TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

DIPLOMARBEIT

Development and Validation of a Material Model with Artificial Neural Networks

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Diplom-Ingenieurs unter Leitung von

Ao. Univ. Prof. Dipl.-Ing. Dr.techn. Helmut Böhm

Nichtlineare Finite Elemente Methoden

E317

Institut für Leichtbau und Struktur-Biomechanik

eingereicht an der Technischen Universität Wien
Fakultät für Maschinenwesen und Betriebswissenschaften

von

Claudia Gaßner

e0725363

Angerweg 17 A-4360 Grein

Wien, am 6.3.2014

Master's Thesis

Development and Validation of a Material Model with Artificial Neural Networks

Claudia Gaßner
e0725363

supervised by:

Ao. Univ. Prof. Dipl.-Ing. Dr.techn. Helmut Böhm,
TU Vienna, Institute of Lightweight Design and Structural Biomechanics

2014, TU Vienna, Vienna

Contents

Acknowledgement

This study was carried out as a master thesis at the Institute of Lightweight Design and Structural Biomechanics (ILSB) at the Vienna University of Technology. The cooperation with ANDATA Development Technologies GmbH, Hallein, Austria and the financial support is gratefully acknowledged.

I especially want to thank my thesis advisor, Prof. Dr. Helmut J. Böhm, for his extensive support and advice .

I also want to thank Dr. Andreas Kuhn for providing the opportunity to do interesting research work.

I would like to thank the ANDATA team, especially Lic. Toni Palau, for the extensive guidance and training.

Last but not least I want to thank my parents Gertraud and Karl Gaßner for enabling me to follow my ambitions, their encouragement and support.

Abbreviations

ANN ... Artificial Neuronal Network

ED ... Eulerian description

LD ... Lagrangian description

RVE ... Representative Volume Element

DOF ... Degree(s) of Freedom

FEM ... Finite Elemente Methods

Notation

0 ... index for initial condition or onset of a certain material behaviour

$\dot{}$ ... time derivative $\frac{d}{dt}$

* ... effective material parameter during damage evolution

max ... index for maximum value

a ... amplitude

$\mathbf{a}$, a_i ... original/reference position

A ... cross-section area

A_D ... effective area of all microcracks and -cavities

A^* ... effective undamaged area

$\mathbf{A}$, A_{ij} ... Almansi-Euler strain tensor

$\mathbf{A}^\mathrm{lin}, A_{ij}^\mathrm{lin}$ ... linearised Almansi-Euler strain tensor

b ... rate at which the size of the yield surface changes during isotropic hardening

b_j ... bias of ANN

$\mathbf{b}$ ... Burger's vector

$\mathbf{B}$ ... Right Cauchy-Green deformation tensor

c ... constant

c_d ... dilatation wave speed

C, C_k ... initial kinematic hardening modulus/i

$\mathbb{C}$, $\mathbb{C}_{ijkl}$ ... material elasticity tensor

$\{\mathbb{C}\}$ ... material elasticity matrix

$\mathbb{C}^\mathrm{ep}$ ... tangential stiffness matrix (plasticity)

$\mathbf{C}$ ... Left Cauchy-Green deformation tensor

d_1, d_2 ... diameter

D ... damage variable

D_1c ... corresponding critical damage variable

e_j ... absolute error

E ... Young's modulus

$\mathrm{E_{MSE}}$ ... mean squared error

$\mathrm{E_{LOG}}$ ... logarithmic error

$\mathrm{E_{SR}}$ ... square root of the mean squared error function

$\mathrm{E_{SSE}}$, E ... sum of squared error function

f ... yield function

f_lin ... purely linear transfer function

f_log ... logistic sigmoid transfer function

f_tan ... tan-sigmoid transfer function

f_tf ... transformation function

f_trans ... transfer function

f_1 ... force component in 1-direction

$\mathbf{f}$ ... force vector

$\mathbf{F}$, F_{ij} ... deformation gradient

g ... plastic potential function

g_in ... function to evaluate the inputs of a neuron

$\mathbf{g}$, g_j ... gradient of the error with respect to the weight

G ... shear modulus

G_f ... dissipated fracture energy per unit area

$\mathbf{G}$, G_{ij} ... Green-Lagrange strain tensor

$\mathbf{G}^\mathrm{lin}$, G_{ij}^lin ... linearised Green-Lagrange strain tensor

h ... constant for the linear isotropic hardening function

$\mathbf{H}^\mathrm{ED}$ ... Hencky or logarithmic strain tensor in ED

$\mathbf{H}^\mathrm{LD}$ ... Hencky or logarithmic strain tensor in LD

$\mathbf{I}$ ... unit matrix

$\mathbf{J}$, J_{ij} ... displacement gradient in LD

J_2 ... second invariant of the stress deviator tensor

k ... shakedown safety coefficient

K ... bulk modulus

$\mathbf{K}$, K_{ij} ... displacement gradient in ED

L ... characteristic element length

M ... number of output nodes

$\mathbf{n}$ ... normal vector

o_j, o_{pj} ... target value of ANN

p ... training pattern

P ... number of training pattern

${}^{I}\mathbf{P}$, ${}^{II}\mathbf{P}$ ... 1^{st} and 2^{nd} Piola-Kirchhoff stress tensor

q ... hydrostatic pressure

Q ... saturated value of r

r ... isotropic hardening function

r_1, r_2 ... radius

$\mathbf{R}$ ... rotation tensor

s ... pseudo-time

s_1 ... thickness

$\mathbf{S}$ ... deviatoric stress tensor

t ... time

t_1 ... time of the onset of shakedown

Δt_stable ... stable time increment

$\mathbf{t}$ ... Cauchy stress vector

${}^{I}\mathbf{t}$, ${}^{II}\mathbf{t}$ ... 1^{st} and 2^{nd} Piola-Kirchhoff stress vector

$\mathbf{u}$, u_i ... displacement vector

$\mathbf{u}^N$, u_i^N ... displacement of node N

$\mathbf{u}_\mathrm{deform}$ ... displacement due to deformation

$\mathbf{u}_\mathrm{rigid}$ ... displacement due to rigid body motion

$\bar{u}_\mathrm{D}^\mathrm{p}$ ... equivalent plastic displacement after the onset of damage

$\bar{u}_\mathrm{f}^\mathrm{p}$ ... equivalent plastic displacement at failure

$\mathbf{U}$ ... right stretch tensor

v_j ... induced local field of ANN

V ... volume

$\mathbf{V}$ ... left stretch tensor

w_D ... state variable for damage initiation

w_{ji} ... weight of ANN

W^p ... plastic work

$\mathbf{W}$ ... Jacobian matrix with the first derivatives of the absolute error with respect to the weights

x_j ... input of ANN

$\mathbf{X}$, X_i ... current position

y_j, y_{pj} ... output of ANN

$\boldsymbol{\alpha}$ ... backstress

$\boldsymbol{\alpha}^\mathrm{dev}$ ... deviatoric backstress

γ, γ_k ... rate(s) at which kinematic hardening modulus/i decrease(s)

δ_j ... local gradient

$\boldsymbol{\varepsilon}$ ... Hencky or logarithmic strain tensor in ED

$\{\varepsilon\}$ ... strain vector

ε^e_{ij}, $\varepsilon^{\mathrm{p}}_{ij}$ ... elastic log. strain, plastic log. strain

$\bar{\varepsilon}^{\mathrm{p}}$ ... effective or accumulated plastic strain

$\bar{\varepsilon}^{\mathrm{p}}_{\mathrm{D}}$ ... effective plastic strain after the onset of damage

$\bar{\varepsilon}^{\mathrm{p}}_{\mathrm{f}}$ ... effective plastic strain at failure

ε^{G}, $\varepsilon^{\mathrm{G^{lin}}}$ ... Green-Lagrange strain, linearised Green-Lagrange strain

$\varepsilon^{\mathrm{H^{ED}}}$, ε ... Hencky or logarithmic strain in ED

$\varepsilon^{\mathrm{H^{LD}}}$ ... Hencky or logarithmic strain in LD

$\boldsymbol{\varepsilon}_{\mathrm{dev}}$ ... deviatoric strain tensor

$\varepsilon_{\mathrm{vol}}$ ... volumetric strain

η ... learning rate

κ_i ... extension ratio

λ ... plastic multiplier

μ ... stress triaxiality

ν ... Poisson ratio

ρ ... density

$\boldsymbol{\sigma}$ ... Cauchy stress tensor

σ_{f} ... stress at material failure

σ_i ... principal stress components

σ_{ij} ... stress components

$\{\sigma\}$ ... stress vector

$\boldsymbol{\sigma}^{\mathrm{el}}$ ... fictitious elastic stress

$\boldsymbol{\sigma}^{\mathrm{r}}(\mathbf{X}, t)$ ... residual stress dependent on space and time

$\bar{\boldsymbol{\sigma}}^{\mathrm{r}}(\mathbf{X})$ ... residual stress dependent on space, independent on time

$\sigma^{\mathrm{M}}_{\mathrm{e}}$, σ_{e} ... equivalent tensile stress of the von Mises yield criterion

$\sigma^{\mathrm{T}}_{\mathrm{e}}$ ... equivalent tensile stress of the Tresca yield criterion

σ_{sat} ... stress threshold of saturation during isotropic or mixed hardening

σ_{y} ... yield stress

τ ... shear component of the logrithmic stress tensor

ϕ ... physical property

φ ... angle

Kurzfassung

Das Ziel dieser Diplomarbeit ist es, ein Materialmodell mit künstlichen Neuronalen Netzen ("artificial neural networks", ANN) zu entwickeln und seine Effizienz zu bewerten. In [22] wurde bereits ein Materialmodell für ein zweidimensionales Kontinuumselements (ebener Verzerrungszustand) mit elasto-plastischem Materialverhalten und isotroper Verfestigung untersucht. Nun stellt sich die Frage, ob es auch möglich ist, ausreichend genaue Ergebnisse für ein Materialmodell eines dreidimensionalen Kontinuumselements mit elasto-plastischem Materialverhalten und kombinierter isotroper und kinematischer Verfestigung, duktiler Schädigung und Materialversagen zu erzielen.

Die ersten beiden Kapitel geben einen Überblick über die Theorie des verwendeten Materialmodells und die Anwendung von ANN. Das letzte Kapitel handelt von der Entwicklung und Bewertung des verwendeten Materialmodells mit ANN. Zuerst werden die benötigten Daten für das Training, die Validation and das Testen der ANN mit dem Finite Elemente (FE) Programm ABAQUS/Explicit generiert. Die verwendeten Materialeigenschaften, Randbedingungen und Lastfälle werden im Detail beschrieben.

Als nächstes werden die Architektur, der Lernprozess und andere Kenngrößen der ANN vorgestellt. Um die Spannungsantworten des ANN-basierenden Materialmodells zu optimieren werden ANN mit einer variierenden Anzahl von Neuronen und "hidden layers" und verschiedene Kombinationen von Eingangs-, Ausgangssignalen und Lastfällen als Trainingsdaten getestet. Die Spannungsantworten der ANN auf die verschiedenen Lastfälle werden verglichen und bewertet. Die Ergebnisse zeigen, dass ANN im Allgemeinen komplexes Materialverhalten erlernen können, aber der Fehler und die Generalisierung für die verschiedenen Lastfälle und auch für einzelnen Spannungskomponenten für die untersuchten Netze von einander abweichen. Daraus kann man schließen, dass es notwendig ist, weitere Untersuchungen durchzuführen um das Ergebnis zu verbessern. Mögliche Lösungsansätze werden im letzen Abschnitt vorgestellt.

Zusätzlich wurde eine ABAQUS/Explicit User-Subroutine, VUMAT, für das ANN-Materialmodell implementiert. Auftretenden Probleme werden ebenfalls beschrieben.

Abstract

The goal of this master's thesis is to develop and evaluate the performance of a material model using artificial neural networks (ANN). In [22] a material model of a two dimensional solid element (plain strain) with elasto-plastic material behaviour with isotropic hardening properties was studied. The question is if it is also possible to obtain suitable results for a material model for three dimensional solid elements with elasto-plastic material behaviour with mixed hardening properties, ductile damage and material failure.

The first two chapters give an overview of the theory concerning the applied material model and the use of ANN. The final chapter deals with the actual development and evaluation of the proposed ANN-based material model. First the necessary data for training, validation and testing of the ANN are generated with the help of the Finite Element (FE) program ABAQUS/Explicit. The used material properties, boundary conditions and applied load cases are discussed in detail.

Next the architecture, learning algorithms and other properties of the ANN are presented. To optimize the stress responses of the ANN-based material model ANN with varying numbers of nodes and hidden layers and different sets of inputs, outputs and load cases for the training data were tested. The stress responses of the ANN for different load cases are compared and evaluated. The obtained results show that the ANN can in general learn complex material behaviour, but for the considered ANN the error and the generalization differ for the various load cases considered as well as for the stress components. This suggests that further studies are necessary to improve the results. Possible solutions and ideas are presented in the final section.

In addition an ABAQUS/Explicit user-subroutine, VUMAT, for the ANN material model was implemented. The occurring issues are presented.

1 Continuum Mechanics and Damage Modelling

In engineering applications the relationship between forces and deformations, or between stresses and strains are essential. Therefore, the mechanical behaviour of a vast range of different materials has been studied and many theories have been developed to describe the occurring phenomena. In this thesis we take a look at the behaviour of metals since the aim of this study is to describe the material behaviour of aluminium with the help of artificial neuronal networks (ANN).
To provide an overview of the general material behaviour of a ductile metal, we use StE690, which shows qualitatively similar behaviour as aluminium, and examine the results of a punch test taken from the literature [2]. During the punch test specimens with a diameter of $d_1 = 8$mm and a thickness of $s_1 = 5$mm are clamped on a die with a bore diameter of $d_2 = 4$mm and an edge radius of $r_1 = 0.5$mm. The punch, with a punch head radius of $r_2 = 1.25$mm is moved downward with a velocity of 0.5 mm/min (cp. fig. 1.1).
The resulting load-displacement curve is shown in Fig. 1.1. The material behaviour can be divided into elastic behaviour (Part I), plastic behaviour (Part II, Part III), damage (Part IV, V) and failure (Part VI). In Part I we can observe the elastic material behaviour, in Part II the transition between elastic and plastic behaviour occurs, in Part III the hardening behaviour of plasticity sets in, in Part IV geometrical softening and damage follow, in Part V a crack, which is detected by CCD-camera monitors, initiates and grows and in Part VI the specimen is finally punched through. [2]

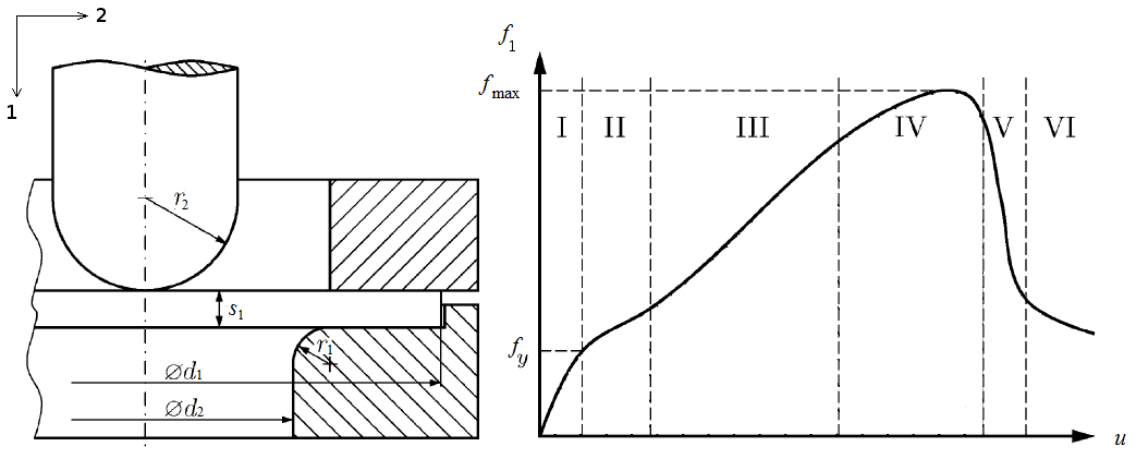

Fig. 1.1: left: Principal sketch of a small punch test. right: Resulting load-displacement curve with load f_1 and displacement u in 1-direction (Part I: Elasticity, Part II: Transition Elasticity-Plasticity with the yield load f_y, Part III: Plasticity - Hardening, Part IV: Damage - Geometrical Softening with the maximal load $f_{\max}$, Part V: Damage - Crack Initiation and Growth, Part VI: Failure) [2]

1.1 Strain and Stress Tensors

To describe the relationship between displacement and force two quantities are established, strain and stress. The theory of continuum mechanics states that a body can be subdivided into infinitesimal elements, called material points. The position vector of a material point P in the three dimensional Euclidean space is $\mathbf{a}$ in the reference configuration at time t_0 and $\mathbf{X}$ in the current configuration at time t (cp. fig. 1.2). There are two different ways of describing the position and physical properties of a material particle. We can either describe them in terms of time and the reference configuration (Lagrangian description, LD) or the current configuration (Eulerian description, ED). A physical property ϕ like the density ρ, or the Cauchy stress tensor $\boldsymbol{\sigma}$ can be described as

$$\text{LD}: \qquad \phi = \phi(\mathbf{a}, t) \tag{1.1}$$

$$\text{ED}: \qquad \phi = \phi(\mathbf{X}, t) \tag{1.2}$$

In the LD an observer would follow a material point and measure the change of the physical properties of the material point, while in the ED an observer would be standing at a fixed place observing the passing of different material points and measuring the change of the physical properties at this position. The LD is often

used in solid mechanics while the ED is usually used in fluid mechanics, but also for plasticity.
The displacement of a material point P can now be described as

$$\text{LD}: \qquad \mathbf{u}(\mathbf{a},t) = \mathbf{X}(\mathbf{a},t) - \mathbf{a} \tag{1.3}$$

$$\text{ED}: \qquad \mathbf{u}(\mathbf{X},t) = \mathbf{X} - \mathbf{a}(\mathbf{X},t) \tag{1.4}$$

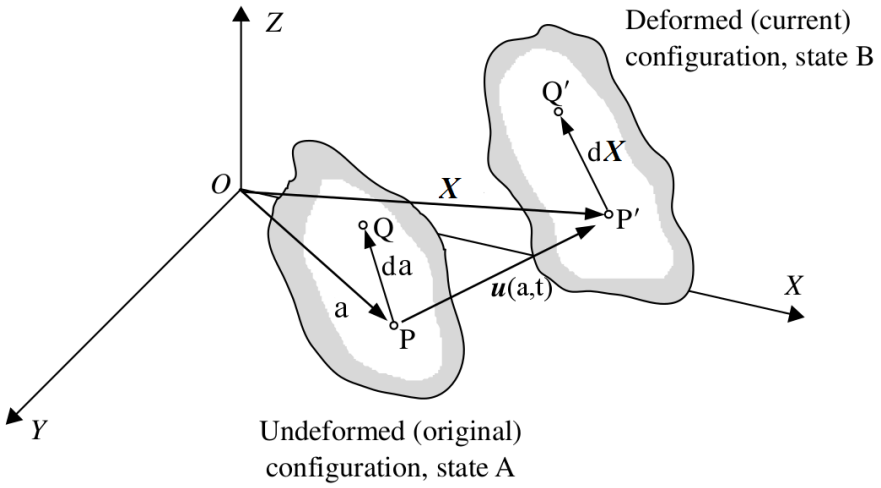

Fig. 1.2: Original and current configurations of a body with the position vectors $\mathbf{a}$ *and* $\mathbf{X}$ *of the material point P at the time* t_0 *and* t *and the displacement* $\mathbf{u}(\mathbf{a},t)$ *in the LD. [9]*

