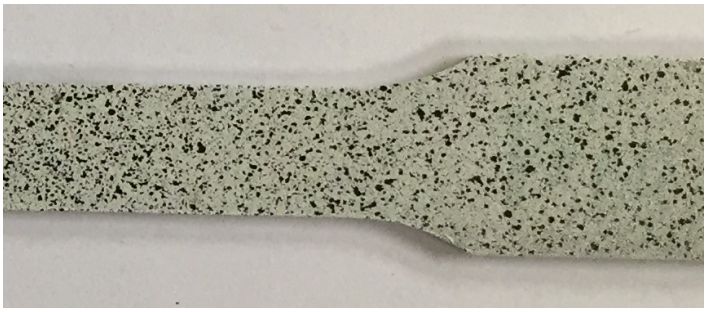


Figure 4.2: Test specimen pattern used for DIC

Table 4.2: Dimensions of the test specimen

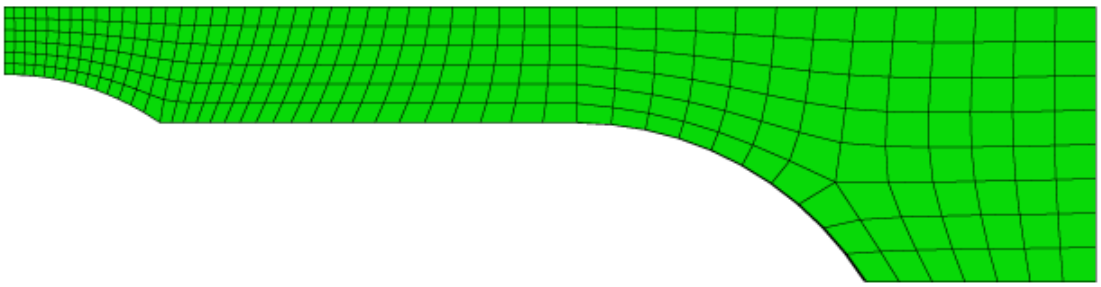
Description	Value [mm]
Overall length	95.3
Width of grip section	24.0
Distance between shoulders	75.3
Reduced section	50.0
Gage length	50.0
Width	10.0
Notch width	5.87
Notch radius	12.0
Thickness	1.82

4.1.2 Abaqus FEA simulation model

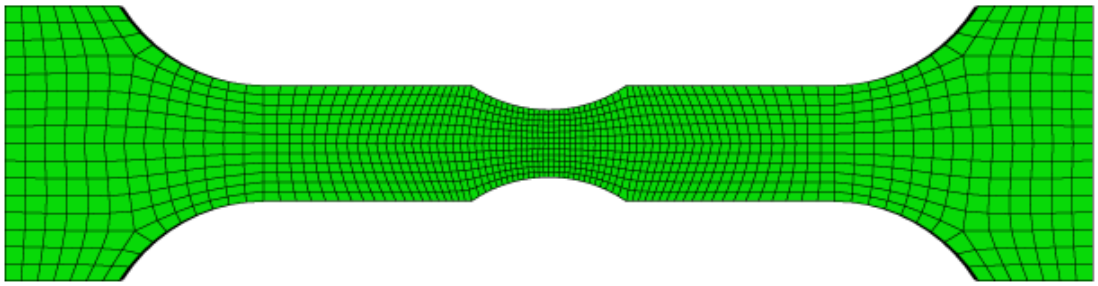
The FEM-model was based on the same script that was used for the Chaboche case study. The geometry of the test specimen is axisymmetric, hence it is sufficient to develop an axisymmetric finite element model. The part of the specimen modelled in Abaqus is presented in figure 4.3a and the entire specimen is shown in figure 4.3b.

The uniaxial plastic flow behaviour was assumed to follow the power law (equation 2.9) and the stress-plastic strain points entered in the FEM model are given in table 4.3.

In order to model fracture in a realistic manner, a command was used for removing elements when the damage variables associated with fiber failure reached a predefined maximum. This is known as element deletion in Abaqus.



(a)



(b)

Figure 4.3: (a) The part of the test specimen modelled in Abaqus FEA (b) The full test specimen displayed using mirroring about the three symmetry planes

Table 4.3: Hardening model

Stress [MPa]	Plastic strain [-]
320.0	0.00000
330.0	0.00023
340.0	0.00056
350.0	0.00103
360.0	0.00168
370.0	0.00258
380.0	0.00381
390.0	0.00549
400.0	0.00775
410.0	0.01079
420.0	0.01482
430.0	0.02016
440.0	0.02715
450.0	0.03628
460.0	0.04812
470.0	0.06338
480.0	0.08295

The material constants used in the simulation are given in table 4.4

Table 4.4: Constants used in the Abaqus FEA model

	Value	Unit	Description
E	210	GPa	Youngs modulus
ν	0.3	-	Poissons number

The material was further assumed to be homogeneous.

