Division of Solid Mechanics

ISRN LUTFD2/TFHF–11/5159–SE (1-97)

INVERSE MODELING OF POLYMER MATERIALS

Master's Dissertation by

Kent Bengtsson and Andreas Holmgren

Supervisors
Lic. Tech. Anders Harrysson, Tetra Pak, Lund
Ph. D. Mathias Wallin, Div. of Solid Mechanics, Lund

Examiner
Prof. Matti Ristinmaa, Div. of Solid Mechanics, Lund

Copyright © by Div. of solid Mechanics, Tetra Pak,
Kent Bengtsson and Andreas Holmgren
Printed by Media-Tryck, Lund, Sweden.

For information, address:
Division of Solid Mechanics, Lund University, Box 118, SE-221 00 Lund, Sweden.
Homepage: http://www.solid.lth.se

Work distribution

In this thesis all of the work has been made by both Andreas Holmgren and Kent Bengtsson.

Preface

This Master's thesis was performed at Tetra Pak Packaging Solutions AB in Lund with supervision from the Division of Solid Mechanics at Lund Institute of Technology. It was started in September 2010 and completed in January 2011.

First, we would like to thank our supervisor from the Division of Solid Mechanics, Ph.D Mathias Wallin for his support, valuable feedback and discussions throughout the project. A special thanks goes to our supervisor at Tetra Pak, Lic. Tech. Anders Harrysson for his encouragement and for always giving us valuable support and positive feedback throughout the project. He has also given us the opportunity to perform the project on site at Tetra Pak for which we are very grateful.

Lund, January 2011
Kent Bengtsson and Andreas Holmgren

Abstract

When performing material testing today, a so called tensile test is usually performed. During this test, a homogeneous displacement field over the entire specimen is assumed and the force versus displacement field is recorded. This is often not enough information to be able to calibrate advanced material models. Before this thesis, experiments were made on two different polymers using the so called ARAMIS system, which have the advantage that it can measure displacements and strains on any point in a specimen. This makes it possible to determine if a material behaves inhomogeneously.

The objective of this thesis is to establish a method which automatically identifies the material parameters for a given material model in order to make an accurate computer simulation.

The parameter identification is based on solving the so called inverse problem where a least square function is to be minimized. This is performed using experimental data from the ARAMIS system as input for a optimization loop that uses the Nelder-Mead simplex algorithm. This optimization loop is implemented in a Python script that uses Abaqus as the FE-solver.

The parameter identification has been tested on some of Abaqus built in material models and also on a toolbox called PolyUMod that contains various material models suitable for simulating polymers.

From the simulations, it was concluded that the parameter identification process works and is applicable. It turned out that the least square function is non-convex and more than one local minima exists. The parameter identification is therefore greatly dependent of the starting values and several parameter identifications with different starting values are needed.

Contents

Preface III

Abstract V

1 Introduction 1

2 Theory 3
- 2.1 Kinematic relations 3
- 2.2 Strain tensor 5
- 2.3 Stress tensors 6
- 2.4 Small strain elasticity 8
- 2.5 Elasto-Plasticity 8
  - 2.5.1 Yield criteria 9
  - 2.5.2 Hardening 10
- 2.6 Visco-plasticity 10
- 2.7 Hyper-elasticity 11
- 2.8 Large strain plasticity 13

3 The finite element method 15
- 3.1 Deriving the FEM-formulation 15
  - 3.1.1 Strong form 16
  - 3.1.2 Weak form 16
  - 3.1.3 Approximation and choice of weight function 17

4 Basics of parameter identification 19
- 4.1 The direct problem 20
- 4.2 The inverse problem 20
- 4.3 The correlation matrix 20
- 4.4 Solution to the inverse problem 22
  - 4.4.1 Constraints 22
- 4.5 The Nelder-Mead (simplex) algorithm 23

5 ARAMIS tensile test 29
- 5.1 Physical test 32
- 5.2 Results and conclusions 33

6 Specific material models 35
6.1 Elasto-plastic . . . 35
6.2 Hyper-elasto-plastic . . . 37
6.2.1 Neo-Hookean form . . . 37
6.2.2 Arruda-Boyce form . . . 37
6.2.3 Mooney-Rivlin form . . . 38
6.3 PolyUMod . . . 39
6.3.1 Bergström-Boyce . . . 39
6.3.2 Hybrid model . . . 41
6.3.3 Three network . . . 43

7 Method 47
7.1 Geometry . . . 47
7.1.1 Optimization procedure . . . 49

8 Test calibration 51
8.1 Problem 1 . . . 52
8.2 Problem 2 . . . 54
8.3 Conclusions . . . 57

9 Results 59
9.1 Elasto-plasticity . . . 60
9.2 Hyper-elasto-plastic . . . 61
9.2.1 Neo-Hookean form . . . 61
9.2.2 Arruda-Boyce form . . . 62
9.2.3 Mooney-Rivlin form . . . 63
9.3 PolyUMod . . . 64
9.3.1 Bergström-Boyce . . . 64
9.3.2 Hybrid . . . 65
9.3.3 Three network . . . 66
9.4 Summary - Results . . . 67

10 Concluding remarks 69
10.1 Discussion . . . 69
10.2 Future work . . . 70

A ARAMIS results 73
A.1 HDPE . . . 74
A.2 LDPE . . . 76

B Results from simulations HDPE 50mm/min 79
B.1 Elasto-plasticity . . . 80
B.2 Neo-Hooke . . . 82
B.3 Arruda-Boyce . . . 84
B.4 Mooney-Rivlin . . . 86
B.5 Bergström-Boyce . . . 88
B.6 Hybrid . . . 90

B.7 Three network 92

C Correlation matrix for monotone versus cyclic loading 95

Bibliography 96

Chapter 1

Introduction

Tetra Pak is a company that was founded in Lund, Sweden 1951. Today it is located all around the world and has almost 20000 employees. It is one of the world's leading manufacturer of various cartons for the food industry.

In order to be competitive in today's market, it is necessary to be both cost and time efficient. When designing a new product, physical prototypes are made to be able to make manual physical testings to know how the product behaves at various loads, i.e. drop-testing. To make prototypes of each idea is a expensive and time consuming approach since prototypes are not easy or cheap to make. If one instead could make virtual tests in a computer using i.e. the finite element method there is a chance to eliminate some of the physical prototypes that does not result in satisfying computer simulations.

Figure 1.1: Different packages from Tetra Pak

In order for the simulations to be consistent with reality, it is necessary to have a material model with corresponding material parameters (e.g. modulus of elasticity) that accurately describes how the material behaves at arbitrary loads.

The objective of this thesis is to establish a method which identifies the material parameters needed in order to make an accurate computer simulation, using i.e. the finite element method. The finite element software Abaqus is used in this thesis as the FE-solver. The set of material parameters varies with the choice of material model. In the present work the material parameters will be identified for two different types of polymers, HDPE (High-density polyethylene) and LDPE (Low-density polyethylene).

The parameters are identified using a uniaxial tensile test on a dog-bone shaped specimen with the so called ARAMIS system. Based on the displacements and strains that are obtained from the experimental tests, the parameters are identified using a Nelder-Mead simplex algorithm optimization loop.

Chapter 2

Theory

The following chapter describes the basics of constitutive modeling. First, some basic continuum mechanics relations are introduced and after this a short description of a few cornerstones of constitutive modeling of mechanical properties during a arbitrary load are presented. This theory will be the foundation in this thesis of which the material models and finite element expression are based on.

Large and small bold letters in this chapter will represent second and first order tensors respectively.

For further motivation and more detailed explanations the interested reader is referred to [9] and [11].

2.1 Kinematic relations

To be able to establish a ground on which strains and stresses are obtained, kinematic relations describing the motion of a body during deformation needs to be presented. In this theory, the flow of particles are described. A particle is described as an infinitesimal part of the material that one wants to be able to describe throughout some deformation. Consider a particle P that is initially located in a so called "reference configuration", where the shape and all initial stresses and strains are known. The subscript $\circ$ will denote the reference configuration. P has the original coordinate $\boldsymbol{x}^\circ$ which are called the material coordinates. The flow of every particle in the body from the reference to the current configuration can be denoted as $\boldsymbol{\varphi}(\boldsymbol{x}^\circ, t)$ where t is the time.

The reference configuration is initially at $t = 0$ and therefore $\boldsymbol{x}^\circ = \boldsymbol{\varphi}(\boldsymbol{x}^\circ, 0)$ and $\boldsymbol{x}^\circ = (x^\circ, y^\circ, z^\circ)$. When describing the motion of the particle originally located at x° in the current configuration it is convenient to refer the displacements to the reference configuration. This gives that

$$\boldsymbol{x}(\boldsymbol{x}^\circ, t) = \boldsymbol{\varphi}(\boldsymbol{x}^\circ, t) = \boldsymbol{x}^\circ + \boldsymbol{u}(\boldsymbol{x}^\circ, t) \tag{2.1}$$

where $\boldsymbol{u} = (u_x, u_y, u_z)$ is the displacement vector.

Let $d\boldsymbol{x}^\circ$ denote a vector between P and a neighboring particle in the reference configuration and $d\boldsymbol{x}$ in the current configuration. The total deformation of $d\boldsymbol{x}^\circ$

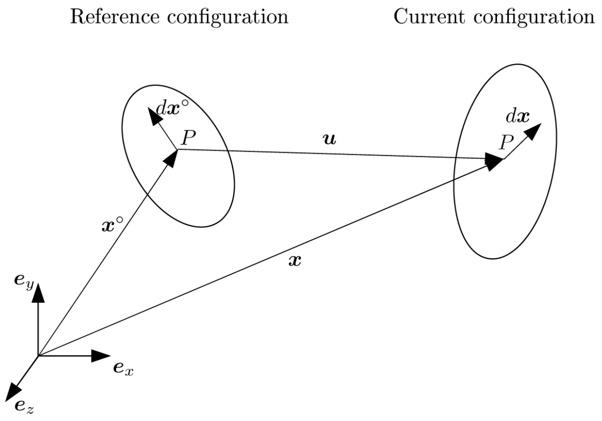


Figure 2.1: Reference configuration and current configuration

is given by

$$d\boldsymbol{x} = \boldsymbol{F} d\boldsymbol{x}^\circ \tag{2.2}$$

where $\boldsymbol{F}$ is the so called deformation gradient. This tensor describes how a line segment in the reference configuration, $d\boldsymbol{x}^\circ$ rotates, translates and changes shape in to $d\boldsymbol{x}$ due to the deformation.

The determinant of the deformation gradient is called the Jacobian, i.e.

$$J = \det \boldsymbol{F} \tag{2.3}$$

Using the assumption that the mass of the body is conserved, i.e. $\rho^\circ dv^\circ = \rho dv$ where dv° and dv is the volume of the undeformed and deformed body respectively, it can be shown that

$$\frac{dv}{dv^\circ} = \frac{\rho}{\rho^\circ} = J > 0 \tag{2.4}$$

which according to (2.3) indicates that $\det \boldsymbol{F} \neq 0$. This means that a unique solution to (2.2) exists and

$$d\boldsymbol{x}^\circ = \boldsymbol{F}^{-1} d\boldsymbol{x} \tag{2.5}$$

holds. This means that every particle in the body has a unique position.

As well as there is a change of the volume in the body, there is also a change of the area due to deformation. Let $d\boldsymbol{x}^{\circ(1)}$ and $d\boldsymbol{x}^{\circ(2)}$ denote vectors between neighboring particles in the reference configuration. These two vectors represents an area, ds° that after deformation transfers to $d\boldsymbol{x}^{(1)}$ and $d\boldsymbol{x}^{(2)}$ with the area ds in the current configuration, see Fig. 2.2.

Define the vector $d\boldsymbol{a}^\circ$ and $d\boldsymbol{a}$ as

$$\begin{aligned} d\boldsymbol{a}^\circ &= \boldsymbol{n}^\circ ds^\circ \\ d\boldsymbol{a} &= \boldsymbol{n} ds \end{aligned} \tag{2.6}$$

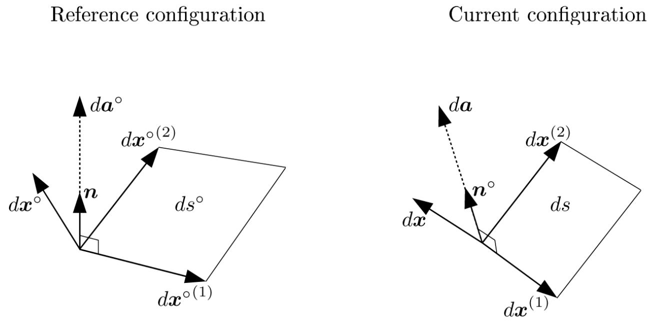

Reference configuration
Current configuration
$d\boldsymbol{a}^\circ$
$d\boldsymbol{x}^{\circ(2)}$
$d\boldsymbol{x}^\circ$
$\boldsymbol{n}$
ds°
$d\boldsymbol{x}^{\circ(1)}$
$d\boldsymbol{a}$
$d\boldsymbol{x}^{(2)}$
$\boldsymbol{n}^\circ$
$d\boldsymbol{x}$
ds
$d\boldsymbol{x}^{(1)}$

Figure 2.2: Infinitesimal surface element in the reference and current configuration

where $\boldsymbol{n}^\circ$, $\boldsymbol{n}$, ds° and ds denote the normal vectors and infinitesimal areas of the reference and current configuration respectively. The vectors $d\boldsymbol{a}^\circ$ and $d\boldsymbol{a}$ are calculated from the definition of vector cross product according to

$$\begin{aligned} d\boldsymbol{a}^\circ &= d\boldsymbol{x}^{\circ(1)} \times d\boldsymbol{x}^{\circ(2)} \\ d\boldsymbol{a} &= d\boldsymbol{x}^{(1)} \times d\boldsymbol{x}^{(2)} \end{aligned} \tag{2.7}$$

Let now $d\boldsymbol{x}^\circ$ denote an arbitrary vector. Multiplying this with (2.6a) gives the volume dv° as $dv^\circ = (d\boldsymbol{x}^\circ)^T d\boldsymbol{a}^\circ$. This volume will after deformation transform to dv according to (2.4) and this gives

$$\underbrace{d\boldsymbol{x}^T d\boldsymbol{a}}_{dv} = J\,\underbrace{(d\boldsymbol{x}^\circ)^T d\boldsymbol{a}^\circ}_{dv^\circ} \tag{2.8}$$

Inserting (2.2) in (2.8) and using that $d\boldsymbol{x}^\circ$ is arbitrary results in

$$d\boldsymbol{a} = J\boldsymbol{F}^{-T} d\boldsymbol{a}^\circ \tag{2.9}$$

This is the so called Nanson's formula that will be used later on.

2.2 Strain tensor

The strain tensor needs to be derived in order to measure the deformation of the body and will be used later in Chapter 3 for the finite element expression. Consider the length of $d\boldsymbol{x}^\circ$ in the reference configuration to be $dl^{\circ 2} = dx^{\circ 2} + dy^{\circ 2} + dz^{\circ 2}$ and in the current configuration $dl^2 = dx^2 + dy^2 + dz^2$, i.e.

$$\begin{aligned} dl^{\circ 2} &= d\boldsymbol{x}^{\circ T} d\boldsymbol{x}^\circ \\ dl^2 &= d\boldsymbol{x}^T d\boldsymbol{x} \end{aligned} \tag{2.10}$$

Using (2.2) in (2.10a) gives

$$dl^2 = d\boldsymbol{x}^{\circ T} \boldsymbol{F}^T \boldsymbol{F} d\boldsymbol{x}^\circ \tag{2.11}$$

Using (2.10) and (2.11) gives

$$dl^2 - dl^{\circ 2} = 2\boldsymbol{x}^{\circ T}\boldsymbol{E}_{\Box} d\boldsymbol{x}^{\circ} \tag{2.12}$$

where the strain tensor is defined by

$$\boldsymbol{E}_{\Box} = \frac{1}{2}(\boldsymbol{F}^T\boldsymbol{F} - \boldsymbol{I}) = \begin{bmatrix} E_{xx} & E_{xy} & E_{xz} \\ E_{yx} & E_{yy} & E_{yz} \\ E_{zx} & E_{zy} & E_{zz} \end{bmatrix} \tag{2.13}$$

This is often called the *Green-Lagrange's* strain tensor. Since $\boldsymbol{E}_{\Box}$ is symmetric it turns out to be appropriate to use a column matrix instead according to

$$\boldsymbol{E} = \begin{bmatrix} E_{xx} \\ E_{yy} \\ E_{zz} \\ 2E_{xy} \\ 2E_{yx} \\ 2E_{xz} \end{bmatrix} \tag{2.14}$$

2.3 Stress tensors

In order to continue the investigation of kinematic relations, some different stresses needs to be presented. These will also be used for the finite element expression in Chapter 3.

The traction vector $\boldsymbol{t}$ needs to be defined to derive the stress tensors and this is defined by

$$\boldsymbol{t} = \left(\frac{\Delta \boldsymbol{P}}{\Delta A}\right)_{\Delta A \to 0} \tag{2.15}$$

where $\Delta \boldsymbol{P}$ is an incremental force vector acting on an incremental surface area ΔA, see Fig. 2.3.

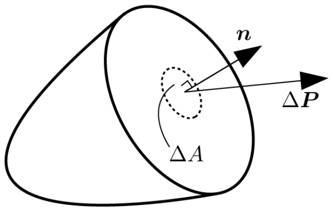


Figure 2.3: Incremental force acting on incremental area

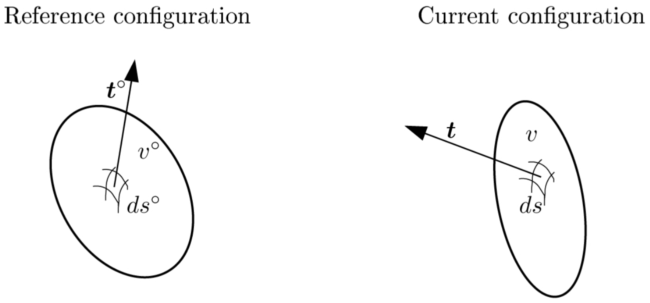


Figure 2.4: Reference configuration and current configuration

For the two different configurations in Fig. 2.4, one can define two different types of stresses, Cauchy and nominal. The Cauchy theorem states that the traction vector, $\boldsymbol{t}$ is linear in $\boldsymbol{n}$ according to

$$\boldsymbol{t} = \boldsymbol{T}\boldsymbol{n} \tag{2.16}$$

where $\boldsymbol{T}$ is the Cauchy stress tensor defined as

$$\boldsymbol{T} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix} \tag{2.17}$$

The Cauchy is defined as the force divided by the deformed surface area in the current configuration as

$$\boldsymbol{t}ds = d\boldsymbol{f} \tag{2.18}$$

and the nominal stress is defined from the reference configuration as the force divided by the undeformed surface area which gives

$$\boldsymbol{t}^\circ ds^\circ = d\boldsymbol{f} \tag{2.19}$$

From this the assumption was made that the force $d\boldsymbol{f}$ is the same in the reference and current configuration as

$$d\boldsymbol{f} = \boldsymbol{t}^\circ ds^\circ = \boldsymbol{t}ds \tag{2.20}$$

Making use of Cauchy's theorem, we find that the following must hold

$$\boldsymbol{t}^\circ = \boldsymbol{P}\boldsymbol{n}^\circ \tag{2.21}$$

where $\boldsymbol{P}$ is known as the *first Piola-Kirchhoff stress tensor*. To know how this is related to Cauchy's stress tensor, we use (2.16) and (2.21) in (2.20) to obtain

$$\boldsymbol{P}\boldsymbol{n}^\circ ds^\circ = \boldsymbol{T}\boldsymbol{n}ds \tag{2.22}$$

We now make use of Nanson's formula (2.9) to find the relation between the first Piola-Kirchhoff and the Cauchy stress tensor to be

$$\boldsymbol{P} = J\boldsymbol{T}\boldsymbol{F}^{-T} \tag{2.23}$$

As it turns out, the relation (2.23) is non-symmetric and it is convenient to obtain a stress that is symmetric. This is done by multiplying (2.23) with $\boldsymbol{F}^{-1}$ to obtain the *second Piola-Kirchhoff stress tensor* defined by

$$\boldsymbol{S}_{\square} = \boldsymbol{F}^{-1}\boldsymbol{P} = J\boldsymbol{F}^{-1}\boldsymbol{T}\boldsymbol{F}^{-T} \tag{2.24}$$

Like the strain tensor, this square matrix is reduced to a column matrix according to

$$\boldsymbol{S} = \begin{bmatrix} S_{xx} \\ S_{yy} \\ S_{zz} \\ S_{xy} \\ S_{yx} \\ S_{xz} \end{bmatrix} \tag{2.25}$$

2.4 Small strain elasticity

If a material behaves only linearly elastic it is known as a *Hooke* material. This can be represented in a rheological model by a spring according to Fig. 2.5. The spring with stiffness E expands linearly according to Hooke's law, $\sigma = E\epsilon$ where in this simple case $\epsilon = \epsilon^e$ where the superscript e refers to that the strain is only elastic.

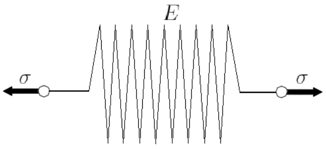


Figure 2.5: Illustration of a elastic material model

2.5 Elasto-Plasticity

As an illustration of the elasto-plastic response, consider the rheological model below. The elastic part is represented by the spring with stiffness E and the plastic, by the friction element with yield stress σ_{y0} in Fig. 2.6.
When $\sigma > 0$ in Fig. 2.6 the spring expands linearly according to Hooke's law and when the stress reaches σ_{y0} the spring has reached its final length and the friction element starts to move, representing the plastic strain, ϵ^p. The total strain ϵ is now defined by $\epsilon = \epsilon^e + \epsilon^p$ and the stress-strain relation is

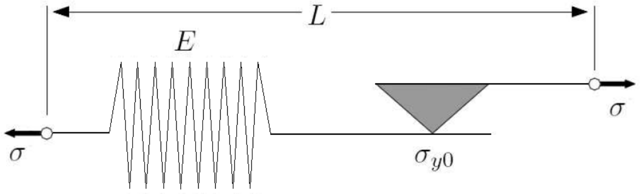
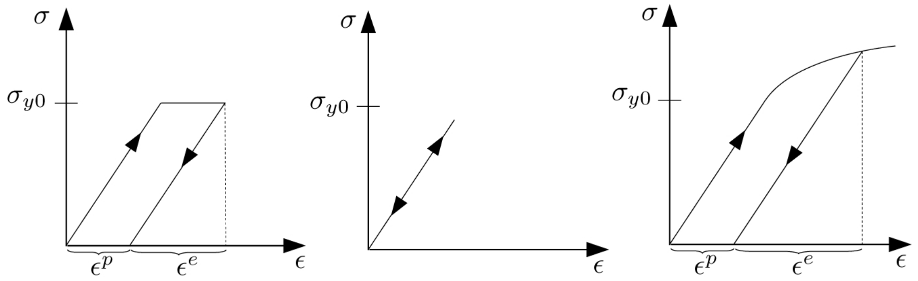


Figure 2.6: Illustration of a elasto-plastic material model

$\sigma = E(\epsilon - \epsilon^p)$. This is known as ideal-plastic behavior and a illustration of this is shown in Fig. 2.7a).

When the material is loaded only within its elastic region ($\sigma < \sigma_{y0}$) it will return to its original state when unloaded, see Fig. 2.7b).



(a) Ideal-plastic response (b) Elastic response (c) Elasto-plastic response

Figure 2.7: Loading and unloading of ideal-plastic, elastic and elasto-plastic response

When introducing the elasto-plastic response in the general case the total strain tensor is defined by

$$\boldsymbol{\epsilon} = \boldsymbol{\epsilon}^e + \boldsymbol{\epsilon}^p \tag{2.26}$$

For a general stress state it turns out to be convenient to define the effective stress, σ_{eff}. When the effective stress reach σ_{y0} the material reaches the plastic region. This threshold stress can also represent the boundary of the elastic region in the stress space, see Fig. 2.8a). As an example, the effective stress of the von Mises criterion is given by

$$\sigma_{eff} = \sqrt{\sigma_{11}^2 + \sigma_{22}^2 + \sigma_{33}^2 - \sigma_{11}\sigma_{22} - \sigma_{33}\sigma_{22} - \sigma_{33}\sigma_{11} + 3\sigma_{12}^2 + 3\sigma_{13}^2 + 3\sigma_{31}^2}$$

2.5.1 Yield criteria

In order to determine if a stress response is elastic or plastic, a yield criteria is introduced. The yield criteria is often represented by a surface in the deviatoric stress space with a function, f describing the surface

$$f(\boldsymbol{T}) = 0 \tag{2.27}$$

If the loading of the material is within the yield surface, $f < 0$, the material behaves elastic. By definition, $f > 0$ can never occur and this means that if $f = 0$, plastic response takes place and therefore plastic deformations will grow and the yield surface might alter shape, change location in the deviatoric stress plane or a combination of both according to different hardening rules.

2.5.2 Hardening

Before any deformation, the initial yield surface is given by

$$F(\boldsymbol{T}) = 0 \tag{2.28}$$

When the deformation is plastic, the yield surface can now be expressed by

$$f(\boldsymbol{T}, K_\alpha) = 0 \tag{2.29}$$

where K_α, $\alpha = 1, 2, \dots, n$ contains hardening parameters that are dependent on the internal parameters. For example, the first hardening parameter can be dependent on the effective plastic strain, i.e. $K_{\alpha=1} = K_{\alpha=1}(\epsilon^p_{eff})$. The internal parameters describe how the yield surface changes shape, size and location during plastic deformation.