The displacement of a material point consists of contributions due to deformation and rigid body motion like translation or rotation. Since we want to investigate the material behaviour we have to establish a quantity that describes only the deformation of the material. Therefore we define the deformation gradient $\mathbf{F}$ as

$$\text{LD}: \qquad \mathbf{F} = [\nabla_{\mathbf{a}} \cdot \mathbf{X}]^T \qquad F_{ij} = \frac{\partial X_i}{\partial a_j} \tag{1.5}$$

$$\text{ED}: \qquad \mathbf{F}^{-1} = [\nabla_{\mathbf{X}} \cdot \mathbf{a}]^T \qquad F_{ij}^{-1} = \frac{\partial a_i}{\partial X_j} \quad , \tag{1.6}$$

and the displacement gradients $\mathbf{J}$ in the LD and $\mathbf{K}$ in the ED as

$$\text{LD}: \qquad \mathbf{J} = [\nabla_{\mathbf{a}} \cdot \mathbf{u}]^T \qquad J_{ij} = \frac{\partial u_i}{\partial a_j} \tag{1.7}$$

$$\text{ED}: \qquad \mathbf{K} = [\nabla_{\mathbf{X}} \cdot \mathbf{u}]^T \qquad K_{ij} = \frac{\partial u_i}{\partial X_j} \quad , \tag{1.8}$$

where $\nabla_{\mathbf{a}}$ and $\nabla_{\mathbf{X}}$ are the nabla operators, which denote the standard derivatives with respect to the original and current position, $\mathbf{a}$ and $\mathbf{X}$, and a superscript T denotes the transpose of a vector or matrix.
Due to equations (1.3) and (1.4) follows

$$\text{LD}: \qquad \mathbf{F} = \mathbf{I} + \mathbf{J} \tag{1.9}$$

$$\text{ED}: \qquad \mathbf{F}^{-1} = \mathbf{I} + \mathbf{K} \quad , \tag{1.10}$$

where $\mathbf{I}$ is the unit tensor of order 2.
Although we have already eliminated the rigid body translation, the deformation gradient can still be split into rigid body rotation and deformation. The polar decomposition of the deformation gradient $\mathbf{F}$ results in

$$\mathbf{F} = \mathbf{RU} = \mathbf{VR} \quad , \tag{1.11}$$

with

$$\mathbf{U} = \left(\mathbf{F}^{\mathrm{T}}\mathbf{F}\right)^{\frac{1}{2}}, \qquad \mathbf{V} = \left(\mathbf{F}\mathbf{F}^{\mathrm{T}}\right)^{\frac{1}{2}} \tag{1.12}$$

The rigid body rotation is described by $\mathbf{R}$ and the deformation by the right or left stretch tensor $\mathbf{U}$ and $\mathbf{V}$.
We can now define the left Cauchy-Green deformation tensor $\mathbf{C}$ and right Cauchy-Green deformation tensor $\mathbf{B}$ as

$$\mathbf{C} = \mathbf{U}^2 = \mathbf{F}^{\mathrm{T}}\mathbf{F}, \qquad \mathbf{B} = \mathbf{V}^2 = \mathbf{F}\mathbf{F}^{\mathrm{T}} \tag{1.13}$$

Finally we get the Green-Lagrange strain tensor $\mathbf{G}$ and the Almansi-Euler strain tensor $\mathbf{A}$

$$\text{LD}: \qquad \mathbf{G} = \frac{1}{2}\left(\mathbf{C} - \mathbf{I}\right) \tag{1.14}$$

$$= \frac{1}{2}\left(\mathbf{F}^{\mathrm{T}}\mathbf{F} - \mathbf{I}\right) \tag{1.15}$$

$$= \frac{1}{2}\left(\mathbf{J} + \mathbf{J}^{\mathrm{T}} + \mathbf{J}\mathbf{J}^{\mathrm{T}}\right) \tag{1.16}$$

$$\text{LD}: \qquad G_{ij} = \frac{1}{2}\left[\frac{\partial u_i}{\partial a_j} + \frac{\partial u_j}{\partial a_i} + \frac{\partial u_k}{\partial a_i}\frac{\partial u_k}{\partial a_j}\right] \tag{1.17}$$

$$\text{ED}: \qquad \mathbf{A} = \frac{1}{2}\left(\mathbf{I} - \mathbf{B}^{-1}\right) \tag{1.18}$$

$$= \frac{1}{2}\left(\mathbf{I} - \left(\mathbf{F}\mathbf{F}^{\mathrm{T}}\right)^{-1}\right) \tag{1.19}$$

$$= \frac{1}{2}\left(\mathbf{K} + \mathbf{K}^{\mathrm{T}} - \mathbf{K}^{\mathrm{T}}\mathbf{K}\right) \tag{1.20}$$

$$\text{ED}: \qquad A_{ij} = \frac{1}{2}\left[\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} - \frac{\partial u_k}{\partial X_i}\frac{\partial u_k}{\partial X_j}\right] \tag{1.21}$$

The Green-Lagrange strain tensor $\mathbf{G}$ is usually used in finite strain theory. However, within the small strain approximation the relationships (1.14) to (1.21) can be linearised if $\frac{\partial u_i}{\partial a_j} \ll 1$ and $\frac{\partial u_i}{\partial X_j} \ll 1$. Then we can assume that $\mathbf{X} \approx \mathbf{a}$ and $\nabla_{\mathbf{a}} \approx \nabla_{\mathbf{X}}$, so that

$$\mathbf{G}^{\text{lin}} \approx \mathbf{A}^{\text{lin}} = \frac{1}{2}\left(\mathbf{F}^{\mathrm{T}} + \mathbf{F}\right) - \mathbf{I} \tag{1.22}$$

$$= \frac{1}{2}\left(\mathbf{J} + \mathbf{J}^{\mathrm{T}}\right) \tag{1.23}$$

$$G_{ij}^{\text{lin}} = \frac{1}{2}\left[\frac{\partial u_i}{\partial a_j} + \frac{\partial u_j}{\partial a_i}\right] \approx \frac{1}{2}\left[\frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i}\right] = A_{ij}^{\text{lin}} \tag{1.24}$$

For plastic behaviour, however, the logarithmic or Hencky strain tensor is often used:

$$\text{LD}: \qquad \mathbf{H}^{\text{LD}}(\mathbf{a}, t) = \ln \mathbf{U} \tag{1.25}$$

$$= \frac{1}{2}\ln\left(\mathbf{F}^{\mathrm{T}}\mathbf{F}\right) \tag{1.26}$$

$$= \frac{1}{2}\ln\left(\mathbf{I} + 2\mathbf{G}\right) \tag{1.27}$$

$$\tag{1.28}$$

$$\text{ED}: \qquad \mathbf{H}^{\text{ED}}(\mathbf{X}, t) = \ln \mathbf{V} \tag{1.29}$$

$$= \frac{1}{2}\ln\left(\mathbf{F}\mathbf{F}^{\mathrm{T}}\right) \tag{1.30}$$

$$= -\frac{1}{2}\ln\left(\mathbf{I} - 2\mathbf{A}\right) \tag{1.31}$$

It is advantageous that consecutive strains can be summed up directly [1]. Furthermore it is possible to split the tensor into a volumetric and a deviatoric strain tensor, which will be discussed later.
If we deform an element in only one direction i (uniaxial tensile strain), we can now summarize and compare the different strain measures in LD: Green-Lagrange strain $\varepsilon_i^{\mathrm{G}}$, linearised Green-Lagrange strain $\varepsilon^{\mathrm{G^{lin}}}$ and Hencky strain $\varepsilon_i^{\mathrm{H^{LD}}}$ with the extension ratio $\kappa_i = \frac{|\mathrm{d}X_i|}{|\mathrm{d}a_i|}$ of the original and current element length $|\mathrm{d}X_i|$, $|\mathrm{d}a_i|$ as

$$\varepsilon_i^{\mathrm{G}} = \frac{1}{2}\left(\kappa_i^2 - 1\right) \tag{1.32}$$

$$\varepsilon_i^{\mathrm{G^{lin}}} = \frac{1}{2}\left(\kappa_i - 1\right) \tag{1.33}$$

$$\varepsilon_i^{\mathrm{H^{LD}}} = \ln\left(\kappa_i\right) \tag{1.34}$$

In the following we will always use the Hencky strain in ED, $\mathbf{H}^{\mathrm{ED}} = \boldsymbol{\varepsilon}$, because it is the logarithmic strain measure used in ABAQUS for the single-element tests.
[3], [10], [17]

If forces are acting on a body and we imagine cutting it into two parts, there must be forces of equal magnitude but opposite direction acting on the dividing surfaces to keep the equilibrium (cp. fig. 1.3 left). We can now calculate the stress vector $\mathbf{t}$ in a point P at the position $\mathbf{X}$ on the resulting section plane as

$$\mathbf{t}\left(\mathbf{X}, \mathbf{n}, t\right) = \lim_{dA \to 0} \frac{d\mathbf{f}}{dA} \quad , \tag{1.35}$$

$\mathbf{f}$ stands for the current force vector, A for the current cross-section area, $\mathbf{n}$ is the normal vector of the surface at the point P and t is the time.
If we imagine that the body is subdivided into infinitesimal cuboid volume elements with edges parallel to the chosen coordinate-system, stresses act on their surfaces which are the components of the Cauchy-stress-tensor $\boldsymbol{\sigma}$ (cp. fig. 1.3 right). The relationship between the stress vector $\mathbf{t}$ and the Cauchy-stress-tensor is

$$\mathbf{t} = \mathbf{n} \cdot \boldsymbol{\sigma} \tag{1.36}$$

---

[1] lin. Green-Lagrange strain: $\varepsilon^{\mathrm{G^{lin}}} = \frac{l - l_0}{l} = \frac{\Delta l}{l_0}$, $\varepsilon_1^{\mathrm{C}} + \varepsilon_1^{\mathrm{C}} = \frac{\Delta l_1}{l_0} + \frac{\Delta l_2}{l_0 + \Delta l_1} \neq \frac{\Delta l_1 + \Delta l_2}{l_0} = \varepsilon_{1+2}^{\mathrm{C}}$

Hencky Strain: $\varepsilon^{\mathrm{H^{LD}}} = \ln\left(\frac{l}{l_0}\right)$, $\varepsilon_1^{\mathrm{H^{LD}}} + \varepsilon_2^{\mathrm{H^{LD}}} = \ln\left(\frac{l_1}{l_0}\right) + \ln\left(\frac{l_2}{l_1}\right) = \ln\left(\frac{l_2}{l_0}\right) = \varepsilon_{1+2}^{\mathrm{H^{LD}}}$

with l_i ... length, l_0 ... initial length

For describing elastic material behaviour the 1^{st} and 2^{nd} Piola-Kirchhoff-stress-tensor ${}^{I}\mathbf{P}$, ${}^{II}\mathbf{P}$ are often used. With the help of the corresponding stress vectors ${}^{I}\mathbf{t}$, ${}^{II}\mathbf{t}$ and the index 0 denoting the initial state they are defined as

$$ {}^{I}\mathbf{t} = \lim_{dA_0 \to 0} \frac{d\mathbf{f}}{dA_0}, \quad {}^{I}\mathbf{t} = \mathbf{n}_0 \cdot {}^{I}\mathbf{P} \tag{1.37} $$

$$ {}^{II}\mathbf{t} = \lim_{dA_0 \to 0} \frac{d\mathbf{f}_0}{dA_0}, \quad {}^{II}\mathbf{t} = \mathbf{n}_0 \cdot {}^{II}\mathbf{P} \tag{1.38} $$

In the following we will always use the Cauchy-stress vector and tensor, $\mathbf{t}$ and $\boldsymbol{\sigma}$, [1], [3], [10], [17].

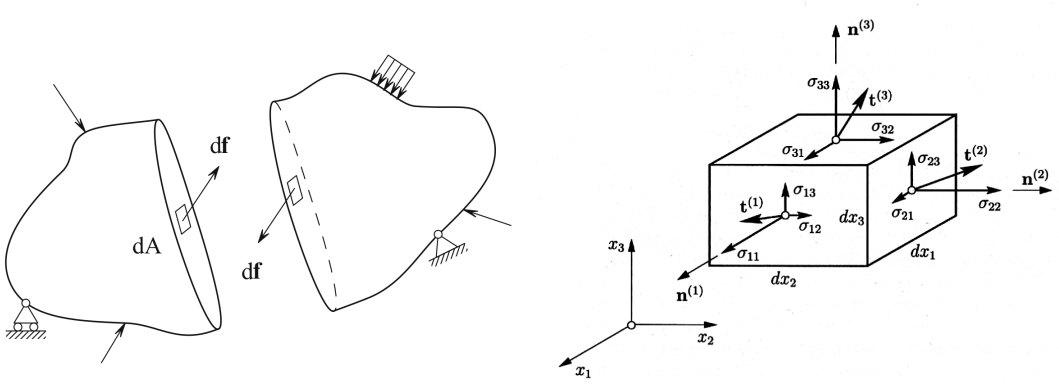

Fig. 1.3: left: Resulting inner force $d\mathbf{f}$ on the cross-section area dA of a body with external loading [3], right: stress components σ_{ij} of the Cauchy stress tensor $\boldsymbol{\sigma}$ on the surface of a cuboid volume element with the resulting stress vectors $\mathbf{t}^{(i)}$ and the normal vectors $\mathbf{n}^{(i)}$ [17]

1.2 Elasto-Plastic Material Behaviour

We stated that Part I of the stress-strain curve displayed in fig. 1.1 shows elastic behaviour. This means that if we remove the applied force the deformation will recover. We can also see that for small deformations the relationship between applied force $\mathbf{f}$ and displacement[2] $\mathbf{u}$ is linear, as well as that between stress $\boldsymbol{\sigma}$ and strain $\boldsymbol{\epsilon}$. This relationship can be described by Hooke's law

$$ \sigma_{ij} = \mathbb{C}_{ijkl}\varepsilon_{kl} \qquad \boldsymbol{\sigma} = \mathbb{C} : \boldsymbol{\varepsilon} \qquad , \tag{1.39} $$

[2]During the punch test the rigid body motion $\mathbf{u}_{\text{rigid}} = \mathbf{0}$. Therefore the displacement $\mathbf{u} = \mathbf{u}_{\text{rigid}} + \mathbf{u}_{\text{deform}}$ is equal to the deformation $\mathbf{u}_{\text{deform}}$

where $\mathbb{C}_{ijkl}$ is the material elasticity tensor. For an isotropic, linear elastic material $\mathbb{C}_{ijkl}$ can be built from two independent material constants. Usually the Young's modulus E and the Poisson ratio ν or the shear modulus G and the bulk modulus K are used. The relationship between these two sets of independent material constants are

$$G = \frac{E}{2(1+\nu)} \tag{1.40}$$

$$K = \frac{E}{3(1-2\nu)} \tag{1.41}$$

We can now write Hooke's law in matrix form as

$$\{\sigma\} = \{\mathbb{C}\}\{\varepsilon\} \qquad , \tag{1.42}$$

with the stress and strain vectors

$$\{\sigma\} = \{\sigma_{11} \quad \sigma_{22} \quad \sigma_{33} \quad \sigma_{12} \quad \sigma_{23} \quad \sigma_{31}\}^{\mathrm{T}} \tag{1.43}$$

$$\{\varepsilon\} = \{\varepsilon_{11} \quad \varepsilon_{22} \quad \varepsilon_{33} \quad 2\varepsilon_{12} \quad 2\varepsilon_{23} \quad 2\varepsilon_{31}\}^{\mathrm{T}} \qquad , \tag{1.44}$$

and the elastic moduli matrix $\{\mathbb{C}\}$

$$\{\mathbb{C}\} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ & 1-\nu & \nu & 0 & 0 & 0 \\ & & 1-\nu & 0 & 0 & 0 \\ & \text{Sym.} & & \frac{1-2\nu}{2} & 0 & 0 \\ & & & & \frac{1-2\nu}{2} & 0 \\ & & & & & \frac{1-2\nu}{2} \end{bmatrix} \tag{1.45}$$

$$\{\mathbb{C}\} = \begin{bmatrix} K+\frac{4}{3}G & K-\frac{2}{3}G & K-\frac{2}{3}G & 0 & 0 & 0 \\ & K+\frac{4}{3}G & K-\frac{2}{3}G & 0 & 0 & 0 \\ & & K+\frac{4}{3}G & 0 & 0 & 0 \\ & & & G & 0 & 0 \\ & \text{Sym.} & & & G & 0 \\ & & & & & G \end{bmatrix} \qquad , \tag{1.46}$$

[12],[23], [6].

In Fig. 1.1 we can see that the elastic behaviour is followed by plastic behaviour, which means that the removal of the applied force results in permanent deformation. Part II of the force-displacement curve describes the transition from elastic to plastic behaviour. The gradual change from elastic to plastic behaviour is typical for many metals and alloys like aluminium. In these cases there is no clear threshold for the onset of yielding. The yield stress σ_y can, for example, be defined as the stress that causes a permanent strain of $\epsilon = 0.2‰$, [23], [12], [9].

The reason for the change from elastic to plastic behaviour of crystalline materials is crystal slip. Most metals are polycrystalline, which means that they consist of many crystallites in which the atoms are placed in a fixed order in a lattice. Two common lattice structures would be the face centered cubic (cp. fig. 1.4 top), where the atoms are placed at the corners and the centers of the faces of the cube and the body centered cubic (cp. fig. 1.4 bottom), where the atoms are centered at the corners and the center of the cube. During elastic deformation the interatomic bonds are not broken and the atoms stay at their original places in the lattice. However, during plastic deformation interatomic bonds are broken and reformed, which results in the motion of atomic planes relative to another (cp. fig. 1.5). This motion is called crystal slip. It preferentially occurs on certain planes and directions of crystals, so called slip systems (cp. fig. 1.4), which are usually the most densely packed planes and directions. If a critical shear stress acts upon such a slip system slip occurs. However, a critical load normal to the slip plane would result in a crack, if there are no other slip systems with an appropriate orientation. Therefore only shear loads at crystal level contribute to plastic deformation.

Since the lattices of metals are usually not perfect, the necessary shear stress to cause yielding of a metal is many orders of magnitude smaller than the theoretical shear strength of a crystal. There are two main types of line defects, so called dislocations, which contribute to the yielding of polycrystalline materials (cp. fig. 1.6). One is the edge dislocation, the other the screw dislocation. Their unit of slip displacement is described by the Burger's vector $\mathbf{b}$, [9], [23].

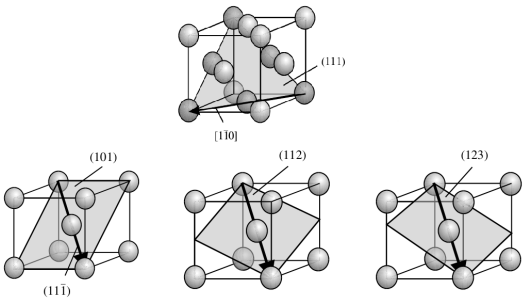

Fig. 1.4: *top: Face centered cubic lattice, bottom: Body centered cubic latice; arrows and grey planes denote different slip systems [9]*

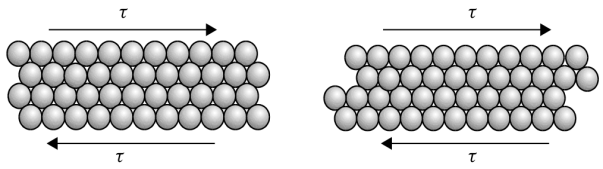

Fig. 1.5: *Crystal slip during plastic deformation with acting shear load* τ *[9]*

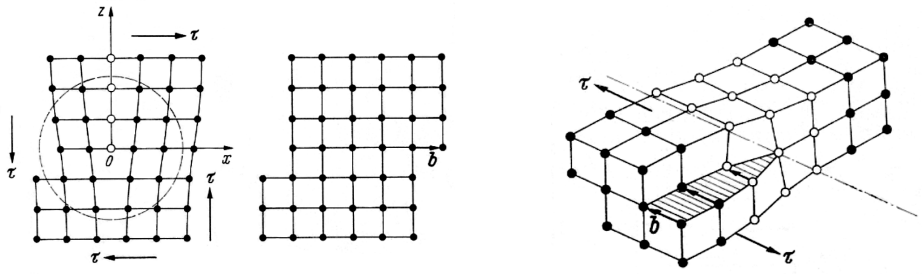

Fig. 1.6: *left: Edge dislocation before and after deformation* $\mathbf{b}$*, right: Screw dislocation with acting shear load* τ *and the Burger's vector* $\mathbf{b}$ *[23]*

To describe elasto-platic material behaviour it is useful to separate the strain tensor $\boldsymbol{\varepsilon}$ into elastic and plastic strains, $\boldsymbol{\varepsilon}^e$ and $\boldsymbol{\varepsilon}^{\mathrm{p}}$. If the strains are small this takes the form

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}^e + \boldsymbol{\varepsilon}^{\mathrm{p}} \tag{1.47}$$

Since we have established that only shear stress contributes to yielding of crystalline metals and plastic slip doesn't lead to volume change, it is logical that hydrostatic stress has no influence on the yielding of such materials. Therefore, it is also useful to establish the terms of volumetric and deviatoric strain and stress. The volumetric

strain ε_{vol} and the deviatoric strain tensor $\boldsymbol{\varepsilon}_{\text{dev}}$ are

$$\varepsilon_{\text{vol}} = \text{tr}\,(\boldsymbol{\varepsilon}) \tag{1.48}$$

$$\boldsymbol{\varepsilon}_{\text{dev}} = \boldsymbol{\varepsilon} - \frac{1}{3}\varepsilon_{\text{vol}}\mathbf{I} \qquad , \tag{1.49}$$

and the hydrostatic pressure q and the deviatoric stress tensor $\mathbf{S}$ (with the volumetric stress tensor $q\mathbf{I}$) are

$$q = -\frac{1}{3}\text{tr}\,(\boldsymbol{\sigma}) \tag{1.50}$$

$$\mathbf{S} = \boldsymbol{\sigma} + q\mathbf{I} \qquad , \tag{1.51}$$

[1], [9], [17].