The FEM model was solved using an explicit solution scheme due to the low computational time when solving quasi-static problems (Zhong et al., 2016). However, the explicit method is not unconditionally stable so it is essential to use small enough time steps.

The parameter values used to develop the database of FEM simulations are given in table 4.5. These were chosen in accordance with previous research (table 4.1).

Table 4.5: Parameter values used to develop the database of FEM simulations

Parameter	1	2	3	4
ϵ_N	0.25	0.30	0.35	-
S_N	0.085	0.095	0.105	-
f_N	0.002	0.008	0.014	0.020
f_f	0.13	0.17	0.21	0.25
f_c	0.004	0.026	0.048	0.070

4.1.3 Artificial neural network

The available data was split randomly into three sample sets:

- Training samples 70 % (404 samples)
- Validation samples 15 % (86 samples)
- Testing samples 15 % (86 samples)

A similar procedure as described in section 3.1.3 was used to determine a suitable number of hidden neurons and it was concluded that 10 hidden neurons was a reasonable choice.

The structure of the ANN is shown in figure 4.4.

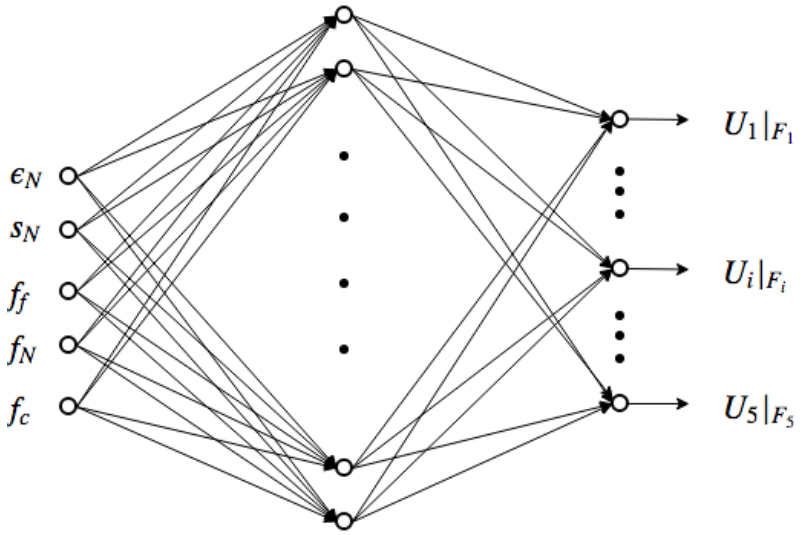


Figure 4.4: Plot showing the load-displacement curve for all the simulations and the experimental load-displacement curve

The ANN has a (5-10-5) structure, consisting of 5 neurons in the input layer, 10 neurons in the hidden layer and 5 neurons in the output layer. The neurons of the input layer corresponds to the material parameters to be identified $\{\epsilon_N, s_N,\ f_f,\ f_N, f_c\}$. The neurons of the output layer represent the displacements corresponding to the five loads given in table 4.6.

It should be noted that the representation of the the output layer is different to the representation that was used in the Chaboche case study ($U_i|_{F_i}$ instead of $F_i|_{U_i}$). This is because load-displacement curve is relatively vertical in the area where the GTN-parameters affect the curve, while it is more horizontal in the hardening part of the curve.

Furthermore, a Levenberg-Marquardt algorithm was used to train the ANN and the Hyperbolic Tangent Sigmoid function was used as the transfer function.

Table 4.6: Load values used for comparing the simulated and the experimental load-displacement curve

Number	Value [N]
1	4700
2	4450
3	4200
4	3950
5	3700

4.1.4 Genetic algorithm

The objective function, was defined as the sum absolute error between the experimental and the simulated displacements corresponding to the load points given in table 4.6, i.e.

$$F_{objective} = \sum_{i=1}^{5} \left| y_{ANN}(i) - y_{EXP}(i) \right| \tag{4.2}$$

Where $y_{ANN}(i)$ is the displacement predicted by the ANN and $y_{EXP}(i)$ is the displacement obtained from the experiment at point i.

The objective function was optimised in a similar way as in the Chaboche case study (section 3.1.4).

However initial tests showed that it was beneficial to increase the mutation rate to ensure that the algorithm found the global minimum. The new generation was then made up as follows:

1. 2 individuals comprises of the best individuals from the previous generation.
2. 34 individuals comprises of individuals created by crossover from selected parents.
3. The remaining 14 individuals were created by mutation.