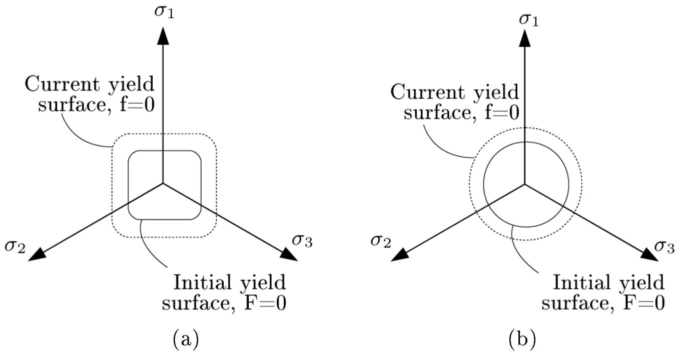

σ_1
Current yield surface, f=0
σ_2
σ_3
Initial yield surface, F=0
(a)
σ_1
Current yield surface, f=0
σ_2
σ_3
Initial yield surface, F=0
(b)

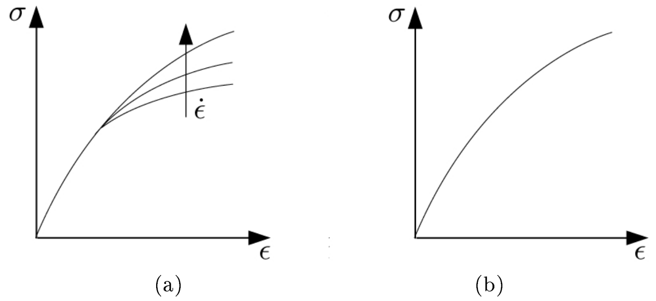
Figure 2.8: a) Arbitrary yield surface b) Yield surface of the von Mises criterion

Figure 2.8a) shows an example of initial and current yield surface after hardening for an arbitrary criterion and Fig. 2.8b) of the von Mises criterion.

If the material have been subjected to plastic loading, it will not return to its original state when unloaded since the plastic deformations will remain, see Fig. 2.7c).

2.6 Visco-plasticity

Many materials, such as rubber and polymers often behaves visco-plastic. This means that the stress-strain response is time dependent and are therefore often called rate-dependent materials. Materials that do not have this property is known as rate-independent materials. Figure 2.9 shows the stress-strain response of these different types of materials.



(a) (b)

Figure 2.9: a) Illustration of a rate-dependent material b) Illustration of a rate-independent material

A rheological model of a visco-plastic material can be seen in Fig. 2.10 where the elastic part is still represented by the spring and the visco-plastic part with the friction element parallel with the dashpot. The dashpot is commonly known as a *Newton* material with the stress-strain relation $\sigma = \eta\dot{\epsilon}$, where η is material constant and $\dot{\epsilon}$ the strain rate. Since the friction element that represents the plastic part is parallel with the dashpot, the total strain rate of these two elements will be referred to as $\dot{\epsilon}^{vp}$. The total strain rate, including the spring will now be $\dot{\epsilon} = \dot{\epsilon}^{e} + \dot{\epsilon}^{vp}$.

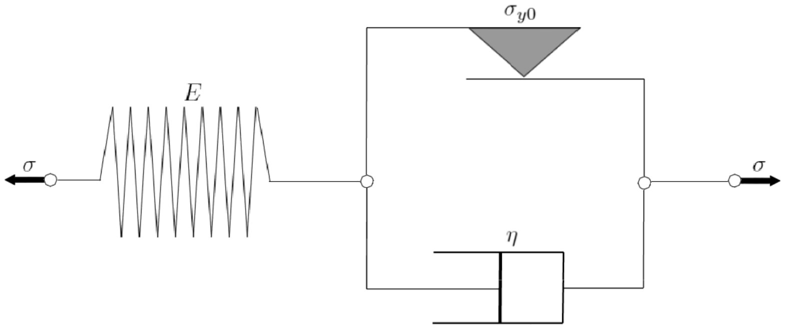


Figure 2.10: Rheological model of visco-plasticity

As can be seen from the experiments in Fig. A.7a) the dashed lines from the greater test speed lies over the lines form the lower test speed. This indicates that the HDPE and LDPE materials behaves visco-plastic.

2.7 Hyper-elasticity

Hyper-elasticity, also called Green-elasticity is a good model for non-linear elasticity. The hyper-elastic model is related to the strain energy W defined per unit reference volume as

$$W(E_{mn}) = \int_0^{E_{mn}} S_{ij}(\widetilde{E}_{kl})d\widetilde{E}_{ij} \tag{2.30}$$

where the integral is performed from the reference configuration to the deformed configuration with present strain value E_{ij} and $\widetilde{E}_{kl}$ is an integration variable. S_{ij} is the second Piola-Kirchoff stress tensor and E_{ij} the Lagrangian strain tensor.

In the one dimensional case the strain energy can be illustrated as the area under the stress-strain curve, see Fig. 2.11.

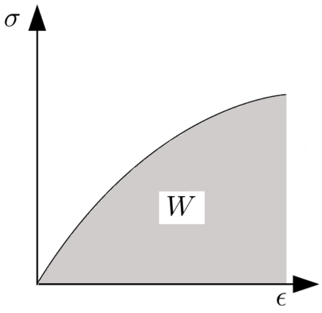


Figure 2.11: Strain energy in a one-dimensional case

Differentiating (2.30) with respect to E_{ij} gives

$$dW = \frac{\partial W}{\partial E_{ij}} dE_{ij} \quad \text{and} \quad dW = S_{ij} dE_{ij} \tag{2.31}$$

Using (2.31) gives

$$(\frac{\partial W}{\partial E_{ij}} - S_{ij})dE_{ij} = 0 \tag{2.32}$$

and since dE_{ij} is an arbitrary strain increment, (2.32) must hold for any strain increment and therefore the following expression for Hyper-elasticity is obtained

$$S_{ij} = \frac{\partial W}{\partial E_{ij}} \quad \text{where} \quad W = W(E_{ij}) \tag{2.33}$$

In Abaqus, the strain energy W is given on the so called polynomial form. One format (cf. [1]) is given by

$$W = \sum_{i+j=1}^{N} C_{ij}(\overline{I}_1 - 3)^i(\overline{I}_2 - 3)^j + \sum_{i=1}^{N} \frac{1}{D_i}(J^{el} - 1)^{2i} \tag{2.34}$$

where N, C_{ij}, D_i are material parameters and J^{el} is the elastic volume ratio. The invariants $\overline{I}_1$ and $\overline{I}_2$ are defined as

$$\overline{I}_1 = \tilde{\lambda}_1^2 + \tilde{\lambda}_2^2 + \tilde{\lambda}_3^2 \quad \text{and} \quad \overline{I}_2 = \tilde{\lambda}_1^{(-2)} + \tilde{\lambda}_2^{(-2)} + \tilde{\lambda}_3^{(-2)} \tag{2.35}$$

which are known as the first and second deviatoric strain invariants, where $\tilde{\lambda}_i = J^{-\frac{1}{3}}\lambda_i$ and λ_i the principal stretches.

Many material models are based from this polynomial form. The ones used in this thesis will be described further in Chapter 6.

2.8 Large strain plasticity

For large deformations, the additative split of the strain tensor (2.26) is not more valid. Instead one uses the so called multiplicative split of the deformation gradient $\boldsymbol{F}$ into one elastic and one plastic part, cf. Fig. 2.12.

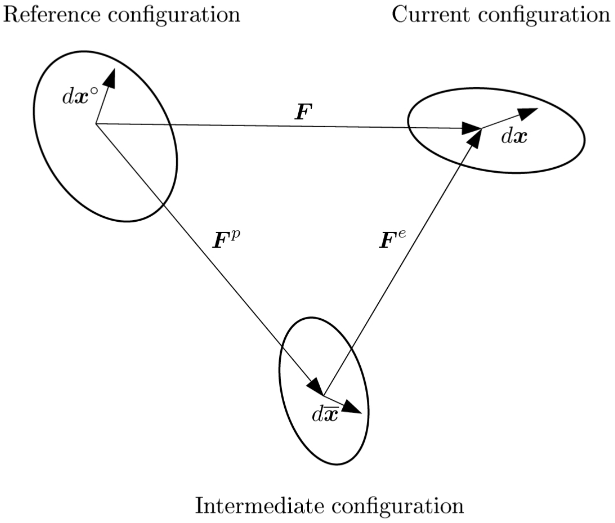

Reference configuration
Current configuration
$d\boldsymbol{x}^\circ$
$\boldsymbol{F}$
$d\boldsymbol{x}$
$\boldsymbol{F}^p$
$\boldsymbol{F}^e$
$d\overline{\boldsymbol{x}}$
Intermediate configuration

Figure 2.12: Illustration of the mapping between configurations with multiplicative split

In Fig. 2.12 an intermediate configuration has been introduced. The intermediate configuration is defined from an elastic unloading from the current configuration. It follows from the definition of the intermediate configuration that the following holds

$$d\boldsymbol{x} = \boldsymbol{F}^e d\overline{\boldsymbol{x}} \tag{2.36}$$

and

$$d\overline{\boldsymbol{x}} = \boldsymbol{F}^p d\boldsymbol{x}^\circ \tag{2.37}$$

Inserting (2.37) in (2.36) gives

$$d\boldsymbol{x} = \boldsymbol{F}^e \boldsymbol{F}^p d\boldsymbol{x}^\circ \tag{2.38}$$

Comparing (2.38) with (2.2) shows that

$$\boldsymbol{F} = \boldsymbol{F}^e \boldsymbol{F}^p \tag{2.39}$$

which is known as the multiplicative split.

For more information about large strain plasticity the interested reader is referred to [1] and [10].

Chapter 3

The finite element method

The finite element method is a commonly used method to derive solutions to a physical problem. The physical problems are usually described by differential equations specified over some area that can be one-, two- or three-dimensional and is very hard to solve with analytical methods. Instead, the finite element method is used as a numerical solution technique to the physical problem. Instead of trying to solve the differential equations over the entire area, the area is divided into smaller parts, called *elements* and the differential equations that describes the physical problem is approximated over each of the elements.

When making use of the finite element method, a few basic steps are performed. These are

1. Establish the strong formulation of the problem.
2. Obtain the weak formulation of the problem.
3. Make an element wise approximation over the entire body of the unknown function.
4. Choose the weight function, w.

These steps are performed in Chapter 3.1 to establish the finite element formulation. For further reading of the finite element method, the reader is referred to [8].

3.1 Deriving the FEM-formulation

In this section the following steps from the list given above are presented. To be able to do this, two theorems (*Gauss* and *Green-Gauss* theorem) needs to be presented from the vector calculus.

Gauss' divergence theorem:

$$\int_V \text{div}\boldsymbol{q} dV = \int_S \boldsymbol{q}^T \boldsymbol{n} dS \tag{3.1}$$

Green-Gauss theorem:

$$\int_V \phi \text{div}\boldsymbol{q} dV = \int_S \phi \boldsymbol{q}^T \boldsymbol{n} dS - \int_V (\boldsymbol{\nabla}\phi)^T \boldsymbol{q} dV \tag{3.2}$$

3.1.1 Strong form

The balance for linear momentum of an arbitrary body can be expressed as

$$\int_s \boldsymbol{t} ds + \int_v \rho \boldsymbol{b} dv = \int_v \rho \ddot{\mathbf{u}} dv \tag{3.3}$$

This is simply Newton's second law of motion ($\sum \boldsymbol{F} = m\boldsymbol{a}$) that states that the sum of all forces acting on the body is equal to the mass of the body times acceleration.

Using (2.16) and (3.1) on the first term of (3.3) results in

$$\int_v (\text{div}\boldsymbol{T} + \rho \boldsymbol{b} - \rho \ddot{\mathbf{u}}) dv = 0 \tag{3.4}$$

Since the volume in (3.4) is arbitrary and should hold for any body gives

$$\boxed{\text{div}\boldsymbol{T} + \rho \boldsymbol{b} = \rho \ddot{\mathbf{u}}} \tag{3.5}$$

This is the strong formulation of the physical problem.

3.1.2 Weak form

Multiplying (3.5) with an arbitrary velocity $\boldsymbol{w}$ and integrate over the body gives the virtual power expression in the current configuration as

$$\int_v \boldsymbol{w}^T \text{div}\boldsymbol{T} dv + \int_v \rho \boldsymbol{w}^T \boldsymbol{b} dv = \int_v \rho \boldsymbol{w}^T \ddot{\mathbf{u}} dv \tag{3.6}$$

Using the Green-Gauss theorem defined by (3.2) and Cauchy's theorem (2.16) on the first term of (3.6) gives

$$\int_v \boldsymbol{w}^T \text{div}\boldsymbol{T} dv = \int_s \boldsymbol{w}^T \boldsymbol{t} ds - \int_v \boldsymbol{\nabla}\boldsymbol{w} : \boldsymbol{T} dv \tag{3.7}$$

where the scalar product between matrices was introduced as

$$\boldsymbol{A} : \boldsymbol{B} = A_{ij} B_{ij} \quad \text{sum over all i,j} \tag{3.8}$$

Inserting (3.7) in (3.6) gives

$$\int_s \boldsymbol{w}^T \boldsymbol{t} ds - \int_v \boldsymbol{\nabla}\boldsymbol{w} : \boldsymbol{T} dv + \int_v \rho \boldsymbol{w}^T \boldsymbol{b} dv = \int_v \rho \boldsymbol{w}^T \ddot{\mathbf{u}} dv \tag{3.9}$$

This equation is given in the current configuration. In many cases it turns out to be convenient to express (3.9) in the reference configuration. It can be proved that

$$\int_s \boldsymbol{w}^T \boldsymbol{t} ds = \int_{s^\circ} \boldsymbol{w}^T \boldsymbol{t}^\circ ds^\circ \tag{3.10}$$

$$\int_v \boldsymbol{\nabla w} : \boldsymbol{T} dv = \int_{v^\circ} \widehat{\boldsymbol{E}}^T \boldsymbol{S} dv^\circ \tag{3.11}$$

$$\int_v \rho \boldsymbol{w}^T \boldsymbol{b} dv = \int_{v^\circ} \rho^\circ \boldsymbol{w}^T \boldsymbol{b} dv^\circ \tag{3.12}$$

$$\int_v \rho \boldsymbol{w}^T \ddot{\boldsymbol{u}} dv = \int_{v^\circ} \rho^\circ \boldsymbol{w}^T \ddot{\boldsymbol{u}} dv^\circ \tag{3.13}$$

The quantities with the ˆ indicates that they are a function of time. Using (3.10)-(3.13) in (3.9) gives

$$\boxed{\int_{v^\circ} \rho^\circ \boldsymbol{w}^T \ddot{\boldsymbol{u}} dv^\circ + \int_{v^\circ} \widehat{\boldsymbol{E}}^T \boldsymbol{S} dv^\circ - \int_{s^\circ} \boldsymbol{w}^T \boldsymbol{t}^\circ ds^\circ - \int_{v^\circ} \rho^\circ \boldsymbol{w}^T \boldsymbol{b} dv^\circ = 0} \tag{3.14}$$

This is the weak formulation of the problem, describing the entire system. The next step is to discretize this in an element-wise solution for the finite element approximation.

3.1.3 Approximation and choice of weight function

The displacement field and the acceleration field are approximated as

$$\boldsymbol{u} = \boldsymbol{N}\boldsymbol{a}, \quad \ddot{\boldsymbol{u}} = \boldsymbol{N}\ddot{\boldsymbol{a}} \tag{3.15}$$

where $\boldsymbol{N}$ are the global shape functions, $\boldsymbol{a}$ contains the nodal displacements and $\boldsymbol{u}$ the displacement vector throughout the body.

Now use the approximation according to Galerkin and put

$$\boldsymbol{w} = \boldsymbol{N}\boldsymbol{c} \tag{3.16}$$

where $\boldsymbol{c}$ is an arbitrary vector. Using that $\widehat{\boldsymbol{E}} = \boldsymbol{B}\boldsymbol{c}$ and inserting (3.15) and (3.16) in (3.14) results in

$$\boldsymbol{c}^T \left(\int_{v^\circ} \rho^\circ \boldsymbol{N}^T \boldsymbol{N} \ddot{\boldsymbol{a}} dv^\circ + \int_{v^\circ} \boldsymbol{B}^T \boldsymbol{S} dv^\circ - \int_{s^\circ} \boldsymbol{N}^T \boldsymbol{t}^\circ ds^\circ - \int_{v^\circ} \rho^\circ \boldsymbol{N}^T \boldsymbol{b} dv^\circ \right) = 0 \tag{3.17}$$

Since $\boldsymbol{c}$ is an arbitrary vector, the finite element method description can be written as

$$\boxed{\boldsymbol{M}\ddot{\boldsymbol{a}} + \boldsymbol{F}_{int} - \boldsymbol{F}_{ext} = \boldsymbol{0}} \tag{3.18}$$

where the mass matrix is defined by

$$\boldsymbol{M} = \int_{v^\circ} \rho^\circ \boldsymbol{N}^T \boldsymbol{N} dv^\circ \tag{3.19}$$

and

$$\boldsymbol{F}_{int} = \int_{v^\circ} \boldsymbol{B}^T \boldsymbol{S} dv^\circ \tag{3.20}$$

$$\boldsymbol{F}_{ext} = \int_{s^\circ} \boldsymbol{N}^T \boldsymbol{t}^\circ ds^\circ + \int_{v^\circ} \rho^\circ \boldsymbol{N}^T \boldsymbol{b} dv^\circ \tag{3.21}$$

Chapter 4

Basics of parameter identification

In order to make realistic simulations, it is necessary to have an appropriate material model with corresponding parameters.

When selecting appropriate material model, physical testing is first needed in order to give an idea about what kind of properties the material are having. For instance, a uniaxial tensile test on a dog-bone shaped specimen could show whether a material could be described by using only an elasto-plastic model, or if it is depended on the strain rate, a visco-elastic model would be more appropriate. Once the choice of material model is made, the parameter identification can begin.

In this chapter the various steps of parameter identification are presented. First, the direct and the inverse method are described in Chapter 4.1 and 4.2 respectively, and it turns out that the inverse method is the most suitable method for parameter identification.

The parameter identification is made by an optimization based on the initial guess. A simulation is made and the results are compared with the results from the experiment. Parameters continuously changes value after each simulation and the optimization ends when the difference between simulated and experimental data has reached a minimum.

A validation of the identified parameters is now needed to know if they can describe the material at some other arbitrary load. This is done by a new simulation and the results are compared with experimental data that was not used for the parameter identification.

An approach on how to solve the inverse problem is described in Chapter 4.4. Finally a detailed description of the Nelder-Mead simplex algorithm, used in the optimization loop is described in Chapter 4.5.

For further information the reader is referred to [5] and [6].

4.1 The direct problem

The field equation of a body in equilibrium during static loading is

$$\mathrm{div}\boldsymbol{T}(\boldsymbol{u}) + \boldsymbol{b} = \boldsymbol{0} \tag{4.1}$$

where $\boldsymbol{T}$ is the Cauchy stress tensor and $\boldsymbol{b}$ body forces. From this, with a combination of boundary conditions and a constitutive model, the displacements can be calculated, giving that the material parameters are known,

$$\boldsymbol{\kappa} \mapsto \boldsymbol{u}(\bullet, \boldsymbol{\kappa}) \tag{4.2}$$

where $\bullet$ represents the boundary conditions and $\boldsymbol{\kappa}$ the material parameters. Since the material parameters here are supposed to be known, (4.2) is not suitable for parameter identification.

4.2 The inverse problem

Let the experimental values of displacements denote $\boldsymbol{u}^{exp} \in \overline{U}$, simulated $\boldsymbol{u}^{sim} \in U$, where $\overline{U}$ represents the observation space for experimental values and U the observation space for simulated values. The reason why these observation spaces are not the same is because the sampling frequency differs from each other. To be able to compare the experimental data with the simulated, an observation operator $\mathcal{M} : U \to \overline{U}$ has to be introduced. This is simply a linear interpolation operator and using this on $\boldsymbol{u}^{sim}$ gives that $\mathcal{M}\boldsymbol{u}^{sim} \in \overline{U}$ and the simulated values are now comparable with the experimental.

The inverse problem of (4.2) can now be described as

$$\text{Find } \boldsymbol{\kappa} : \mathcal{M}\boldsymbol{u}^{sim}(\bullet, \boldsymbol{\kappa}) = \boldsymbol{u}^{exp} \tag{4.3}$$

Generally, this problem can not be satisfied exactly. Instead one tries to minimize the difference between simulated and experimental values using a least-square function of (4.3) in an optimization loop. This gives

$$f(\boldsymbol{\kappa}) = \frac{1}{2}\left\|\mathcal{M}\boldsymbol{u}^{sim}(\bullet, \boldsymbol{\kappa}) - \boldsymbol{u}^{exp}\right\|^2 \to \underbrace{min}_{\kappa} \tag{4.4}$$

In general, this equation is non-convex and does therefore not have a unique solution.

A description of the least square function can be found in [4].

4.3 The correlation matrix

Sometimes it is hard to find the solution to (4.4). This is often the case when small variations of $\boldsymbol{u}^{exp}$ may lead to large variations of the material parameters, $\boldsymbol{\kappa}$. The reason for this can either be that the material model contains many parameters, or that one or many parameters are linearly dependent of each other.

Parameters that are linearly dependent implies that different combinations of the parameter values can give the same results for the error function, cf. (4.4). This has no significance when investigating the load case that was used for the parameter identification, but to be sure that the parameters can describe other load cases as well, the importance of a validation is understood.

To be able to calculate the parameters linear dependency, the so called correlation matrix can be introduced. This is build from the definition of a scalar product where the angle between to vectors are calculated from

$$\cos \Phi_{ij} = \frac{\Delta \boldsymbol{u}_i^{sim} \Delta \boldsymbol{u}_j^{sim}}{||\Delta \boldsymbol{u}_i^{sim}||||\Delta \boldsymbol{u}_j^{sim}||} \in [-1, 1] \tag{4.5}$$

where

$$\Delta \boldsymbol{u}_i^{sim} = \frac{\partial \boldsymbol{u}^{sim}}{\partial \kappa_i} \tag{4.6}$$

The correlation matrix indicates the variation of how the displacement $\boldsymbol{u}^{sim}$ varies with respect to the i:th parameter in $\boldsymbol{\kappa}$.

If $|\cos \Phi_{ij}| = 1$ the parameters are linearly dependent and no unique solution might exist. $\cos \Phi_{ij} = 0$ implies that the parameters are linearly independent and they describe different properties of the deformation which lead to that they can individually be assigned values, independent of each other. Another case is when the derivative in (4.6) is equal to zero. This means that the simulated result is independent of κ_i and this will be denoted by NaN. When this happens, the parameters are either not activated, or insufficient data was recorded during the experiment to capture a certain behavior that would be needed.

Example 3.1 Consider a uniaxial test where only the elongation and force is recorded, see Fig. 4.1. With this data alone it is impossible to describe even the simplest linearly elastic model, since this both contains Young's modulus and Poission's ratio, see (6.1). Since no information about the elongation perpendicular to the force was recorded, the Poission's ratio can not be determined. This would lead to

$$\cos \Phi_{E\nu} = \frac{\Delta \boldsymbol{u}_E \Delta \boldsymbol{u}_\nu}{||\Delta \boldsymbol{u}_E||||\Delta \boldsymbol{u}_\nu||} = NaN \tag{4.7}$$

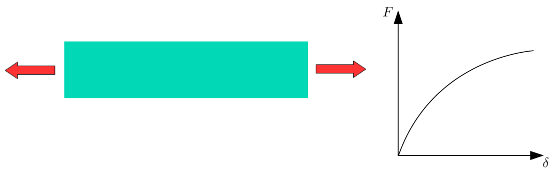
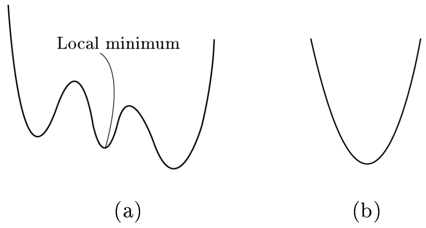


Figure 4.1: Uniaxial tensile test with corresponding global force-displacement diagram

The correlation matrix can be calculated even though no physical data exists. Generally speaking the correlation matrix could be a helpful tool when designing experiments if a specific material model is to be used. This is a good way to determine what kind of experiment that needs to be performed to be able to determine all parameters.

4.4 Solution to the inverse problem

To solve the inverse problem, an optimization on (4.4) needs to be done and one can use several different minima search algorithms. In general, (4.4) is non-convex which indicates that several local minima may exist, see Fig.4.2a). This requires the use of an algorithm that may overcome local minima in order to find an optimal solution. If one would use a gradient-based method, local minima would be difficult to overcome, instead a good approach is to use the Nelder-Mead simplex algorithm. This is a so called deterministic algorithm which means that it always gives the same result when using the same starting values. It does not use gradients to find the solution, instead it uses function evaluations which leads to that local minima may be overcome. On the other hand, this requires more calculations than a gradient based method and therefore converges more slowly. A more detailed description of the Nelder-Mead algorithm is found in Chapter 4.5.



(a) (b)

Figure 4.2: a) Non-convex function b) Convex function

4.4.1 Constraints

It is convenient to introduce constraints on each material parameter to ensure that no parameters are assigned values that is not allowed due to physical restrictions, i.e. that Young's modulus never can be negative. Doing this enables each parameter to have its own lower and upper value of what it can adopt.

Since the magnitude of the parameters can vary, i.e. Young's modulus is significantly higher than Poisson's ratio, it is convenient to normalize the parameters according to

$$\overline{\kappa}_i = \frac{\kappa_i - \kappa_u}{\kappa_l - \kappa_u} \in [0, 1] \tag{4.8}$$

From (4.8) the normalized parameter $\overline{\kappa}_i$ is obtained from its original parameter

κ_i and can now only adopt values between zero and one. κ_u and κ_l corresponds to the upper and lower limits of the parameter.

Different type of experimentally sampled variables can be used in (4.4), e.g. displacements, forces and strains. Since the magnitude of the different variables can differ greatly from each other, a normalization of (4.4) is done according to

$$f(\bar{\boldsymbol{\kappa}}) = \sum_i \left| \frac{\mathcal{M} a_i^{sim}(\bullet, \bar{\boldsymbol{\kappa}}) - a_i^{exp}}{a_i^{exp}} \right| \tag{4.9}$$

where a contains the variables.

To make the optimization loop search only within the boundaries, a penalty function is introduced to (4.9) as

$$f \leftarrow f + \underbrace{\sum_k^l \left[e^{-c(\bar{\kappa}_k - 0)} + e^{c(\bar{\kappa}_k - 1)} \right]}_{Penalty} \tag{4.10}$$

where l is the number of parameters in $\boldsymbol{\kappa}$. The penalty term increases the function value drastically when the optimization loop is searching near a parameters boundaries, and a solution close to or outside the boundaries is thus avoided. This forces the optimization loop to only search within the boundaries.