1.2.1 Yield Criteria for Metals

Determining the onset of yielding for a uniaxial tensile load case in 11-direction would be straightforward. If the measured stress in 11-direction is smaller than the yield stress, $\sigma_{11} < \sigma_{\text{y}}$, the material behaviour is elastic, otherwise ($\sigma_{11} \geq \sigma_{\text{y}}$) the material behaviour is plastic. However, for a multiaxial stress state a suitable yield criterion must be found.

For isotropic yielding independent of hydrostatic stress we introduce two common yield criteria, the Tresca yield criterion and the von Mises yield criterion.

The Tresca yield criterion states that yielding occurs when the maximum shear stress at a point reaches a critical value. For principal stress σ_i the equivalent tensile stress $\sigma_{\text{e}}^{\text{T}}$ is

$$\sigma_{\text{e}}^{\text{T}} = 2\tau_{\text{max}} = \max\left(|\,\sigma_1 - \sigma_2\,|, |\,\sigma_2 - \sigma_3\,|, |\,\sigma_3 - \sigma_1\,|\right) \tag{1.52}$$

The von Mises yield criterion also considers the influence of the intermediate principal stresses. Its equivalent tensile stress $\sigma_{\text{e}}^{\text{M}} = \sigma_{\text{e}}$ in terms of principal stresses σ_i and in terms of direct and shear stresses σ_{ij} is

$$\begin{aligned}\sigma_{\text{e}} &= \left[\frac{1}{2}\left((\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2\right)\right]^{\frac{1}{2}} \qquad (1.53)\\ &= \left[\frac{1}{2}\left((\sigma_{11} - \sigma_{22})^2 + (\sigma_{22} - \sigma_{33})^2 + (\sigma_{33} - \sigma_{11})^2 + 6\left(\sigma_{12}^2 + \sigma_{23}^2 + \sigma_{31}^2\right)\right)\right]^{\frac{1}{2}}\end{aligned}$$

It is also possible to write the von Mises equivalent stress in terms of the deviatoric stress tensor $\mathbf{S}$ or the second invariant of the stress deviator tensor J_2

$$\sigma_{\mathrm{e}} = \left(\frac{3}{2}\mathbf{S} : \mathbf{S}\right)^{\frac{1}{2}} = \sqrt{3J_2} \tag{1.54}$$

If the von Mises criterion is applied, the plastic behaviour is also referred to as J_2-plasticity. In analogy to the effective Mises stress σ_{e} we can also define an effective plastic strain increment $d\bar{\varepsilon}^{\mathrm{p}}$

$$d\bar{\varepsilon}^{\mathrm{p}} = \left(\frac{2}{3} d\boldsymbol{\varepsilon}^{\mathrm{p}} : d\boldsymbol{\varepsilon}^{\mathrm{p}}\right)^{\frac{1}{2}} \tag{1.55}$$

We can now define a yield function f as

$$f(\boldsymbol{\sigma}) = \sigma_{\mathrm{e}} - \sigma_{\mathrm{y}} \qquad , \tag{1.56}$$

which describes the yield condition as

$$\text{elastic behaviour:} \qquad f < 0 \tag{1.57}$$

$$\text{plastic behaviour:} \qquad f = 0 \tag{1.58}$$

We can now imagine that the yield function describes a surface in the principal stress space, called the yield surface. The space enclosed by the yield surface is the elastic region. In the Tresca criterion the yield surface is the surface of a hexagonal prism and in the von Mises criterion it is the surface of a circular cylinder in the principal stress space (cp. fig. 1.7). The axes in both criteria are along the line $\sigma_1 = \sigma_2 = \sigma_3$, also called the hydrostatic axis. If we look at these criteria in a deviatoric plane π and in a two-dimensional principal stress space with $\sigma_3 = 0$, the Tresca criterion describes a regular hexagon or a distorted hexagon and the Mises criterion describes a circle or an ellipse respectively (cp. fig. 1.8), [9], [1], [6], [12].

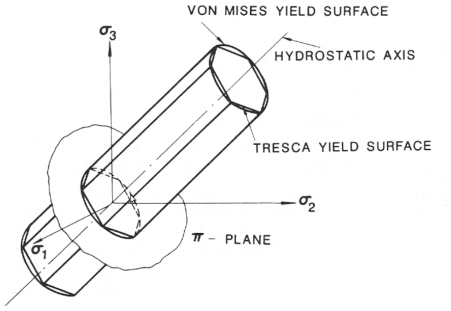

Fig. 1.7: Tresca and von Mises yield surfaces in the principal stress space [6]

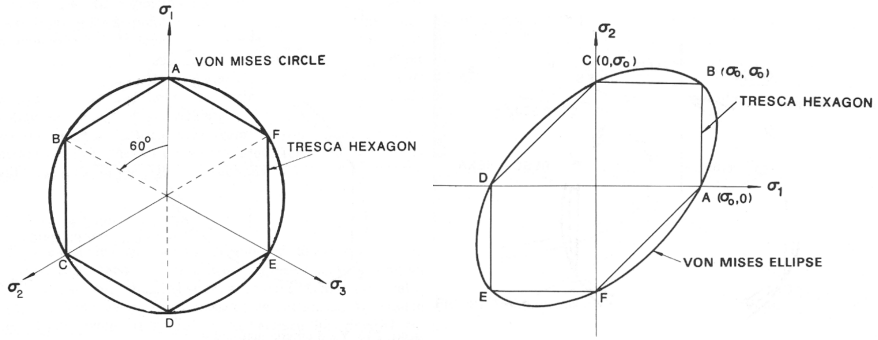

Fig. 1.8: Tresca and von Mises yield surfaces in left: a deviatoric plane π and right: a two-dimensional principal stress space with $\sigma_3 = 0$ [6]

1.2.2 Flow Rule, Normality Hypothesis, Drucker's Stability Postulate

If yielding is initiated, a point in stress space reaches the yield surface. Continuing the loading results in plastic flow, which leads to an increase in the plastic strain tensor. The plastic strain increment can be written in terms of the plastic potential function $g(\boldsymbol{\sigma})$ as

$$d\boldsymbol{\varepsilon}^{\mathrm{p}} = d\lambda \frac{\partial g}{\partial \boldsymbol{\sigma}} \tag{1.59}$$

For metals the yield function usually equals the plastic potential function $f = g$, which leads to the associated flow rule [3]

$$d\boldsymbol{\varepsilon}^{\mathrm{p}} = d\lambda \frac{\partial f}{\partial \boldsymbol{\sigma}} \tag{1.60}$$

[3] $f \neq g$ ... nonassociated flow rule

For this condition the normality hypothesis is true. According to [9] the normality hypothesis states *"[...] that the increment in the plastic strain tensor is in a direction [...] which is normal to the tangent to the yield surface at the load point."* (cp. fig. 1.9). The direction of the strain is given by $\frac{\partial f}{\partial \boldsymbol{\sigma}}$ and the magnitude by the plastic multiplier $d\lambda \geq 1$.
Furthermore the yield surface must be convex, since Drucker's stability postulate states, that the increment of plastic work dW^{p} must be positive for plasticity with work hardening or zero for ideal plasticity. This also shows the irreversible character of plastic deformation

$$dW^{\mathrm{p}} = d\boldsymbol{\sigma} : d\boldsymbol{\varepsilon}^{\mathrm{p}} \geq 0 \tag{1.61}$$

Combining eqn. (1.55) for the effective plastic strain and eqn. (1.60) for the associated flow rule we get

$$d\lambda = d\bar{\varepsilon}^{\mathrm{p}} \tag{1.62}$$

Using eqn. (1.62) and the von Mises criterion, eqns. (1.53), (1.54), for the yield function f, eqn.(1.56), the flow rule can be written as

$$d\boldsymbol{\varepsilon}^{\mathrm{p}} = \frac{3}{2} d\bar{\varepsilon}^{\mathrm{p}} \frac{\mathbf{S}}{\sigma_{\mathrm{e}}} \qquad , \tag{1.63}$$

[9], [6], [17], [1].

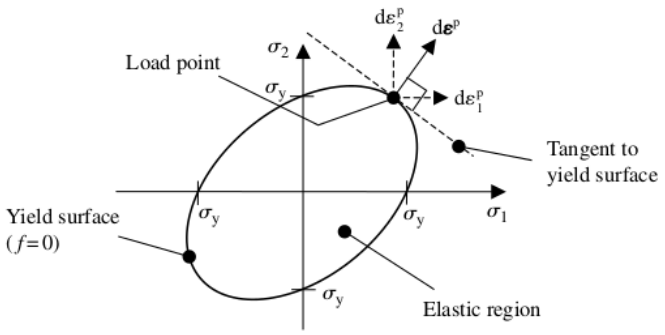

Fig. 1.9: Increment of the plastic strain tensor $d\boldsymbol{\varepsilon}^{\mathrm{p}}$ in a load point of a von Mises yield surface for plane stress condition [9]

1.2.3 Consistency Condition and Ideal Plastic Behaviour

We have already established that yielding occurs as soon as the load point reaches the yield surface. The consistency condition states, that the load point remains on the yield surface during further plastic deformation and the yield stress σ_{y} remains

constant for ideal plastic behaviour (cp. fig. 1.10). Since the yield function f for a load point on the yield surface must be zero we can write

$$f(\boldsymbol{\sigma} + d\boldsymbol{\sigma}) = f(\boldsymbol{\sigma}) + df = f(\boldsymbol{\sigma}) = 0 \tag{1.64}$$

Therefore the consistency condition for ideal plastic behaviour follows as

$$df = \frac{\partial f}{\partial \boldsymbol{\sigma}} : d\boldsymbol{\sigma} = 0 \tag{1.65}$$

If we now combine Hooke's law, eqn. (1.39), the equation for strain separation, eqn. (1.47) and the flow rule (1.60) we get

$$d\boldsymbol{\sigma} = \mathbb{C} : (d\boldsymbol{\varepsilon} - d\boldsymbol{\varepsilon}^{\mathrm{p}}) = \mathbb{C} : \left(d\boldsymbol{\varepsilon} - d\lambda \, \frac{\partial f}{\partial \boldsymbol{\sigma}} \right) \tag{1.66}$$

Substituting eqn. (1.66) into the consistency condition (1.65) leads to the equation for $d\lambda$

$$d\lambda = \frac{\frac{\partial f}{\partial \boldsymbol{\sigma}} : \mathbb{C}}{\frac{\partial f}{\partial \boldsymbol{\sigma}} : \mathbb{C} : \frac{\partial f}{\partial \boldsymbol{\sigma}}} : d\boldsymbol{\varepsilon} \tag{1.67}$$

Inserting this result into eqn. (1.66) we finally get the incremental relationship between stress and strain with the tangential stiffness matrix $\mathbb{C}^{\mathrm{ep}}$

$$d\boldsymbol{\sigma} = \left(\mathbb{C} - \frac{\mathbb{C} : \frac{\partial f}{\partial \boldsymbol{\sigma}} : \frac{\partial f}{\partial \boldsymbol{\sigma}} : \mathbb{C}}{\frac{\partial f}{\partial \boldsymbol{\sigma}} : \mathbb{C} : \frac{\partial f}{\partial \boldsymbol{\sigma}}} \right) : d\boldsymbol{\varepsilon} = \mathbb{C}^{\mathrm{ep}} : d\boldsymbol{\varepsilon} \qquad , \tag{1.68}$$

[6], [9], [17].

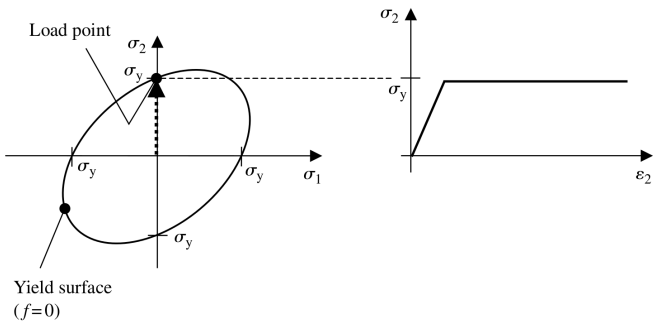

Fig. 1.10: *Ideal plastic behaviour of a material under uniaxial tensile loading, left: load point on the von Mises surface, right: stress-strain relationship [9]*

1.2.4 Isotropic hardening

For many metals the yield stress σ_{y} increases with further plastic deformation. This is called work hardening. During isotropic hardening the yield surface expands uniformly in all directions in stress space. The surfaces subsequent to the initial yield surface are also called loading surfaces. Additionally to the established relationships for ideal plasticity, a hardening rule is necessary to describe the evolution of the yield surface. Furthermore, the yield function becomes dependent on the effective or accumulated plastic strain $\bar{\varepsilon}^{\mathrm{p}}$, which can be deduced from its incremental form (cp. eqn. (1.55)) by appropriate integration or summation. We can write the yield function for isotropic hardening as

$$f\left(\boldsymbol{\sigma}, \bar{\varepsilon}^{\mathrm{p}}\right) = \sigma_{\mathrm{e}} - \sigma_{\mathrm{y}}\left(\bar{\varepsilon}^{\mathrm{p}}\right) \qquad , \tag{1.69}$$

with

$$\sigma_{\mathrm{y}}\left(\bar{\varepsilon}^{\mathrm{p}}\right) = \sigma_{y0} + r\left(\bar{\varepsilon}^{\mathrm{p}}\right) \qquad , \tag{1.70}$$

σ_{y0} is the initial yield function and $r(\bar{\varepsilon}^{\mathrm{p}})$ is the isotropic hardening function, which describes the expansion of the yield surface. Some common approaches for $r(\bar{\varepsilon}^{\mathrm{p}})$ would be a linear hardening function (cp. fig. 1.11)

$$dr\left(\bar{\varepsilon}^{\mathrm{p}}\right) = h \; d\bar{\varepsilon}^{\mathrm{p}} \qquad r\left(\bar{\varepsilon}^{\mathrm{p}}\right) = h \; \bar{\varepsilon}^{\mathrm{p}} \qquad , \tag{1.71}$$

or an exponential hardening function with saturation (cp. fig. 1.12).

$$dr\left(\bar{\varepsilon}^{\mathrm{p}}\right) = b\left(Q - r\right) d\bar{\varepsilon}^{\mathrm{p}} \qquad r\left(\bar{\varepsilon}^{\mathrm{p}}\right) = Q\left(1 - \mathrm{e}^{-b\bar{\varepsilon}^{\mathrm{p}}}\right) \qquad , \tag{1.72}$$

h is a constant, Q is the saturated value of r which leads to a stress threshold of $\sigma_{\mathrm{sat}} = \sigma_{y0} + Q$ and the constant b is the rate at which the size of the yield surface changes. The initial condition for the integration is $r(0) = 0$.
The consistency condition can be written as

$$df = \frac{\partial f}{\partial \boldsymbol{\sigma}} : d\boldsymbol{\sigma} + \frac{\partial f}{\partial \bar{\varepsilon}^{\mathrm{p}}} \; d\bar{\varepsilon}^{\mathrm{p}} \tag{1.73}$$

The equivalent strain $\bar{\varepsilon}^{\mathrm{p}}$, eqn. (1.55), can also be written as

$$\bar{\varepsilon}^{\mathrm{p}} = \left(\frac{2}{3} d\lambda \frac{\partial f}{\partial \boldsymbol{\sigma}} : d\lambda \frac{\partial f}{\partial \boldsymbol{\sigma}}\right)^{\frac{1}{2}} \tag{1.74}$$

In analogy to section 1.2.3 the relationship between the stress and strain increments, $d\boldsymbol{\sigma}$ and $d\boldsymbol{\varepsilon}$, can be found with Hooke's law (1.66), the consistency condition (1.73) and equation (1.74) for the effective strain $\bar{\varepsilon}^{\mathrm{p}}$, [1], [9], [6].