The objective function was optimised within the intervals given in table 4.7. These are the same ranges that was applied for developing the database.

Table 4.7: Parameter ranges used for optimisation of material parameters

Parameter	Minimum	Maximum
ϵ_N	0.25	0.35
S_N	0.085	0.105
f_N	0.002	0.020
f_f	0.13	0.25
f_c	0.004	0.07

4.2 Results

Figure 4.5 shows the load-displacement curve for all the 576 different simulations and the load-displacement points recorded during the experiment.

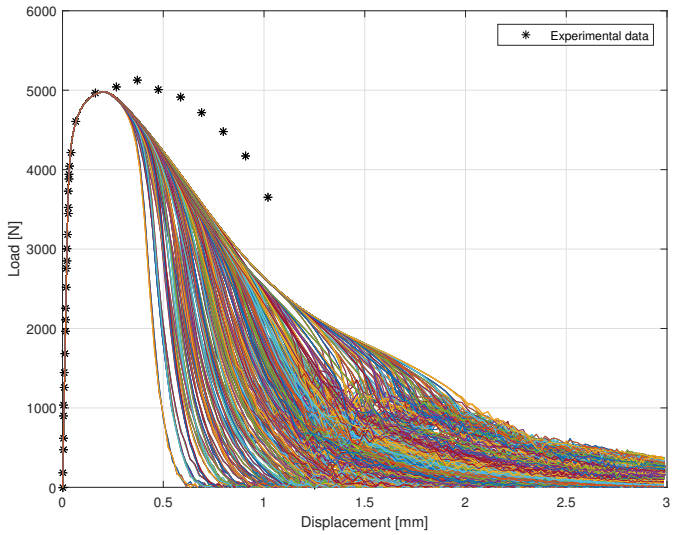


Figure 4.5: Plot showing the load-displacement curve for all the simulations and the experimental load-displacement points

From the plot it can be seen that the experimental test shows a completely different behaviour than the simulated tests. Consequently, it will be impossible to identify the material parameters using the information obtained from the simulations and the experiment.

Possible reasons for the discrepancy between the experimental and the simulated results could be:

- Poor choice of model parameters
- Limitations of the finite element formulation
- Hardening model unable to represent post-necking material response

From the plot the it can be seen that all the load-displacement curves obtained from the simulations coincide until the displacement reach approximately 0.3 [mm]. It can therefore be concluded that the discrepancy is not related to the parameters varied throughout the simulations ($\epsilon_N, s_N, f_f, f_N$ and f_c). However the discrepancy could still be related to the fixed parameters (q_1, q_2, q_3, f_0). So in order test the effect of theses parameters, each parameter was varied while the others were kept constant. The results from the tests showed that despite the parameters were varied way outside the range recommended in the literature it only had minor effect on the load-displacement curves.

Figure 4.6 and 4.7 shows the axial strain in the vertical direction for the experimental specimen and the FEM model for a somewhere similar load just prior to failure.

From the figures it can be observed that the two strain fields look really similar at the given load. It was also observed that the FEM model exhibited a realistic response before local necking.

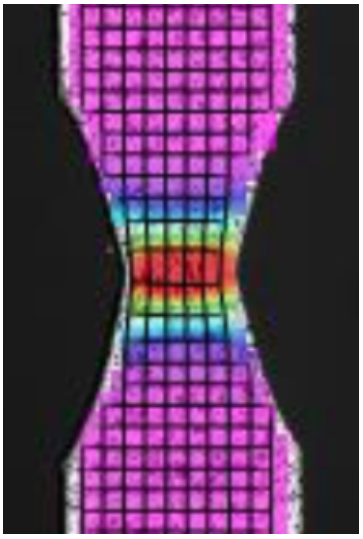


Figure 4.6: DIC - axial strain in notched region prior to failure

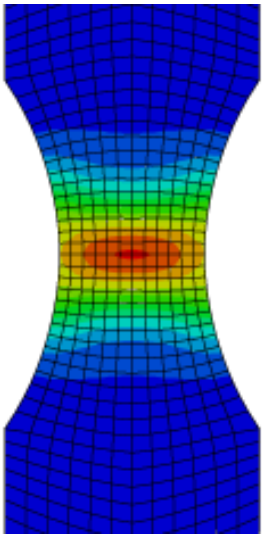


Figure 4.7: FEM - axial strain in notched region prior to failure

However it is still difficult to exclude that the difference between the results was due to limitations of the finite element formulation. It should also be kept in mind that the experiment was a part of a demonstration so only a limited amount of data was recorded during the test, making it even harder to compare the results.