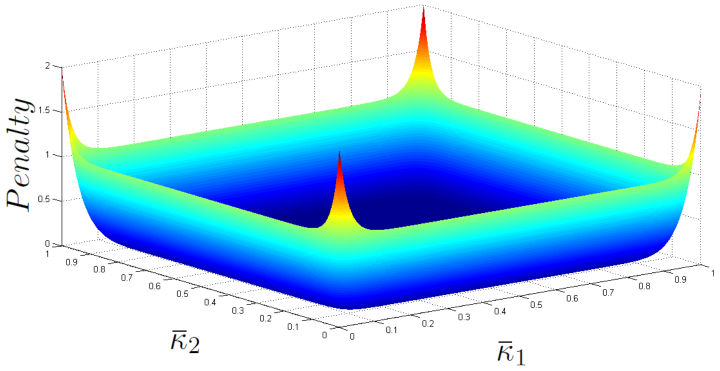
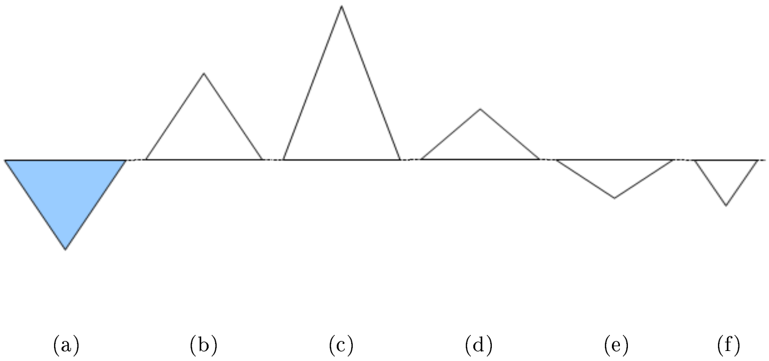


Figure 4.3: Illustration of the penalty function with two parameters

4.5 The Nelder-Mead (simplex) algorithm

As previously mentioned, the Nelder-Mead (simplex) algorithm is a so called deterministic, non-gradient method of searching for a minimum of a given function. This means that no derivatives needs to be calculated and it always gives the same result when using same initial values. The function is continuously searching within an area called a simplex. A simplex in $\mathbb{R}^n$ is a set of n+1 points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ in $\mathbb{R}^n$ such that the set of vectors $\{\boldsymbol{x}_i - \boldsymbol{x}_0 :: i = 1 \ldots n\}$ is linearly independent in $\mathbb{R}^n$. This simply means that if the function is in the two-dimensional space, the simplex becomes a triangle, in the three-dimensional space a tetrahedron etc. During the parameter identification, the dimension (n) of the space is equal to the number of material parameters.



(a) (b) (c) (d) (e) (f)

Figure 4.4: a) Original b) Reflection c) Expansion d) Outward contraction e) Inward contraction f) Shrink

The simplex can transform in five different ways that can be seen in Fig. 4.4 where the simplex is represented by a triangle. The transformations are based on the vertices function values that are all calculated and evaluated during the optimization process. A more detailed description of how this is done is found below.

Working procedure of the Nelder-Mead algorithm

The simplex with $n+1$ vertices are initially generated from an initial input point $\boldsymbol{\kappa}_0 \in \mathbb{R}^n$. From this point, the remaining n vertices are usually calculated from

$$\boldsymbol{\kappa}_j = \boldsymbol{\kappa}_0 + h_j e_j, \qquad j = 1, \dots, n$$

where h_j is a step size in the direction of the unit vector $e_j \in \mathbb{R}^n$.

The objective function to be minimized is denoted as f. $\boldsymbol{\kappa}_m$ and $\boldsymbol{\kappa}_M$ denotes the smallest and largest function value of the corner points in the simplex, i.e.

$$f(\boldsymbol{\kappa}_m) = \min_{1 \le j \le n+1} f(\boldsymbol{\kappa}_j), \qquad f(\boldsymbol{\kappa}_M) = \max_{1 \le j \le n+1} f(\boldsymbol{\kappa}_j)$$

The second largest value of the corner points is represented by the point $\boldsymbol{\kappa}_{\overline{m}}$.

As previously mentioned, five different transformations of the simplex can be performed during the optimization according to Fig. 4.4. These are controlled by four parameters, α for reflection, β for expansion, γ for contraction and δ for shrinkage. These numerical parameters are chosen so that $\alpha > 0$, $\beta > 1$, $\beta > \alpha$, $0 < \gamma < 1$ and $0 < \delta < 1$. Standard values used in most implementations are

$$\alpha = 1, \quad \beta = 2, \quad \gamma = \frac{1}{2}, \quad \delta = \frac{1}{2}$$

The first step in one iteration is to calculate the function value of each corner point $\boldsymbol{\kappa}_j$ to determine which has the worst (highest) value; $f(\boldsymbol{\kappa}_M)$. After this, the center of gravity of the face of the simplex opposite to this point is calculated as

$$\boldsymbol{\kappa}_g = \frac{1}{n} \sum_{\boldsymbol{\kappa}_j \neq \boldsymbol{\kappa}_M} \boldsymbol{\kappa}_j$$

From this, the so called "reflection point" is calculated from

$$\boldsymbol{\kappa}_r = \boldsymbol{\kappa}_g + \alpha(\boldsymbol{\kappa}_g - \boldsymbol{\kappa}_M)$$

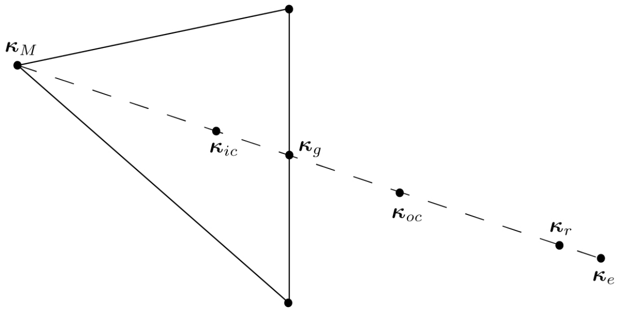


Figure 4.5: Initial simplex

Depending on the function value of $f(\boldsymbol{\kappa}_r)$, different transformations will be made of the simplex and the function value is evaluated according to

- Reflection
  If $f(\boldsymbol{\kappa}_m) < f(\boldsymbol{\kappa}_r) < f(\boldsymbol{\kappa}_M)$ put

  $$\boldsymbol{\kappa}_{new} := \boldsymbol{\kappa}_r$$

  and terminate the iteration.

- Expansion
  If $f(\boldsymbol{\kappa}_r) \leq f(\boldsymbol{\kappa}_m)$ it seems that the simplex is moving in a good direction and can therefore try to expand the simplex according to

  $$\boldsymbol{\kappa}_e = \boldsymbol{\kappa}_g + \beta(\boldsymbol{\kappa}_r - \boldsymbol{\kappa}_g)$$

  and put

  $$\boldsymbol{\kappa}_{new} := \begin{cases} \boldsymbol{\kappa}_e & \text{if } f(\boldsymbol{\kappa}_e) < f(\boldsymbol{\kappa}_r) \\ \boldsymbol{\kappa}_r & \text{otherwise} \end{cases}$$

  and terminate the iteration.

- Contraction
  If $f(\boldsymbol{\kappa}_r) \geq f(\boldsymbol{\kappa}_{\overline{m}})$, calculate the contraction point $\boldsymbol{\kappa}_c$ in two cases.

  1. Outward If $f(\boldsymbol{\kappa}_{\overline{m}}) \leq f(\boldsymbol{\kappa}_r) < f(\boldsymbol{\kappa}_M)$ calculate

     $$\boldsymbol{\kappa}_{oc} = \boldsymbol{\kappa}_g + \gamma(\boldsymbol{\kappa}_r - \boldsymbol{\kappa}_g)$$

     If now $f(\boldsymbol{\kappa}_{oc}) \leq f(\boldsymbol{\kappa}_r)$ put

     $$\boldsymbol{\kappa}_{new} := \boldsymbol{\kappa}_{oc}$$

     and terminate the iteration. Otherwise, perform a shrink transformation.

  2. Inward
     If $f(\boldsymbol{\kappa}_r) \geq f(\boldsymbol{\kappa}_M)$ calculate

     $$\boldsymbol{\kappa}_{ic} = \boldsymbol{\kappa}_g + \gamma(\boldsymbol{\kappa}_M - \boldsymbol{\kappa}_g)$$

     If now $f(\boldsymbol{\kappa}_{ic}) < f(\boldsymbol{\kappa}_M)$ put

     $$\boldsymbol{\kappa}_{new} := \boldsymbol{\kappa}_{ic}$$

     and terminate the iteration. Otherwise, perform a shrink transformation.

- Shrinkage
  If contraction is unsuccessful a shrinkage transformation is made and n new vertices are calculated as

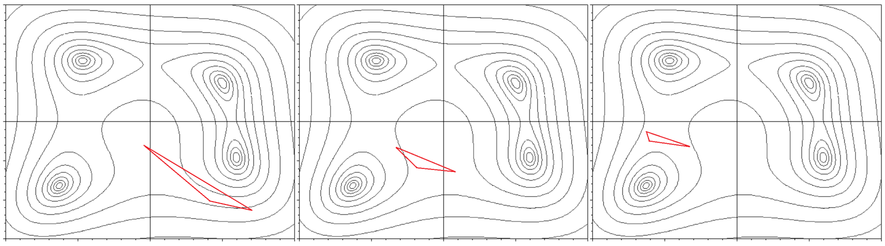
  $$\boldsymbol{\kappa}_j := \delta(\boldsymbol{\kappa}_j + \boldsymbol{\kappa}_m), \quad j = 1, \ldots, n$$

  and then restart the iteration step.

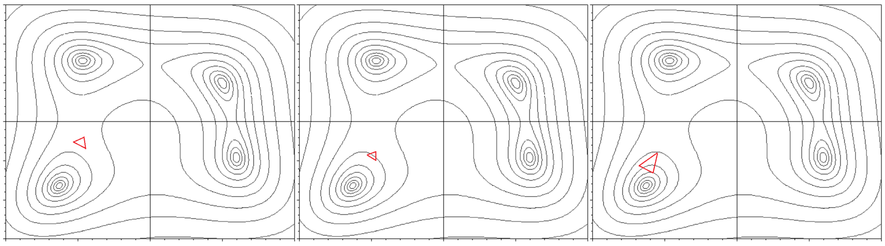
The optimization loop ends when $|f(\kappa_m^n) - f(\kappa_m^{n+1})| < tol$ where tol is a tolerance value for accepted convergence.

As an illustration of how the simplex alters shape during a minima search, see Fig. 4.6.

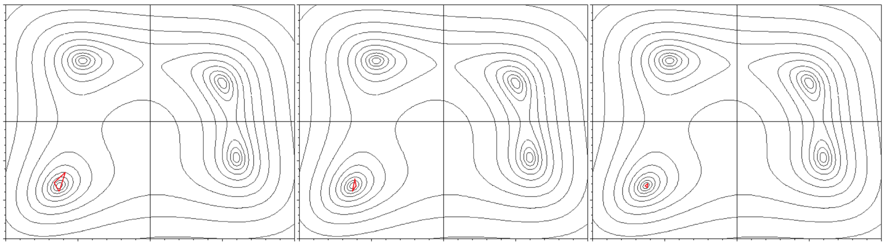
The interested reader is referred to [4] and [7].



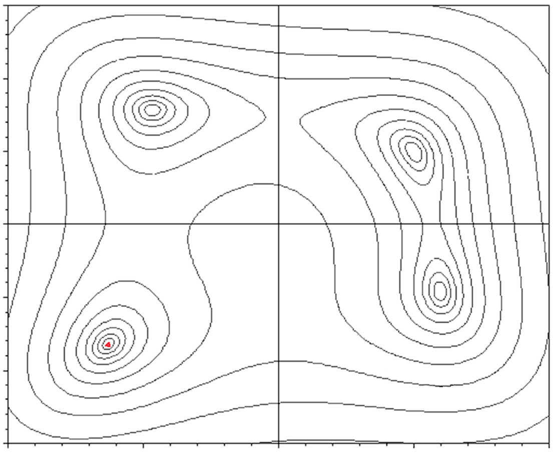
(a) It. 1 (b) It. 4 (c) It. 7



(d) It. 10 (e) It. 13 (f) It. 16



(g) It. 19 (h) It. 22 (i) It. 25



(j) It. 28

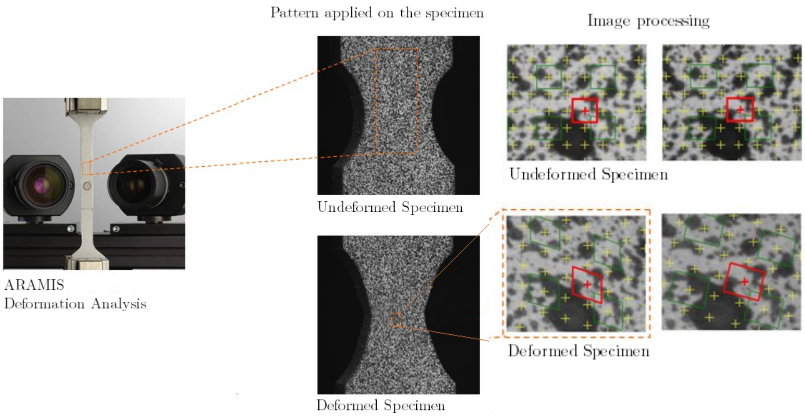
Figure 4.6: Illustration of the Nelder-Mead algorithm with two parameters where the simplex changes shape and moves towards a minima

Chapter 5

ARAMIS tensile test

The most common material testing performed in the industry today is a so called uniaxial tensile test. At these types of test, a homogeneous displacement field is assumed. To be able to have a controlled displacement field, a dog-bone shaped specimen is most commonly used. This is though often not the case for many types of polymers. During the loading, force and elongation is recorded. This is done today at Tetra Pak using a vision system that only record force and elongation in one direction with a limited number of points.

In order to record more information from the tensile test, a more advanced test needs to be performed. Therefore a cooperation with the external supplier, GOM (Gesellschaft für Optische Messtechnik) in Germany was started. The reason for the cooperation is because GOM has a so called ARAMIS system that can record displacements in more than one direction at any given point on a test specimen. This makes it possible to more accurately capture the deformation of the specimen and determine any inhomogeneous deformation.


ARAMIS
Deformation Analysis
Pattern applied on the specimen
Undeformed Specimen
Deformed Specimen
Image processing
Undeformed Specimen
Deformed Specimen

(a) (b) (c)

Figure 5.1: a) A mounted test in the ARAMIS system b) Applied pattern on specimen c) Image processing

The goal of the cooperation with GOM is to evaluate their testing equipment and see if the ARAMIS system can help improve the simulations by finding better material parameters compared to the one used today at Tetra Pak.

The ARAMIS system is a vision system that uses a technique called digital image correlation (DIC) with two cameras that continuously records a predefined pattern that follows the deformation of the specimen during the load. This has the benefit that no contact with the test specimen exists, minimizing the risk of measurement error. The setup of the ARAMIS system is shown in Fig. 5.1. When the loading is complete, it is possible to extract data from the points of interest to a text file. In this thesis the points shown in Fig. 5.2 are extracted and evaluated. The reason for the evenly distributed points in the four quadrants is to know if the test specimen is straightly attached in the test rig.

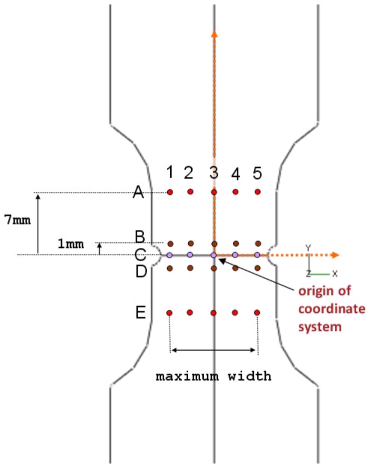
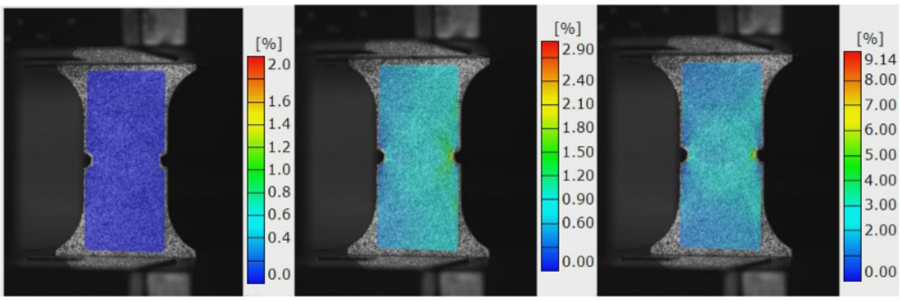


Figure 5.2: Points extracted in the ARAMIS system

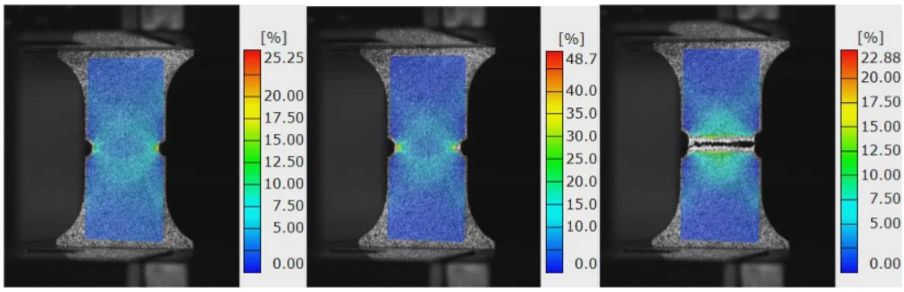
The idea behind using the ARAMIS system instead of the one Tetra Pak is currently using today is to more accurately capture the deformation of a specimen. The data obtained from the experiment can be used to i.e. calibrate a specific material model such that the simulations becomes more consistent with reality.

In this report, tests have been made on two different polymers; HDPE (High-density polyethylene) and LDPE (Low-density polyethylene) at two different test speeds (50 mm/min and 100 mm/min). Four tests on each polymer and test speed was made to account for measuring errors/deviations.

Figure 5.3 - 5.4 shows the longitudinal strain on a specimen in the ARAMIS system on a HDPE and LDPE polymer with a test speed of 50 mm/min.

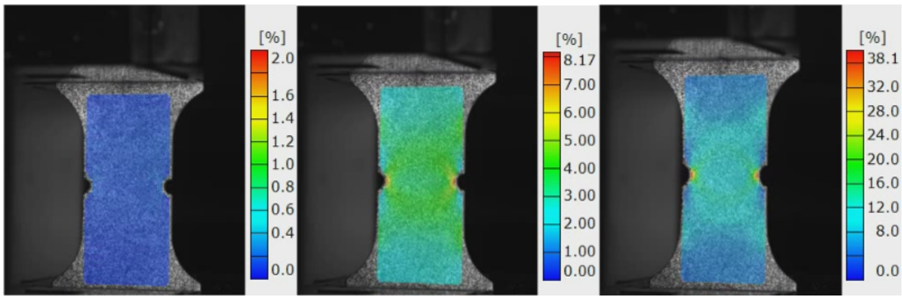

[%]
2.0
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.0
[%]
2.90
2.40
2.10
1.80
1.50
1.20
0.90
0.60
0.00
[%]
9.14
8.00
7.00
6.00
5.00
4.00
3.00
2.00
0.00

(a) 0.0 sec. (b) 0.6 sec. (c) 1.2 sec.

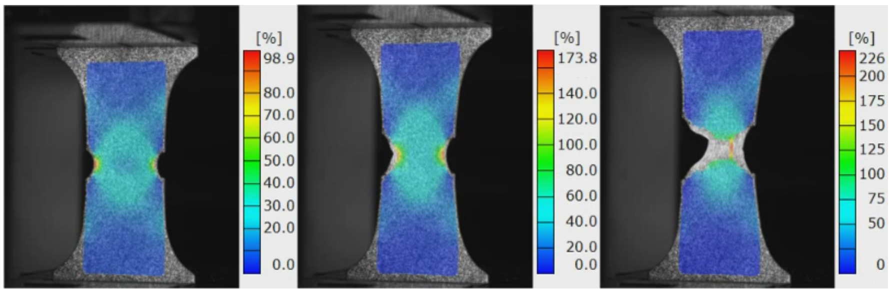

[%]
25.25
20.00
17.50
15.00
12.50
10.00
7.50
5.00
0.00
[%]
48.7
40.0
35.0
30.0
25.0
20.0
15.0
10.0
0.0
[%]
22.88
20.00
17.50
15.00
12.50
10.00
7.50
5.00
0.00

(d) 1.8 sec. (e) 2.4 sec. (f) 3.2 sec.

Figure 5.3: Longitudinal strain, HDPE 50 mm/min


[%]
2.0
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.0
[%]
8.17
7.00
6.00
5.00
4.00
3.00
2.00
1.00
0.00
[%]
38.1
32.0
28.0
24.0
20.0
16.0
12.0
8.0
0.0

(a) 0.0 sec. (b) 1.6 sec. (c) 3.6 sec.

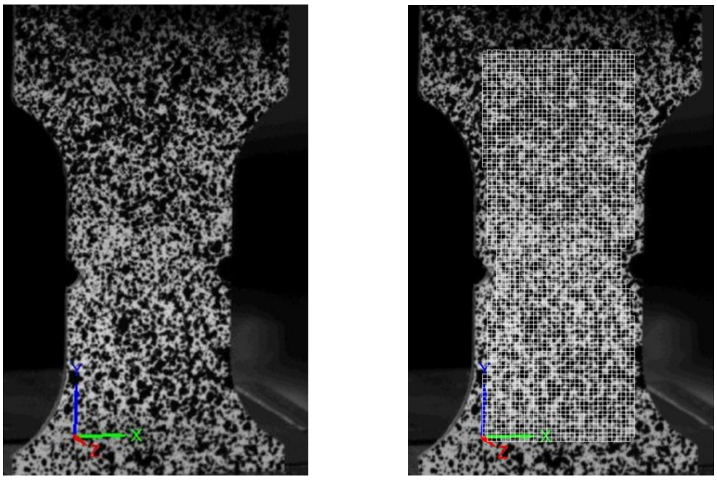

[%]
98.9
80.0
70.0
60.0
50.0
40.0
30.0
20.0
0.0
[%]
173.8
140.0
120.0
100.0
80.0
60.0
40.0
20.0
0.0
[%]
226
200
175
150
125
100
75
50
0

(d) 5.6 sec. (e) 8.6 sec. (f) 10.6 sec.

Figure 5.4: Longitudinal strain, LDPE 50 mm/min

5.1 Physical test

The test is displacement controlled from one of the clamps whereas the other remain fixed and the force from the clamps are measured during the load. In Fig. 5.5 the specimen has been sprayed with a stochastic pattern on which the cameras applies a grid. With this grid, the cameras can identify each points movement during the deformation. The points of interest, see Fig. 5.2 are extracted after the experiment.



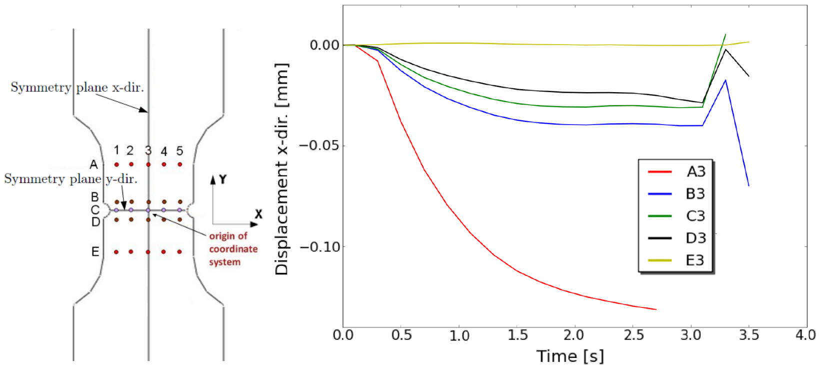
(a) (b)

Figure 5.5: a) Stochastic pattern on dog-bone specimen b) Grid applied on the specimen

When choosing points to extract data from it is appropriate to do this according to Fig. 5.2. With these points and the symmetry planes, see Fig. 5.6a) it is possible to detect if the specimen is straightly attached in the test rig. The alignment is of vital importance when comparing simulated and experimental data since the boundary conditions in the computer model states that the specimen is completely straight, more about this in Chapter 7. All of the tests have been made until fracture occurs and as mentioned before, four tests on each polymer and test speeds was made to account for measuring errors/deviations.

5.2 Results and conclusions

To check if the specimen is straightly attached, the points with index 3 can be checked so that their displacements in the x-direction remain unchanged during the loading.



(a) (b)

Figure 5.6: a) Symmetry planes of the specimen b) x-displacements of middle points (A3-E3)

Looking at Fig. 5.6b) one can see that the displacements at the symmetry line in the x-direction is non-zero which indicates that the specimen is not straightly attached. To compensate for this, the relative displacement is calculated from

$$u_{ij}^{x,rel} = u_{ij} - u_{i3} \tag{5.1}$$

where the index i refers to A-E and j to 1-5. This has also been done in the y-direction[1]. The results from the ARAMIS tensile test are presented in Appendix A. The results shows that both HDPE and LDPE have large deformation for all test speeds near the nodge. It can also be concluded that the displacement field varies greatly from each individual point and the material is therefore inhomogeneous as expected. There is also a significant difference between the two test speeds which indicates that the material response is time-dependent.

---

[1] The motivation for this compensation is thoroughly described in Chapter 7

Chapter 6

Specific material models

The following chapter describes the different material models used in the parameter identification. A few of Abaqus own built in models have been used and also a toolbox for Abaqus made by Ph. D. Jörgen Bergström called *PolyUMod*. The PolyUMod material library contains a set of material models which are often used for describing polymers.

6.1 Elasto-plastic

The simplest material model is given by the case when the material behaves only linearly elastic. According to Hooke's law the constitutive matrix $\boldsymbol{D}$ is defined by

$$\boldsymbol{D} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2}(1-2\nu) & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2}(1-2\nu) & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2}(1-2\nu) \end{bmatrix} \tag{6.1}$$

The plastic part is added with linearly splitted yield stresses with corresponding fixed effective plastic strains according to Fig. 6.1 where

$$\sigma_y = \sigma_y(\epsilon^p_{eff}) \tag{6.2}$$

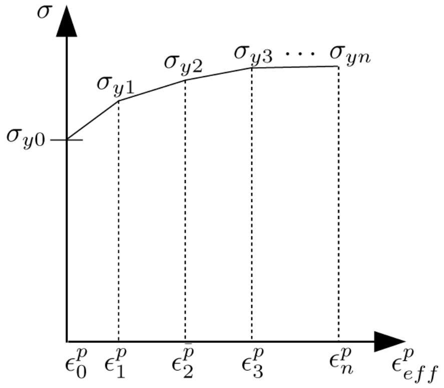


Figure 6.1: Illustration of the yield stresses as function of effective plastic strains

The material parameters that are used in the optimization for the elasto-plastic model can be found in Table 6.1.