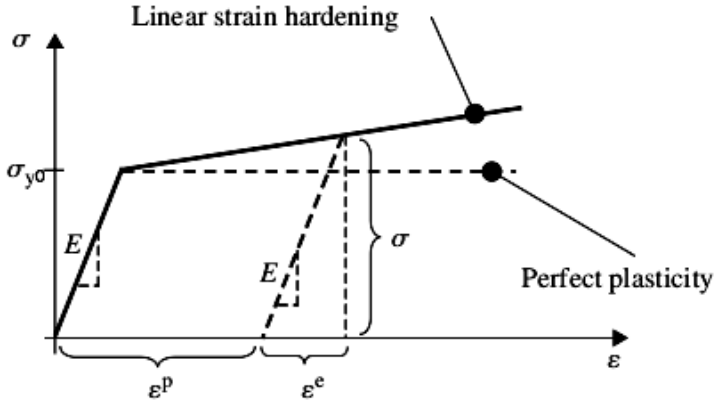

Fig. 1.11: *Isotropic hardening with a linear hardening function, an initial yield stress σ_{y0}, the decomposition of the strain into elastic and plastic components ε^e and ε^{p} and the Young's modulus E [9]*

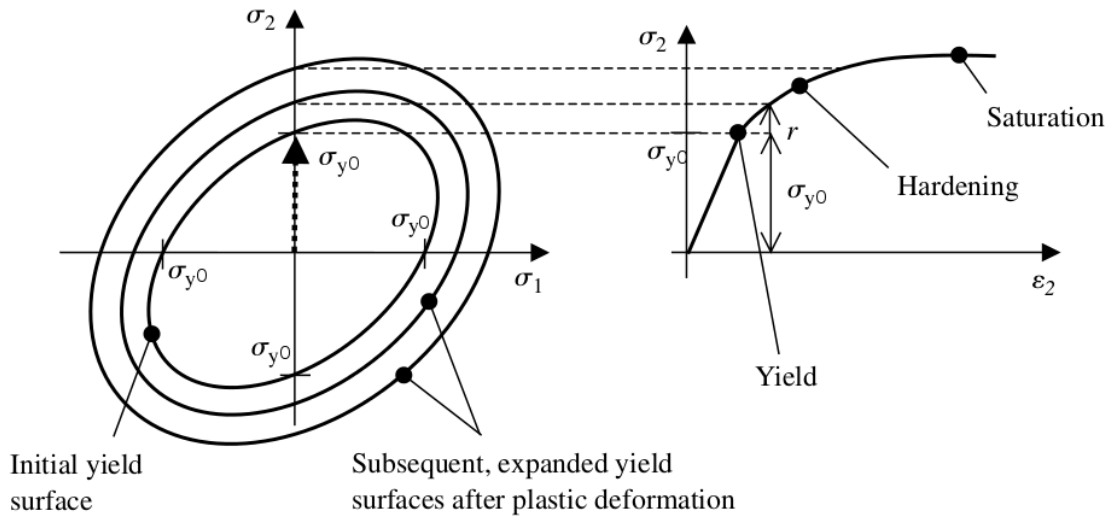

Fig. 1.12: *Isotropic hardening with an exponential hardening function r with saturation and the initial yield stress σ_{y0} [9]*

1.2.5 Kinematic Hardening

Let's assume that a material with isotropic hardening deforms plastically under tensile loading until the load is reversed. The material deforms elastically until the load point reaches the yield surface again. Due to the expansion of the yield surface during the isotropic hardening of the material the elastic region increases (cp. fig. 1.13 top). However, experiments show that this load case doesn't usually result in an increased elastic region for metals. To avoid the expansion of the elastic region the yield surface is translated in the stress space, which is called kinematic hardening. The resulting increase of yield strength in the direction of plastic flow and its decrease in the opposite direction is also referred to as Bauschinger effect (cp. fig. 1.13 bottom).
The yield function with the von Mises yield criterion can be written as

$$f\left(\boldsymbol{\sigma}-\boldsymbol{\alpha}\right)=\left[\frac{3}{2}\left(\mathbf{S}-\boldsymbol{\alpha}^{\mathrm{dev}}\right):\left(\mathbf{S}-\boldsymbol{\alpha}^{\mathrm{dev}}\right)\right]^{\frac{1}{2}}-\sigma_{y0} \qquad , \tag{1.75}$$

with the so-called back stress $\boldsymbol{\alpha}$, which describes the translation in the stress space and its deviatoric part $\boldsymbol{\alpha}^{\mathrm{dev}}$. For our following calculation we will use the evolution equation of the kinematic hardening component defined by ABAQUS as

$$d\boldsymbol{\alpha}_k = C_k\frac{\left(\boldsymbol{\sigma}-\boldsymbol{\alpha}\right)}{\sigma_{y0}}\;d\bar{\varepsilon}^{\mathrm{p}}-\gamma_k\boldsymbol{\alpha}_k\;d\bar{\varepsilon}^{\mathrm{p}} \tag{1.76}$$

$$\boldsymbol{\alpha}=\sum_{k=1}^{N}\boldsymbol{\alpha}_k \tag{1.77}$$

This gives the option to define and superpose several backstresses $\boldsymbol{\alpha}_k$ for complex kinematic material behaviour. The first term in equation (1.76) is the linear Ziegler hardening law, a purely kinematic term with the initial kinematic hardening moduli C_k and the second additive term is the recall term which describes relaxation with the rate γ_k at which the kinematic hardening moduli decreases. With the consistency condition

$$df=\frac{\partial f}{\partial\boldsymbol{\sigma}}:d\boldsymbol{\sigma}+\frac{\partial f}{\partial\boldsymbol{\alpha}}:d\boldsymbol{\alpha} \qquad , \tag{1.78}$$

the relationship between stress and strain increment can be found in analogy to sections 1.2.3 and 1.2.4, [1], [9], [6].

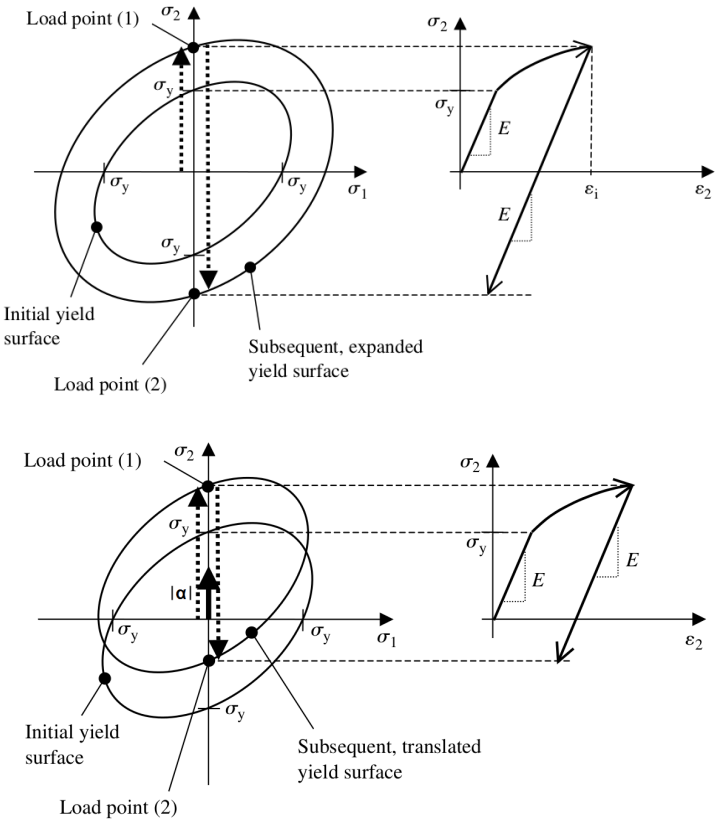

Fig. 1.13: top: Material behaviour of an isotropically hardening material with increase of the elastic region during reverse loading. bottom: Bauschinger effect of a kinematically hardening material with the backstress $\boldsymbol{\alpha}$ results in a smaller elastic region during unloading [9]

1.2.6 Mixed Mode Hardening

For the material behaviour of metals it is sometimes useful to combine isotropic and kinematic hardening behaviour. This results in the following yield function and consistency condition

$$f\left(\boldsymbol{\sigma}-\boldsymbol{\alpha},\bar{\varepsilon}^{\mathrm{p}}\right)=\sigma_{\mathrm{e}}\left(\boldsymbol{\sigma}-\boldsymbol{\alpha}\right)-r\left(\bar{\varepsilon}^{\mathrm{p}}\right)-\sigma_{y0} \tag{1.79}$$

$$df=\frac{\partial f}{\partial\boldsymbol{\sigma}}:d\boldsymbol{\sigma}+\frac{\partial f}{\partial\boldsymbol{\alpha}}:d\boldsymbol{\alpha}+\frac{\partial f}{\partial\bar{\varepsilon}^{\mathrm{p}}}\,d\bar{\varepsilon}^{\mathrm{p}} \qquad , \tag{1.80}$$

[9], [6].

1.2.7 Shakedown and Ratchetting

If a solid yields under some variable load, such as plastic cyclic loading, two alternative responses may be observed over time. One is that the plastic strain increases with every cycle until the material fails. This phenomenon is called ratchetting.
On the other hand, the plastic strain may saturate towards a constant c

$$\lim_{t\to\infty} \boldsymbol{\varepsilon}^{\mathrm{p}}(t) = c \qquad , \tag{1.81}$$

which means that after a time t_1 no further plastic strain is accumulated and the behaviour of the material becomes purely elastic. This is referred to as shakedown and occurs if the strain amplitude is small enough or the hardening effect is strong enough.

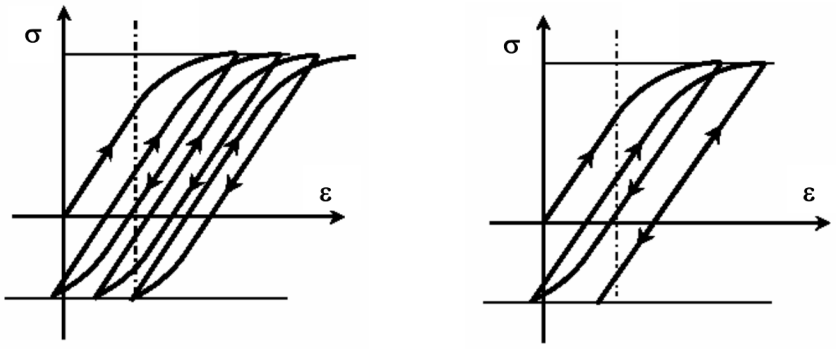

Fig. 1.14: left: Ratchetting (increase of the plastic strain during every load cycle), right: Shakedown (plastic strain saturates after a number of load cycles and the material behaviour becomes purely elastic) [20]

To describe shakedown we introduce the relationship

$$\boldsymbol{\sigma}(\mathbf{X},t) = \boldsymbol{\sigma}^{\mathrm{el}}(\mathbf{X},t) + \boldsymbol{\sigma}^{\mathrm{r}}(\mathbf{X},t) \tag{1.82}$$

The actual stress tensor $\boldsymbol{\sigma}$ is split into a fictitious elastic stress tensor $\boldsymbol{\sigma}^{\mathrm{el}}$, which depicts the stress response of a purely elastic material under the actual load, and a self-stress or residual stress tensor $\boldsymbol{\sigma}^{\mathrm{r}}$.
For perfect plasticity the classical static theorem or Melan's theorem [19] states, that shakedown occurs if there exist a residual stress field $\boldsymbol{\sigma}^{\mathrm{r}}(\mathbf{X},t)$, a safety coefficient

$k > 1$ and a time t_1 for which the yield function is

$$f\left(k\boldsymbol{\sigma}^{\mathrm{r}}(\mathbf{X},t)+\boldsymbol{\sigma}^{\mathrm{el}}(\mathbf{X},t),\sigma_{\mathrm{y}}\right)\leq 0 \qquad \forall\mathbf{X},\forall t > t_1 \tag{1.83}$$

If we assume that $t > t_1$ and therefore shakedown has already occurred the plastic strain field $\boldsymbol{\varepsilon}^{\mathrm{p}}$ is constant over time resulting in a residual stress field $\bar{\boldsymbol{\sigma}}^{\mathrm{r}}(\mathbf{X})$, which is independent of time. If we use the yield function defined in eqn. (1.56) we get

$$\sigma_{\mathrm{e}}\left(k\bar{\boldsymbol{\sigma}}^{\mathrm{r}}+\boldsymbol{\sigma}^{\mathrm{el}}\right)-\sigma_{\mathrm{y}}\leq 0 \tag{1.84}$$

Since a purely isotropic hardening material cannot account for the Bauschinger effect, we consider kinematic or mixed hardening for further shakedown theorems. We can adopt the static shakedown theorem for kinematic hardening and the yield function (1.56) by simply including the backstress $\boldsymbol{\alpha}(\mathbf{X},t)$

$$\sigma_{e}\left(k\bar{\boldsymbol{\sigma}}^{\mathrm{r}}+\boldsymbol{\sigma}^{\mathrm{el}}-\boldsymbol{\alpha}\right)-\sigma_{\mathrm{y}}\leq 0 \tag{1.85}$$

If the kinematic hardening has an upper stress limit σ_{sat} we can either use equation (1.85) or

$$\sigma_{e}\left(k\bar{\boldsymbol{\sigma}}^{\mathrm{r}}+\boldsymbol{\sigma}^{\mathrm{el}}\right)-\sigma_{\mathrm{sat}}\leq 0 \tag{1.86}$$

$$\sigma_{e}\left(\boldsymbol{\alpha}\right)-\left(\sigma_{\mathrm{sat}}-\sigma_{\mathrm{y}}\right)\leq 0 \tag{1.87}$$

For mixed hardening the yield stress σ_{y} also depends on the accumulated plastic strain $\bar{\varepsilon}^{\mathrm{p}}$.

For some applications also a kinematic shakedown theorem or Koiter's thoerem [13] is used. It states that shakedown occurs if the dissipation $\boldsymbol{\sigma}:\dot{\boldsymbol{\varepsilon}}^{\mathrm{p}}$ of a solid of the volume V exceeds the fictitious elastic dissipation $\boldsymbol{\sigma}^{\mathrm{el}}:\dot{\boldsymbol{\varepsilon}}^{\mathrm{p}}$

$$k\int_0^{\mathrm{T}}dt\int_V\boldsymbol{\sigma}^{\mathrm{el}}:\dot{\boldsymbol{\varepsilon}}^{\mathrm{p}}dV\leq\int_0^{\mathrm{T}}dt\int_V\boldsymbol{\sigma}:\dot{\boldsymbol{\varepsilon}}^{\mathrm{p}}dV \tag{1.88}$$

Again a safety coefficient $k > 1$ is established. The kinematic theorem can also be extended to work hardening materials. In some applications a combination of the static theorem for the lower bound and the kinematic theorem for the upper bound of shakedown is used, [20], [21], [8], [14], [7].

1.3 Damage and Material Failure

The phenomenon of damage can be described on different length scales. On the microscopic scale we can observe two main damage mechanisms in metals. During cold deformation decohesion between inclusions or second-phase particles and the matrix occurs, which is also called ductile fracture. Since dislocations wander through the lattice structure of metals during plastic deformation, they can be stopped at microdefects, which results in the growth of microcracks or microvoids. During deformation at temperatures greater than about half the melting temperature, the main cause of damage development is intergranular damage.
On the mesoscale we introduce a Representative Volume Element (RVE) for which the damage discontinuities, like microcracks and microvoids, are small enough to be continuously distributed within the material. For metal its size would be about 0.1mm^3. On this scale the constitutive equations for the damage continuum mechanics are formulated.
The macroscale is the scale of the engineering structures. The limit of damage continuum mechanics is reached, when macrocracks appear. To describe the growth of macrocracks and structural failure the theory of fracture mechanics is applied. In fracture mechanics the macrocracks are treated as discontinuities with stress singularities at the crack tips.
In this chapter we will focus on the material model for damage and material failure provided by ABAQUS, [15], [16], [28].

1.3.1 Damage Initiation

In the ABAQUS manual [1] the equivalent plastic strain at damage initiation $\bar{\varepsilon}^{\mathrm{p}} = \bar{\varepsilon}^{\mathrm{p}}_{D0}$ is assumed to be a function of stress triaxiality μ and equivalent plastic strain rate $\dot{\bar{\varepsilon}}^{\mathrm{p}}$, which can be found from experiments, for a crystalline metal. The stress triaxiality is defined as the negative ratio of the hydrostatic pressure q and the von Mises equivalent stress σ_{e}

$$\mu = -\frac{q}{\sigma_{\mathrm{e}}} \tag{1.89}$$

We can now define a damage indicator w_{D} for the onset of damage

$$w_{\mathrm{D}} = \int \frac{d\bar{\varepsilon}^{\mathrm{p}}}{\bar{\varepsilon}^{\mathrm{p}}_{D0}(\mu, \dot{\bar{\varepsilon}}^{\mathrm{p}})} \qquad , \tag{1.90}$$

which increases monotonically with plastic deformation ($0 \leq w_{\mathrm{D}} < 1$ no damage) and meets the threshold for damage initiation if $w_{\mathrm{D}} = 1$, [1].

1.3.2 Damage Variables and Effective Stress

To describe damage evolution we have to establish a variable that captures the state of damage during the loading process. Since ductile damage is the result of microcracks and microcavities it is reasonable to define a variable comparing the area of planes intersecting the RVE, δA, with the effective area of all microcracks and microcavities δA_{D} on this intersection. For a plane with the normal $\mathbf{n}$ the damage variable would be

$$D(\mathbf{n}) = \frac{\delta A_{\mathrm{D}}}{\delta A} \tag{1.91}$$

For isotropic damage the dependence on the normal vector $\mathbf{n}$ disappears and we get with the effective undamaged area $A^* = A - A_{\mathrm{D}}$ (for simplicity we drop δ)

$$D = \frac{A_{\mathrm{D}}}{A} = 1 - \frac{A^*}{A} \tag{1.92}$$

The effective stress in a damaged case can be deduced from the the damage variable D. Since the internal forces acting on a damaged section must equal the forces on the undamaged section we get

$$A\ \mathbf{n}\cdot\boldsymbol{\sigma} = A^*\ \mathbf{n}\cdot\sigma^* = (1-D)\ A\ \mathbf{n}\cdot\boldsymbol{\sigma}^* \qquad , \tag{1.93}$$

which results in the effective stress $\boldsymbol{\sigma}^*$

$$\boldsymbol{\sigma}^* = \frac{\boldsymbol{\sigma}}{1-D} \tag{1.94}$$

In addition to the definition of the damage variable D in eqn. (1.92) other equations have been established to estimate the evolution of the damage variable. ABAQUS uses an approach for the evolution of the ductile damage variable, considering the equivalent plastic strain (cp. eqn. (1.55)). The equivalent plastic strain at failure $\bar{\varepsilon}^{\mathrm{p}} = \bar{\varepsilon}^{\mathrm{p}}_{\mathrm{f}}$ is dependent on the characteristic element length L. This leads to a mesh dependency of the result, such that a refined mesh results in a decrease of dissipated energy. To reduce the mesh dependency the equivalent plastic displacement $\bar{u}^{\mathrm{p}}_{\mathrm{D}}$ is used instead of equivalent plastic strain. Once the damage initiation criterion $w_{\mathrm{D}} = 1$ and $\bar{\varepsilon}^{\mathrm{p}} = \bar{\varepsilon}^{\mathrm{p}}_{\mathrm{D0}}$ have been reached the equation

$$\dot{\bar{u}}^{\mathrm{p}}_{\mathrm{D}} = L\ \dot{\bar{\varepsilon}}^{\mathrm{p}}_{\mathrm{D}} \qquad , \tag{1.95}$$

is used, where the index D stands for damage, and D0 for the onset of damage. Now the damage variable can be defined directly as a function of the equivalent plastic displacement $\bar{u}^{\mathrm{p}}_{\mathrm{D}}$ (with $\bar{u}^{\mathrm{p}}_{\mathrm{f}}$ at failure) or in term of the dissipated fracture energy per unit area G_{f}

$$G_{\mathrm{f}} = \int_{\bar{\varepsilon}^{\mathrm{p}}_{D0}}^{\bar{\varepsilon}^{\mathrm{p}}_{\mathrm{f}}} L\sigma_{\mathrm{y}}\, d\bar{\varepsilon}^{\mathrm{p}} = \int_{0}^{\bar{u}^{\mathrm{p}}_{\mathrm{f}}} \sigma_{\mathrm{y}}\, d\bar{u}^{\mathrm{p}} \tag{1.96}$$

In this thesis an exponential evolution law of the damage variable is used, which is based on the dissipated fracture energy

$$D = 1 - \exp\left(- \int_{0}^{\bar{u}^{\mathrm{p}}_{\mathrm{D}}} \frac{\sigma^{*}_{\mathrm{y}}\, \dot{\bar{u}}^{\mathrm{p}}_{\mathrm{D}}}{G_{\mathrm{f}}} \right) \qquad , \tag{1.97}$$

where σ^{*}_{y} denotes the effective yield stress (cp. eqn. (1.94)), [1], [16],[28], [15].

1.3.3 Hypothesis of Strain Equivalence for Elastic and Plastic Material Behaviour

As we can see in fig. 1.1 the load increases more slowly with respect to the applied displacement than before in region IV and eventually decreases in regions V and VI, though the displacements keep increasing. Moreover, if we decrease the load once damage initiation has occurred, the Young's modulus for elastic behaviour has a smaller value. These two phenomena are known as softening and degradation of elasticity and can be described with the hypothesis of strain equivalence (cp. fig. 1.15). According to [28] we can assume that *" [...] the strain associated with a damaged state under the applied stress is equivalent to the strain associated with its fictitious undamaged state under the effective stress."*
This leads to the equations

$$\begin{aligned} \boldsymbol{\sigma} &= \mathbb{C}^{*}\, \boldsymbol{\varepsilon} \\ \boldsymbol{\sigma}^{*} &= \mathbb{C}\, \boldsymbol{\varepsilon} \end{aligned} \tag{1.98}$$

Using eqn. (1.94) we get

$$\mathbb{C}^{*} = (1 - D)\, \mathbb{C} \qquad , \tag{1.99}$$

for the effective material elasticity tensor $\mathbb{C}^{*}$. We can now write for the effective Young's modulus E^{*} and the effective Poisson ratio ν^{*}

$$E^{*} = (1 - D)\, E, \quad \nu^{*} = \nu \tag{1.100}$$

In analogy to the hypothesis of strain equivalence other hypotheses have been formulated, like the hypotheses of stress, elastic strain energy or complementary energy equivalence. One should be aware that for the elastic energy equivalence hypotheses the degradation of the Young's modulus follows a quadratic law $E^* = (1-D)^2 E$ and the Poisson ratio equals the effective Poisson ratio as before.
For plasticity with mixed hardening we can derive the yield function f for a damaged state from eqn. (1.79) and the effective stress (1.94) as

$$f\left(\boldsymbol{\sigma}^* - \boldsymbol{\alpha}, \bar{\varepsilon}^{\mathrm{p}}\right) = \sigma_{\mathrm{e}}\left(\frac{\boldsymbol{\sigma}}{1-D} - \boldsymbol{\alpha}\right) - r\left(\bar{\varepsilon}^{\mathrm{p}}\right) - \sigma_{y0} \qquad , \tag{1.101}$$

[1], [16],[28], [15].