Another possible reason for difference is that the applied hardening model was insufficient to model the plastic flow behaviour of the material. This behaviour was obtained by defining a hardening curve (table 4.3) in the region before necking and then extrapolate this curve to the post necking region. Obviously, this simple extrapolation method do not guarantee a correct prediction of the post necking hardening behaviour. However, the combination of the power law and the extrapolation method is known to provide good results for less ductile steel types.

Multiple adjustments were performed in order to improve the hardening model. For example simulations were performed with different values of the hardening parameters (k and n) in the power law, tests were also performed with a Chaboche hardening law instead of the power law, however non of these adjustment improved the result.

A possible reason that the applied hardening laws (Power law and Chaboche) were not able to model the plastic flow behaviour of the material is because they have a steep stress-strain gradient. As mentioned in section 2.1 the material's resistance will increase when it is deformed from the yield strength until the point of the ultimate strength. When the stress-strain gradient is large the point of the ultimate strength will occur at small a strain, hence giving a less ductile response. On the other hand, a hardening law having a less steep gradient (for example the Voce law) would result in a more ductile response. No tests were performed with a hardening law having a less steep gradient because it would be difficult to calibrate a more complex hardening function using the limited amount of available data.

Another frequently used method for improving the post-necking hardening model is to apply a correction. For example Bridgman (1964) proposed a correction method for symmetrical test specimen based on geometric considerations of the deformation pattern and analytical formulas

for the stress distribution at the neck (Cabezas & Celentano, 2004). However some of the variables included in the model were not measured during the experiment (these are extremely difficult to measure) so it was not possible to apply the correction on the given problem.

The aim of the case study was to investigate how the proposed method can be used predict the GTN-model. So in order to test the proposed method on the GTN-model, all the recorded displacement points were divided by two so that the adjusted load-displacement points were localised within the range of the simulations.

Figure 4.8 shows the adjusted experimental load-displacement points and the two simulations giving the minimum and maximum displacement.

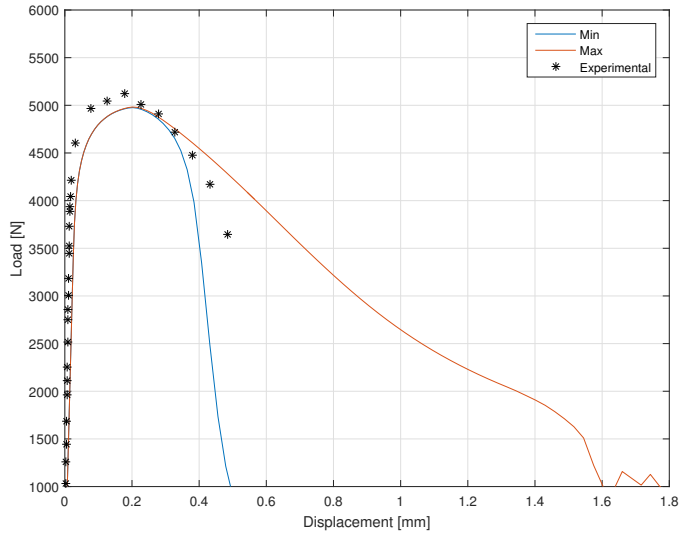


Figure 4.8: Plot of the adjusted experimental load-displacement points and the two simulations giving the minimum and maximum displacement

Figure 4.9 shows the same quantities in the region of consideration.

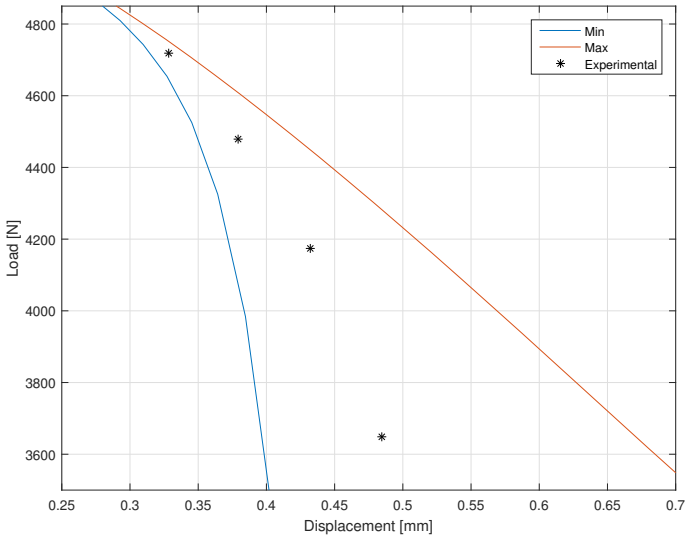


Figure 4.9: Pot of the adjusted experimental load-displacement points and the two simulations giving the minimum and maximum displacement in the region of consideration

4.2.1 Artificial neural network

The mean square errors (MSE) between the target and the output from the ANN for the different samples are given in table 4.8.