Table 6.1: Material parameters used in the elasto-plastic model

Elasto-plastic model

Index	**Symbol**	**Unit**	**Description**
1	E	Pa	Young's modulus
2	ν	-	Poisson's ratio
3	σ_{y0}	Pa	Initial yield stress
4	σ_{y1}	Pa	Second yield stress
5	σ_{y2}	Pa	Third yield stress
$\vdots$	$\vdots$	$\vdots$	$\vdots$
n+3	σ_{yn}	Pa	n:th* yield stress

*n is the number of yield stresses chosen to be optimized

6.2 Hyper-elasto-plastic

From the theory of hyper-elasticity in Chapter 2.7 it is known that the strain energy W serves as a potential function for the stresses according to (2.33). In this section the various strain energy expressions for the different hyper-elastic material models that are used in the simulations are presented [1]. These describes the elastic behavior of the material model and the plastic deformations are added in the same manner as the elasto-plastic model according to (6.2) where Abaqus uses the multiplicative split according to (2.39).

6.2.1 Neo-Hookean form

The strain energy function for the Neo-Hookean material model is defined as

$$W = C_{10}(\bar{I}_1 - 3) + \frac{1}{D_1}(J^{el} - 1)^2 \tag{6.3}$$

where C_{10} and D_1 are elastic material parameters that depend on the initial shear stress μ_0 and bulk modulus K_0 respectively according to

$$C_{10} = \frac{\mu_0}{2} \quad \text{and} \quad D_1 = \frac{2}{K_0} \tag{6.4}$$

The material parameters that are used in the optimization can be found in Table 6.2.

Table 6.2: Material parameters used in the Neo-Hookean model

Neo-Hookean model

Index	**Symbol**	**Unit**	**Description**
1	C_{10}	Pa	Elastic material parameter 1
2	D_1	Pa^{-1}	Elastic material parameter 2
3	σ_{y0}	Pa	Initial yield stress
4	σ_{y1}	Pa	Second yield stress
5	σ_{y2}	Pa	Third yield stress
⋮	⋮	⋮	⋮
n+3	σ_{yn}	Pa	n:th* yield stress

*n is the number of yield stresses chosen to be optimized

6.2.2 Arruda-Boyce form

The strain energy function for the Arruda-Boyce material model is defined as

$$\begin{aligned} W \quad = \quad & \mu\left\{\frac{1}{2}(\bar{I}_1 - 3) + \frac{1}{20\lambda_m^2}(\bar{I}_1^2 - 9) + \frac{11}{1050\lambda_m^4}(\bar{I}_1^3 - 27) + \frac{19}{7000\lambda_m^6}\right. \\ & \left.\cdot(\bar{I}_1^4 - 81) + \frac{519}{673750\lambda_m^8}(\bar{I}_1^5 - 243)\right\} + \frac{1}{D}\left(\frac{(J^{el})^2 - 1}{2} - \ln J^{el}\right) \end{aligned} \tag{6.5}$$

where μ, λ_m and D are elastic material parameters that depend on the initial shear stress μ_0 and bulk modulus K_0 according to

$$\mu = \mu_0 \left(1 + \frac{3}{5\lambda_m^2} + \frac{99}{175\lambda_m^4} + \frac{513}{875\lambda_m^6} + \frac{42039}{67375\lambda_m^8} \right)^{-1} \quad \text{and} \quad D = \frac{2}{K_0} \tag{6.6}$$

The material parameters that are used in the optimization can be found in Table 6.3.

Table 6.3: Material parameters used in the Arruda-Boyce model

Arruda-Boyce model

Index	Symbol	Unit	Description
1	μ	Pa	Elastic material parameter 1
2	λ_m	-	Elastic material parameter 2
3	D	Pa^{-1}	Elastic material parameter 3
4	σ_{y0}	Pa	Initial yield stress
5	σ_{y1}	Pa	Second yield stress
6	σ_{y2}	Pa	Third yield stress
⋮	⋮	⋮	⋮
n+4	σ_{yn}	Pa	n:th* yield stress

*n is the number of yield stresses chosen to be optimized

6.2.3 Mooney-Rivlin form

The strain energy function for the Mooney-Rivlin material model is defined as

$$W = C_{10}(\overline{I}_1 - 3) + C_{01}(\overline{I}_2 - 3) + \frac{1}{D_1}(J^{el} - 1)^2 \tag{6.7}$$

where C_{10}, C_{01} and D_1 are elastic material parameters that depend on the initial shear stress μ_0 and bulk modulus K_0 according to

$$C_{10} + C_{01} = \frac{\mu_0}{2} \quad \text{and} \quad D_1 = \frac{2}{K_0} \tag{6.8}$$

The material parameters that are used in the optimization can be found in Table 6.4.

Table 6.4: Material parameters used in the Mooney-Rivlin model

Mooney-Rivlin model

Index	**Symbol**	**Unit**	**Description**
1	C_{10}	Pa	Elastic material parameter 1
2	C_{01}	Pa	Elastic material parameter 2
3	D_1	Pa^{-1}	Elastic material parameter 3
4	σ_{y0}	Pa	Initial yield stress
5	σ_{y1}	Pa	Second yield stress
6	σ_{y2}	Pa	Third yield stress
$\vdots$	$\vdots$	$\vdots$	$\vdots$
n+4	σ_{yn}	Pa	n:th* yield stress

*n is the number of yield stresses chosen to be optimized

6.3 PolyUMod

The following section will describe the material models used in the simulations from Jörgen Bergströms material library *PolyUMod*. As a helpful tool for the calibration of these models, the PolyUMod comes with a calibration tool. This tool uses the global force versus displacement field data from the experiments as input and give good starting points of the initial values of the parameters with corresponding constraints. Each material model is briefly explained and for a more detailed explanation of how each specific model is build the reader is referred to [2] and [3]. The superscript e, p and v in the equations refers to the elastic, plastic and viscous part. Subscripts in capital letters refers to the network it represents, i.e. A refers to network A. The various networks are illustrated in each models rheological representation.

6.3.1 Bergström-Boyce

The Bergström-Boyce model has been shown to give good results for predicting large-strain and time-dependent behavior for both soft biomaterials and traditional engineering rubbers.

The model is build from two parallel networks as can be seen in the rheological model in Fig. 6.2.

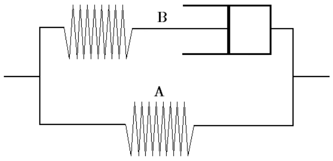


Figure 6.2: Illustration of the Bergström Boyce material model

The stresses acting in network A is obtained from

$$\boldsymbol{T}_A = \frac{\mu}{J\overline{\lambda^*}} \frac{\mathcal{L}^{-1}(\overline{\lambda^*}/\lambda_L)}{\mathcal{L}^{-1}(1/\lambda_L)} \mathrm{dev}[\mathbf{b}^*] + \kappa(J-1)\mathbf{I} \tag{6.9}$$

and network B

$$\boldsymbol{T}_B = \frac{s\mu}{J_B^e\overline{\lambda_B^{e*}}} \frac{\mathcal{L}^{-1}(\overline{\lambda_B^{e*}}/\lambda_L)}{\mathcal{L}^{-1}(1/\lambda_L)} \mathrm{dev}[\mathbf{b}_B^{e*}] + \kappa(J_B^e-1)\mathbf{I} \tag{6.10}$$

where $\mathcal{L}(x) = \coth(x) - 1/x$, known as the Lengevin function. $\overline{\lambda^*} = (\mathrm{tr}[\boldsymbol{b}^*]/3)^{1/2}$ where $\boldsymbol{b}^* = J^{-2/3}\boldsymbol{F}\boldsymbol{F}^T$, $\boldsymbol{b}_B^{e*} = (J_B^e)^{-2/3}\boldsymbol{F}_B^e(\boldsymbol{F}_B^e)^T$, $J_B^e = \det[\boldsymbol{F}_B^e]$ and $\overline{\lambda_B^{e*}} = (\mathrm{tr}[\boldsymbol{b}_B^{e*}]/3)^{1/2}$.

The total stress response is given by

$$\boldsymbol{T} = \boldsymbol{T}_A + \boldsymbol{T}_B \tag{6.11}$$

Since there is a viscous dashpot in network B, a equation describing the viscous flow is needed. The rate-equation for this viscous flow is given by

$$\dot{\gamma}_B^v = \dot{\gamma}_0(\overline{\lambda_B^v} - 1 + \xi)^C \left[R\left(\frac{\tau}{\tau_{base}} - \hat{\tau}_{cut}\right)\right]^m \tag{6.12}$$

where $\dot{\gamma}_0 \equiv 1/s$ is a constant introduced to ensure dimensional consistency. R is a ramp function defined as $R(x) = (x + |x|)/2$, $\overline{\lambda_B^v} = (\mathrm{tr}[(\boldsymbol{F}\boldsymbol{F}^T)_B^v]/3)^{1/2}$ and finally the effective stress driving the viscous flow is $\tau = ||\mathrm{dev}[\boldsymbol{T}_B]||_F$.

The material parameters used in the Bergström-Boyce model are summarized in Table 6.5.

Table 6.5: Material parameters used in the Bergström-Boyce model

Bergström Boyce model

Index	Symbol	Unit	Description
1	μ	Pa	Shear modulus of network A
2	λ_L	-	Locking stretch
3	κ	Pa	Bulk modulus
4	s	-	Relative stiffness of network B
5	ξ	-	Strain adjustment factor
6	C	-	Strain exponential
7	τ_{Base}	Pa	Flow resistance
8	m	-	Stress exponential
9	$\widehat{\tau}_{cut}$	Pa	Cut-off stress for flow

6.3.2 Hybrid model

The Hybrid model is a material model developed for predicting the large strain time-dependent behavior of ultra-high molecular weight polyethylene. It is also a good model for other types of thermoplastics.

The model can be illustrated in a rheological model according to Fig. 6.3.

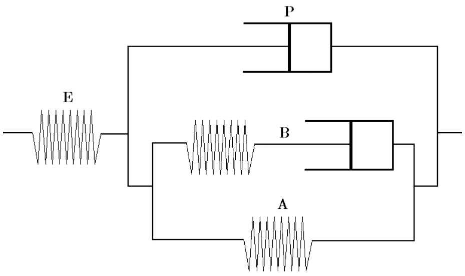


Figure 6.3: Illustration of the Hybrid material model

For a given deformation state, the Cauchy stress expression is given by

$$\boldsymbol{T} = \frac{1}{J^e}(2\mu\mathbf{E}^e + \lambda\mathrm{tr}[\mathbf{E}^e]\mathbf{I}) \tag{6.13}$$

where

$$\mu = \frac{E}{2(1+\nu)}, \qquad \lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}, \qquad J^e = \det[\boldsymbol{F}^e] \tag{6.14}$$

The stresses acting in network A is obtained from

$$\begin{aligned} \boldsymbol{T}_A \quad = \quad & \frac{1}{1+q}\left\{\frac{\mu_A}{J^p\overline{\lambda^{p*}}}\frac{\mathcal{L}^{-1}(\overline{\lambda^{p*}}/\lambda_L)}{\mathcal{L}^{-1}(1/\lambda_L)}\mathrm{dev}[\mathbf{b}^{p*}] + \kappa(J^p-1)\mathbf{I}\right. \\ & \left. +q\frac{\mu_A}{J}\left[I_1^*\mathbf{b}^{p*} - \frac{2I_2^{p*}}{3}\mathbf{I} - (\mathbf{b}^{p*})^2\right]\right\} \end{aligned} \tag{6.15}$$

and network B

$$\begin{aligned}\boldsymbol{T}_B &= \frac{s_B}{1+q}\left\{\frac{\mu_A}{J_B^e\overline{\lambda_B^{e*}}}\frac{\mathcal{L}^{-1}(\overline{\lambda_B^{e*}}/\lambda_L)}{\mathcal{L}^{-1}(1/\lambda_L)}\mathrm{dev}[\mathbf{b}_B^{e*}]+\kappa(J_B^e-1)\mathbf{I}\right. \\ &\quad \left.+q\frac{\mu_A}{J}\left[I_1^*\mathbf{b}^{e*}-\frac{2I_2^{e*}}{3}\mathbf{I}-(\mathbf{b}^{e*})^2\right]\right\}\end{aligned} \tag{6.16}$$

where $J^p = \det[\boldsymbol{F}^p]$, $\overline{\lambda_B^{p*}} = (\mathrm{tr}[\boldsymbol{b}_B^{p*}]/3)^{1/2}$, $\boldsymbol{b}^{p*} = (J^p)^{-2/3}\boldsymbol{F}\boldsymbol{F}^T$, $\boldsymbol{b}^{e*} = (J^e)^{-2/3}\boldsymbol{F}\boldsymbol{F}^T$, $I_1^* = \mathrm{tr}[\boldsymbol{b}^*]$ and $I_2^* = \boldsymbol{b}^* : \boldsymbol{b}^*$. s_B is a parameter that specifies the relative stiffness of network B compared to network A. During deformation s_B evolves according to

$$\dot{s}_B = -\alpha_B(s_B - s_{Bf})\dot{\gamma}_p \tag{6.17}$$

where s_B in the undeformed state is s_{Bi}, and s_{Bf} in the fully transformed state.

The time derivative of the visco-elastic deformation gradient of network B is given by

$$\begin{aligned}\dot{\mathbf{F}}_B^v = \mathbf{L}_B^v\mathbf{F}_B^v &= \dot{\gamma}_0\left(\frac{\tau_B}{\tau_B^{base}[1+R(p_B/\hat{p})]}\right)^{m_B} \\ &\quad \cdot(\mathbf{F}_B^e)^{-1}\frac{\mathrm{dev}[\boldsymbol{T}_B]}{\tau_B}\mathbf{F}^p\end{aligned} \tag{6.18}$$

where $\tau_B = ||\mathrm{dev}[\boldsymbol{T}_B]||_F$ is the effective shear stress driving the viscoelastic flow and $p_B = -\mathrm{tr}[\boldsymbol{T}_B]/3$ is the hydrostatic pressure.

The time rate of change of network P is given by

$$\begin{aligned}\dot{\mathbf{F}}^p = \mathbf{L}^p\mathbf{F}^p &= \dot{\gamma}_0\left(\frac{\tau^p}{\tau_{base}^p[1+R(p^p/\hat{p})]}\right)^{m^p} \\ &\quad \cdot(\mathbf{R}^e)^\top\frac{\mathrm{dev}[\mathbf{T}^p]}{\tau^p}\mathbf{R}^e\mathbf{F}^p \equiv \dot{\gamma}^p\mathbf{N}^p\end{aligned} \tag{6.19}$$

where $\tau^p = ||\mathrm{dev}[\mathbf{T}^p]||_F$ is the effective shear stress driving the plastic flow and $p^p = -\mathrm{tr}[\mathbf{T}^p]/3$ is the hydrostatic pressure. $\mathbf{T}^p = \boldsymbol{T} - [\boldsymbol{F}^e(\boldsymbol{T}_A + \boldsymbol{T}_B)\boldsymbol{F}^{eT}]/J^e$ is the stress acting on the relaxed configuration convected to the current configuration.

The material parameters used in the Hybrid model are summarized in Table 6.6.

Table 6.6: Material parameters used in the Hybrid model

Hybrid model

Index	Symbol	Unit	Description
1	E	Pa	Young's modulus
2	ν	-	Poisson's ratio
3	μ_A	Pa	Shear modulus A
4	λ_L	-	Locking stretch
5	q	-	Relative contribution of I_2 of network A
6	κ	Pa	Bulk modulus
7	s_{Bi}	-	Initial stiffness B
8	s_{Bf}	-	Final stiffness B
9	α_B	-	Transition rate stiffness B
10	τ_{base}^{B}	Pa	Flow resistance B
11	m_B	-	Stress exponent B
12	$\widehat{p}$	Pa	Pressure dependence of flow
13	τ_{base}^{p}	Pa	Flow resistance p
14	m^p	-	Stress exponent p

6.3.3 Three network

The Three network model is a material model developed for thermoplastic materials. It is similar to the Hybrid model but is designed to be more numerically efficient.

The model can be illustrated in a rheological model according to Fig. 6.4.

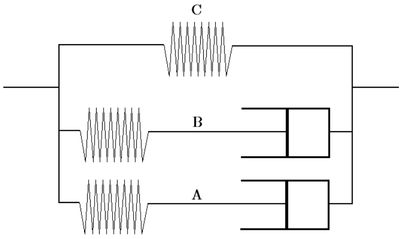


Figure 6.4: Illustration of the three network material model

The stress acting in network A is obtained from

$$\boldsymbol{T}_A = \frac{\mu_A}{J_A^e \overline{\lambda_A^{e*}}} \left[1 + \frac{\theta - \theta_0}{\hat{\theta}}\right] \frac{\mathcal{L}^{-1}(\overline{\lambda_A^{e*}}/\lambda_L)}{\mathcal{L}^{-1}(1/\lambda_L)} \operatorname{dev}[\mathbf{b}_A^{e*}] + \kappa(J_A^e - 1)\mathbf{1} \tag{6.20}$$

and network B

$$\boldsymbol{T}_B = \frac{\mu_B}{J_B^e \overline{\lambda_B^{e*}}} \left[1 + \frac{\theta - \theta_0}{\hat{\theta}}\right] \frac{\mathcal{L}^{-1}(\overline{\lambda_B^{e*}}/\lambda_L)}{\mathcal{L}^{-1}(1/\lambda_L)} \operatorname{dev}[\mathbf{b}_B^{e*}] + \kappa(J_B^e - 1)\mathbf{1} \tag{6.21}$$

where θ is the current temperature and θ_0 is a reference temperature. The effective shear modulus is taken to evolve with plastic strain according to

$$\dot{\mu}_B = -\beta[\mu_B - \mu_{Bf}]\dot{\gamma}_A \tag{6.22}$$

where μ_B in the undeformed state is μ_{Bi}, and μ_{Bf} in the fully transformed state. The stress acting in network C is given by

$$\begin{aligned} \boldsymbol{T}_C &= \frac{1}{1+q}\left\{\frac{\mu_C}{J\overline{\lambda^*}}\left[1+\frac{\theta-\theta_0}{\hat{\theta}}\right]\frac{\mathcal{L}^{-1}\left(\frac{\overline{\lambda^*}}{\lambda_L}\right)}{\mathcal{L}^{-1}\left(\frac{1}{\lambda_L}\right)}\mathrm{dev}[\mathbf{b}^*]\right. \\ &\qquad \left. +\kappa(J-1)\mathbf{1} + q\frac{\mu_C}{J}\left[I_1^*\mathbf{b}^* - \frac{2I_2^*}{3}\mathbf{I} - (\mathbf{b}^*)^2\right]\right\} \end{aligned} \tag{6.23}$$

The total stress response in the system is given by

$$\boldsymbol{T} = \boldsymbol{T}_A + \boldsymbol{T}_B + \boldsymbol{T}_C \tag{6.24}$$

The effective deviatoric flow rate for network A is given by

$$\dot{\gamma}_A = \dot{\gamma}_0\left(\frac{\tau_A}{\hat{\tau}_A + aR(p_A)}\right)^{m^A}\left(\frac{\theta}{\theta_0}\right)^n \tag{6.25}$$

and network B

$$\dot{\gamma}_B = \dot{\gamma}_0\left(\frac{\tau_B}{\hat{\tau}_B + aR(p_B)}\right)^{m^B}\left(\frac{\theta}{\theta_0}\right)^n \tag{6.26}$$

where $\tau_A = ||\mathrm{dev}[\boldsymbol{T}_A]||_F$, $\tau_B = ||\mathrm{dev}[\boldsymbol{T}_B]||_F$, $p_A = -\mathrm{tr}[\boldsymbol{T}_A]/3$ and $p_B = -\mathrm{tr}[\boldsymbol{T}_B]/3$.

The material parameters used in the Three network model are summarized in Table 6.7.

Table 6.7: Material parameters used in the three network model

Three network model

Index	**Symbol**	**Unit**	**Description**
1	μ_A	Pa	Shear modulus of network A
2	$\widehat{\theta}$	T	Temperature factor
3	λ_L	-	Locking stretch
4	κ	Pa	Bulk modulus
5	$\widehat{\tau}_A$	Pa	Flow resistance of network A
6	a	-	Pressure dependence of flow
7	m_A	-	Stress exponential of network A
8	n	-	Temperature exponential
9	μ_{Bi}	Pa	Initial shear modulus of network B
10	μ_{Bf}	Pa	Final shear modulus of network B
11	β	-	Evolution rate of μ_B
12	$\widehat{\tau}_B$	Pa	Flow resistance of network B
13	m_B	-	Stress exponential of network B
14	μ_C	Pa	Shear modulus of network C
15	q	-	Relative contribution of I_2 of network C
16	α	T^{-1}	Thermal expansion coefficient
17	θ_0	T	Thermal expansion reference temperature

Chapter 7

Method

In this chapter a description of the method and geometry will be performed. A discussion of the assumptions made on the geometry will also be described.

7.1 Geometry

The geometry of the specimen is a so called dog-bone with a nodge in the middle according to Fig. 7.1.

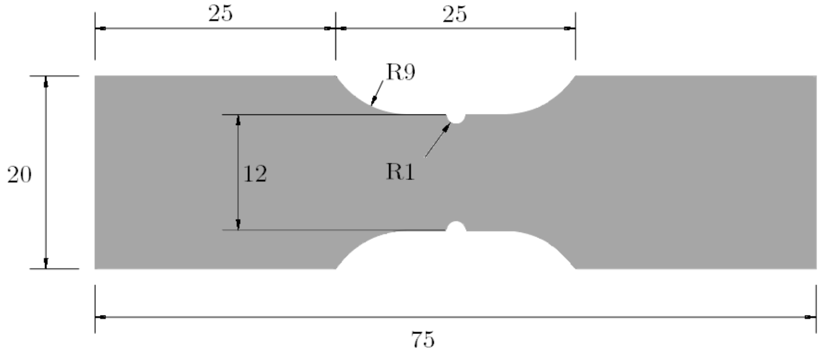
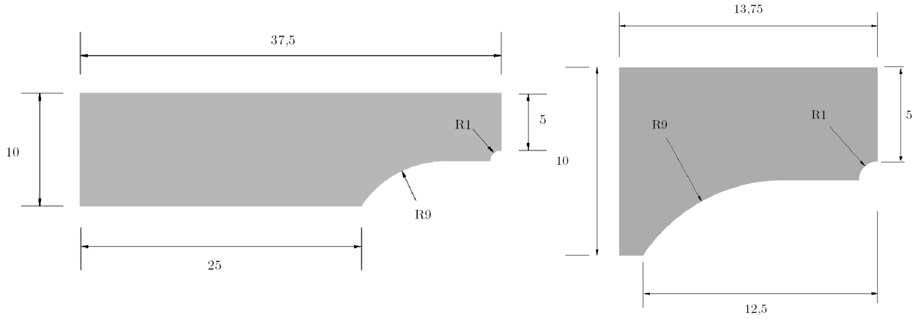


Figure 7.1: Dog-bone specimen [mm]

To be able to save time and computer power during the simulations, symmetry on the geometry in Fig. 7.1 will be used. Figure 7.2a) shows the symmetric part of the geometry. In order for the ARAMIS system to perform the tensile test, the clamps on the machine holds on a relatively big part of the geometry. To account for this, a smaller part of the symmetric geometry will be considered as seen in Fig. 7.2b) where the material held by the clamps is removed. From this geometry, the left side will be subjected to a displacement controlled load in Abaqus.



(a) (b)

Figure 7.2: a) Symmetry part b) Part used in simulations [mm]

The part will be meshed according to Fig. 7.3 where three- and four-node isoparametric elements are used. To save even more computer power and simulation time and due to the fact that the specimen is relatively thin, the body will be treated as a two-dimensional body with plane-stress conditions and a thickness of 0.6mm.

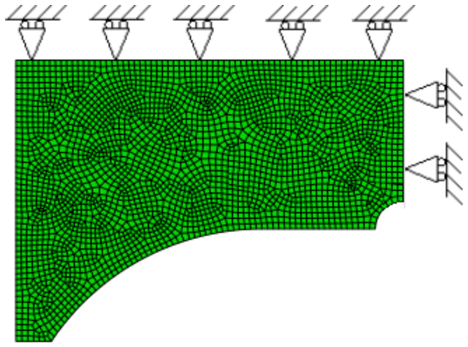


Figure 7.3: Finite element mesh consisting of 70 three-node elements and 2688 four-node elements with plane stress condition with applied boundary conditions

Table 7.1 shows the displacements of the clamps from the ARAMIS system just before the material begins to break in the material[1]. When simulating the tensile test for the optimization, the half average value will be considered[2]. Since symmetry is assumed, a boundary condition will enforce that the y-directions in C points will remain zero at all times. This has to be accounted for in the measurement data since thus is not the case from the ARAMIS system.

---

[1] Tests 2-4 on HDPE 100mm/min where unsuccessful and only incomplete data was recorded during the tests

[2] The half value is used because only half of the body is considered in the simulations

Table 7.1: Displacement of the clamps for all tests

Displacement clamps [mm]

Test:	1	2	3	4	Avg
HDPE 50mm/min	1.99	1.97	1.99	1.91	1.97
HDPE 100mm/min	1.86	-	-	-	1.86
LDPE 50mm/min	5.97	5.88	5.85	5.78	5.87
LDPE 100mm/min	4.25	4.10	4.23	4.23	4.2

The relative displacements will be calculated in the same manner as (5.1) according to

$$u_{kl}^{y,rel} = u_{kl} - u_{Cl} \tag{7.1}$$

where index k and l refers to A-C and 1-3 respectively[3].

7.1.1 Optimization procedure

The optimization procedure is done according to the method described in Chapter 4. A description of the procedure can be seen in Fig. 7.4 where initial values are put in to Python that writes the values to the material file and uses Abaqus as the FE-solver. When analysis is completed, the values are extracted in to Python where the error measure f is calculated. If $f >$ tol Python writes new values to the material file and the optimization loop continues until $f <$ tol and the values are accepted.