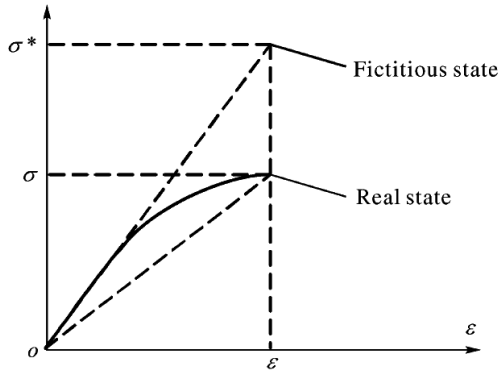

Fig. 1.15: *Hypothesis of strain equivalence: the strain ε in a damaged state under the stress σ and in an undamaged fictitious state under the effective stress σ^* is equal [28]*

1.3.4 Material Failure

Material failure for ductile materials occurs, when the damage variable reaches a value of $D = 1$. This can be interpreted as a crack occupying the whole surface of the volume element, which means the effective undamaged area $A^* = 0$. However, for many materials the critical damage variable is set from $D = 0.2$ to 0.5 and for brittle fracture to $D = 0$. The critical value of the damage variable can be estimated experimentally in a pure monotonic tensile test with the relation

$$D_{1\mathrm{c}} = 1 - \frac{\sigma_{\mathrm{f}}}{\sigma_{\max}} \qquad , \tag{1.102}$$

and is called corresponding critical damage $D_{1\mathrm{c}}$. σ_{f} is the stress at material failure and $\sigma_{\max}$ is the maximum stress value.

In ABAQUS a solid element is removed from the mesh if its damage variable reaches its critical value. However, it is also possible to keep the element throughout the simulation, its residual stiffness being set to at least 1% of its original stiffness to avoid numerical issues, [1], [15].

2 Artificial Neural Networks

The original idea of Artificial Neural Networks (ANN) was to design a model capable of simulating the information processing of the nervous system. An ANN is basically a nonlinear mapping system consisting of connected processing units. The modelling of a system with a behaviour close to the real process in the nervous system is quite complex and requires a high number of units. In this thesis a simpler approach will be used. The goal is to design an ANN capable of learning a specified material behaviour by training it with a data set of corresponding inputs and outputs, [24], [26].

2.1 Weighted Networks

Every ANN consists of a number of processing units, so called neurons (cp. fig. 2.1). Each neuron j has a number of inputs x_i which are multiplied with a weight w_{ji}, where the first index indicates the number of the neuron an the second index the number of the input. Often an internal bias b_j is introduced, by adding a channel with a constant input value x_0 leading to $b_j = w_{j0}x_0$. The inputs are evaluated with a suitable function g_{in} to obtain a scalar value. Usually g_{in} is simply the addition function

$$g_{\text{in}}(w_{j1}x_{j1}, ..., w_{jm}x_{jm}, b_j) = \sum_{i=1}^{m} w_{ji}x_i + b_j = \sum_{i=0}^{m} w_{ji}x_i \tag{2.1}$$

The result of the integration function g_{in} is the induced local field v_j, which is evaluated with a non-linear transfer function f_{trans} to get the final output y_j of the neuron j.

$$y_j = f_{\text{trans}}(\sum_{i=0}^{m} w_{ji}x_i) = f_{\text{trans}}(v_j) \tag{2.2}$$

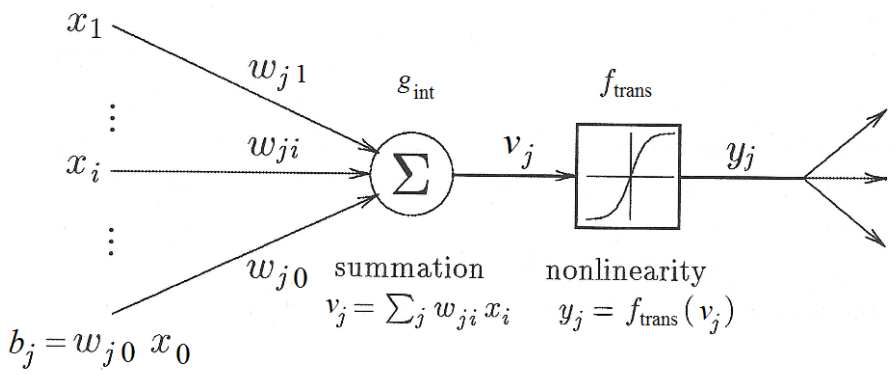

Fig. 2.1: In the neuron j the inputs x_i, multiplied with weights w_{ji}, and the bias b_j, represented by a constant input x_0, are reduced with the integration function g_{in} to a scalar value, the induced field v_j, and passed through the transfer function f_{trans} to obtain the output y_j [24]

Some commonly used transfer functions would be the logistic sigmoid function $f_{\log}$, the tan-sigmoid function $f_{\tan}$ or a linear funtion f_{lin}

$$f_{\log}(v_j) = \frac{1}{1+\mathrm{e}^{-cv_j}} \tag{2.3}$$

$$f_{\tan}(v_j) = \frac{\mathrm{e}^{cv_j} - \mathrm{e}^{-cv_j}}{\mathrm{e}^{cv_j} + \mathrm{e}^{-cv_j}} \tag{2.4}$$

$$f_{\text{lin}}(v_j) = cv_j + d \quad , \tag{2.5}$$

where c and d are constants. In fig. 2.2 the above three transfer functions are depicted with different values of c [5], [11], [24], [26].

An ANN consists of many connected neurons. The architecture of the ANN and the values of the weights and biases are responsible for correctly implementing different functions. Furthermore the connection of the neurons is linked to the used learning algorithms, which will be discussed in the next section.
The structure of ANN can be arbitrary, but a layered structure is conventionally used, where the neurons are organized in different connected layers. The most common structures are single-layer and multi-layer feedforward or recurrent networks.
A single-layer network consists of an input and an output layer, while a multi-layer network has additional layers between the input and output layers, which are called hidden layers (cp. fig. 2.3).
In a feedforward network the inputs of a layer are equal to the outputs of the layer in front of it. This means that there are no feedback loops, which would allow to

feed outputs of a layer back as inputs for neurons of the same layer or layers before. Networks with feedback loops are referred to as recurrent networks. They can have one or several feedback loops. If all nodes of a layer are connected to all nodes of the next layer the network is fully connected, otherwise it is called partially connected, [11], [24].

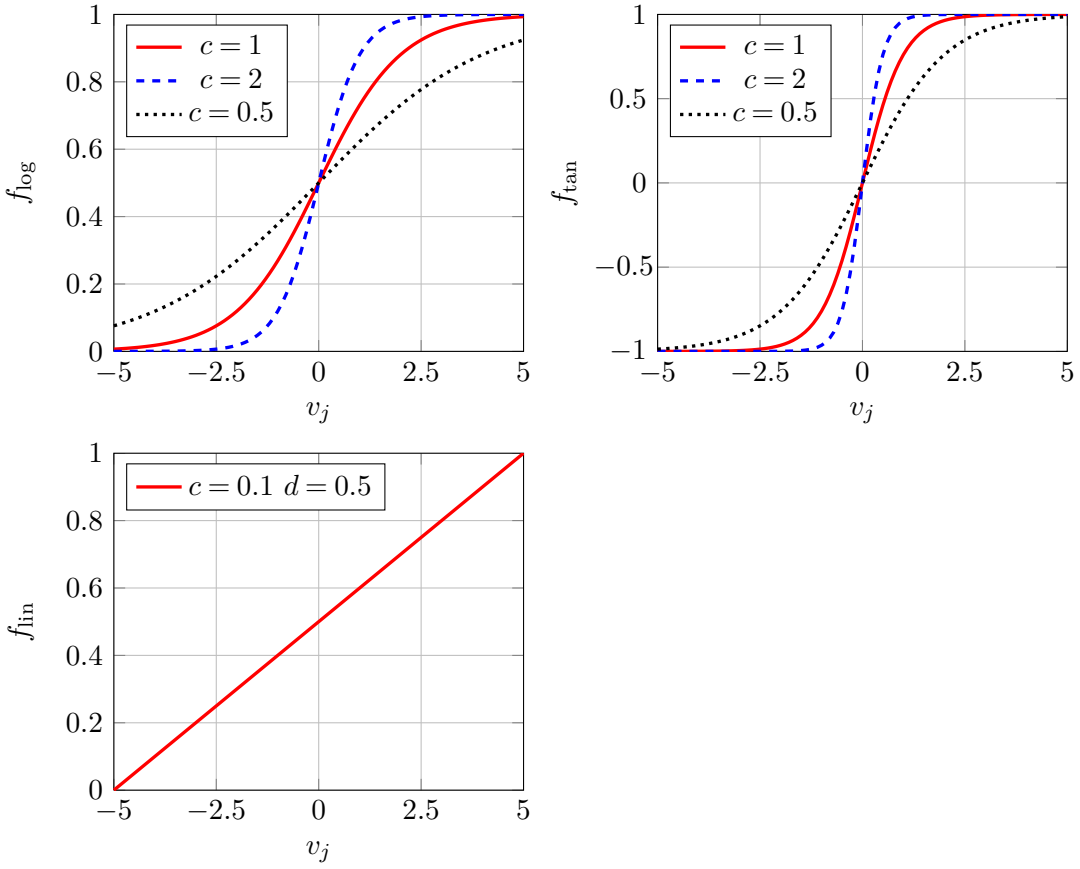

Fig. 2.2: Transfer functions: logistic sigmoid function $f_{\log}$, tan-sigmoid function $f_{\tan}$ and the purely linear function f_{lin} with different constants

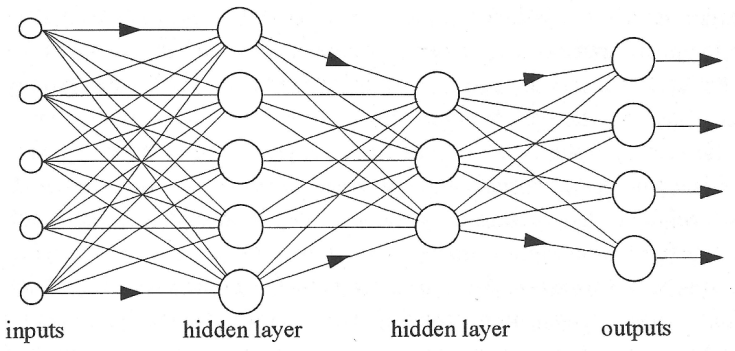

Fig. 2.3: *Multi-layered, fully connected feedforward network with an input, two hidden and an output layer. [24]*

2.2 Learning Algorithms

To find the proper weights and biases of the ANN a learning algorithm must be used. In general one can distinguish between supervised and unsupervised learning. During supervised learning the training data consists of inputs x_i and desired outputs or targets o. We can now apply an error-correction learning rule, which compares the actual computed outputs y with the desired targets o by estimating the error with the help of an appropriate objective function. Some common objective function are the sum of the squared error $\mathrm{E_{SSE}}$, the mean squared error $\mathrm{E_{MSE}}$ and the logarithmic error function $\mathrm{E_{LOG}}$

$$\mathrm{E_{SSE}} = \sum_{p=1}^{P}\sum_{j=1}^{M}(o_{pj} - y_{pj})^2 \tag{2.6}$$

$$\mathrm{E_{MSE}} = \frac{\mathrm{E_{SSE}}}{PM} \tag{2.7}$$

$$\mathrm{E_{LOG}} = \sum_{p=1}^{P}\sum_{j=1}^{M} o_{pj}\ln y_{pj} + (1-o_{pj})\ln(1-y_{pj}) \quad , \tag{2.8}$$

where p is the index and P the number of training patterns and j is the index and M the number of the output nodes. A control mechanism is activated to reduce the error by adjusting the weights and biases between training cycles, called epochs, to minimise the objective function.
During unsupervised learning there are no target values. Special learning rules are used. In this thesis we will only use supervised learning algorithms, [5], [11], [24].

2.2.1 The Back Propagation Algorithm

A common algorithm for training multilayered feedforward networks is back propagation. This algorithm consists of two passes. The first pass is called forward propagation, during which the ANN simply computes the outputs according to its current inputs following eqn. (2.2). Now we can compare the results with the desired outputs and calculate the error with an appropriate objective function. We use the sum of the squared error function $\mathrm{E} = \mathrm{E}_{\mathrm{SSE}}$ and write eqn. (2.6) for a single pattern p as

$$\mathrm{E}(p) = \frac{1}{2}\sum_j (o_j(p) - y_j(p))^2 = \frac{1}{2}\sum_j e_j^2(p) \qquad , \tag{2.9}$$

where the factor $1/2$ cancels out a factor 2 in eqn. (2.12) and (2.19), j stands for the neurons of the output layer, and $e_j = o_j - y_j$ is the error between the targets o_j and computed outputs y_j.

During the second pass, called back propagation, the weights are adjusted to minimize the calculated error $\mathrm{E}(p)$. To do so it is necessary to compute the derivative of the error with respect to the weights

$$\frac{\partial \mathrm{E}(p)}{\partial w_{ji}(p)} = \underbrace{\frac{\partial \mathrm{E}(p)}{\partial v_j(p)}}_{\delta_j(p)} \underbrace{\frac{\partial v_j(p)}{\partial w_{ji}(p)}}_{x_i(p)=y_i(p)} \tag{2.10}$$

$$\frac{\partial \mathrm{E}(p)}{\partial w_{ji}(p)} = \delta_j(p)\; y_i(p) \qquad , \tag{2.11}$$

where δ_j is the local gradient and x_i are the inputs of node j, which equal the outputs y_i of node i.

We now have to distinguish between a node in the output or in a hidden layer. For an output node its error e_j is determined with the given target values o_j and we can write

$$\frac{\partial \mathrm{E}(p)}{\partial v_j(p)} = \underbrace{\frac{\partial \mathrm{E}(p)}{\partial e_j(p)}}_{e_j(p)} \underbrace{\frac{\partial e_j(p)}{\partial y_j(p)}}_{-1} \underbrace{\frac{\partial y_j(p)}{\partial v_j(p)}}_{\frac{\partial f_{\mathrm{trans}}}{\partial v_j} = f'_{\mathrm{trans}}(v_j)} \tag{2.12}$$

$$\delta_j(p) = -e_j(p)\; f'_{\mathrm{trans}}(v_j(p)) \tag{2.13}$$

For a hidden node we have to consider that the error e_j has to be determined recursively. It depends on all nodes k to which it feeds information. We can write

$$\frac{\partial \mathrm{E}(p)}{\partial v_j(p)} = \frac{\partial \mathrm{E}(p)}{\partial y_j(p)} \underbrace{\frac{\partial y_j(p)}{\partial v_j(p)}}_{f'_{\mathrm{trans}}(v_j)} \tag{2.14}$$

$$\frac{\partial \mathrm{E}(p)}{\partial y_j(p)} = \underbrace{\sum_k e_k(p) \underbrace{\frac{\partial e_k(p)}{\partial v_k(p)}}_{-f'_{\mathrm{trans}}(v_k)}}_{\delta_k} \underbrace{\frac{\partial v_k(p)}{\partial y_j(p)}}_{w_{kj}} \tag{2.15}$$

$$\delta_j(p) = f'_{\mathrm{trans}}(v_j(p)) \sum_k \delta_k(p) w_{kj}(p) \tag{2.16}$$

We can see now that the second pass starts at the output layer of the ANN and recursively determines the derivative $\frac{\partial \mathrm{E}}{\partial w_{ji}}$ for every node of the different layers, [11], [24].

To minimize the error E over a number of epochs, we have to correct the weights by adding the correction term $\Delta w_{ji}(p)$

$$w_{ji,\mathrm{new}} = w_{ji,\mathrm{old}} + \Delta w_{ji} \tag{2.17}$$

$$\Delta w_{ji}(p) = -\eta \frac{\partial \mathrm{E}(p)}{\partial w_{ji}(p)} = -\eta \ \delta_j(p) \ y_i(p) \qquad , \tag{2.18}$$

where η is the learning rate which represents the gradient descent in weight space and usually ranges between $0.05 < \eta < 0.75$. This method is also called steepest descent algorithm, [11], [24].

MATLAB [18] offers a number of different training functions for the weight update. Since we achieved the best results with the Levenberg-Marquardt algorithm [18], we will only discuss this algorithm.

The gradient $g_j = \frac{\partial E(p)}{\partial w_{ij}(p)}$ can also be written as (cp. eqn. (2.9))

$$g_j = \frac{\partial E(p)}{\partial w_{ji}(p)} = \sum e_j(p) \frac{\partial e_j(p)}{\partial w_{ji}(p)} \tag{2.19}$$

$$\mathbf{g} = \mathbf{W}^{\mathrm{T}} \ \mathbf{e} \qquad , \tag{2.20}$$

using the Jacobian matrix $\mathbf{W}$ of the first derivatives of the absolute errors e_j with respect to the weights w_{ji} and the vectors of the gradient $\mathbf{g}$ and of the absolute errors $\mathbf{e}$. In the Levenberg-Marquardt algorithm the vector of correction terms $\Delta\mathbf{w}$

is calculated as follows

$$\Delta \mathbf{w} = -[\mathbf{W}^{\mathrm{T}}\mathbf{W} + \theta \mathbf{I}]^{-1}\mathbf{W}^{\mathrm{T}}\mathbf{e} \tag{2.21}$$

It can be interpreted as a combination of the Gauss-Newton method ($\theta = 0$), which converges quickly near a minimum but can diverge, if the starting values are too far off the minimum, and the steepest descent algorithm (θ is large), whose convergence is slow but independent of the starting values, [24], [18].

2.2.2 Training, Testing and Validation

Let us now summarize the outline of an ANN. An ANN consists of a number of neurons, which are usually grouped in layers starting with the input layer, followed by hidden layers and ending with the output layer.
The inputs of a neuron are multiplied with weights and reduced to a scalar value, called induced field, with the help of a suitable function, which is commonly the addition function. The induced field is evaluated with a proper transfer function and results in an output, which either acts as input for another neuron or as actual output of the network.
During the training process a set of input and output data is presented to the ANN. For the applied back propagation algorithm it is necessary to evaluate the error of the computed and the target outputs. The goal of the training is to minimize the error by adjusting the weights.
Besides the architecture, the function to evaluate the inputs, the transfer and error functions and the learning algorithm some other crucial questions concern the training data, its selection and preparation, and the stopping criteria and evaluation of the ANN's performance.
Usually the provided data is split up into training, validation and test data. Only the training data is presented to the ANN to adjust the weights. The validation set is used to control the performance of the ANN during the learning process and to help establishing a proper stopping criteria, which will be discussed later. The test data is unknown to the ANN and is used to finally validate the trained ANN.
But before any data is presented to the ANN it is necessary to perform data scaling. During the training process data with large magnitude can override data with small magnitude. To prevent this from happening, data normalization is applied within a specified interval.