Table 4.8: Mean square error on sample sets

Sample	MSE [-]
Training	$9.75 * 10^{-7}$
Validation	$9.85 * 10^{-7}$
Testing	$9.75 * 10^{-7}$

The MSE gives the mean square difference between the simulated displacement and the displacement predicted by the ANN. From the table it can be observed that the error on all the sample sets are very small.

In figure 4.10 the MSE for the three samples are plotted against the number of epochs.

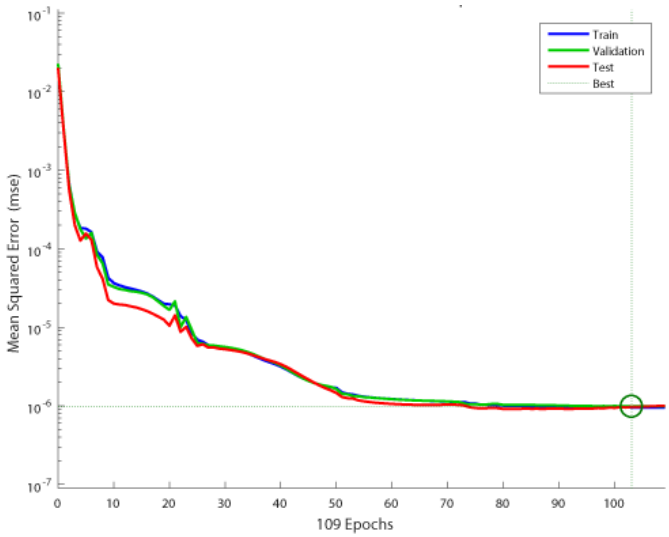


Figure 4.10: Plot of the performance of the ANN versus the number of epochs

The figure shows that the best validation performance was obtained after 104 epochs. The training was therefore stopped at this point to prevent over fitting. Furthermore, it can be seen that all the curves have similar characteristics, indicating that the ANN neither over fits or extrapolate. It can also be observed that the final errors are small so it can be expected that the ANN will perform well on new similar data it has never seen before.

Figure 4.11 shows the distribution of the network errors.

The figure shows that errors are distributed within a reasonable range around zero.

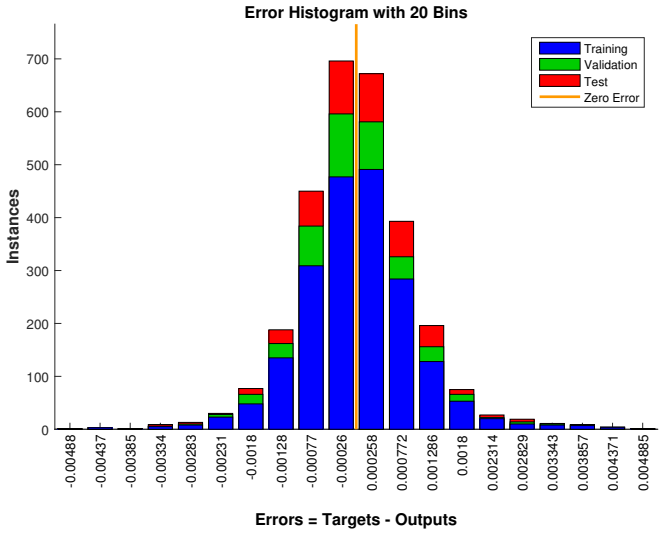


Figure 4.11: Plot of the distribution of the network errors

4.2.2 Parameter identification

Figure 4.12 shows the evolution of the sum absolute error for the best individuals in each generation.

From the figure it can be seen that the solution stabilised after approximately 100 generations. It can further be observed that the final sum absolute error is small indicating that the discrepancy between the simulated and adjusted experimental results is small. The minimum error corresponds to the parameters given in table 4.9.