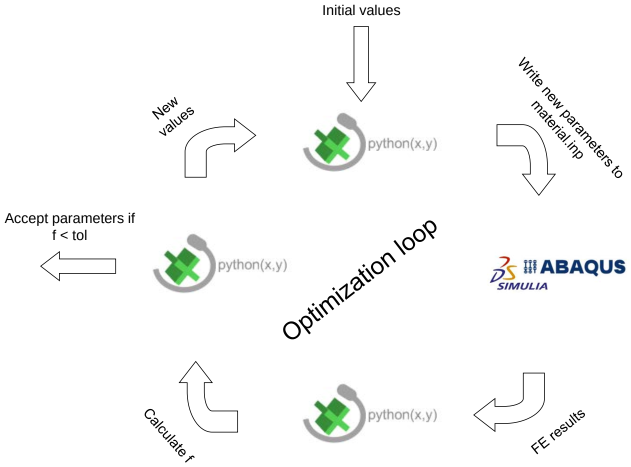

Initial values
Write new parameters to material.inp
New values
python(x,y)
Accept parameters if f < tol
python(x,y)
Optimization loop
ABAQUS
SIMULIA
Calculate f
python(x,y)
FE results

Figure 7.4: Flow chart

---

[3] Why only these points are considered is because symmetry is used and only the points in the second quadrant will be used in the simulations

Chapter 8

Test calibration

In this chapter a calibration of two example problems will be performed. The motivation for this is to validate the underlying theory for the parameter identification process as described in Chapter 4. The conclusions of these examples will be presented in the end of this chapter. The calibration will be based on the global force versus displacement field and also displacements on a number of chosen points in the body. The geometries of the two examples are shown in Fig. 8.1.

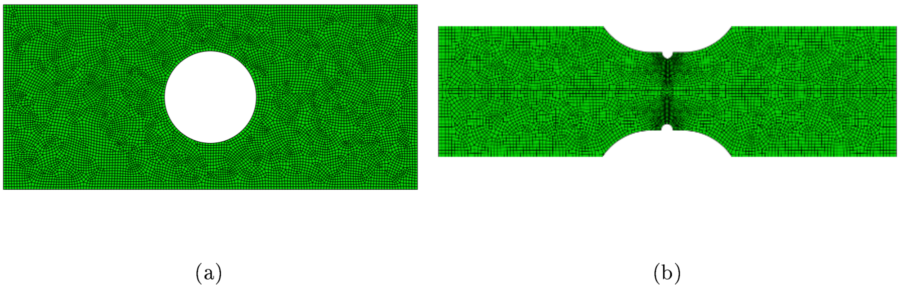


Figure 8.1: a) Problem 1 b) Problem 2

Usually, experimental data are subjected to measurement errors, making it complicated to solve the inverse problem given by (4.4). Instead, when verifying if the theory of parameter identification works, numerically generated experimental data will be used in the calibrations. The data will be created by a uniaxial tensile test on both examples where symmetry will be used. The specimens are given an elasto-plastic material model with von Mises isotropic hardening, see Chapter 6.1. When calibrating the material parameters, the exact same geometry is used with the same material model, but with a different set of material parameters. If the calibration is successful, this would result in the same set of parameters with a small function value, f given by (4.10).

8.1 Problem 1

The experimental data for problem 1 will be generated from the points on the geometry shown in Fig. 8.2 and the material parameters are given known values according to Table 8.1.

Table 8.1: Material parameters used to create virtual experimental data

E	ν	σ_{y0}	σ_{y1}	σ_{y2}	σ_{y3}	σ_{y4}
3000	0.27	8	15	25	35	50

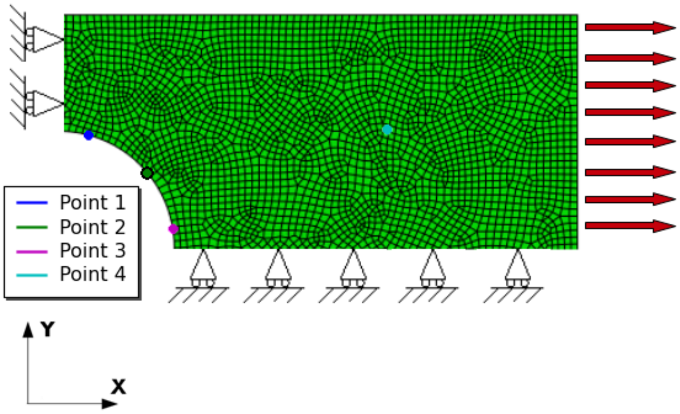


Figure 8.2: Geometry and boundary conditions. Measurement points in colored dots

The results from the parameter identification with the highest and lowest function value are shown in Fig. 8.3. Both runs are shown to give an idea of how much the simulation with the highest function value differs from the simulation with the best. The start and end values for the parameters are given in Table 8.2 and 8.3 respectively. The correlation matrix is given by (8.1).

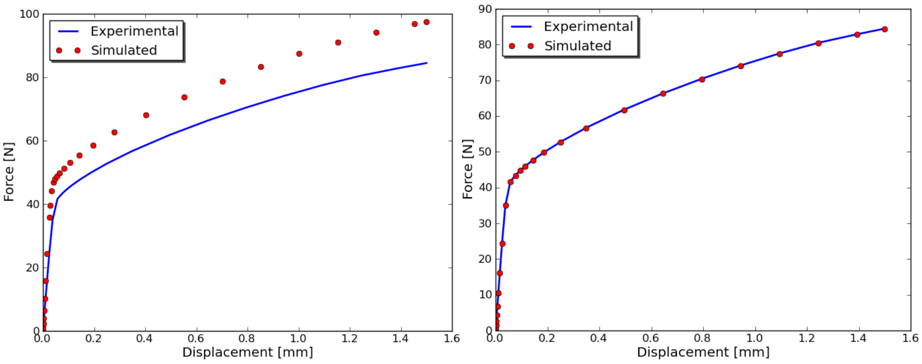
Table 8.2: Material parameters at the start of the parameter identification process

	E	ν	σ_{y0}	σ_{y1}	σ_{y2}	σ_{y3}	σ_{y4}
Start^1	3500	0.22	12	19	29	39	54
Start^2	3300	0.25	10	17	27	37	52

$$\cos\Phi_{ij} = \begin{bmatrix} 1 & -0.029 & 0.27 & -0.226 & -0.116 & NaN & NaN \\ -0.029 & 1 & -0.03 & 0.056 & 0.006 & NaN & NaN \\ 0.27 & -0.03 & 1 & -0.765 & -0.792 & NaN & NaN \\ -0.226 & 0.056 & -0.765 & 1 & 0.282 & NaN & NaN \\ -0.116 & 0.006 & -0.792 & 0.282 & 1 & NaN & NaN \\ NaN & NaN & NaN & NaN & NaN & NaN & NaN \\ NaN & NaN & NaN & NaN & NaN & NaN & NaN \end{bmatrix} \tag{8.1}$$

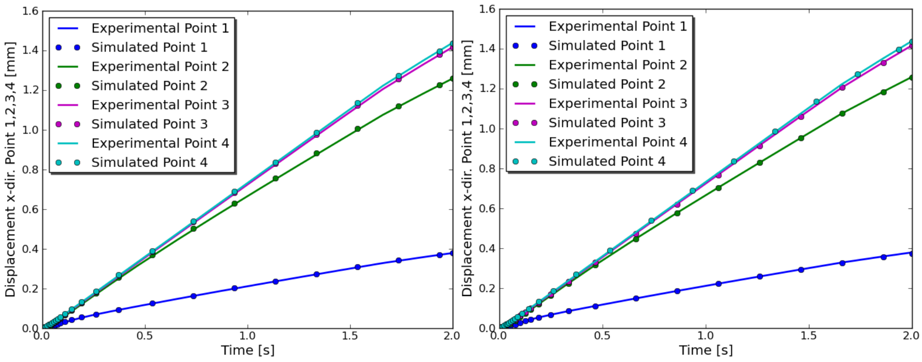
Table 8.3: Material parameters at the end of the parameter identification process

	E	ν	σ_{y0}	σ_{y1}	σ_{y2}	σ_{y3}	σ_{y4}	f
End^1	4544	0.338	9.17	17.27	28.7	40.23	46.68	0.57
End^2	3007	0.2706	7.99	14.99	24.92	39.73	48.78	0.025



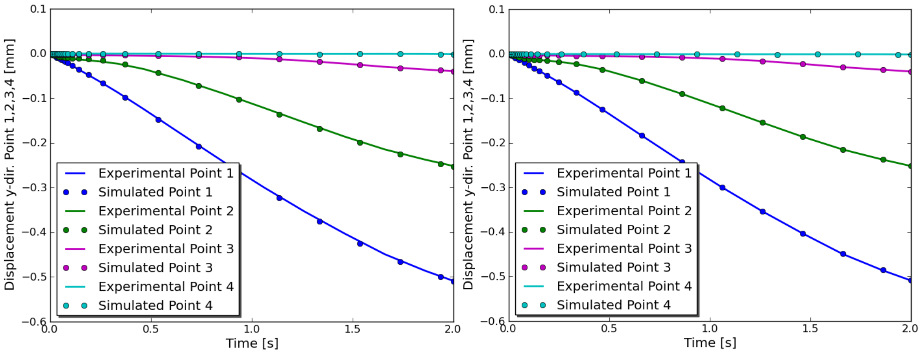
(a) Global force vs. displacement run 1

(b) Global force vs. displacement run 2



(c) x-displacement vs. time run 1

(d) x-displacement vs. time run 2



(e) y-displacement vs. time run 1

(f) y-displacement vs. time run 2

Figure 8.3: Results from simulations compared to virtual experimental data

8.2 Problem 2

The experimental data for problem 2 will be generated from the points on the geometry shown in Fig. 8.4 and the material parameters are given known values according to Table 8.4.

Table 8.4: Material parameters used to create virtual experimental data

E	ν	σ_{y0}	σ_{y1}	σ_{y2}	σ_{y3}	σ_{y4}	σ_{y5}
1000	0.22	10	15	20	25	35	50

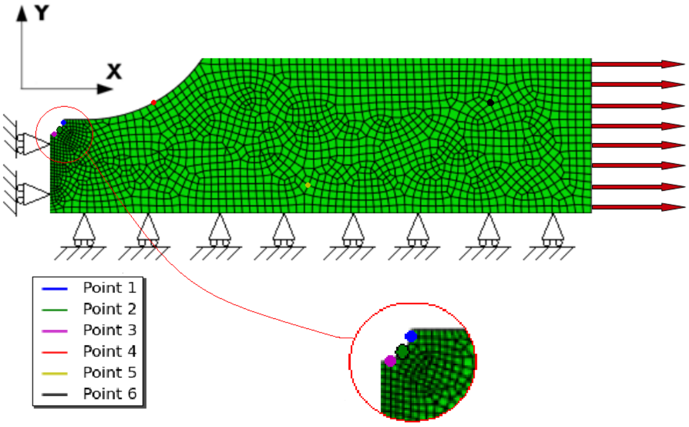


Figure 8.4: Geometry and boundary conditions. Measurement points in colored dots

The results from the parameter identification with the highest and lowest function value are shown in Fig. 8.5 and 8.6. The parameters start and end values are given in Table 8.5 and 8.6 respectively. The first and fourth run are shown to give an idea of how much the simulation with the highest function value differs from the simulation with the best. The correlation matrix is given by (8.2).

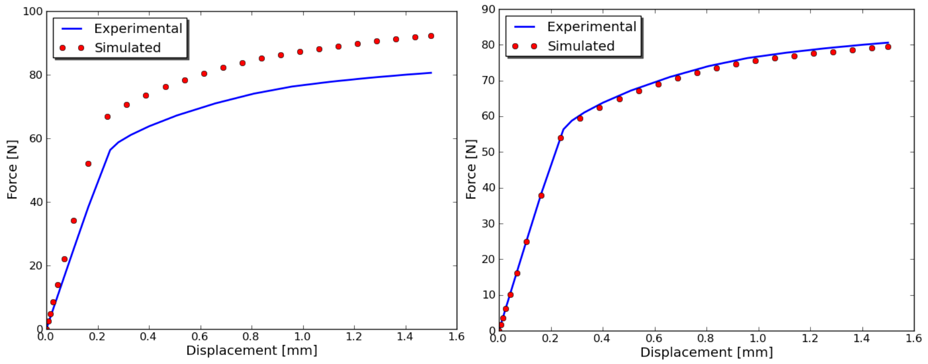
Table 8.5: Material parameters at the start of the parameter identification process

	E	ν	σ_{y0}	σ_{y1}	σ_{y2}	σ_{y3}	σ_{y4}	σ_{y5}
Start^1	800	0.27	13	18	23	28	38	53
Start^2	900	0.2	8	13	18	23	33	48
Start^3	1100	0.24	12	17	22	27	37	52
Start^4	1050	0.2	9	16	19	26	34	51

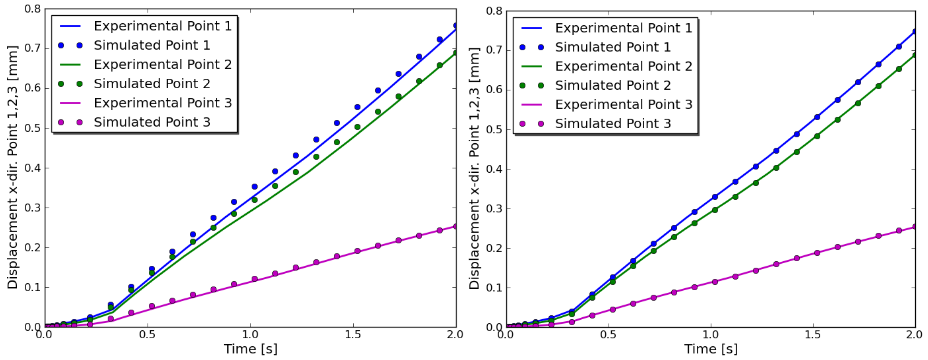
Table 8.6: Material parameters at the end of the parameter identification process

	E	ν	σ_{y0}	σ_{y1}	σ_{y2}	σ_{y3}	σ_{y4}	σ_{y5}	f
End[1]	1364	0.265	11.4	16.7	23.2	28.6	38.8	54.1	0.39
End[2]	884	0.219	8.8	13.3	17.6	21.7	33.8	46.5	0.15
End[3]	1200	0.24	11.1	16.4	22.5	27.1	36.4	55	0.29
End[4]	990	0.219	9.85	14.78	19.7	24.7	34	51.75	0.038

$$\cos\Phi_{ij} = \begin{bmatrix} 1 & 0.126 & 0.594 & -0.41 & -0.39 & -0.056 & -0.063 & NaN \\ 0.126 & 1 & 0.002 & -0.052 & 0.063 & 0.043 & 0.048 & NaN \\ 0.594 & 0.002 & 1 & -0.358 & -0.76 & -0.257 & -0.121 & NaN \\ -0.41 & -0.052 & -0.358 & 1 & 0.158 & -0.131 & -0.092 & NaN \\ -0.39 & 0.063 & -0.76 & 0.158 & 1 & 0.567 & 0.341 & NaN \\ -0.056 & 0.043 & -0.257 & -0.131 & 0.567 & 1 & 0.438 & NaN \\ -0.063 & 0.048 & -0.121 & -0.092 & 0.341 & 0.438 & 1 & NaN \\ NaN & NaN & NaN & NaN & NaN & NaN & NaN & NaN \end{bmatrix} \tag{8.2}$$

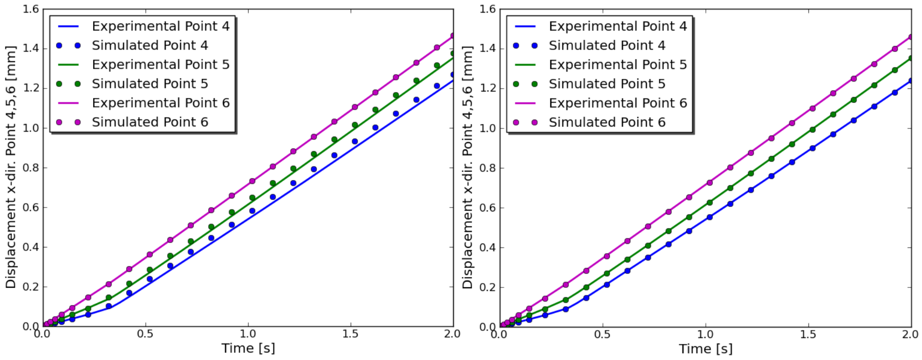


(a) Global force vs. displacement run 1 (b) Global force vs. displacement run 4

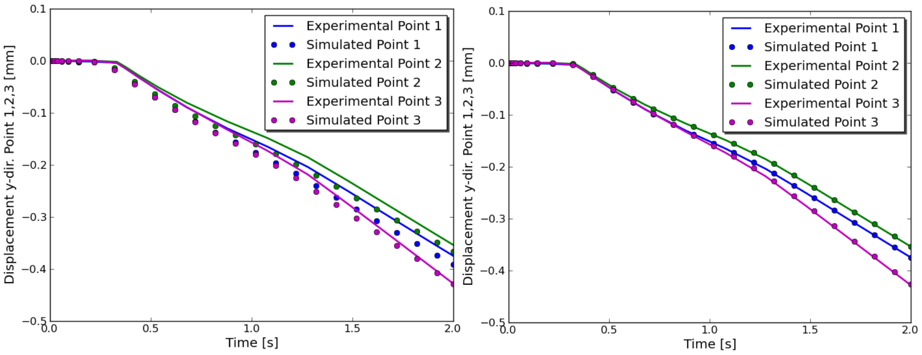


(c) x-displacement vs. time, point 1,2,3 run 1 (d) x-displacement vs. time, point 1,2,3 run 4

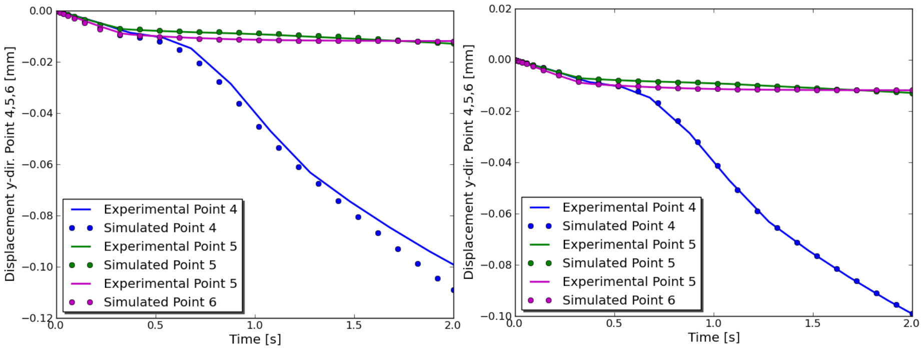
Figure 8.5: Results from simulations compared to virtual experimental data



(a) x-displacement vs. time, point 4,5,6 run 1 (b) x-displacement vs. time, point 4,5,6 run 4



(c) y-displacement vs. time, point 1,2,3 run 1 (d) y-displacement vs. time, point 1,2,3 run 4



(e) y-displacement vs. time, point 4,5,6 run 1 (f) y-displacement vs. time, point 4,5,6 run 4

Figure 8.6: Results from simulations compared to virtual experimental data

8.3 Conclusions

Figure 8.3, 8.5 and 8.6 shows how well adjusted the simulated data from the best run is with the experimental results of the two problem examples. Since the simulated curves fit the experimental results very well, the parameter identification seems to be successful which can be confirmed from Table 8.3 and 8.6 where there is a small function value on the second and fourth run respectively. Why some of the parameters deviates significantly more from the correct value than the other parameters can be explained by (8.1) and (8.2) where the NaN expression indicates that these parameters are not activated during this test. There is simply not enough plastic deformation in the body to activate these and therefore impossible to calibrate.

From (8.1) and (8.2) one can see there is a very small correlation between Young's modulus and Poisson's ratio. This confirms that they describe two different behaviors of the material and are possible to identify with this type of model which confirms that the correlation matrix is a good tool for determining the parameters linear dependency.

One discovery that was made during this calibration is the great importance of what the initial values are. Table 8.3 and 8.6 shows that if a start value is far away from the correct value the Nelder-Mead algorithm have trouble finding the most optimal solution. Once it have converged to a local minima it can never find the global minima. As can be seen in Fig. 8.3, 8.5 and 8.6 there is a noticeable difference between the runs which indicates it is important to try a number of different starting values and in the end, choose the parameters that gives the lowest function value. It can also be seen in these figures that it is the global force versus displacement field that is most dependent of the final parameters, while the displacements in the specimen do not differ as much. Therefore it is also concluded that a normalization of the variables according to (4.9) is important so that every variable gives the same percentage deviation for the optimization.

Chapter 9

Results

This chapter presents the results from all the various simulations with the material models described in Chapter 6. The geometry that was used is the one shown in Fig. 7.3. Due to lack of time, only simulations on HDPE 50mm/min were performed. Every material parameter with its start and end values with corresponding function value are presented in tables in each section. For some material models, the plastic strains are needed as indata, these are given as subscripts on the yield stresses in brackets. Each table ends with a summary of how many iterations that was needed for the optimization to converge and the total time for each simulation with the lowest function value. The motivation for this is to present an estimation of how long time[1] each simulation with each material model takes. Figure 9.1 shows an example of where a simulation is assumed to have converged indicated with a red dot. For each simulation the force versus displacement is presented. The simulations have been performed on a Windows based computer, running a dual core with 4GB RAM using Abaqus Standard. Figures showing the displacement in each individual point in the specimen are given in Appendix B.

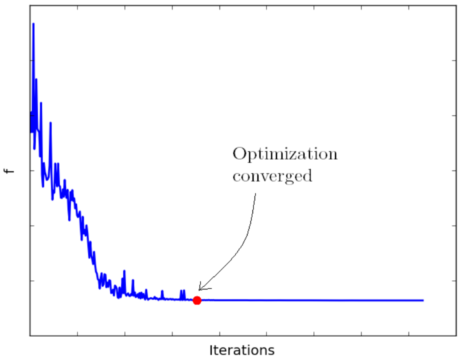


Figure 9.1: Illustration of the convergence point in the results

---

[1] This is a rough approximation since the number of iterations is greatly dependent of the starting values

9.1 Elasto-plasticity

Table 9.1: Start/End values from simulations of the elasto-plastic model

Elasto-plastic model

Mat. par.	**Start**^{1}	**End**^{1}	**Start**^{2}	**End**^{2}	**Start**^{3}	**End**^{3}
E	420	409.1	850	573.5	560	573.2
ν	0.22	0.25	0.225	0.247	0.25	0.256
σ_{y0}	$7.7_{[0.0]}$	7.1	$8.5_{[0.0]}$	6.56	$8_{[0.0]}$	7.54
σ_{y1}	$14.7_{[0.005]}$	14.4	$13.5_{[0.005]}$	13.2	$13_{[0.005]}$	17.46
σ_{y2}	$21.3_{[0.01]}$	21.1	$22_{[0.01]}$	18.85	$20_{[0.01]}$	19.1
σ_{y3}	$24.9_{[0.1]}$	25.8	$27_{[0.05]}$	29.5	$36_{[0.1]}$	49.7
σ_{y4}	$33.8_{[0.6]}$	33.5	$35_{[0.1]}$	42.35	$37_{[0.3]}$	26.8
σ_{y5}	-	-	$41_{[0.3]}$	40.4	$42_{[0.6]}$	50.8
	$f^1 = 94.49$		$f^2 = 68.93$		$f^3 = 68.89$	

$Iterations : 300 \mid Time/iteration : 36\ sec. \mid \sum time : 3h : 00min.$

$Time/iteration\ 3D : 48\ sec. \mid f^3_{3D} = 67.6$

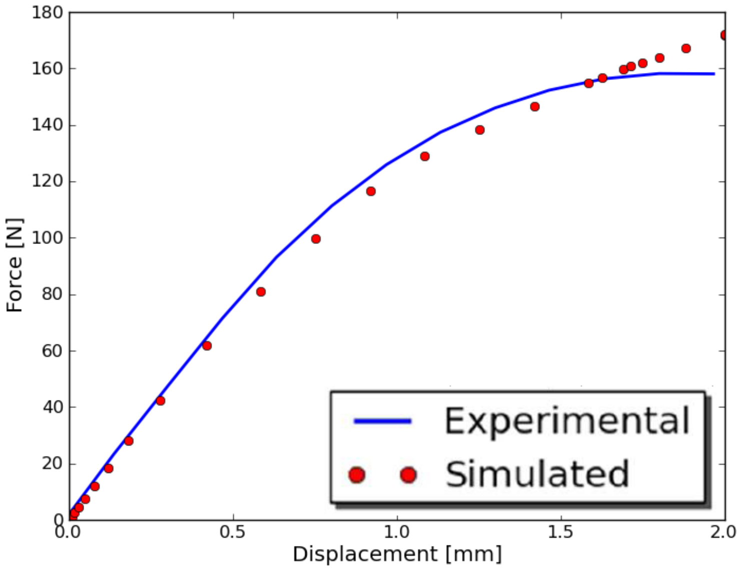


Figure 9.2: Global force vs. displacement with the elasto-plastic material model, run 3

9.2 Hyper-elasto-plastic

9.2.1 Neo-Hookean form

Table 9.2: Start/End values from simulations of the Neo-Hooke model

Neo-Hooke model

Mat. par.	**Start**^{1}	**End**^{1}	**Start**^{2}	**End**^{2}	**Start**^{3}	**End**^{3}
C_{10}	70	50.6	110	114.3	70	107.1
$D1$	0.01	0.013	0.005	0.0053	0.01	0.005
σ_{y0}	$10_{[0.0]}$	5.7	$8_{[0.0]}$	5.9	$10_{[0.0]}$	8.6
σ_{y1}	$18_{[0.2]}$	19.5	$13_{[0.005]}$	16.3	$18_{[0.005]}$	18.4
σ_{y2}	$20_{[0.4]}$	17.3	$20_{[0.01]}$	18.2	$20_{[0.01]}$	19.1
σ_{y3}	$25_{[0.6]}$	20.9	$38_{[0.1]}$	49.8	$25_{[0.1]}$	51.6
σ_{y4}	$36_{[0.8]}$	59.8	$40_{[0.3]}$	36.2	$36_{[0.3]}$	35.1
σ_{y5}	$45_{[1.0]}$	64.8	$45_{[0.6]}$	42.1	$45_{[0.6]}$	48.6
	$f^1 = 82.89$		$f^2 = 69.06$		$f^3 = 69.01$	
$Iterations : 220 \mid Time/iteration : 34\ sec. \mid \sum time : 2h : 05min.$						