In advance data mining is also useful and necessary. Since the data may contain errors or may be superimposed with perturbations, its quality is a factor. Moreover the complexity of the ANN design depends on the quantity of available data and the problem to be solved. The tricky question is to find an ANN, which is complex enough, and combine it with the right set of training data, which captures all significant features of the target function and is appropriate in quality and quantity. As already mentioned the goal of the training process is to minimize the error of the computed output y with respect to the desired output o, which may be interpreted as an optimization process. However, it is possible that the back propagation converges to a local minimum instead of a global minimum (cp. fig. 2.4). One way of dealing with this problem is to train ANN with different sets of initial weights, [5],[24].

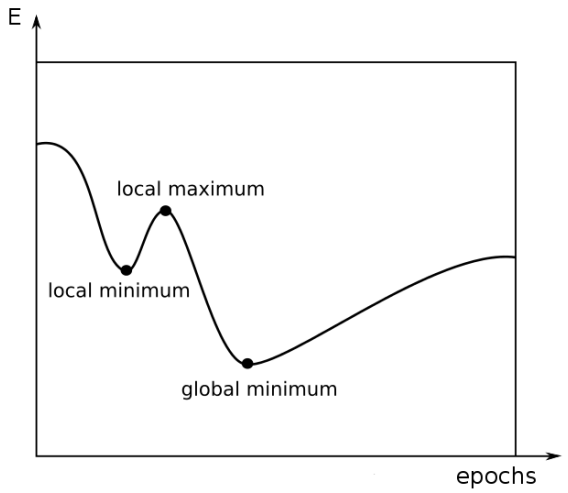

Fig. 2.4: *Example of the error E over a number of epochs showing a local minimum and maximum as well as a global minimum. [5]*

However, a minimum of the error function doesn't guarantee good generalization. This means that it is possible to find a function that minimizes the error but doesn't represent the general behaviour of the training data and therefore can't give the correct outputs for new patterns. Two possible reasons for bad generalization are over- and underfitting. Overfitting occurs if the number of hidden nodes, training data or training cycles is too high, while underfitting occurs at the exact opposite condition (cp. fig. 2.5). To avoid overfitting due to a high number of training cycles an early stopping criterion based on a validation set can be used to stop the training process (cp. fig. 2.6). The samples of the training data are split into a training set, used to modify the weights, and a validation set, to control the performance of the

ANN towards unknown data, without changing the ANN itself. After a number of training cycles the error per sample on the validation set is estimated. If the error of the validation set increases the training is stopped. Unfortunately, an increase of the validation error can also be due to surpassing a local minimum of the error space (cp. fig. 2.6). Other stopping criteria could be determined with thresholds of the error function, the number of training cycles or the calculation time, [5], [24].

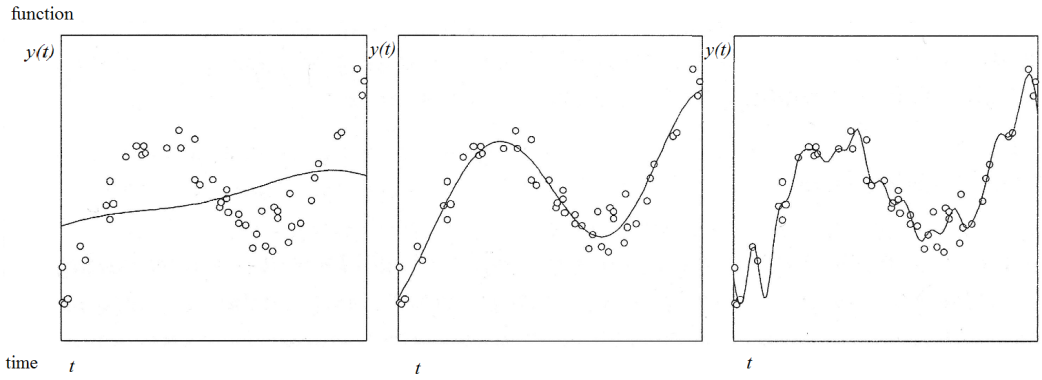

Fig. 2.5: *The dots represent the given data set and the full line the response of the ANN. left: underfitting of the desired function, middle: well fitted function, right: overfitting of the desired function [24]*

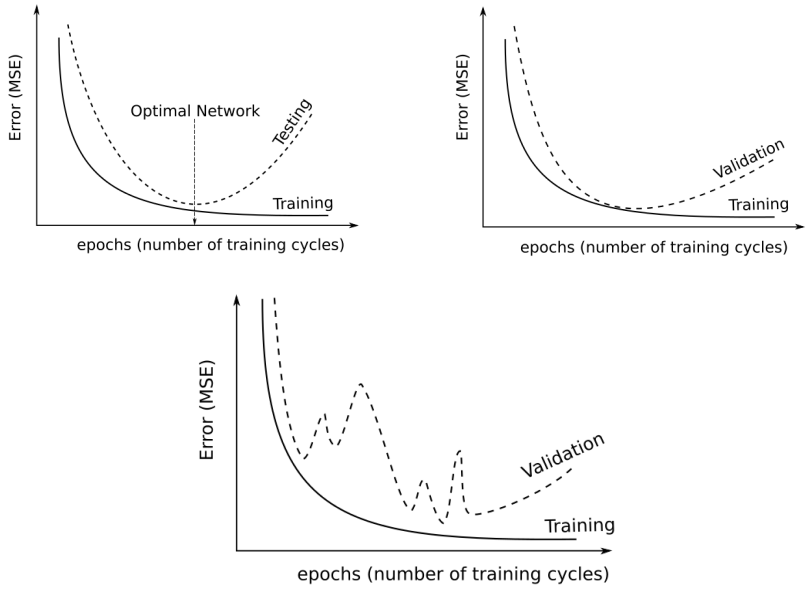

Fig. 2.6: *top left: due to overfitting the error of the testing data increases while the error of the training data decreases, top right: the training is stopped when the error of the validation data starts increasing bottom right: error of the validation data over epochs can have local minima [5]*

3 Development of a Material Model with Artificial Neural Networks

The development of ANN-based material models can be divided into several steps. First it is necessary to generate a set of data which can be used to train the ANN and evaluate its performance. For this step we use the Finite Element Method (FEM) program ABAQUS/Explicit [1].
For the training and the validation of the ANN we use the numerical computing environment MATLAB NN toolbox plus additional tools for ANN programming, Stipulator & Brainer, provided by the engineering company ANDATA [18].
In addition an implementation of the ANN material model for the FEM program ABAQUS/Explicit was coded as a VUMAT user-subroutine.

3.1 Numerical Generation of Load Cases with ABAQUS

For the training of an ANN we need a set of training data. In our case this would be the correlated time histories of the strain and stress components resulting from an ABAQUS analysis. To produce the necessary information, we use a single element model with a deterministic elasto-plastic material behaviour with damage as provided by the FEM program ABAQUS.
For the single element a three dimensional solid element (C3D8R, 8-node linear brick) with reduced integration (one integration point) and hourglass control was chosen. To avoid rigid body motion it is necessary to constrain the degrees of freedom (DOF). Therefore the following constraints were applied (cp. fig. 3.1 blue)

$$\begin{aligned} &\text{node } 1 - \text{fixed DOF}: 1,\ 2,\ 3, \quad \text{i.e.}: u_1^1 = u_2^1 = u_3^1 = 0 \\ &\text{node } 2 - \text{fixed DOF}: 2,3, \qquad \text{i.e.}: u_2^2 = u_3^2 = 0 \\ &\text{node } 4 - \text{fixed DOF}: 3, \qquad \text{i.e.}: u_3^4 = 0 \end{aligned} \tag{3.1}$$

Displacements were applied to the master nodes 2,4 and/or 5 as required (cp. fig. 3.1). The other degrees of freedom were controlled by periodicity boundary conditions, which ensure that pairs of initially parallel, homologous faces deform in a compatible way (cp. fig. 3.2). The displacements of the master nodes fully determine the displacements of the coupled slave nodes. In Fig. 3.2 the displacements $\mathbf{u}^2$ and $\mathbf{u}^4$ of the unit cell are applied at the corresponding master nodes 2 and 4, which result in the displacement $\mathbf{u}^3 = \mathbf{u}^2 + \mathbf{u}^4$ of the slave node 3. Since we use a single element with linear interpolation the surfaces will stay plane. The equations for the periodic boundary conditions of a three-dimensional single element can be written as

$$\begin{aligned} u_i^8 - u_i^5 - u_i^4 &= 0 \\ u_i^3 - u_i^2 - u_i^4 &= 0 \\ u_i^6 - u_i^2 - u_i^5 &= 0 \\ u_i^7 - u_i^2 - u_i^4 - u_i^5 &= 0 \end{aligned} \tag{3.2}$$

The displacement u_i^N denotes the displacement of the node N in the direction $i = 1,2,3$. [1], [4]

Due to the periodic coupling conditions (3.2) and the boundary conditions (3.1) only the displacement components u_1^2, u_1^4, u_2^4, u_1^5, u_2^5 and u_3^5 are potential active degrees of freedom in this system. For any of these six displacement components some time dependent behaviour $u_i^N = u_i^N(t)$ (with the special case $u_i^N = 0$) may be specified or it may be left free to be evaluated by the program. The latter case is identified as u_i^N =free.

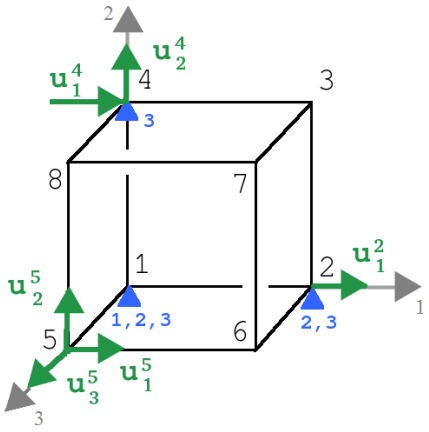

Fig. 3.1: 3-dimensional solid element (C3D8R) with node numbers (black), fixed DOF to avoid rigid body motion (blue) and applied displacements u_i^N (green)

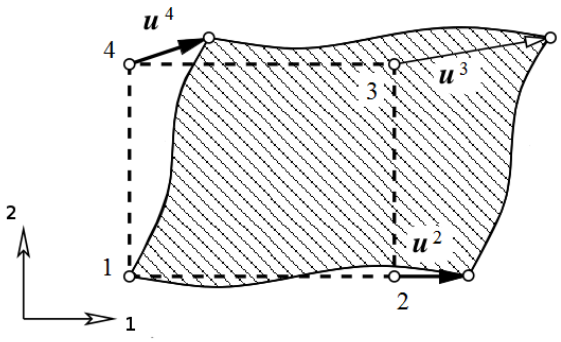

Fig. 3.2: 2-dimensional unit cell with periodic boundary conditions to ensure that parallel faces in the undeformed state deform equally. The displacement $\mathbf{u}^3$ *of the slave node 3 results from the applied displacement* $\mathbf{u}^2$ *and* $\mathbf{u}^4$ *of the master nodes 2 and 4. (dashed line: undeformed state, solid line: deformed state) [4]*

The used material behaviour combines elasticity, plasticity with mixed mode hardening properties (cp. section 1.2) and material damage resulting in final material failure (cp. section 1.3). The material parameters are chosen to approximate the behaviour of aluminium. Whereas the properties for elasticity and damage were obtained from [1], the material characteristics for plasticity were extracted from [27], the damage behaviour was modified in order to achieve a later onset of damage and a slower damage evolution.

The properties used for the elasto-plastic material behaviour are listed in tab 3.1 (cp. eqn. (1.72) and (1.76)). According to [1] a ductile damage initiation criterion is used and the onset of damage is defined in a table correlating the equivalent plastic strain at damage initiation $\bar{\varepsilon}^{\mathrm{p}}_{D0}$, the stress triaxiality μ and the equivalent plastic strain rate $\dot{\bar{\varepsilon}}^{\mathrm{p}}$ (cp. fig. 3.3). A stress triaxiality of $\mu = 1/3$ would be a uniaxial tensile load case, $\mu < 0$ would be the compressive loading region, in which no ductile failure occurs, and $\mu > 1.5$ would describe behaviour dominated by hydrostatic tensile loading.

The evolution of the damage parameter D is defined by an exponential equation (cp. eqn. 1.97). The dissipated fracture energy per unit area G_{f} is set to $1 \cdot 10^7$ J/m^2. Material failure occurs if the damage variable reaches the value $D = 1$.

Tab. 3.1: Used elasto-plastic material properties for aluminum in SI units

Density ρ	2700	$\mathrm{kg/m^3}$
Young's Modulus E	7×10^{10}	$\mathrm{N/m^2}$
Poisson ratio ν	0.33	-
Initial yield stress σ_0	1.54×10^8	$\mathrm{N/m^2}$
Max. change in the size of the yield surface Q	1.402×10^8	$\mathrm{N/m^2}$
Rate at which the yield surfaces changes b	7.094	-
initial kinematic hardening modulus C	7.019×10^9	$\mathrm{N/m^2}$
rate at which C decreases γ	118.6	-

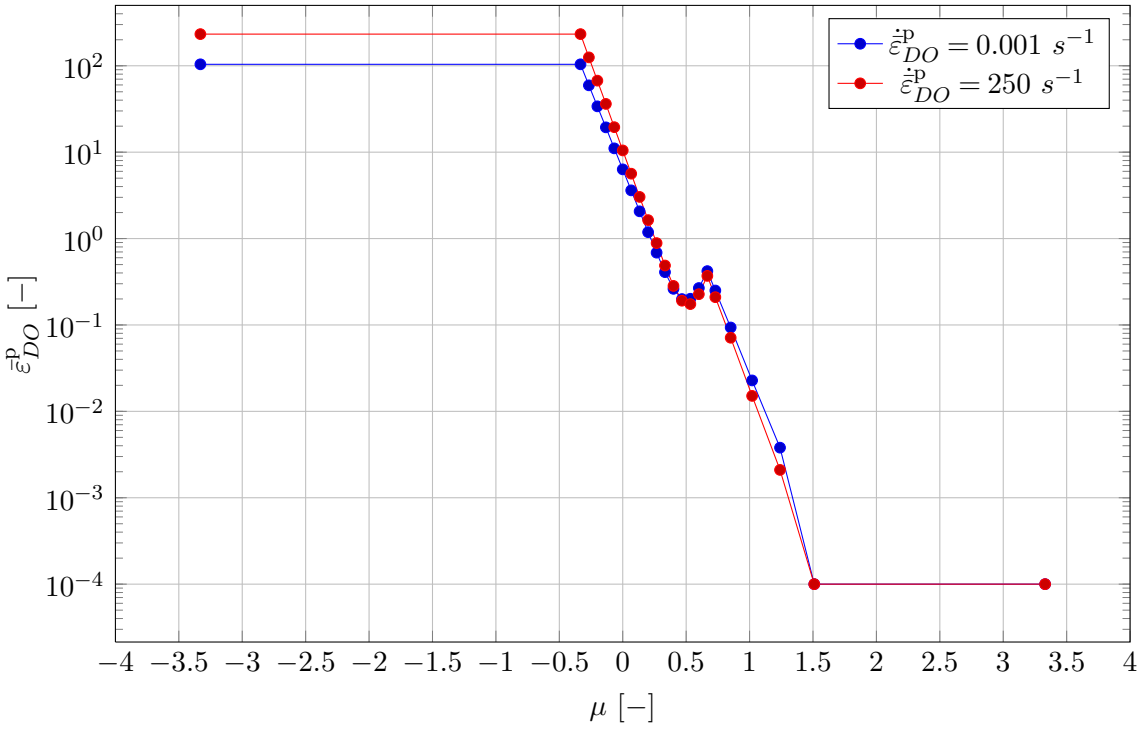

Fig. 3.3: Equivalent plastic strain at damage initiation $\bar{\varepsilon}^{\mathrm{p}}_{DO}$ as a function of stress triaxiality μ and equivalent plastic strain rate $\dot{\bar{\varepsilon}}^{\mathrm{p}}_{DO}$ [1].

Several typical load cases were applied, e.g., uniaxial and biaxial tensile and compressive loading, cyclic, "butterfly" and random load cases. In tab. 3.2 the load cases with their applied displacements, their number and the number used in the training of the ANN are listed. The resulting strain and stress components of several different load cases will be presented later.
In the tensile, compressive and hydrostatic load cases displacements were only monotonically applied in the principal directions, the active components being u_1^2, u_2^4 and/or u_3^5 (while the other possible displacement components were free). For the random load cases all six possible displacement components were prescribed and for the other load cases we differentiate between cases where all six possible displacement components, displacement components $u_1^2,\ u_2^4,\ u_1^4,\ u_2^5$ ("free") and only displacement components $u_1^2,\ u_2^4$ ("free free") were prescribed.
The cyclic load cases can be split into load cases with increasing displacement amplitudes over every cycle ("Cyclic"), with constant displacement amplitudes ("Constant Cyclic") and with constant displacement amplitudes leading to shakedown behaviour ("Shakedown"). Plotting the displacements u_1^2 over u_1^4, u_2^4 over u_2^5 or u_3^5 over u_1^5 results in straight lines, which can be defined by amplitude a and angle φ in the corresponding displacement planes (cp. fig. 3.4). We determine the displacement components by choosing these amplitudes and angles. The rate of the applied displacement components is constant and the number of cycles are chosen for a total time of one second. To achieve load cases with increasing displacement amplitudes (cp. fig. 3.5) we use the following equation

$$a(t_{n+1}) = c \cdot a(t_n) \qquad \text{with } c > 1 \tag{3.3}$$

The amplitudes a, accordingly, form a geometric series, where t_n is the discrete time and $c > 1$ a constant .
For the butterfly load cases the displacement components u_i^N are applied over time t according to the following equations

$$u_i^N = a_1 \cos\varphi - a_2 \sin\varphi \qquad \text{for } u_1^2,\ u_2^4,\ u_3^5 \tag{3.4}$$

$$u_i^N = a_1 \sin\varphi + a_2 \cos\varphi \qquad \text{for } u_1^4,\ u_2^5,\ u_1^5 \tag{3.5}$$

$$a_1 = -1 + \exp\left(c_1\ \sin(2\pi\ t)\right) \tag{3.6}$$

$$a_2 = -1 + \exp\left(\sqrt{3}\ c_1\ \sin(4\pi\ t)\right) \tag{3.7}$$

Plotting the displacements u_1^2 over u_1^4, u_2^4 over u_2^5 or u_3^5 over u_1^5 results in a butterfly

like shape (cp. fig. 3.6), where c_1 is a constant, which can be compared with the amplitude a in the cyclic load cases, and φ is an angle in the corresponding displacement plane.