Table 4.9: Identified material parameters

Parameter	Value
ϵ_N	0.30967
S_N	0.08850
f_N	0.01975
f_f	0.20397
f_c	0.00537

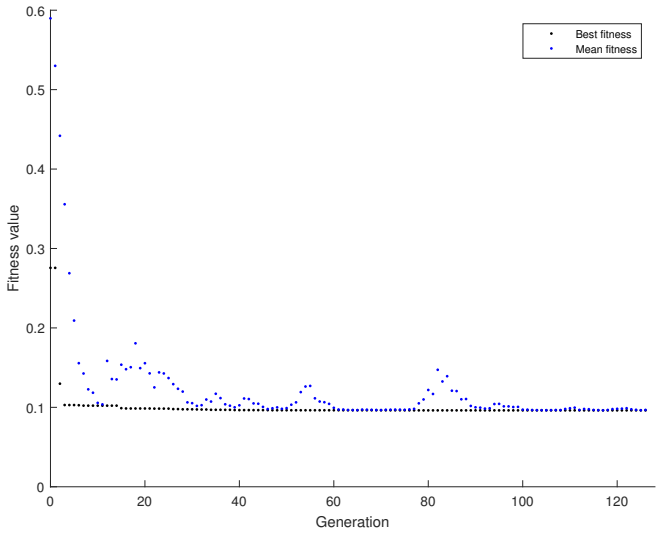


Figure 4.12: Plot showing the evolution of the sum absolute error for the best individuals in each generation

Figure 4.13 shows a comparison of the load-displacement curve corresponding to the identified parameters and the adjusted experimental load-displacement curve. The discrepancy between the two curves is minimised within the region delimited by the black rectangle. From the figure it can be seen that the two curves matches well within the region of consideration.

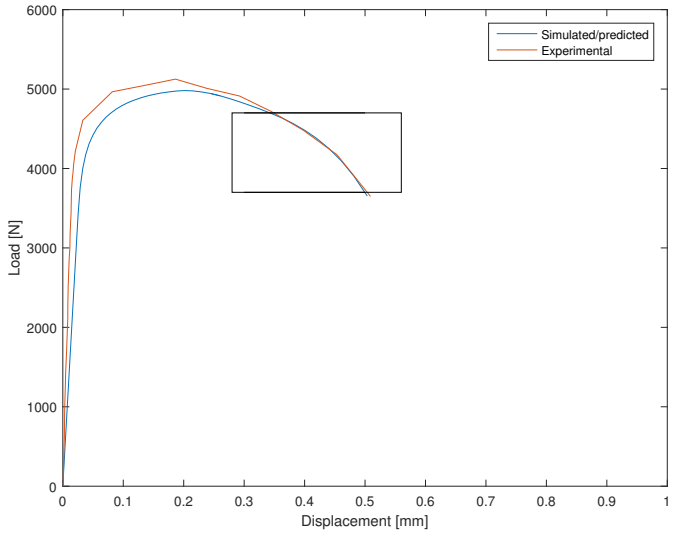


Figure 4.13: Comparison of the experimental load-displacement curve and the curve corresponding to the identified parameters

Figure 4.14 shows a closer look of the region of consideration.

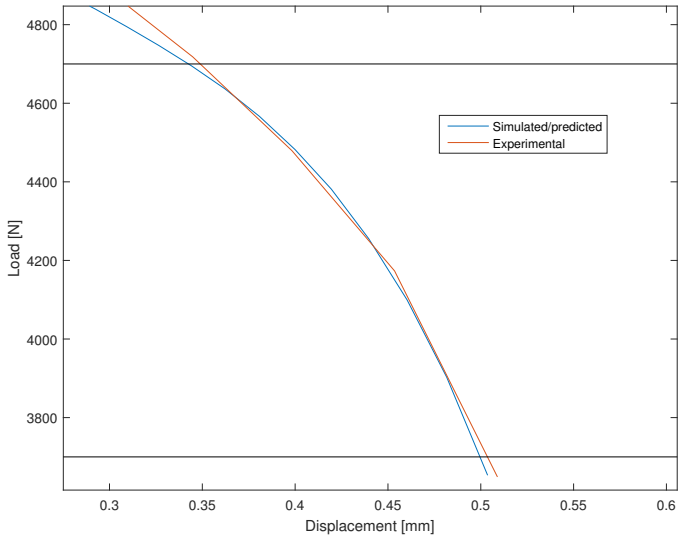


Figure 4.14: Comparison of the experimental load-displacement curve and the curve corresponding to the identified parameters within the area of consideration

By running the optimisation algorithm multiple times it was yet again observed that different set of material parameters could lead to a nearly identical load-displacement curve.

4.3 Concluding remarks

In this chapter a case study was carried out aiming to illustrate how the identification method proposed in chapter 3 can be used to identify the parameters in the GTN-model.

By comparing the load-displacement curves obtained from the simulations and from the experiment it was observed that the experimental test showed a completely different behaviour than the simulated tests. The discrepancy between the experimental and the simulated results could have several possible reasons, however, it seems reasonable to believe that it was due to that the hardening model was unable to represent the post-necking material response.

Due a to large discrepancy between the simulated and experimental results it would be impossible to identify the material parameters using the obtained data. So in order to evaluate how the proposed method perform on the GTN-model the experimental data was adjusted so that it was located within the range of the simulations. The results showed that the load-displacement curve corresponding to the identified parameters matched the adjusted experimental curve well, however, it was yet again observed that different set of material parameters could lead to a nearly identical load-displacement curve.