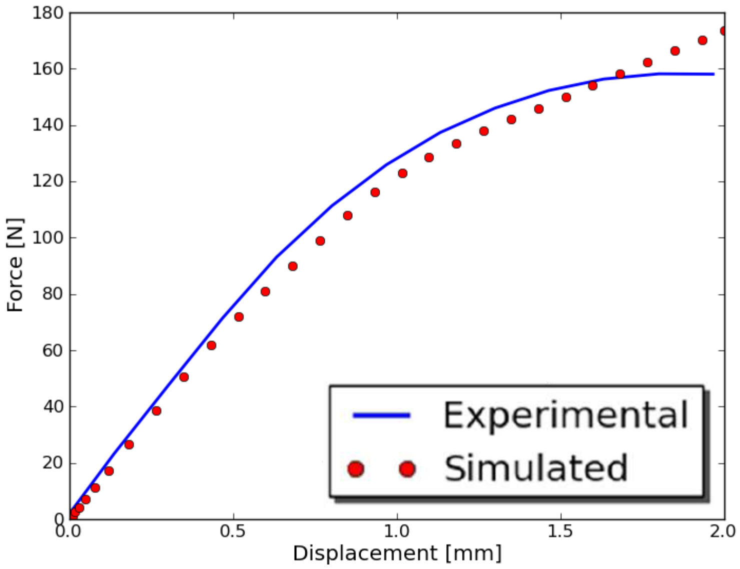


Figure 9.3: Global force vs. displacement with the Neo-Hooke material model, run 3

9.2.2 Arruda-Boyce form

Table 9.3: Start/End values from simulations of the Arruda-Boyce model

Arruda-Boyce model

Mat. par.	**Start**^{1}	**End**^{1}	**Start**^{2}	**End**^{2}	**Start**^{3}	**End**^{3}
μ	175	222	250	87.8	250	220.5
λ_m	5	6.22	3	1.1	3	5
D	0.015	0.006	0.01	0.005	0.01	0.005
σ_{y0}	$8_{[0.0]}$	4.6	$5_{[0.0]}$	8.3	$5_{[0.0]}$	5.5
σ_{y1}	$13_{[0.005]}$	12.8	$11_{[0.005]}$	13.8	$11_{[0.005]}$	13.0
σ_{y2}	$20_{[0.01]}$	18.5	$15_{[0.01]}$	19.9	$15_{[0.01]}$	20.0
σ_{y3}	$30_{[0.1]}$	37.9	$40_{[0.1]}$	44.9	$40_{[0.1]}$	38.0
σ_{y4}	$36_{[0.6]}$	41.7	$45_{[0.3]}$	48.2	$45_{[0.3]}$	41.6
σ_{y5}	$42_{[1.0]}$	45.9	$50_{[0.6]}$	49.0	$50_{[0.6]}$	50.5
	$f^1 = 70.5$		$f^2 = 69.25$		$f^3 = 69.25$	

$Iterations : 90 \mid Time/iteration : 25\ sec. \mid \sum time : 0h : 38min.$

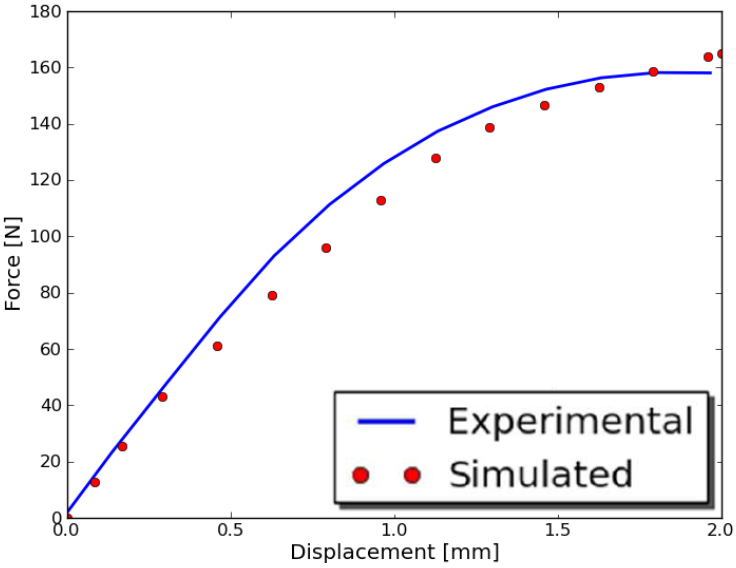


Figure 9.4: Global force vs. displacement with the Arruda-Boyce material model, run 3

9.2.3 Mooney-Rivlin form

Table 9.4: Material parameters used in the Mooney-Rivlin model

Mooney-Rivlin model

Mat. par.	**Start^1**	**End^1**	**Start^2**	**End^2**	**Start^3**	**End^3**
C_{10}	300	217.9	150	298	300	292.7
C_{01}	100	103.3	125	96.1	100	96.7
D_1	0.005	0.0057	0.01	0.0051	0.005	0.0052
σ_{y0}	$8_{[0.0]}$	8.2	$5_{[0.0]}$	2.1	$4_{[0.0]}$	1.1
σ_{y1}	$10_{[0.01]}$	14.9	$11_{[0.005]}$	10	$10_{[0.005]}$	11.5
σ_{y2}	$30_{[0.1]}$	49.8	$15_{[0.01]}$	16.7	$15_{[0.01]}$	15.6
σ_{y3}	$50_{[0.3]}$	36.6	$40_{[0.1]}$	46.5	$50_{[0.1]}$	52.1
σ_{y4}	$40_{[0.6]}$	47.1	$45_{[0.3]}$	41.4	$40_{[0.3]}$	45.5
σ_{y5}	$45_{[1.0]}$	43.5	$50_{[0.6]}$	47.4	$45_{[0.6]}$	44.4
	$f^1 = 69.82$		$f^2 = 68.99$		$f^3 = 68.82$	

$Iterations : 170 \mid Time/iteration : 30\ sec. \mid \sum time : 01h : 25min.$

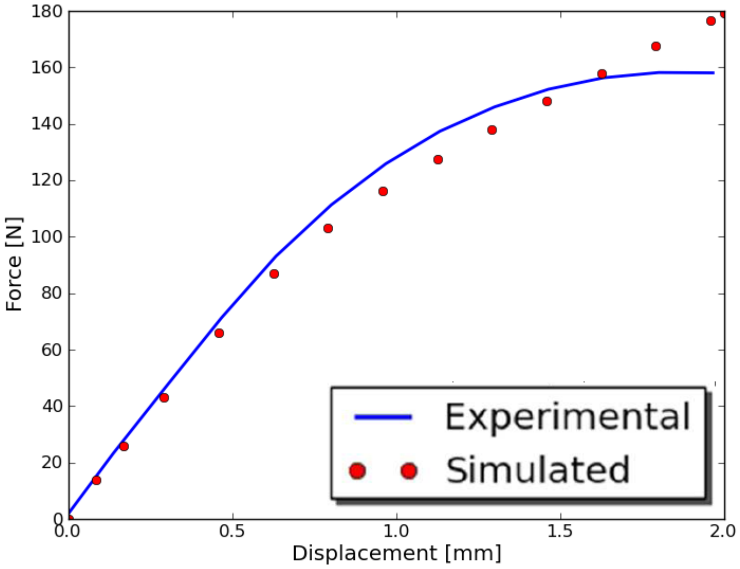


Figure 9.5: Global force vs. displacement with the Mooney-Rivlin material model, run 3

9.3 PolyUMod

9.3.1 Bergström-Boyce

Table 9.5: Start/End values from simulations of the Bergström-Boyce model

Bergström-Boyce model

Mat. par.	**Start**^{1}	**End**^{1}	**Start**^{2}	**End**^{2}	**Start**^{3}	**End**^{3}
μ	1	2.6	150	129	6.62	3.52
λ_L	4	3.98	4	3.38	4.72	5.55
κ	100	81.5	2500	2287	100	80.4
s	3	3.23	3	4.87	19.85	39.4
ξ	0.05	0.052	0.05	0.049	0.15	0.22
C	-0.5	-0.52	-0.5	-0.46	-0.47	-0.457
τ_{Base}	4	3.55	4	3.79	24.36	29.2
m	5	5.44	5	3.96	23.5	24.5
$\hat{\tau}_{cut}$	0.01	-0.002	0.01	0.011	0.01	0.0017
	$f^1 = 139.17$		$f^2 = 130$		$f^3 = 127.4$	

$Iterations : 250 \mid Time/iteration : 195\ sec. \mid \sum time : 13h : 32min.$

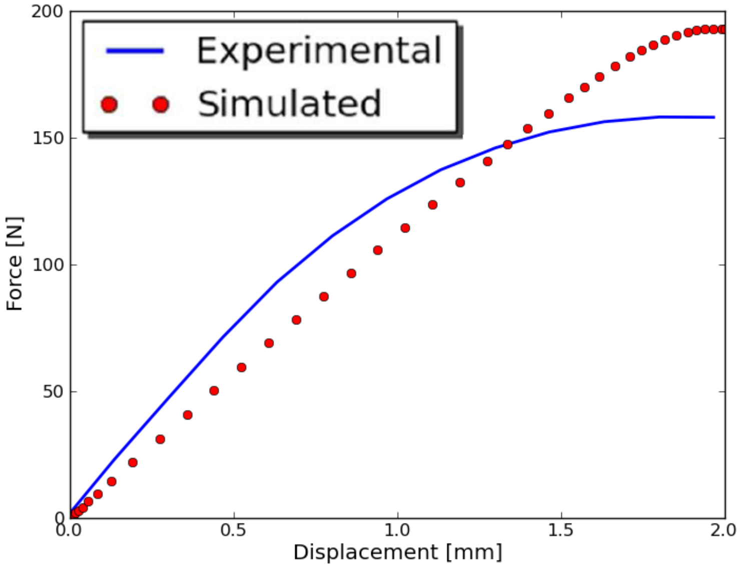


Figure 9.6: Global force vs. displacement with the Bergström-Boyce material model, run 3

9.3.2 Hybrid

Table 9.6: Start/End values from simulations of the Hybrid model

Hybrid model

Mat. par.	**Start^1**	**End^1**	**Start^2**	**End^2**	**Start^3**	**End^3**
E	500	449	2000	1012	728	569.3
ν	0.46	0.39	0.4	0.25	0.21	0.26
μ_A	13.32	18.1	20	28.8	22.5	26.3
λ_L	2.25	1.58	3	3.8	3.15	3.9
q	0.01	0.11	0.03	0.04	0.055	0.04
κ	5000	7459	4000	4882	6000	5781
s_{Bi}	10	4	30	10.3	36	37.1
s_{Bf}	20.6	88.7	8	5.4	21.5	18.6
α_B	223.2	228.2	17	18.4	20	18.1
τ^B_{base}	26.6	0	38	34.8	35	33.2
m_B	6.28	19.7	12	12.7	4	2.6
$\hat{p}$	200	92	300	252.3	300	280.5
τ^p_{base}	6.6	26	20	22.4	15	18.0
m^p	5	3.8	9	9.3	3.38	4.8
	$f^1 = 104$		$f^2 = 77$		$f^3 = 68.9$	

$Iterations : 450 \mid Time/iteration : 150\ sec. \mid \sum time : 18h : 45min.$

$Time/iteration\ 3D : 900\ sec. \mid f^3_{3D} = 68.99$

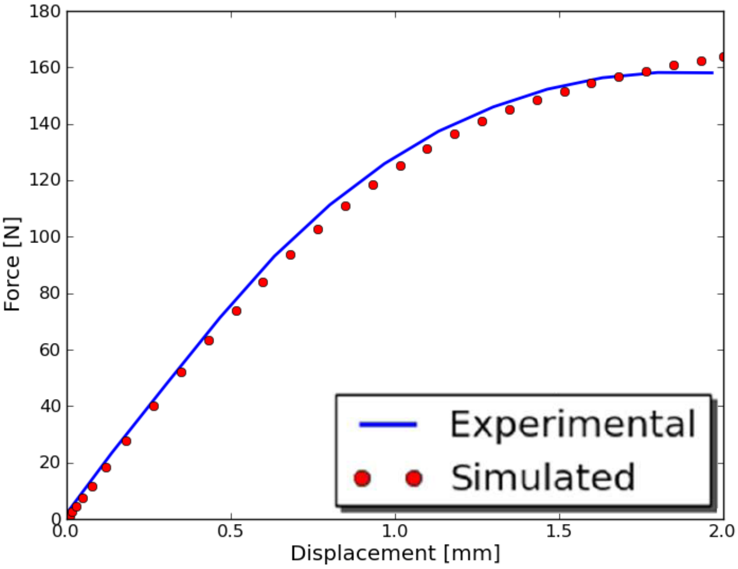


Figure 9.7: Global force vs. displacement with the Hybrid material model, run 3

9.3.3 Three network

Table 9.7: Material parameters used in the three network model

Three network model						
Mat. par.	**Start^1**	**End^1**	**Start^2**	**End^2**	**Start^3**	**End^3**
μ_A	238.9	227.9	192	185.2	100	75.9
$\widehat{\theta}$	999	999.7	999	991	800	843.4
λ_L	2.28	2.17	3.1	3.0	7.0	7.3
κ	2000	1946	2000	2003	2500	2575
$\widehat{\tau}_A$	6.05	5.58	7.3	6.7	12.0	11.5
a	1.4E-4	7.4E-3	1.2E-4	5.2E-3	0.007	0.0064
m_A	6.9	2.91	9.7	3.1	3	1.41
n	0.01	0.009	0.01	0.008	0.001	0.0018
μ_{Bi}	76.5	92.5	131.9	133.7	150	145.8
μ_{Bf}	69.84	69.82	48.3	48.8	60	61.4
β	12.4	12.0	11.1	10	10	9.6
$\widehat{\tau}_B$	22.6	21.3	25.4	25.5	25	23.2
m_B	11.1	10.8	9.7	10.1	15	14.2
μ_C	7.2	9.6	8.3	9.4	12	13.3
q	0.2	0.12	0.2	0.1	0.1	0.244
α	E-7	E-4	E-7	4E-4	2E-5	2.8E-4
θ_0	293	293	293	293	293	293
	$f^1 = 129.6$		$f^2 = 129.0$		$f^3 = 126.9$	
$Iterations : 200 \mid Time/iteration : 210\ sec. \mid \sum time : 11h : 35min.$						

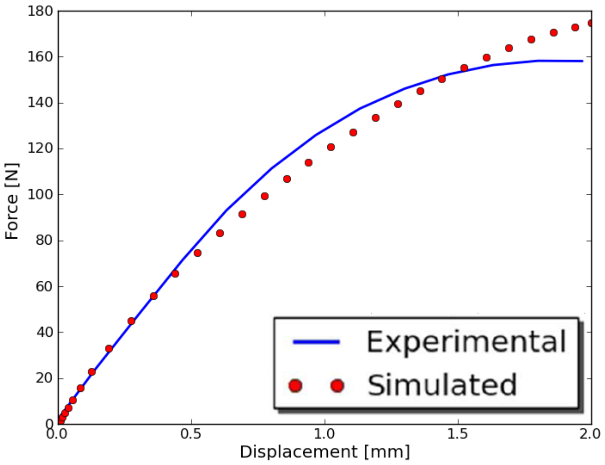
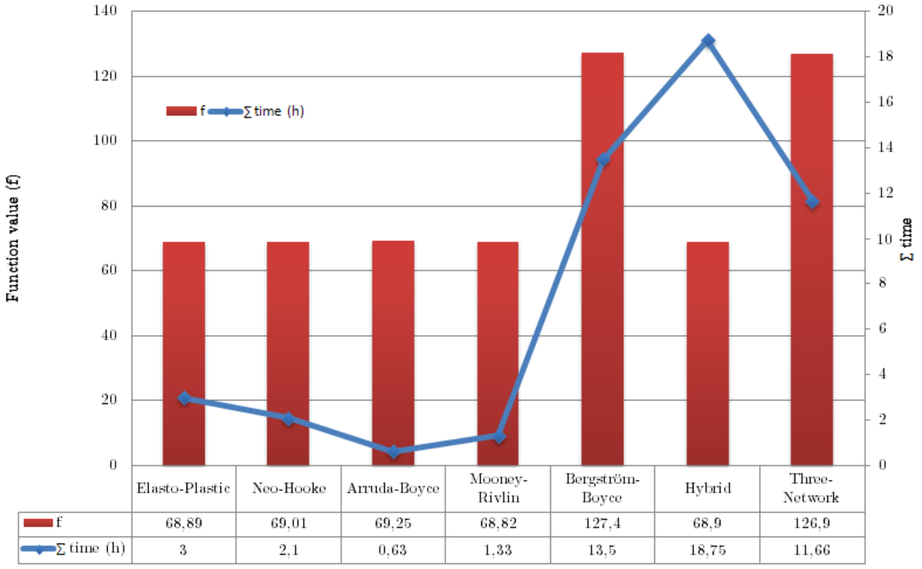


Figure 9.8: Global force vs. displacement with the Three-Network material model, run 3

9.4 Summary - Results

Figure 9.9 shows a summary from all the simulations, containing the final function values and the simulation times for the runs with the lowest function value of each material model.



	Elasto-Plastic	Neo-Hooke	Arruda-Boyce	Mooney-Rivlin	Bergström-Boyce	Hybrid	Three-Network
f	68,89	69,01	69,25	68,82	127,4	68,9	126,9
∑ time (h)	3	2,1	0,63	1,33	13,5	18,75	11,66

Figure 9.9: Summarize of time and function values for the different material models

Chapter 10

Concluding remarks

This final chapter summarizes the various problem and difficulties that was encountered during this thesis. Discussions regarding the simulations and results are given. Finally, suggestions on further work that could be performed are presented.

10.1 Discussion

As previously mentioned, only simulations on HDPE 50mm/min have been performed. All the simulations have been performed on the same computer, so that the simulation time is comparable with every material model. As can be seen in Chapter 9, the final function values does not differ much between the various material models. The material models that have the most problem simulating the polymer were the Bergström-Boyce and Three-Network. This is a rather mysterious behavior since these models contain many parameters that should be able to be calibrated to get even better results than Abaqus built in models. Two possible explanations for this are that either are not enough simulations performed or there could be a numerical error in the Python scipt.

The main difference between the simulations is the simulation time for the optimization to converge. Turning the interest towards Fig. 9.9 one can see that the Bergström models are taking significantly more computer time than Abaqus own built in models. One possible reason for this is because they are far more advanced than Abaqus built in models. Therefore they need more time to complete the numerical calculations due to a large amount of parameters. The models showing best results are the Arruda-Boyce and Mooney-Rivlin model since they shows low function values with short simulation times.

It can be seen from the results in Appendix B that every material model have trouble capturing the large deformations close to the nodge, specifically point B1 in the y-direction. On the other hand, the contractions in the x-direction for every point are better captured in the simulations. One possible explanation for this is that the deformation in the x-direction is much smaller than in the y-direction.

To treat the body as a two-dimensional geometry was successful. Table 9.6 shows that the function value is close to the three-dimensional value, and that the three-dimensional geometry needs more simulation time for one iteration.

Once the script for the parameter identification was established it was easy to use different material models. The big difficulty was to assign good initial values with corresponding constraints. This have been an important discovery since the final parameters are greatly dependent on this which can be seen from the results. Therefore it is important to try out a number of different sets of starting values of the material parameters. As can be seen from the figures in Appendix B.5 and Appendix B.7 for the Bergström-Boyce and Three-Network models, these results do not seem to be optimal solutions since all the simulated curves indicates that the material is not stiff enough. Therefore, further simulations with different starting values on the parameters describing the stiffness should be performed. The material models that were most difficult to calibrate were the ones from the PolyUMod library. This because they are very advanced and to know each parameters property requires extensive knowledge in both mathematics and constitutive modeling. The PolyUMod calibration tool was used for these advanced models to make sure that constraints were given reasonable values within physical limitations.

Once the parameters are assigned with reasonable values, the optimization procedure with the Nelder-Mead simplex algorithm seems to be very robust and a solution to the inverse problem was always found.

One important discussion is the possibility of performing another type of experiment. The experiment used in this thesis was a simple monotone loading of the specimen. With this type of experiment it is impossible to evaluate how the material has hardened during the loading. Therefore an experiment with cyclic loading which initially loads, unloads and loads again could be made to know how the material hardens during the loading. This is an important property to know since many types of packages are subjected to this kind of load, i.e. the opening procedure of a cap. In Appendix C, (C.1) and (C.2) shows two correlation matrices from a monotone and a cyclic loading respectively with the Mooney-Rivlin model. It can be seen that the correlation matrix from the cyclic load triggers more parameters and lowers their linear dependency which could mean that this type of test can calibrate both Abaqus built in models and especially the models from the PolyUMod material library better than the monotone load.

10.2 Future work

Since there is a time limit of this work there are areas that could not be covered. During the work new questions came up and ideas developed which could be of interest to examine.

- Verify the implementation of the Python script.
- Perform new experiments containing cyclic loading. This to be able to

calibrate the material models better, which in the end can give better simulations and more accurately capture the behavior of the material.

- Try out different geometries of the test-specimen for the experiments. This could as well as a cyclic loading help calibrating more parameters in the material models. This can be examined without a physical experiment in Abaqus by deriving the correlation matrix for different geometries.
- Run simulations that contain both test speeds to be able to better calibrate the parameters describing the rate-dependent response.
- Due to the lack of time, simulations on all test speeds and materials was not made during this work. This could also be done in a future work.

Appendix A

ARAMIS results

The following sections presents the results from the ARAMIS tensile test for HDPE and LDPE respectively. Each section starts with the global force versus displacement registered from the clamps. After this, the displacement fields for the x and y-directions (relative due to symmetry) for the five points A1,A2,B1,B2,C1 in the second quadrant are presented. The y-displacement for point C1 is always zero due to the relative value and is therefore not presented.

A.1 HDPE

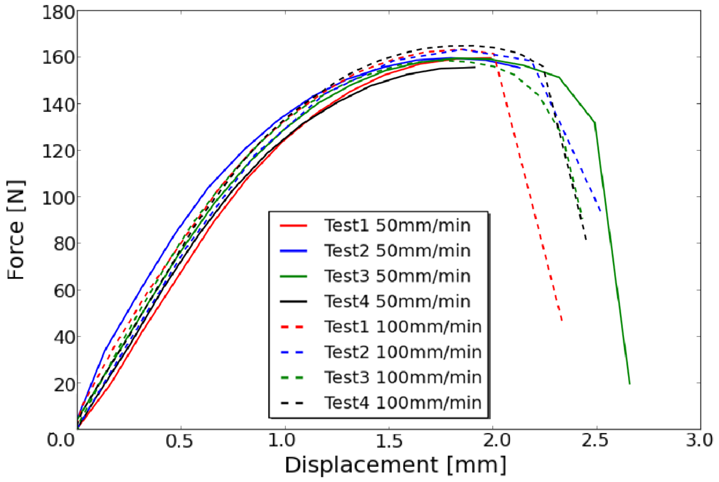
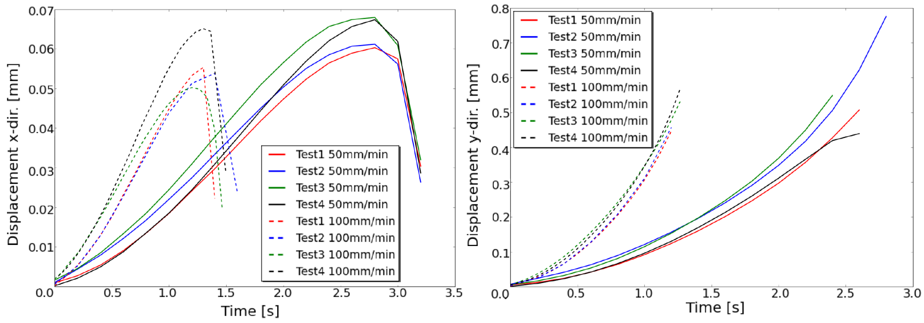
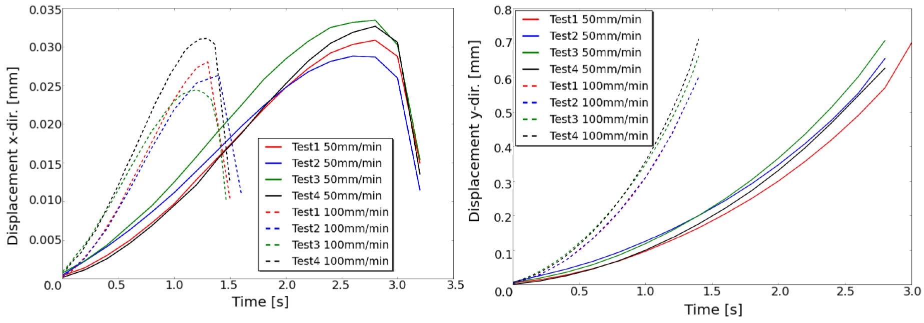