Tab. 3.2: Applied Load Cases

Load Case	**Applied Displacements**	**Number**	**Used in Training** [4]
Uniaxial Tensile Strain	u_1^2	10	5/3
Biaxial Tensile Strain	u_1^2, u_2^4	10	2/3
Uniaxial Compressive Strain	u_1^2	10	4/1
Biaxial Compressive Strain	u_1^2, u_2^4	10	2/4
Hydrostatic Load	u_1^2, u_2^4, u_3^5	20	4/5
Cyclic	all [5]	64	23
Cyclic free	$u_1^2, u_2^4, u_1^4, u_2^5$	64	22
Cyclic free free	u_1^2, u_1^4	60	19
Constant Cyclic	all	81	21
Constant Cyclic free	$u_1^2, u_2^4, u_1^4, u_2^5$	108	25
Constant Cyclic free free	u_1^2, u_1^4	60	21
Shakedown	all	6	0
Shakedown free	$u_1^2, u_2^4, u_1^4, u_2^5$	7	0
Shakedown free free	u_1^2, u_1^4	7	0
Butterfly	all	135	29/26
Butterfly free	$u_1^2, u_2^4, u_1^4, u_2^5$	432	25
Butterfly free free	u_1^2, u_1^4	100	23/20
Random 1	all	180	36/35
Random 2	all	180	26
In total:		1544	287/279

[4] first number: number of load cases for ANN of normal stress components, second number: number of load cases for ANN of shear stress components; only one number: same number of load cases for all ANN

[5] all pertinent displacement components, i.e.,: $u_1^2,\ u_1^4,\ u_2^4, u_2^5,\ u_3^5,\ u_1^5$

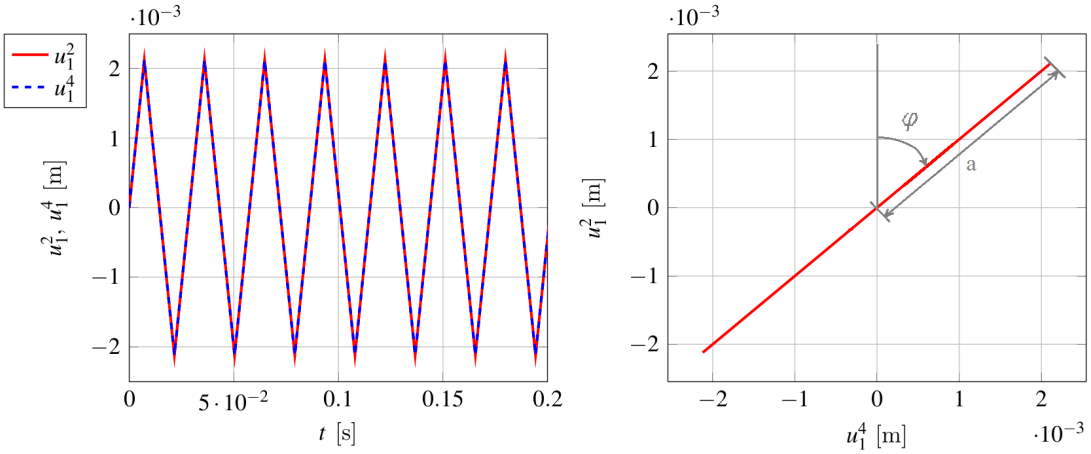

Fig. 3.4: Displacements u_1^2 and u_1^4 of a cyclic load case. left: displacement components over time t, right: straight line in displacement plane with amplitude a and angle φ

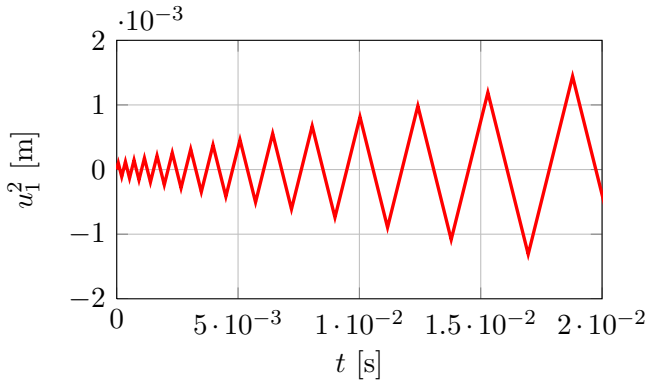

Fig. 3.5: Displacements u_1^2 of a cyclic load case with increasing amplitude over time t

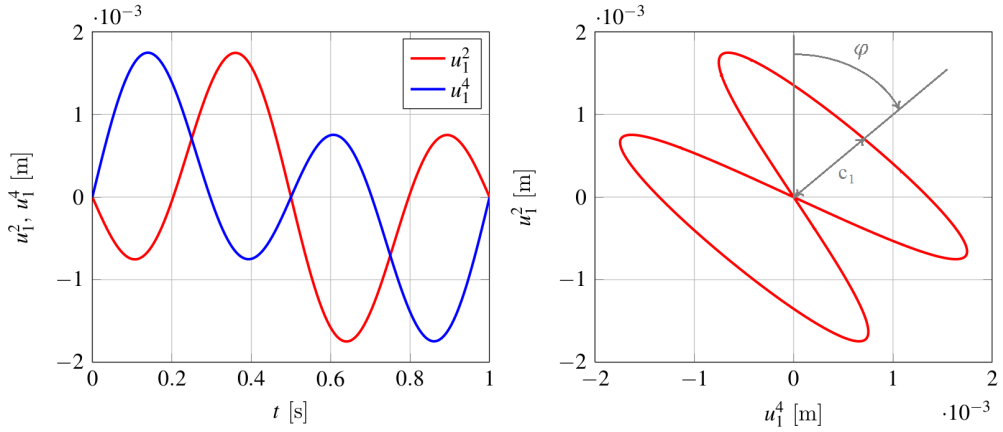

Fig. 3.6: Displacements u_1^2 and u_1^4 of a butterfly load case. left: displacements over time t, right: butterfly shape in displacement plane with constant c_1 and angle φ

There are two sets of random load cases. In set "Random 1" the displacement component u_i^N at time t_n is determined by adding up the random displacement steps Δu_i^N in the pertinent direction

$$u_i^N(t_n) = \Delta u_i^N(t_1) + \Delta u_i^N(t_2) + ... + \Delta u_i^N(t_n) \tag{3.8}$$

In "Random 2" a random displacement component u_i^N is introduced for every, discrete time t_n. In fig. 3.7 we can see that both sets of random load cases could produce equal displacement paths over time. The advantage of using two different methods is to determine different thresholds. In "Random 1" we chose a maximum displacement step $\Delta u_{i,\max}^N$ and in "Random 2" a maximum displacement $u_{i,\max}^N$.

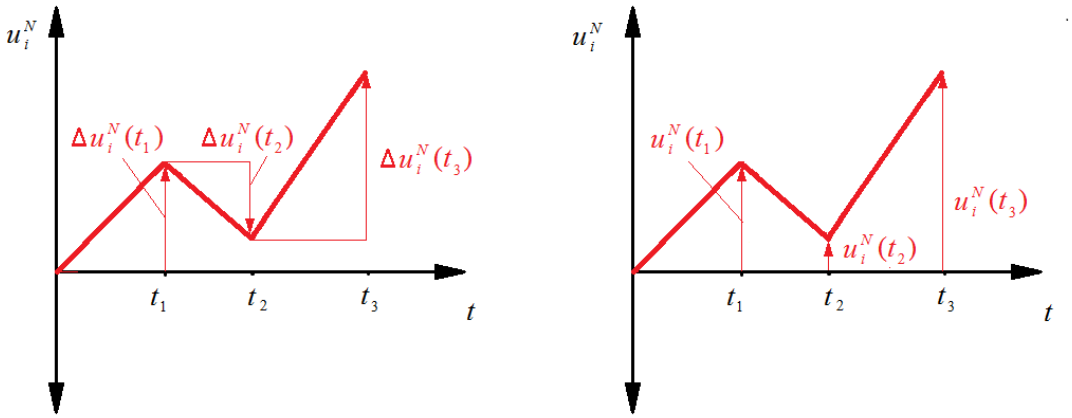

Fig. 3.7: Example of a random load case generated in set "Random 1" (left) by adding up displacement steps $\Delta u_i^N(t_n)$ and in set "Random 2" (right) by introducing displacement components $u_i^N(t_n)$ for every discrete time t_n

3.2 Development and Training of an Artificial Neural Network

The generated data sets of strain-stress relationships have to undergo further signal processing before they can be used for the training of an ANN. During the FEM simulation in ABAQUS the current stress state $\boldsymbol{\sigma}(t_n)$ is evaluated from the previous strain state $\boldsymbol{\varepsilon}(t_{n-1})$, the current strain state $\boldsymbol{\varepsilon}(t_n) = \boldsymbol{\varepsilon}(t_{n-1}) + \Delta\boldsymbol{\varepsilon}$ and the previous stress state $\boldsymbol{\sigma}(t_{n-1})$ at discrete time steps t_n. Once yielding or damage occurs

the current stress state also depends on the previous stress and strain states for a time $t < t_{n-1}$. This information is provided via state variables like the equivalent plastic strain or damage variables. The current state variables do not depend on the physical time when first yielding or damage initiation occurred and only their value is of importance. It is useful to describe the history of the strain and stress state not over the time, but, as introduced in [22], via the arc length along the strain dimensions, which may be referred to as pseudo-time s_n, by adding up the incremental strain components $\Delta\varepsilon_{ij}(t_i)$ as follows

$$s_n(t_n) = \sum_{i=0}^{n} \left(\sqrt{\Delta\varepsilon_{11}^2(t_i) + \Delta\varepsilon_{22}^2(t_i) + \Delta\varepsilon_{33}^2(t_i) + \Delta\varepsilon_{12}^2(t_i) + \Delta\varepsilon_{23}^2(t_i) + \Delta\varepsilon_{31}^2(t_i)} \right) \tag{3.9}$$

The stress and strain curves are reparameterized with the inverse function $t_n^{-1}(s_n)$ of $s_n(t_n)$ as follows

$$\boldsymbol{\varepsilon}(s_n) = \boldsymbol{\varepsilon}\left(t_n^{-1}(s_n)\right) \tag{3.10}$$

$$\boldsymbol{\sigma}(s_n) = \boldsymbol{\sigma}\left(t_n^{-1}(s_n)\right) \tag{3.11}$$

In the following we will use $\boldsymbol{\varepsilon}(s_n) = \boldsymbol{\varepsilon}^{(n)}$.
The next question is what inputs for the ANN are required for correctly predicting the stress outputs. In [22] two approaches were suggested. The first one is a recurrent parametrization where previous stress and strain states are used to predict the current stress state for a given strain state

$$\boldsymbol{\varepsilon}^{(n)}, \boldsymbol{\varepsilon}^{(n-1)}, \boldsymbol{\sigma}^{(n-1)} \rightarrow \boldsymbol{\sigma}^{(n)} \tag{3.12}$$

The indices n and $n-1$ represent the current and the previous state. The second approach is a filter-based parametrization, where a window integral $\hat{\boldsymbol{\varepsilon}}$ over previous strain states is used. For a window length of l_w we get

$$\hat{\varepsilon}_{ij} = \int_{s_n - l_\mathrm{w}}^{s_n} \varepsilon_{ij}(\xi)\ d\xi \tag{3.13}$$

The input-output relationship of the ANN is

$$\boldsymbol{\varepsilon}^{(n)}, \hat{\boldsymbol{\varepsilon}}^{(n)} \rightarrow \boldsymbol{\sigma}^{(n)} \tag{3.14}$$

Considering these recommendations several different combinations of input sets were tested to obtain the best prediction of the current stress state. Previous strain and

stress states as well as previous window integrals where used at $n-0.01$ and $n-0.005$. Window integrals with a window length l_w of 0.001, 0.002, 0.003, 0.005, 0.01, 0.05 and 0.1 were considered. The best results were achieved with the following sets of inputs for the ANN

$$\boldsymbol{\varepsilon}^{(n)}, \hat{\boldsymbol{\varepsilon}}^{(n)}_{l_\mathrm{w}=0.01}, \hat{\boldsymbol{\varepsilon}}^{(n)}_{l_\mathrm{w}=0.05}, \hat{\boldsymbol{\varepsilon}}^{(n)}_{l_\mathrm{w}=0.1}, \hat{\boldsymbol{\varepsilon}}^{(n-0.01)}_{l_\mathrm{w}=0.01}, \hat{\boldsymbol{\varepsilon}}^{(n-0.01)}_{l_\mathrm{w}=0.05}, \hat{\boldsymbol{\varepsilon}}^{(n-0.01)}_{l_\mathrm{w}=0.1} \tag{3.15}$$

Another question was how many ANN are necessary for the prediction of the stress components. Options of one ANN for all six stress components, two ANN for the three normal stress and the three shear stress components, three ANN for the stress components $(\sigma_{11},\ \sigma_{12})$, $(\sigma_{22},\ \sigma_{23})$ and $(\sigma_{33},\ \sigma_{31})$ as well as six ANN, one for each stress component, were tested. The best results were obtained with six individual ANN.

The architecture of the ANN itself was another important aspect. In the tests, the number of nodes in the hidden layers ranged from 10 to 180 and the number of hidden layers from 1 to 5. The finally chosen ANN have four hidden layers with 25 nodes each.

For the transformation of the input and output data a linear transformation of the form

$$f_\mathrm{tf} = \frac{a_1 \zeta + a_2}{b} \tag{3.16}$$

and the interval $[-0.5, 0.5]$ was chosen, where ζ is substituted with the input or output data and a_1, a_2 and b are coefficients to fit the data onto the used interval. As transfer functions for the hidden layers the tan-sigmoid function f_tan (cp. eqn. (2.4)) was used and for the output layer the chosen transfer function was the linear function f_lin (cp. eqn. (2.5)).

Furthermore all training algorithms provided by Matlab were considered. The best results were obtained by the Levenberg-Marquardt algorithm and the Bayesian regularization, which enhances the Levenberg-Marquardt algorithm by minimizing a linear combination of squared errors and weights to obtain a good generalization [18]. Although the resulting error of the Bayesian Regulation was slightly smaller, the Levenberg-Marquardt algorithm was preferred due to its computational efficiency. The mean squared error $\mathrm{E_{MSE}}$ was used as error function. As stopping criterion the error was not used directly but minimum thresholds of its performance gradient were set to 1×10^{-5} and 1×10^{-4} for normal and shear stress components, respectively.

Another essential part is the selection of the training and validation data. From

the selected data 70% of the samples are used for training, 15% for validation and 15% for testing. Three different strategies were tested, based on separating the load cases into random and deterministically defined load cases. If only load cases from the random or deterministically defined set were used, the performance in the other set was found to be bad. Therefore load cases from both sets were used to train the ANN. The training procedures started with a set of arbitrarily chosen load cases representing all types of load cases listed in tab. 3.2 except uniaxial tensile, hydrostatic, compressive and shakedown loads. If the generalization of certain load cases was particularly bad, they were added to the training set. This is the reason why uniaxial tensile, compressive and hydrostatic loads had to be included. The goal was to achieve a wide representation of all possible load cases.

3.3 Performance and Validation

The six ANN used in the following have four hidden layers with 25 neurons each, 42 inputs consisting of current strain components, window integrals of the strain components at the current and previous states as defined in eqn. (3.15) and one output, which is one of the stress components. Using a recurrent ANN with previous stress states led to high offsets and noisy oscillating stress responses. In fig. 3.8 the stress response σ_{12} of a recurrent network is depicted over the pseudo-time s. We can see oscillating behaviour at approximately $0.3 \leq s \leq 0.6$ and $s > 1.6$ and an offset at approximately $1.2 \leq s \leq 1.6$. Therefore recurrent networks were excluded from further studies.

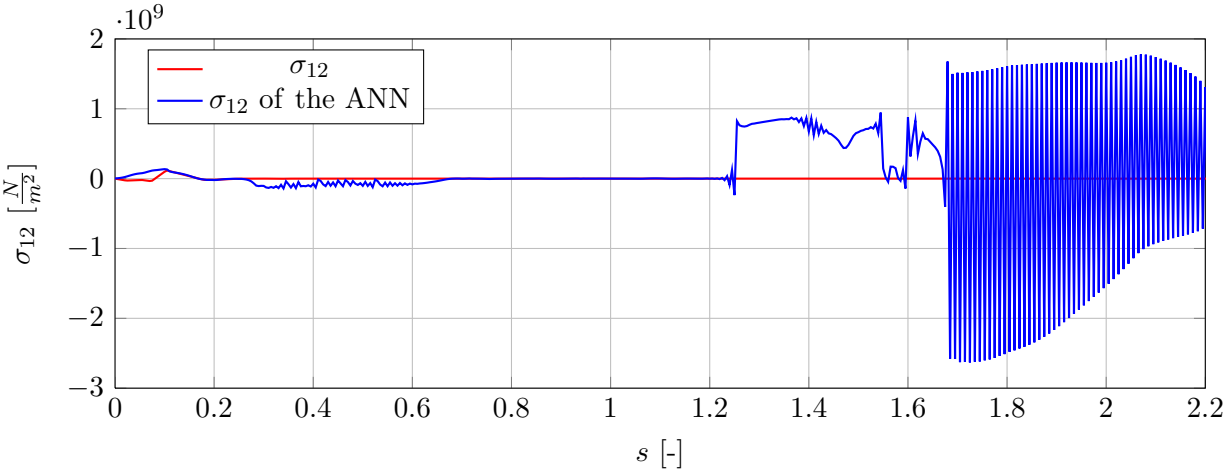

Fig. 3.8: Stress response σ_{12} of a recurrent ANN showing offsets and noisy oscillating behaviour.

To train the ANN we used the mean squared error E_{MSE} (cp. eqn. (2.7)). In order to deduce values, that can be compared more easily with the output of the stress components we calculate

$$E_{SR} = \sqrt{E_{MSE}} \tag{3.17}$$

It should be noted that with one neuron in the output layer and a number of P patterns, the error includes a factor of $\sqrt{P}$ accounting for the number of patterns. In table 3.3 the error E_{SR} is listed for different load cases and in table 3.4 the absolute maximum values of the stress components $|\sigma_{ij,\ \max}|$ are listed for the load cases from figs. 3.9 to 3.14. If we compare the values of these two tables, we can observe that the absolute maximum stress values are of the same magnitude or one magnitude smaller than the error E_{SR}. We can also see that the error in the shear stress components is much smaller than in the normal stress components, and that the error for a certain load case can vary for the different stress components.

Tab. 3.3: Square Root of the Mean Squared Error of the Stress Components for Different Load Cases

	Constant Cyclic	Cyclic	Shakedown	Butterfly	Tensile Strain
σ_{11}	8.049×10^{8}	6.753×10^{8}	1.004×10^{8}	4.277×10^{9}	1.076×10^{8}
σ_{22}	3.610×10^{8}	7.404×10^{8}	1.313×10^{8}	2.925×10^{9}	1.214×10^{8}
σ_{33}	4.426×10^{8}	5.955×10^{8}	1.458×10^{8}	3.451×10^{9}	1.805×10^{8}
σ_{12}	1.4057×10^{7}	1.064×10^{7}	1.359×10^{7}	2.315×10^{7}	2.174×10^{6}
σ_{23}	1.131×10^{7}	8.263×10^{6}	9.824×10^{6}	2.318×10^{7}	1.046×10^{6}
σ_{31}	9.063×10^{6}	2.306×10^{6}	1.050×10^{7}	1.513×10^{7}	4.858×10^{5}

	Compressive Strain	Hydrostatic Load	Random	All Loadcases
σ_{11}	1.290×10^{8}	9.922×10^{8}	2.026×10^{9}	3.103×10^{9}
σ_{22}	9.552×10^{7}	8.602×10^{8}	2.836×10^{9}	2.317×10^{9}
σ_{33}	1.770×10^{8}	1.203×10^{9}	2.428×10^{9}	2.588×10^{9}
σ_{12}	1.800×10^{6}	1.532×10^{6}	2.989×10^{7}	2.066×10^{7}
σ_{23}	1.226×10^{6}	1.278×10^{6}	3.829×10^{7}	2.206×10^{7}
σ_{31}	5.728×10^{5}	1.752×10^{6}	2.814×10^{7}	1.542×10^{7}

Tab. 3.4: Absolute Maximum Values of the Stress Components for Typical Load Cases Depicted in Figs. 3.9 to 3.14

	Shakedown	Random	Butterfly	Tensile Strain	Tensile Strain
$\lvert\sigma_{11,\ \max}\rvert$	9.761×10^7	1.770×10^8	5.223×10^9	5.923×10^8	2.332×10^8
$\lvert\sigma_{22,\ \max}\rvert$	1.185×10^8	1.801×10^8	3.243×10^9	2.879×10^8	8.570×10^7
$\lvert\sigma_{33,\ \max}\rvert$	1.481×10^8	2.442×10^8	5.550×10^9	2.429×10^8	1.611×10^8
$\lvert\sigma_{12,\ \max}\rvert$	2.893×10^7	3.646×10^7	1.457×10^8	4.479×10^6	5.338×10^6
$\lvert\sigma_{23,\ \max}\rvert$	2.457×10^6	3.432×10^7	1.663×10^8	3.638×10^6	1.154×10^6
$\lvert\sigma_{31,\ \max}\rvert$	2.181×10^5	1.595×10^7	9.595×10^7	6.767×10^5	3.363×10^5

If we look at the stress response plot of the shakedown load case (cp. fig. 3.9) driven by all pertinent displacement components u_1^2, u_1^4, u_2^4, u_2^5, u_3^5 and u_1^5, we can see that though the stress component σ_{31} is not zero, the stress response is three magnitudes smaller in comparison to the normal stress components and therefore acceptable. The stress components σ_{11}, σ_{12}, σ_{23} and σ_{31} were well reproduced by the ANN, though shakedown load cases were not included in the training set. This is an indication that the ANN have actually learned the material behaviour. However, the performance for the stress components σ_{22} and especially σ_{33} is poor in comparison. This coincides with the observation from the mean squared errors (which, however, covers a number of shakedown load cases), that the performance in the shear stress components is better than in the normal stress components and that the performance for the same load case can vary in the different stress components, since they were trained in different ANN. Furthermore it should be noted that load cases with "free" and "free free" displacements show zero-stress responses in the stress components σ_{33}, σ_{31} and/or σ_{22}, σ_{23}. This means that the ANN for these stress components have seen fewer non-zero stress responses. For example 25 "Constant Cyclic free" load cases were used in training. Therefore the ANN for σ_{11} would be trained with 25 load cases showing a stress response with cyclic behaviour, while the ANN for σ_{33} was trained with 25 load cases with a zero-stress response due to the "free" displacement in this direction. This could also explain the drop in the generalization quality. Moreover it seems difficult to reproduce a perfect zero-stress response, thus a small stress response in comparison with the non-zero-stress responses would be sufficient.