Chapter 5

Conclusion and recommendations for further work

5.1 Summary and conclusion

In this thesis a method for identifying material parameters based on ANN, GA, numerical simulations and experimental tests is presented. The main advantage of the proposed method compared to a more classical inverse identification method is the reduction in computational time. Furthermore, the method provides the user with simple tools for identifying and removing simulations showing an undesirable behaviour, these could possibly lead to a slow convergence rate or even cause errors in a classical identification method.

Two case studies were carried out in order to test the method and to illustrate how it can be applied. The aim of the first case was to identify the hardening parameters in the Chaboche model. The results showed that the load-displacement curve corresponding to the identified parameters matched the experimental curve well within the area of consideration. However, the material model was unable to model the behaviour of the material for large strains.

The second case aimed at finding the parameter in the GTN-model. By comparing the load-displacement curves obtained from the simulations and from the experiment, it was observed that the experimental test showed a completely different behaviour than the simulated tests. It was concluded that the discrepancy between the experimental and the simulated results was most likely due to that the hardening model was unable to represent the post-necking material response.

Due to a large discrepancy between the simulated and experimental results it would be impossible to identify the material parameters using the obtained data. So in order to evaluate how the proposed method perform on the GTN-model the experimental data was adjusted so that it was located within the range of the simulations. The results showed that the load-displacement curve corresponding to the identified parameters matched the adjusted experimental curve well within the region of consideration.

In both cases it was observed that different set of material parameters could give a nearly identical load-displacement curve, hence it is recommended to extend the method so that it uses

more quantities in the comparison between the simulated and experimental result. However, by making this adjustment it is expected that the method can outperform conventional methods when computational time and accurate results are both emphasised. Therefore this strategy should be considered in the future.

5.2 Recommendations for further work

From the two case studies it was observed that different set of material parameters could lead to a nearly identical load-displacement curve. It is therefore recommended to extend the method so that it uses more quantities in the comparison between the simulated and experimental results. A possible way to include this in the proposed method is to use more than one experimental test in the identification procedure. Another possible option is to include more parameters in the comparison between the experimental and the simulated response; for example the actual width of the specimen at different points could be used in addition to the load-displacement curve.

During the process of running the simulations and training the ANN, lack of guidelines for design of methodology made it difficult to know how many simulations were necessary to obtain accurate results. For further work it is therefore recommended to perform some sensitivity analysis to determine the number of necessary simulations for identifying the parameters in different material models.

Chapter 6

References

References

Abbassi, F., Belhadj, T., Mistou, S., & Zghal, A. (2013). Parameter identification of a mechanical ductile damage using artificial neural networks in sheet metal forming. *Materials and Design*, *45*.

Aguir, H., Belhadjsalah, H., & Hambli, R. (2011). Parameter identification of an elasto-plastic behaviour using artificial neural networks-genetic algorithm method. *Materials and Design*, *32*(1).

ANSYS. (2009). *Theory reference for the mechanical apdl and mechanical applications* (12.0 ed.).

AUTODESK. (2018). *von mises material properties.* `https://knowledge.autodesk.com/support/simulation-mechanical/learn-explore/caas/CloudHelp/cloudhelp/2018/ENU/SimMech-UsersGuide/files/GUID-8904EDCE-0565-4530-8CFA-24919DD5BF01-htm.html`. (Accessed: 2018-20-05)

Bell, K. (2013). *An engineering approach to finite element analysis of linear structural mechanics problems.* Akademika Publishing.

Benseddiq, N., & Imad, A. (2008). A ductile fracture analysis using a local damage model. *International Journal of Pressure Vessels and Piping*, *85*(4).

Betegon, C., Rodriguez, C., & Belzunce, F. J. (1997). Analysis and modelisation of short crack growth by ductile fracture micromechanisms. *Fatigue and Fracture of Engineering Materials and Structures*, *20*(5), 633–644.

Bishop, C. M. (1995). *Neural networks for pattern recognition.* Oxford University Press.

Bishop, C. M. (2006). *Pattern recognition and machine learning.* Springer.

Bridgman, P. (1964). *Studies in large plastic flow and fracture : with special emphasis on the effects of hydrostatic pressure.* Cambridge, Mass: Harvard Univ. Press.

Brocks, W., & Steglich, D. (2007). *10.05 - hybrid methods* (Vol. 10). Pergamon ; Elsevier.

Cabezas, E. E., & Celentano, D. J. (2004). Experimental and numerical analysis of the tensile test using sheet specimens. *Finite Elements in Analysis and Design*, *40*(5), 555–575.