Figure A.1: Global force vs. displacement



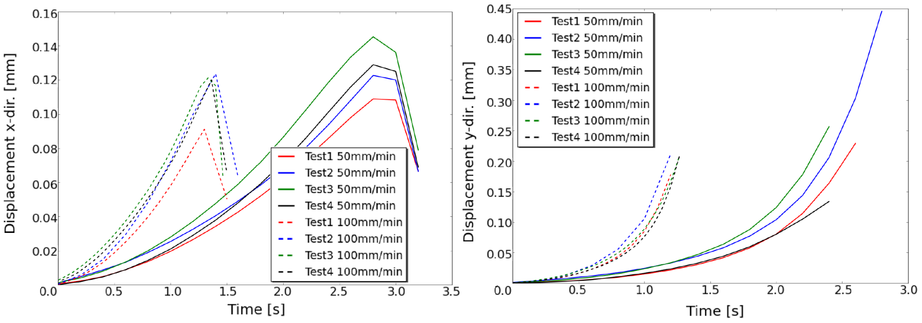
(a) x-direction (b) y-direction

Figure A.2: Displacement vs. time A1



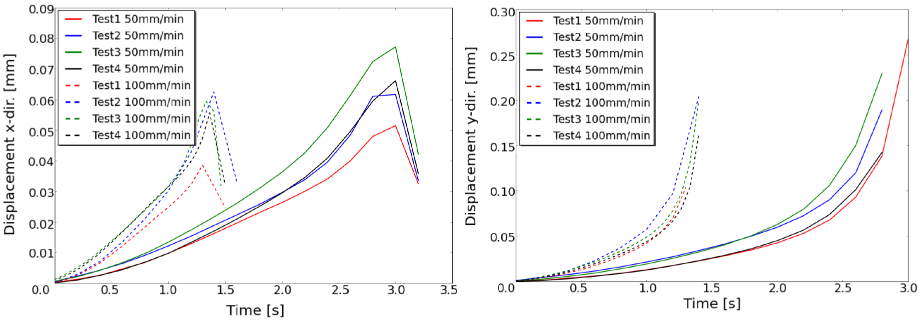
(a) x-direction (b) y-direction

Figure A.3: Displacement vs. time A2



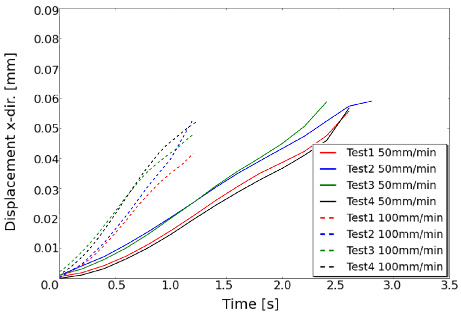
(a) x-direction (b) y-direction

Figure A.4: Displacement vs. time B1



(a) x-direction (b) y-direction

Figure A.5: Displacement vs. time B2



(a) x-direction

Figure A.6: Displacement vs. time C1

A.2 LDPE

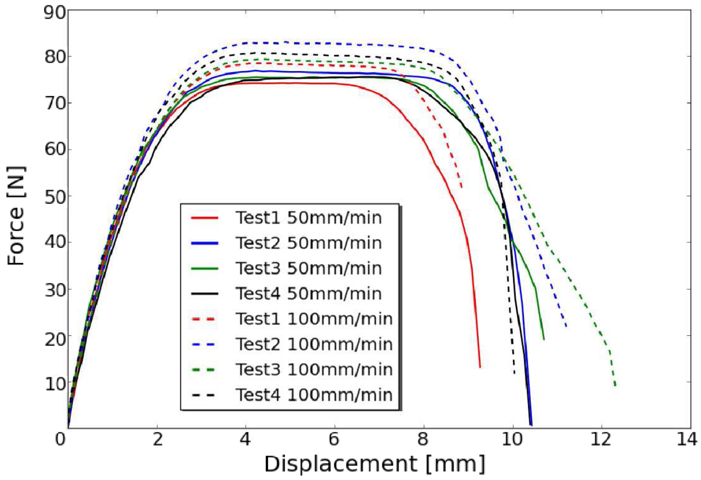
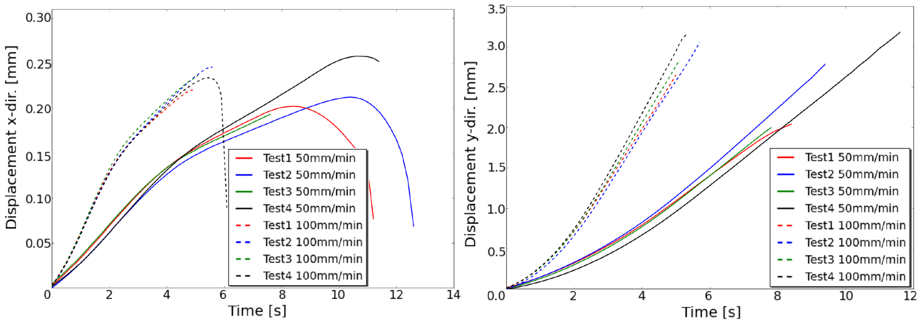


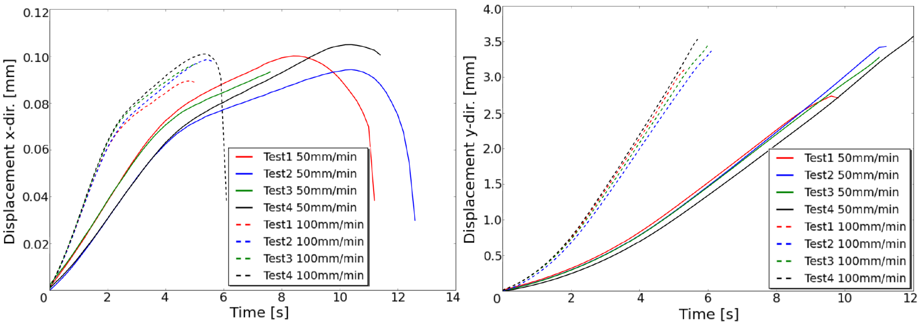
Figure A.7: Global force vs. displacement



(a) x-direction

(b) y-direction

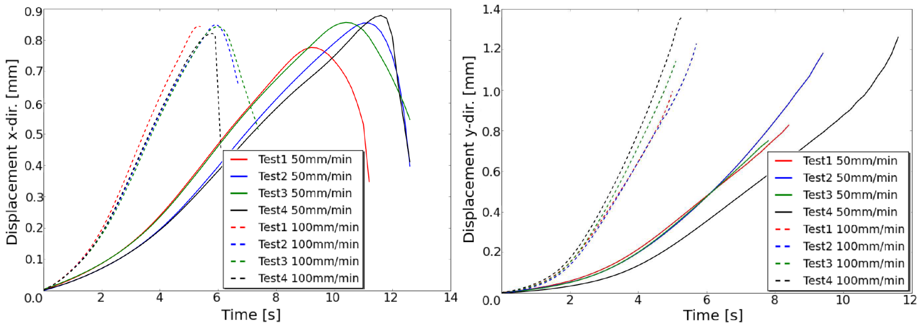
Figure A.8: Displacement vs. time A1



(a) x-direction

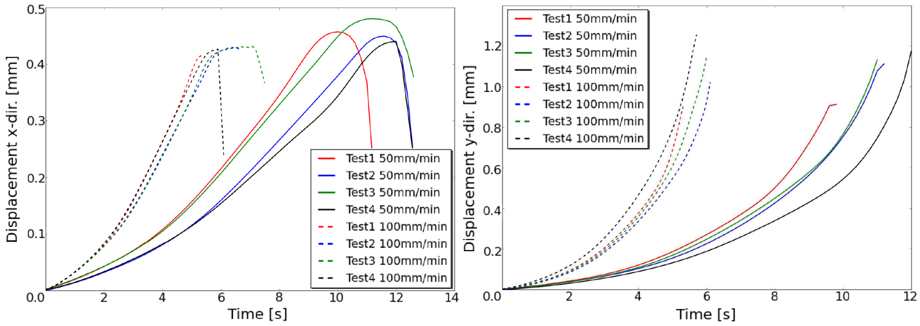
(b) y-direction

Figure A.9: Displacement vs. time A2



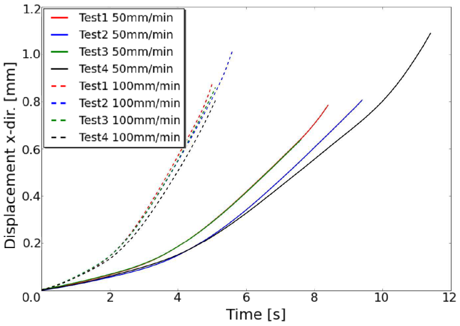
(a) x-direction (b) y-direction

Figure A.10: Displacement vs. time B1



(a) x-direction (b) y-direction

Figure A.11: Displacement vs. time B2



(a) x-direction

Figure A.12: Displacement vs. time C1

Appendix B

Results from simulations HDPE 50mm/min

The following sections presents the results from the the simulations in Abaqus with all the material models. The figures shows the results from the simulations with the lowest function value.

B.1 Elasto-plasticity

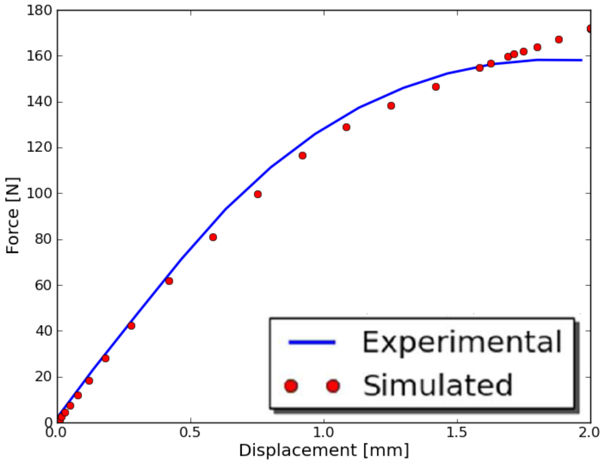
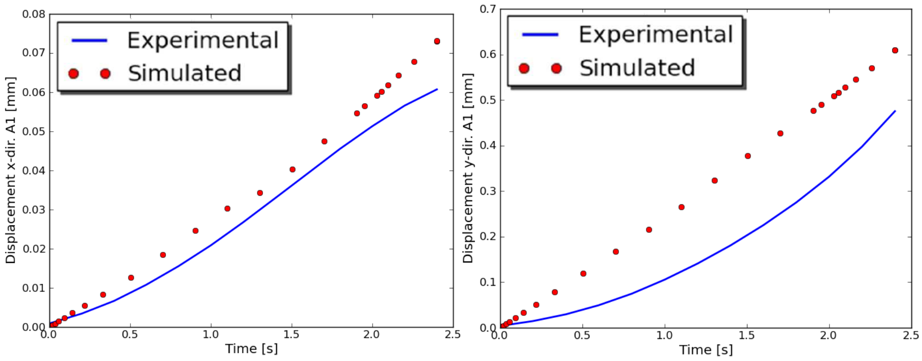
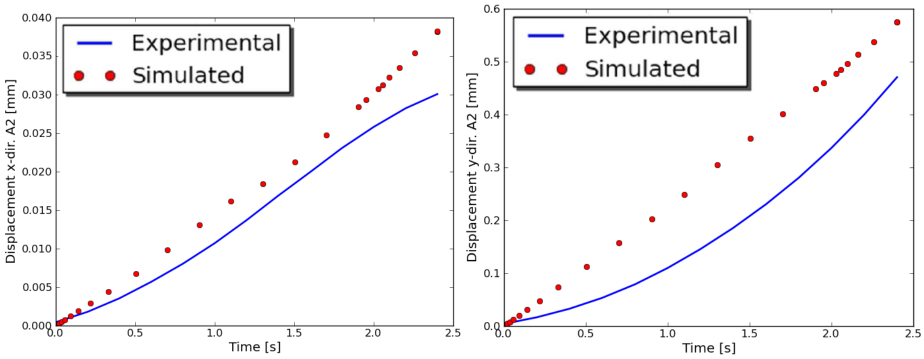


Figure B.1: Global force vs. displacement



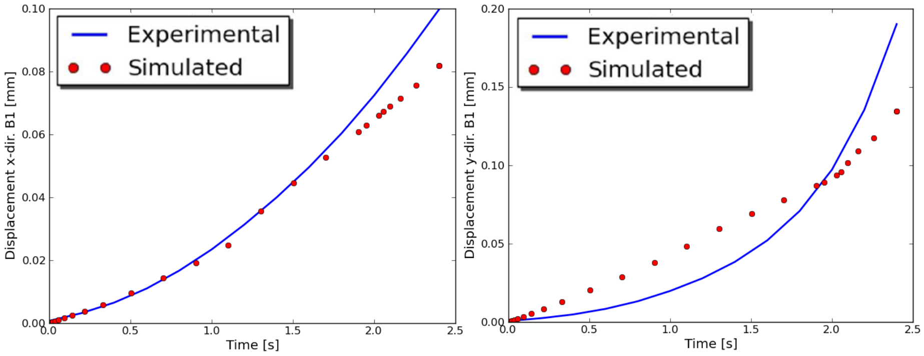
(a) x-direction (b) y-direction

Figure B.2: Displacement vs. time A1



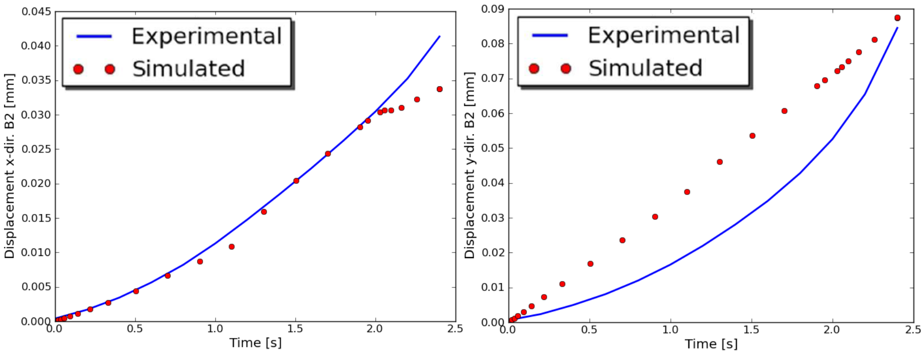
(a) x-direction (b) y-direction

Figure B.3: Displacement vs. time A2



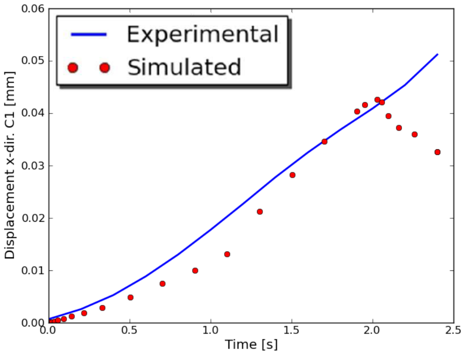
(a) x-direction (b) y-direction

Figure B.4: Displacement vs. time B1



(a) x-direction (b) y-direction

Figure B.5: Displacement vs. time B2



(a) x-direction

Figure B.6: Displacement vs. time C1

B.2 Neo-Hooke

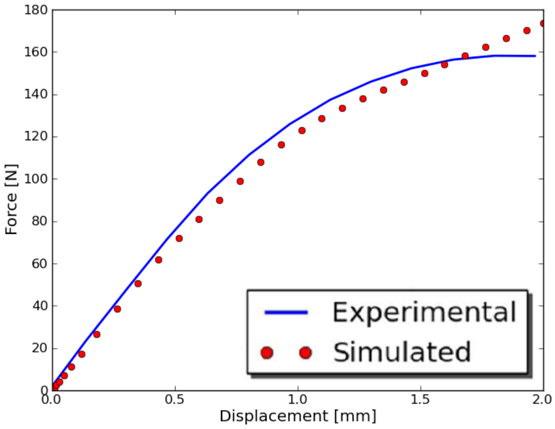
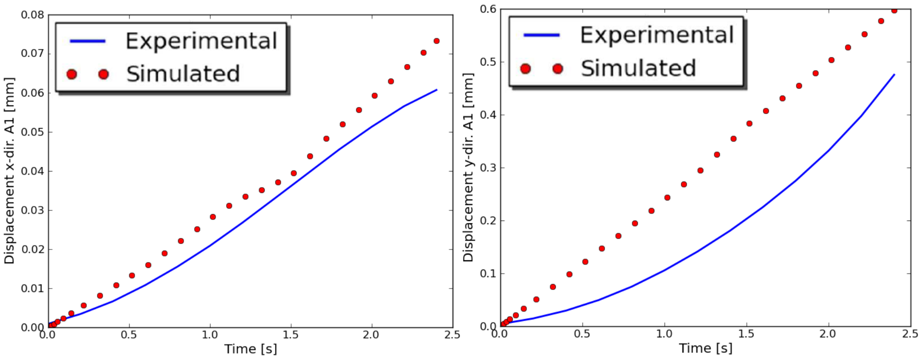
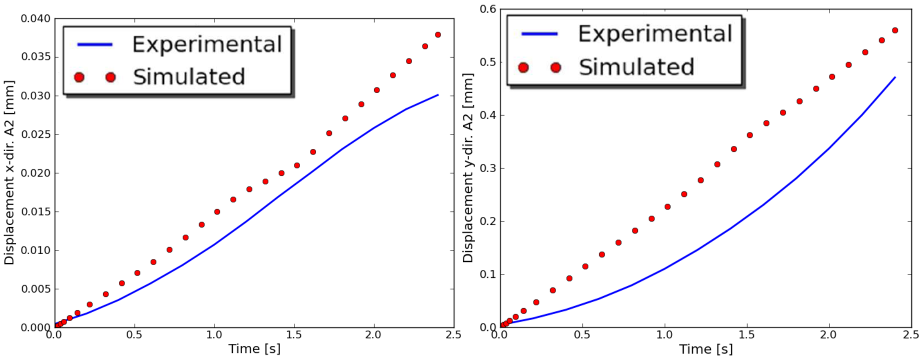


Figure B.7: Global force vs. displacement



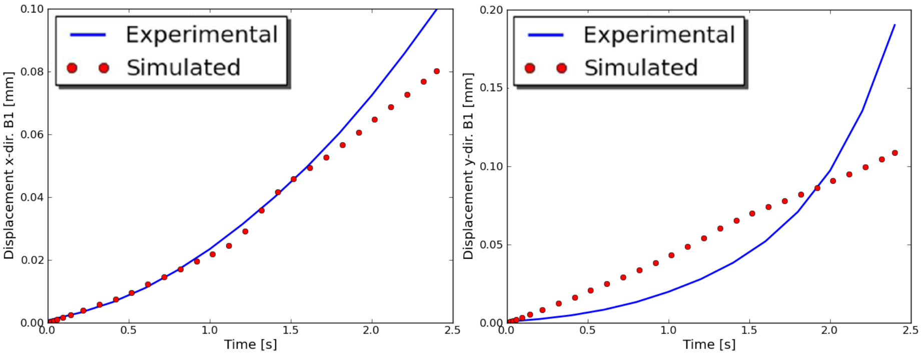
(a) x-direction (b) y-direction

Figure B.8: Displacement vs. time A1



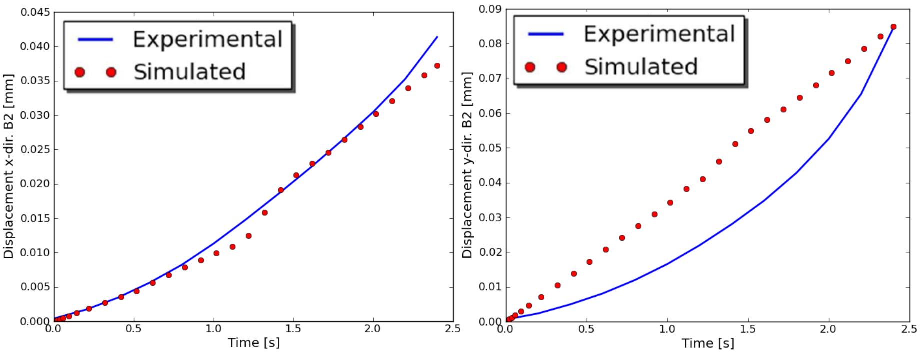
(a) x-direction (b) y-direction

Figure B.9: Displacement vs. time A2



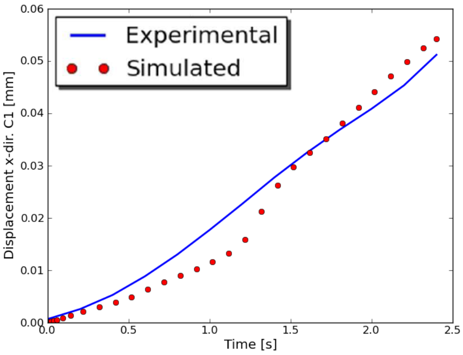
(a) x-direction (b) y-direction

Figure B.10: Displacement vs. time B1



(a) x-direction (b) y-direction

Figure B.11: Displacement vs. time B2



(a) x-direction

Figure B.12: Displacement vs. time C1

B.3 Arruda-Boyce

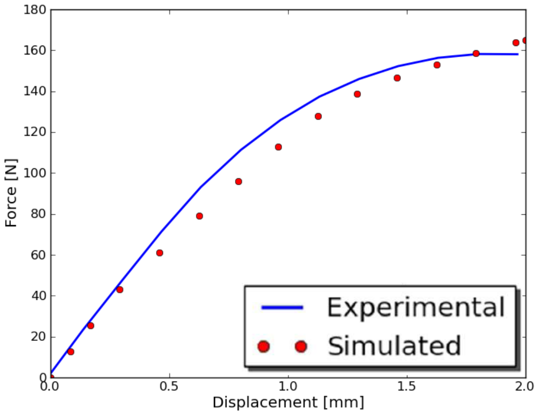
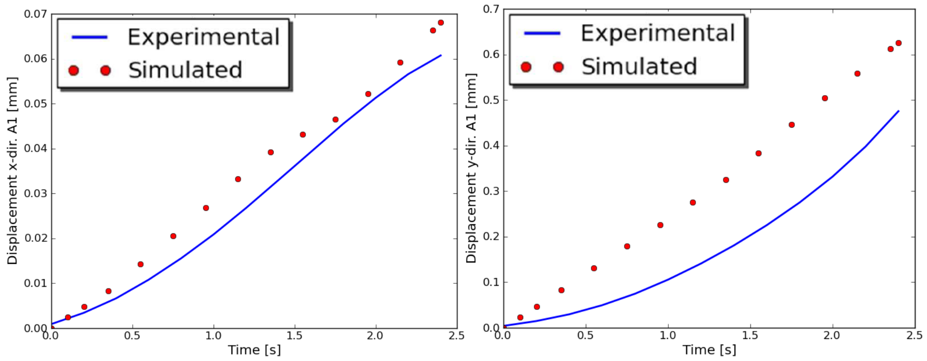


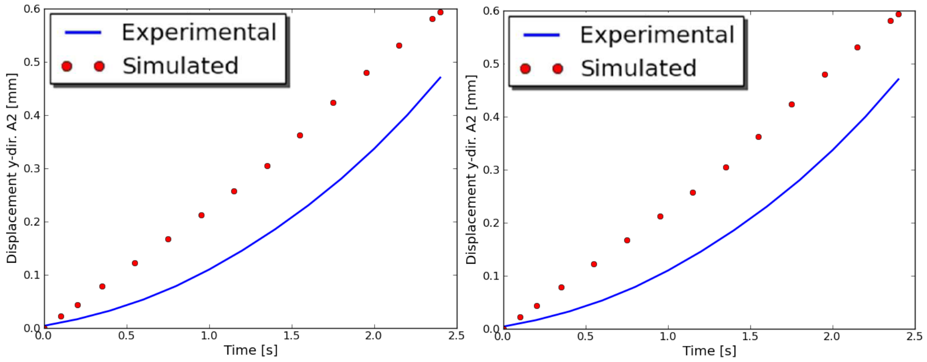
Figure B.13: Global force vs. displacement



(a) x-direction

(b) y-direction

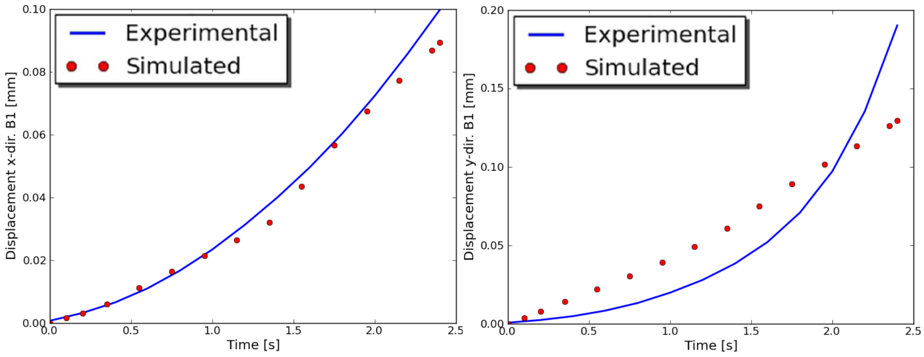
Figure B.14: Displacement vs. time A1



(a) x-direction

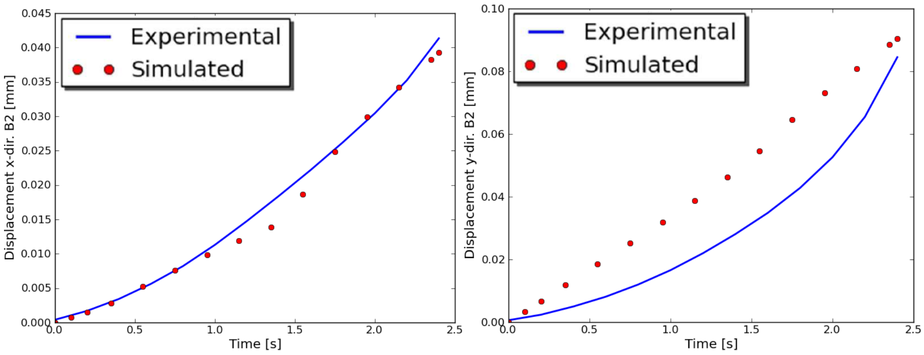
(b) y-direction

Figure B.15: Displacement vs. time A2



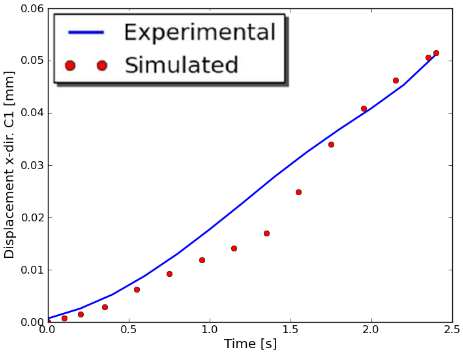
(a) x-direction (b) y-direction

Figure B.16: Displacement vs. time B1



(a) x-direction (b) y-direction

Figure B.17: Displacement vs. time B2



(a) x-direction

Figure B.18: Displacement vs. time C1

B.4 Mooney-Rivlin

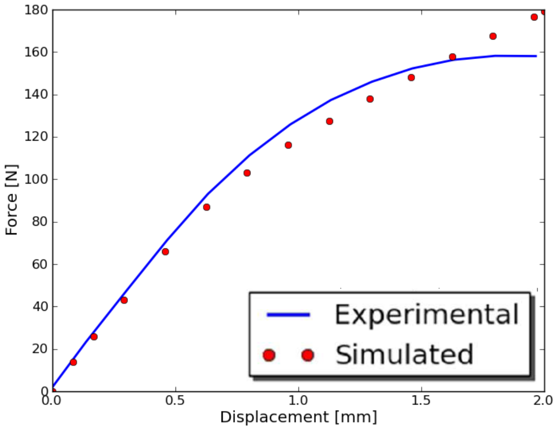
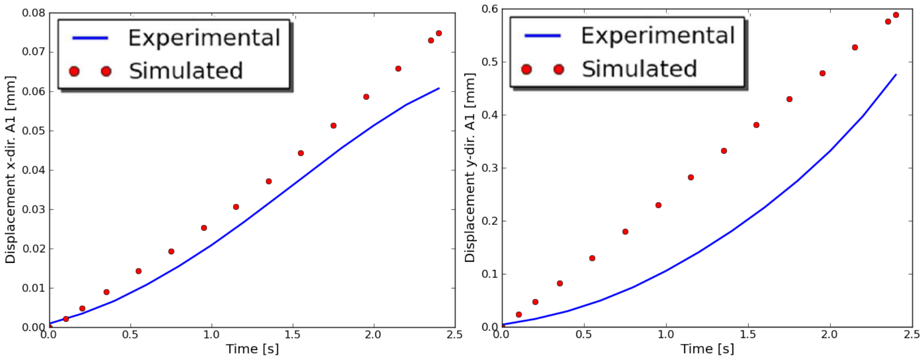
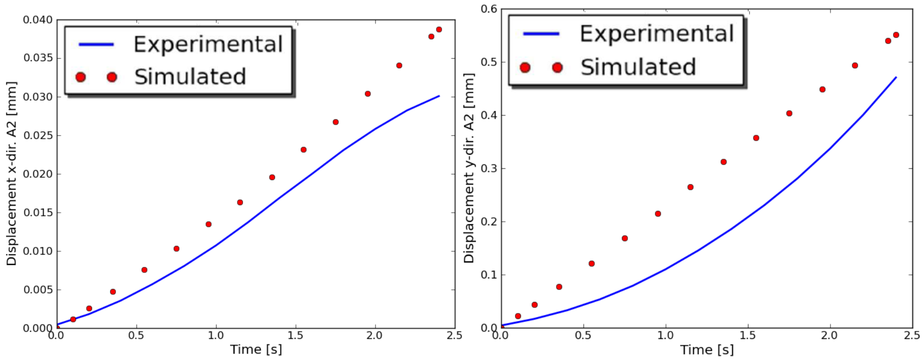