Cuesta, I. I., Alegre, J. M., & Lacalle, R. (2010). Determination of the gurson-tvergaard damage model parameters for simulating small punch tests. *Fatigue and Fracture of Engineering Materials and Structures*, *33*(11).

Demuth, H. B., Beale, M. H., De Jess, O., & Hagan, M. T. (2014). *Neural network design* (2nd ed.). Martin Hagan.

Fanaie, N., Ghalamzan Esfahani, F., & Soroushnia, S. (2015). Analytical study of composite beams with different arrangements of channel shear connectors. *Steel and Composite Structures*, *19*.

Faurholdt, T. G. (2000). Inverse modelling of constitutive parameters for elastoplastic problems. *The Journal of Strain Analysis for Engineering Design*, *35*(6).

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning.* Addison-Wesley.

Gurson, A. (1977). Continuum theory of ductile rupture by void nucleation and growth: Part 1 - yield criteria and flow rules for porous ductile media. *Journal of Engineering Materials and Technology, Transactions of the ASME*.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*.

Irgens, F. (2008). *Continuum mechanics.* Springer Berlin Heidelberg.

Jiang, Y., & Kurath, P. (1996). Characteristics of the armstrong-frederick type plasticity models. *International Journal of Plasticity*, *12*(3).

Kalnins, A., Rudolph, J., & Willuweit, A. (2015). Using the nonlinear kinematic hardening material model of chaboche for elastic-plastic ratcheting analysis. *Journal of Pressure Vessel Technology*, *137*(3).

Karpathy, A. (2018). *Neural networks part 1: Setting up the architecture.* Retrieved 2018-08-04, from `http://cs231n.github.io/neural-networks-1/`

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In (Vol. 4). IEEE Publishing.

Lagace, P. (2008). *Yield (and failure) criteria.* `http://web.mit.edu/16.unified/www/SPRING/materials/Lectures/M5.3-Unified09.pdf`. (Accessed: 2018-07-05)

Lemaitre, J., & Chaboche, J.-L. (1990). *Mechanics of solid materials.* Cambridge University Press.

Lostado, R., Martínez-De-Pisón, F. J., Fernández, R., & Fernández, J. (2010). Using genetic algorithms to optimize the material behaviour model in finite element models of processes with cyclic loads. *The Journal of Strain Analysis for Engineering Design*, *46*(2).

Mahnken, R. (1999). Aspects on the finite-element implementation of the gurson model including parameter identification. *International Journal of Plasticity*, *15*(11).

Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, *11*(2).

MathWorks. (2018). *Improve neural network generalization and avoid overfitting.* `https://se.mathworks.com/help/nnet/ug/improve-neural-network-generalization-and-avoid-overfitting.html`. (Accessed: 2018-07-05)

Moan, T. (2003). *Tmr 4190: Finite element modelling and analysis of marine structures.* Norwegian University of Science and Technology.

Mohd, R. N., & Geraghty, J. (2011). Genetic algorithm performance with different selection strategies in solving tsp. International Conference of Computational Intelligence and Intelligent Systems (ICCIIS'11).

Muñoz-Rojas, P. A., Cardoso, E., & Vaz, M. J. . (2010). Parameter identification of damage models using genetic algorithms. *Experimental Mechanics*, *50*(5).

Needleman, A., & Tvergaard, V. (1984). An analysis of ductile rupture in notched bars. *Journal of the Mechanics and Physics of Solids*, *32*(6).

Purshouse, R., & Fleming, P. (2002). Why use elitism and sharing in a multi-objective genetic algorithm? GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference.

Schmitt, W., Sun, D., & Blauel, J. (1997). Damage mechanics analysis (gurson model) and experimental verification of the behaviour of a crack in a weld-cladded component. *Nuclear Engineering and Design*, *174*(3).

Skallerud, B., & Zhang, Z. (1997, August). A 3d numerical study of ductile tearing and fatigue crack growth under nominal cyclic plasticity. *International Journal of Solids and Structures*, *34*(24).

Tvergaard, V., & Needleman, A. (1984). Analysis of the cup-cone fracture in a round tensile bar. *Acta Metallurgica*, *32*(1).

Widrow, B., & Hoff, M. E. (1988). Neurocomputing: Foundations of research. In J. A. Anderson & E. Rosenfeld (Eds.), (chap. Adaptive Switching Circuits). MIT Press.

Wriggers, P. (2008). *Nonlinear finite element methods*. Springer Berlin Heidelberg.

Zhong, J., Xu, T., Guan, K., & Zou, B. (2016). Determination of ductile damage parameters using hybrid particle swarm optimization. *Experimental Mechanics*, *56*(6).