Figure B.19: Global force vs. displacement



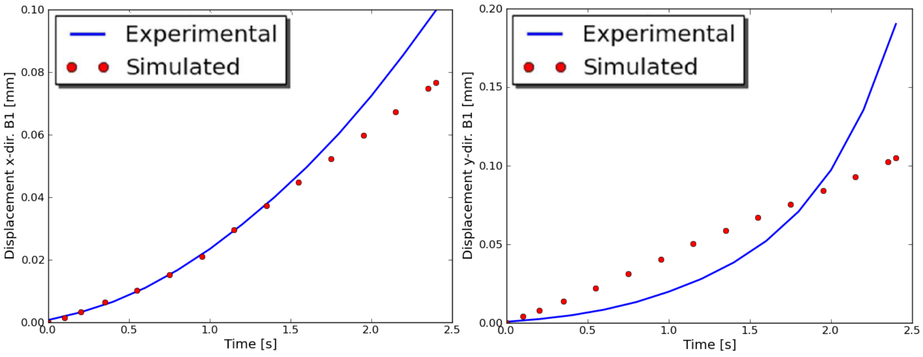
(a) x-direction (b) y-direction

Figure B.20: Displacement vs. time A1



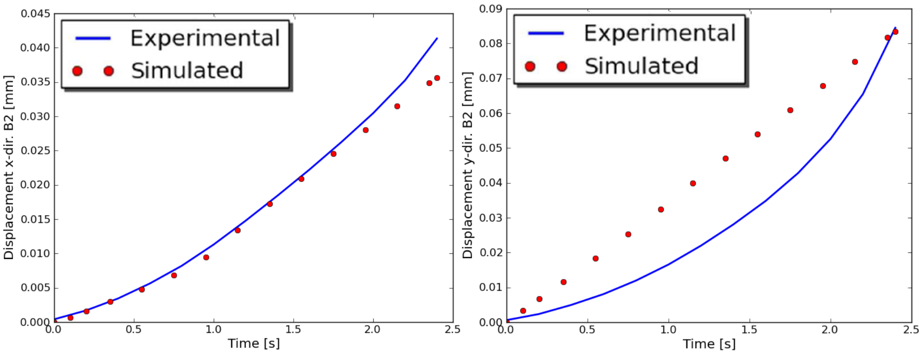
(a) x-direction (b) y-direction

Figure B.21: Displacement vs. time A2



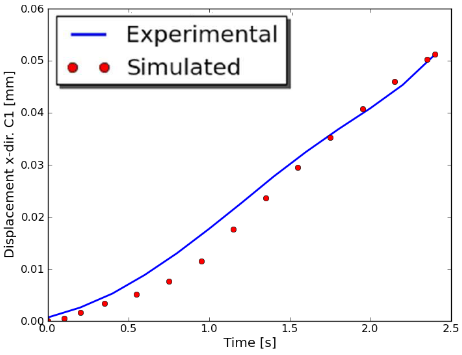
(a) x-direction (b) y-direction

Figure B.22: Displacement vs. time B1



(a) x-direction (b) y-direction

Figure B.23: Displacement vs. time B2



(a) x-direction

Figure B.24: Displacement vs. time C1

B.5 Bergström-Boyce

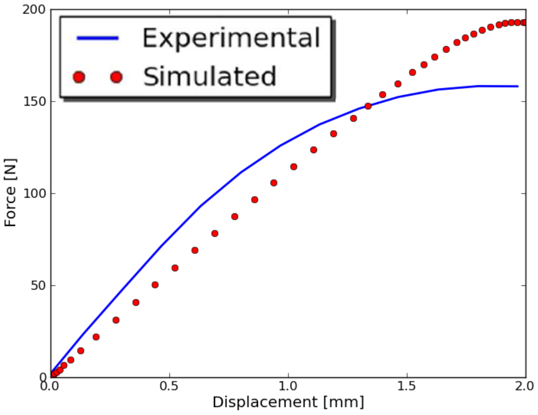
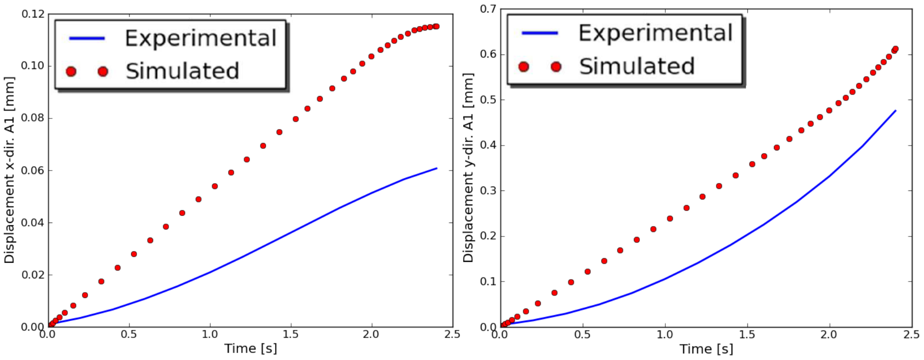
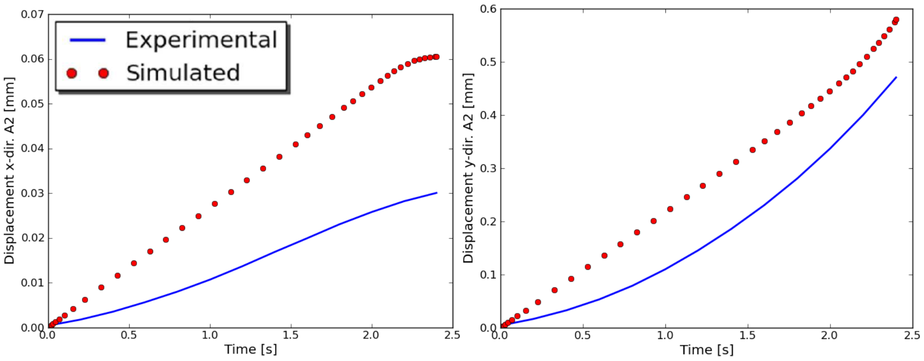


Figure B.25: Global force vs. displacement



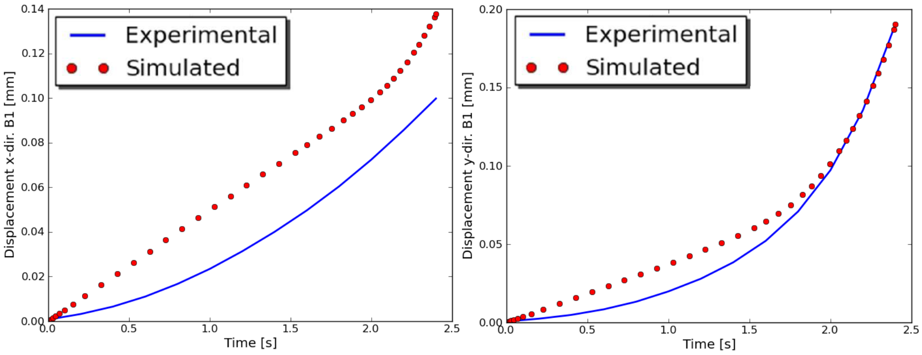
(a) x-direction (b) y-direction

Figure B.26: Displacement vs. time A1



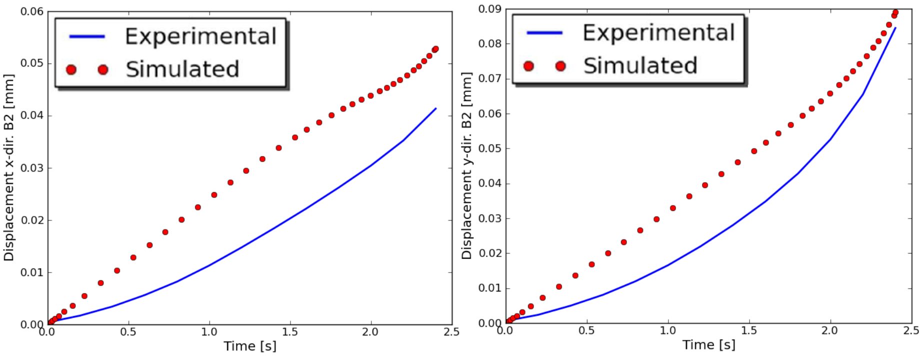
(a) x-direction (b) y-direction

Figure B.27: Displacement vs. time A2



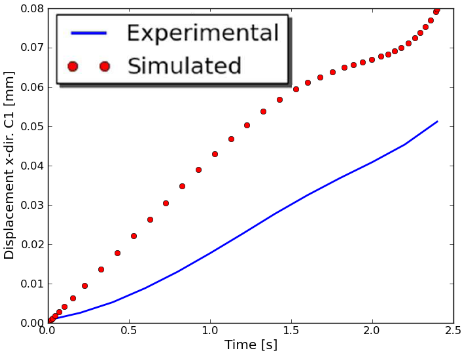
(a) x-direction (b) y-direction

Figure B.28: Displacement vs. time B1



(a) x-direction (b) y-direction

Figure B.29: Displacement vs. time B2



(a) x-direction

Figure B.30: Displacement vs. time C1

B.6 Hybrid

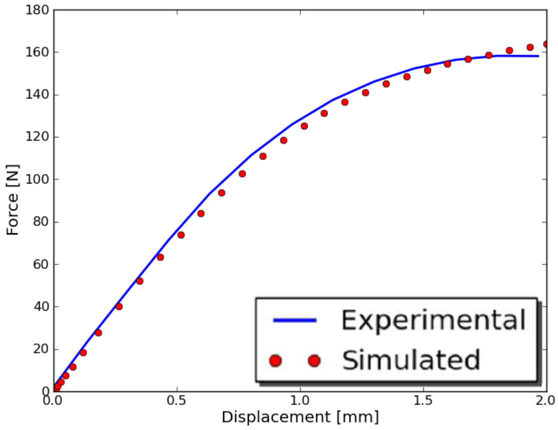
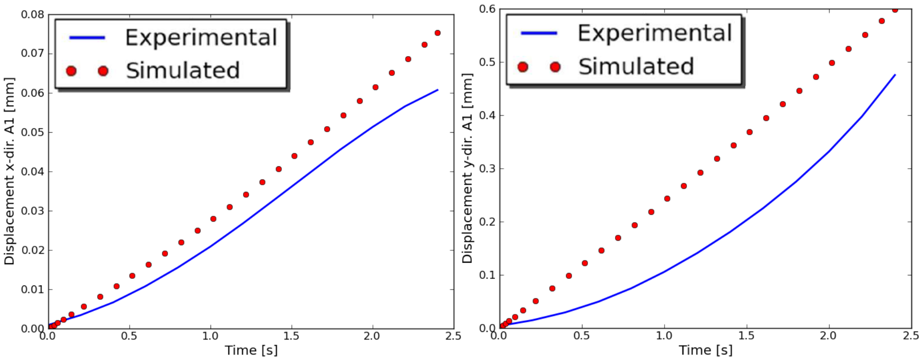
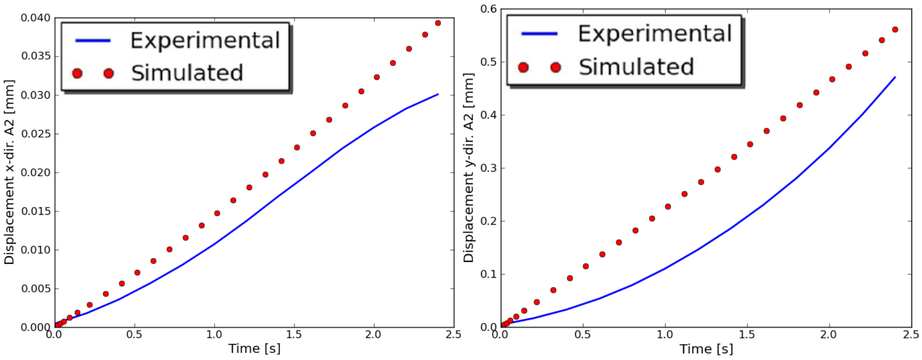


Figure B.31: Global force vs. displacement



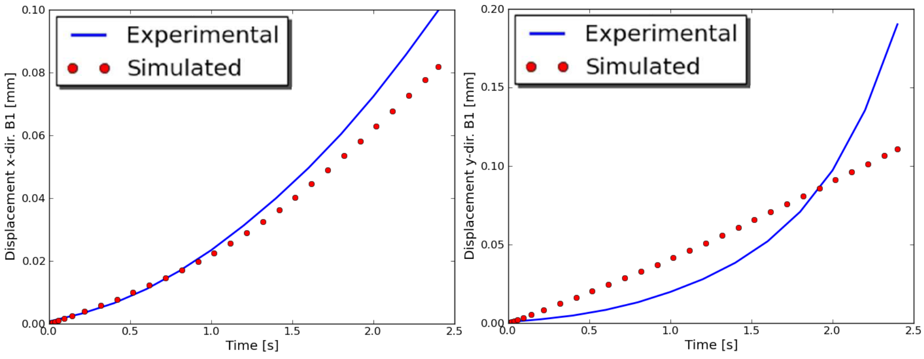
(a) x-direction (b) y-direction

Figure B.32: Displacement vs. time A1



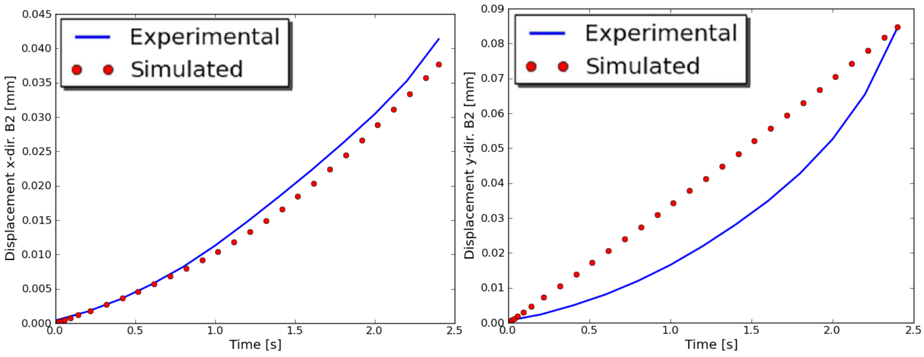
(a) x-direction (b) y-direction

Figure B.33: Displacement vs. time A2



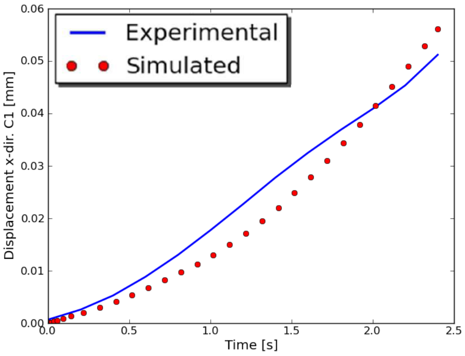
(a) x-direction (b) y-direction

Figure B.34: Displacement vs. time B1



(a) x-direction (b) y-direction

Figure B.35: Displacement vs. time B2



(a) x-direction

Figure B.36: Displacement vs. time C1

B.7 Three network

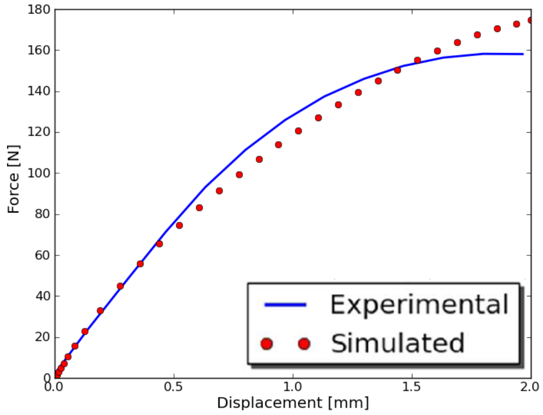
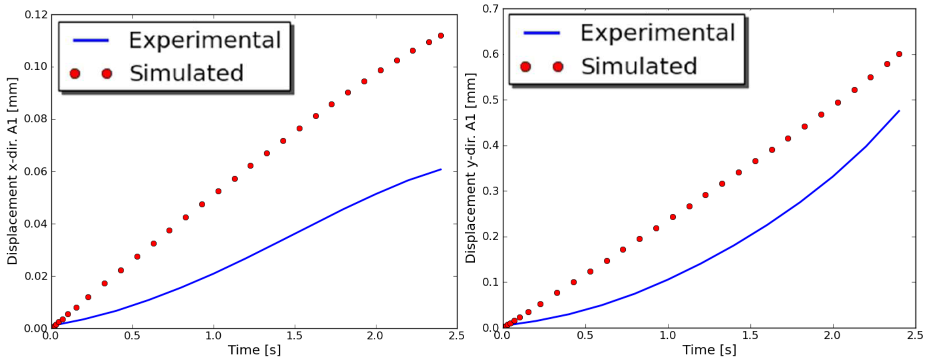


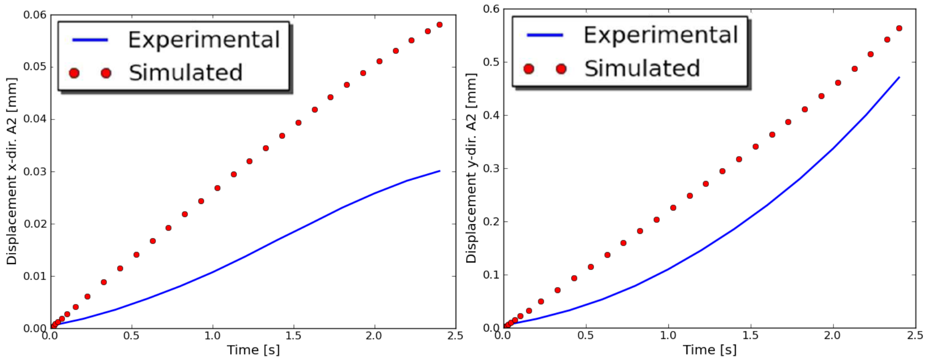
Figure B.37: Global force vs. displacement



(a) x-direction

(b) y-direction

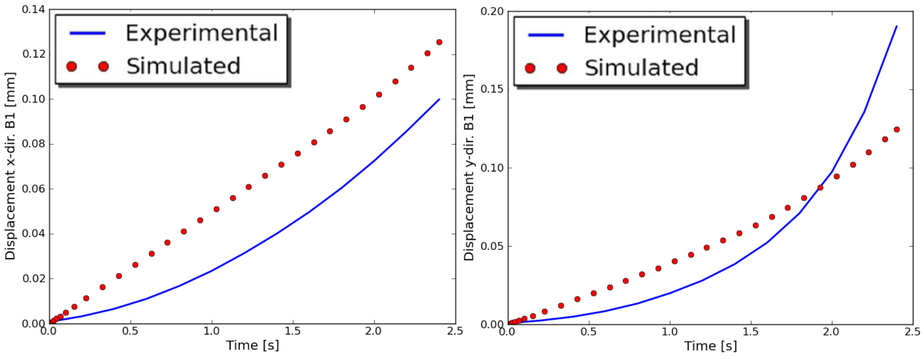
Figure B.38: Displacement vs. time A1



(a) x-direction

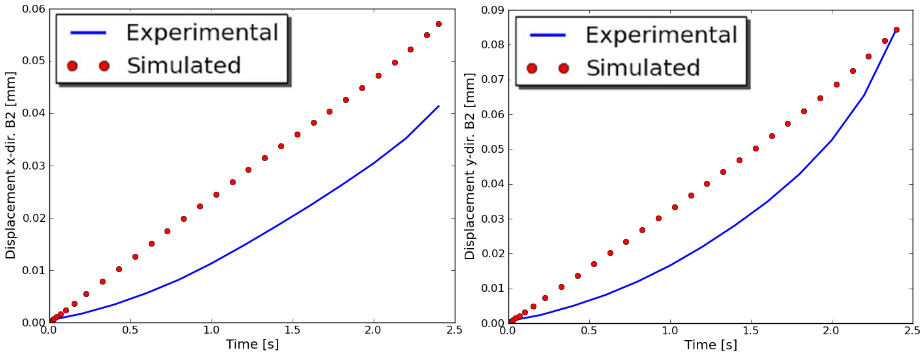
(b) y-direction

Figure B.39: Displacement vs. time A2



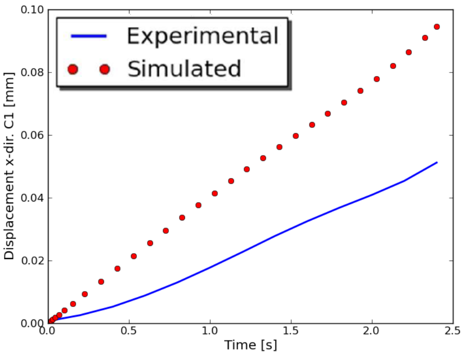
(a) x-direction (b) y-direction

Figure B.40: Displacement vs. time B1



(a) x-direction (b) y-direction

Figure B.41: Displacement vs. time B2



(a) x-direction

Figure B.42: Displacement vs. time C1

Appendix C

Correlation matrix for monotone versus cyclic loading

Equation (C.1) shows the correlation matrix for the experiment used in this thesis and (C.2) from a simulation on the same dog-bone specimen during a cyclic loading using the Mooney-Rivlin material model.

$$\cos\Phi_{ij} = \begin{bmatrix} 1 & -0.182 & 0.843 & 0.315 & 0.473 & 0.416 & -0.048 & -0.402 & NaN \\ -0.182 & 1 & -0.149 & -0.195 & -0.042 & -0.175 & 0.098 & 0.179 & NaN \\ 0.843 & -0.149 & 1 & 0.197 & 0.187 & 0.510 & 0.228 & -0.262 & NaN \\ 0.315 & -0.195 & 0.197 & 1 & 0.162 & 0.014 & -0.010 & -0.025 & NaN \\ 0.473 & -0.042 & 0.187 & 0.162 & 1 & -0.112 & -0.234 & -0.040 & NaN \\ 0.416 & -0.175 & 0.510 & 0.014 & -0.112 & 1 & -0.576 & -0.419 & NaN \\ -0.048 & 0.098 & 0.228 & -0.010 & -0.234 & -0.576 & 1 & 0.239 & NaN \\ -0.402 & 0.179 & -0.262 & -0.025 & -0.040 & -0.419 & 0.239 & 1 & NaN \\ NaN & NaN & NaN & NaN & NaN & NaN & NaN & NaN & NaN \end{bmatrix} \tag{C.1}$$

$$\cos\Phi_{ij} = \begin{bmatrix} 1 & -0.420 & 0.484 & 0.023 & 0.084 & -0.112 & -0.317 & -0.215 & -0.153 \\ -0.420 & 1 & -0.280 & -0.079 & 0.003 & 0.032 & 0.070 & 0.167 & 0.111 \\ 0.484 & -0.280 & 1 & 0.170 & 0.158 & 0.275 & 0.148 & -0.256 & -0.157 \\ 0.023 & -0.079 & 0.170 & 1 & 0.192 & 0.080 & 0.013 & -0.055 & -0.023 \\ 0.084 & 0.003 & 0.158 & 0.192 & 1 & 0.139 & -0.211 & -0.093 & -0.038 \\ -0.112 & 0.032 & 0.275 & 0.080 & 0.139 & 1 & -0.629 & -0.482 & -0.202 \\ -0.317 & 0.070 & 0.148 & 0.013 & -0.211 & -0.629 & 1 & 0.317 & 0.100 \\ -0.215 & 0.167 & -0.256 & -0.055 & -0.093 & -0.482 & 0.317 & 1 & 0.679 \\ -0.153 & 0.111 & -0.157 & -0.023 & -0.038 & -0.202 & 0.100 & 0.679 & 1 \end{bmatrix} \tag{C.2}$$

Bibliography

[1] *Abaqus documentation manual*, 6.10 edition, 2010.

[2] Jörgen Bergström. *PolyUMod - A library of User Materials for Abaqus*, 2009-02 edition, 2009.

[3] Jörgen S. Bergström. An augmented hybrid constitutive model for simulation of unloading and cyclig loading behavior of conventional and highly crosslinked uhmwpe, 2003.

[4] Lars-Christer Böiers. Lectures on optimization. Lund University, Sweden, 2009.

[5] Rolf Mahnken and Erwin Stein. *Parameter identification for finite deformation elasto-plasticity in principal directions.* 1996.

[6] Rolf Mahnken and Erwin Stein. *A unified approach for parameter identification of inelastic material models in the frame of the finite element method.* 1996.

[7] J.A. Nelderand and R. Mead. A simplex method for function minimization. *Comput*, J, 7:308313.

[8] Niels Saabye Ottosen and Hans Petersson. *Introduction to the FINITE ELEMENT METHOD.* Prentice Hall, University of Lund, Sweden, 1992.

[9] Niels Saabye Ottosen and Matti Ristinmaa. *The Mechanics of Constitutive Modeling.* ELSEVIER, Division of Solid Mechanics, Lund University, Sweden, 2005.

[10] Niels Saabye Ottosen and Matti Ristinmaa. *Introduction to Large Strain Plasticity.* Division of Solid Mechanics, Lund University, Sweden, 2008.

[11] J.C. Simo and T.J.R Hughes. *Computational inelasticity*, volume 7. Springer.