In fig. 3.10 we see the stress and strain histories generated within a type 2 random load case and in fig. 3.11 the same load case is depicted on the interval $[0, 0.2]$ of the pseudo-time s to show its behaviour in more detail. We can observe that the ANN generally follow the stress responses, but over- or underestimate stress peaks. In the stress component σ_{33} we can find an offset in the ANN response similar to the shakedown load case.
In the stress response plot of a butterfly load case (cp. fig. 3.12) we can see that the ANN capture the initial material behaviour reasonably well but have problems with depicting the zero-stress response after material failure, which occurs at a pseudo-time of approximately $s = 0.5$. In the random load case the stress responses of the ANN do not quite follow the actual stress responses, especially σ_{33}, σ_{12} and σ_{13} (cp. fig. 3.10 and 3.11). However, the behaviour in the butterfly load case is too noisy, especially for the σ_{12} component. The same holds true for the uniaxial tensile strain load case depicted in fig. 3.13. An even more extreme behaviour in this respect is evidenced by the uniaxial tensile load case shown in fig. 3.13, where spurious excursions at amplitudes exceeding that of the actual signal are present for all three normal stress components. Figure. 3.14, in contrast, shows another uniaxial tensile load case which exhibits much better but still not fully satisfactory behaviour. Two tensile load cases, one with a bad performance (cp. fig. 3.13) and one with a good performance in the σ_{11} stress component (cp. fig. 3.14), were chosen to be presented in this thesis.
These observations lead to the conclusion that the ANN described in section 3.2 have problems reproducing zero-stress responses. The resulting oscillations can have the magnitude of the actual stress response in the non-zero stress components (cp. fig. 3.13) which is not an acceptable error. However, especially for random and cyclic load cases the responses are quite good, though they may vary in the different stress components, due to the separate training of the corresponding ANN.

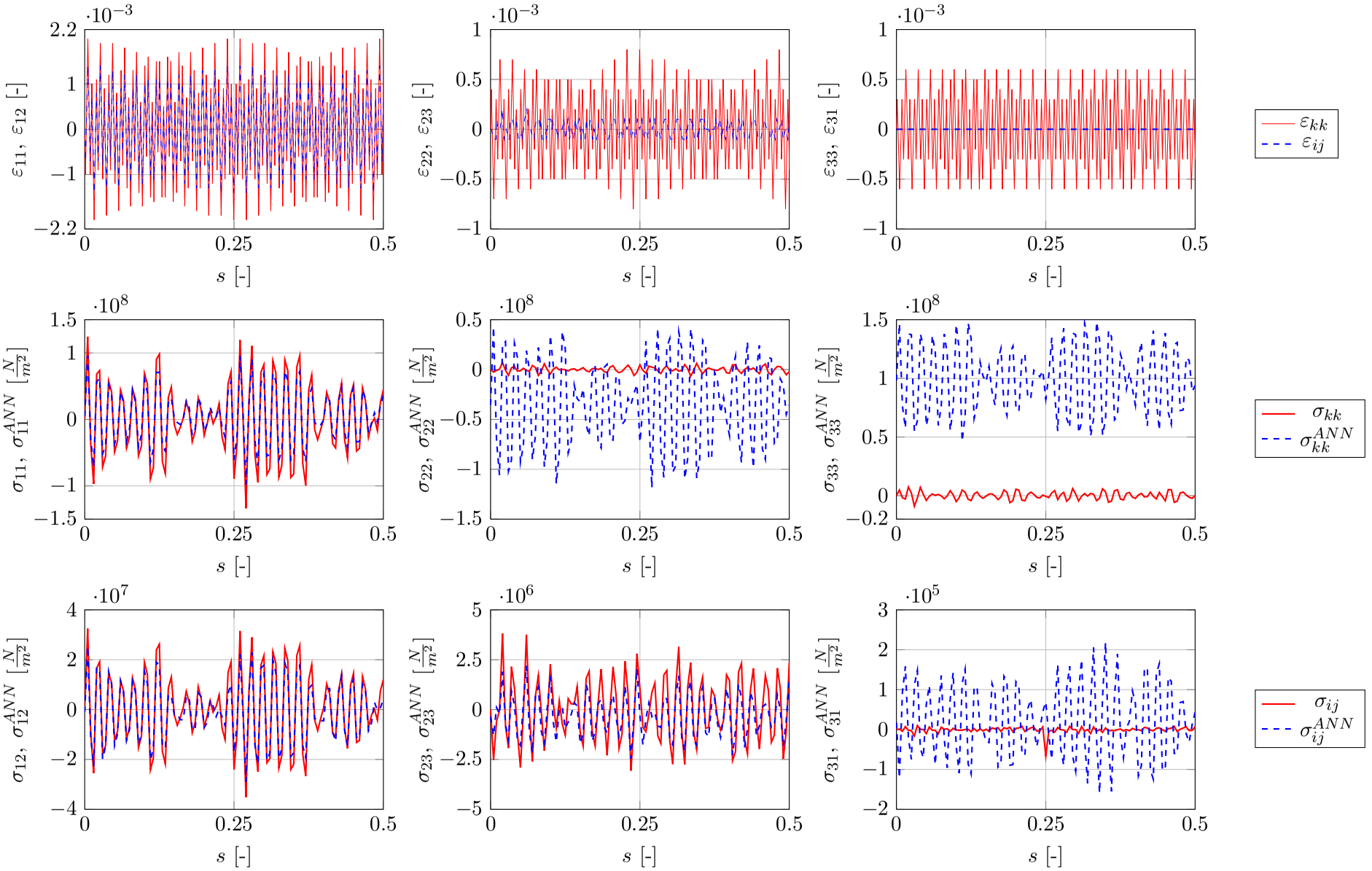

Fig. 3.9: Shakedown load case over pseudo-time s (all pertinent displacement components); first row: solid red lines = strain components in normal direction ε_{kk}, dashed blue lines = strain components in shear direction ε_{ij}; second and third rows: solid red lines = stress components of deterministic material model σ_{kk}, σ_{ij}, dashed blue lines = stress components of ANN material model σ_{kk}, σ_{ij}^{ANN} $(i,j,k=1,2,3;\ i \neq j)$

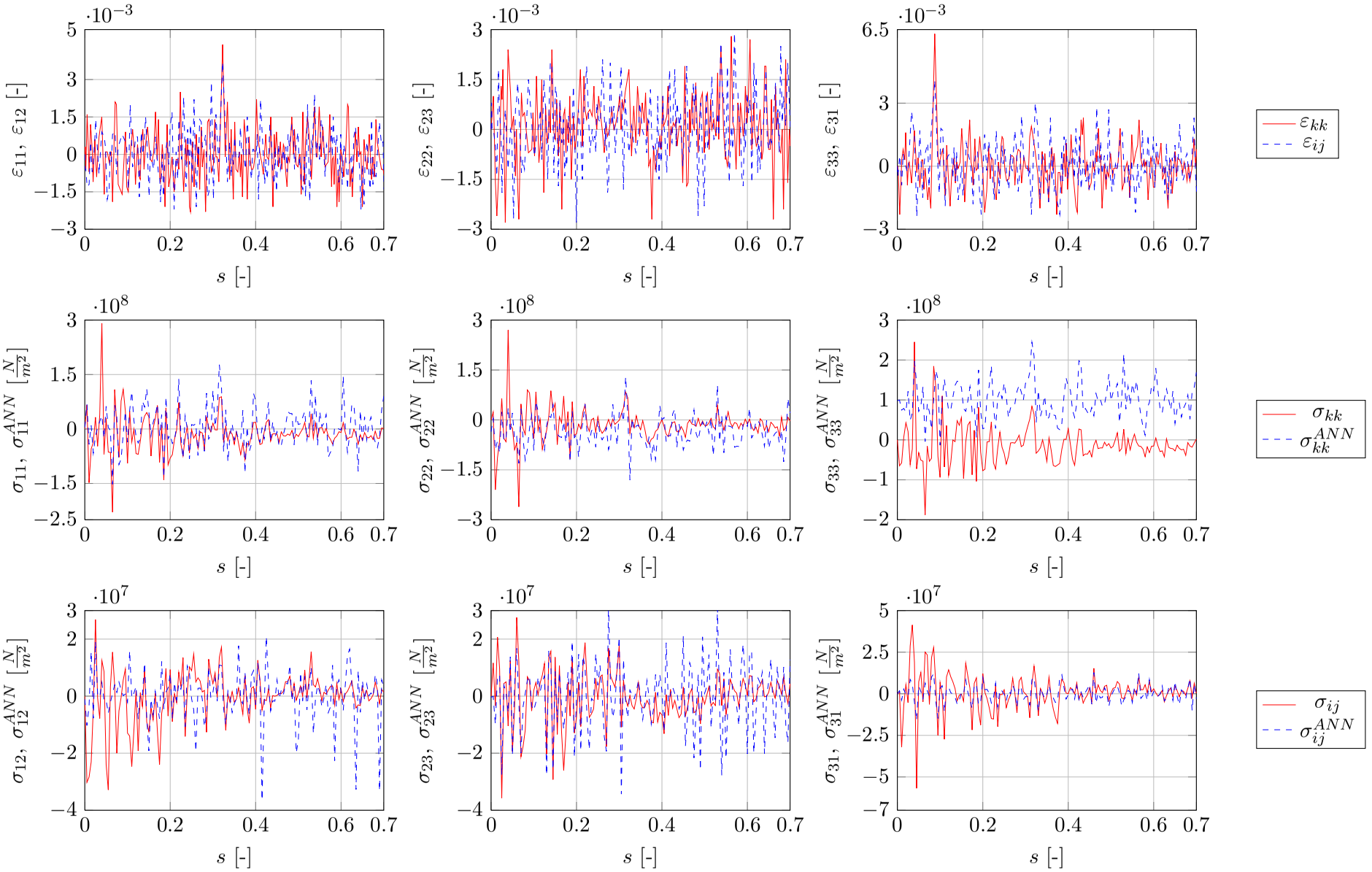

Fig. 3.10: *Random load case over pseudo-time s (all pertinent displacement components); first row: solid red lines = strain components in normal direction ε_{kk}, dashed blue lines = strain components in shear direction ε_{ij}; second and third rows: solid red lines = stress components of deterministic material model σ_{kk}, σ_{ij}, dashed blue lines = stress components of ANN material model σ_{kk}, σ_{ij}^{ANN} $(i,j,k = 1,2,3;\ i \neq j)$*

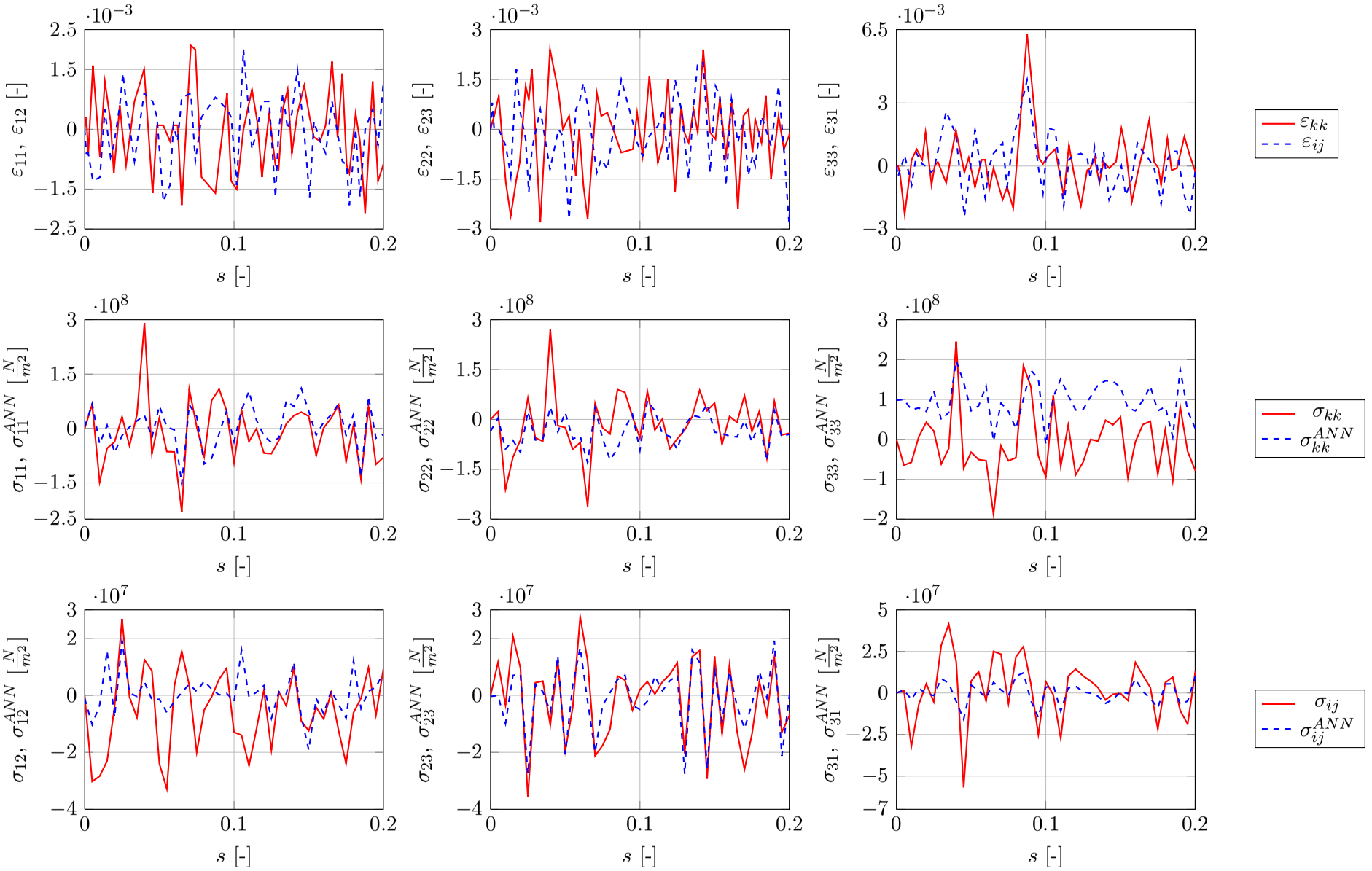

Fig. 3.11: *Detail of random load case over pseudo-time s (all pertinent displacement components); first row: solid red lines = strain components in normal direction ε_{kk}, dashed blue lines = strain components in shear direction ε_{ij}; second and third rows: solid red lines = stress components of deterministic material model σ_{kk}, σ_{ij}, dashed blue lines = stress components of ANN material model σ_{kk}, σ_{ij}^{ANN} $(i,j,k = 1,2,3;\ i \neq j)$*

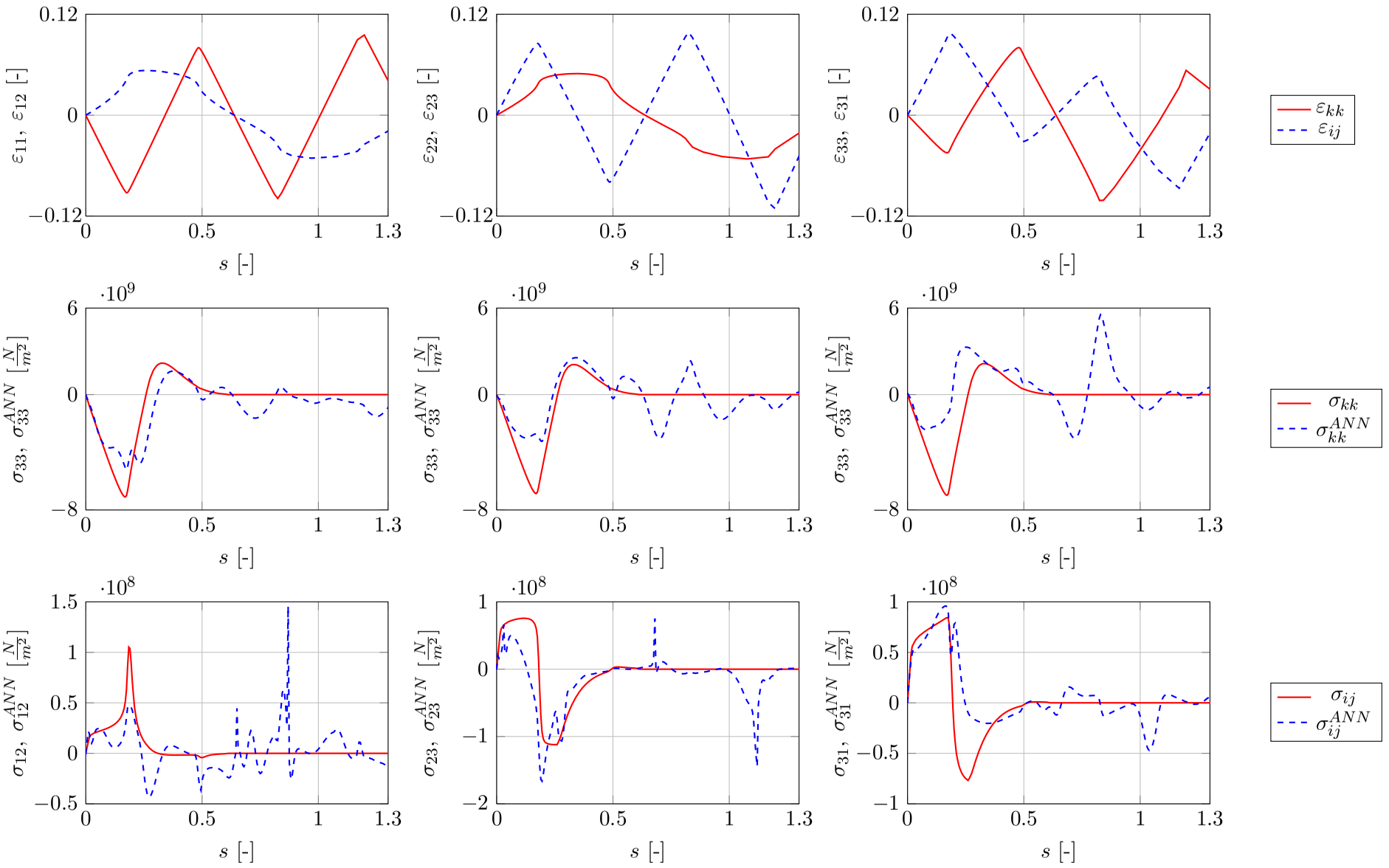

Fig. 3.12: Butterfly load case over pseudo-time s (all pertinent displacement components); first row: solid red lines = strain components in normal direction ε_{kk}, dashed blue lines = strain components in shear direction ε_{ij}; second and third rows: solid red lines = stress components of deterministic material model σ_{kk}, σ_{ij}, dashed blue lines = stress components of ANN material model σ_{kk}, σ_{ij}^{ANN} $(i,j,k = 1,2,3;\ i \neq j)$

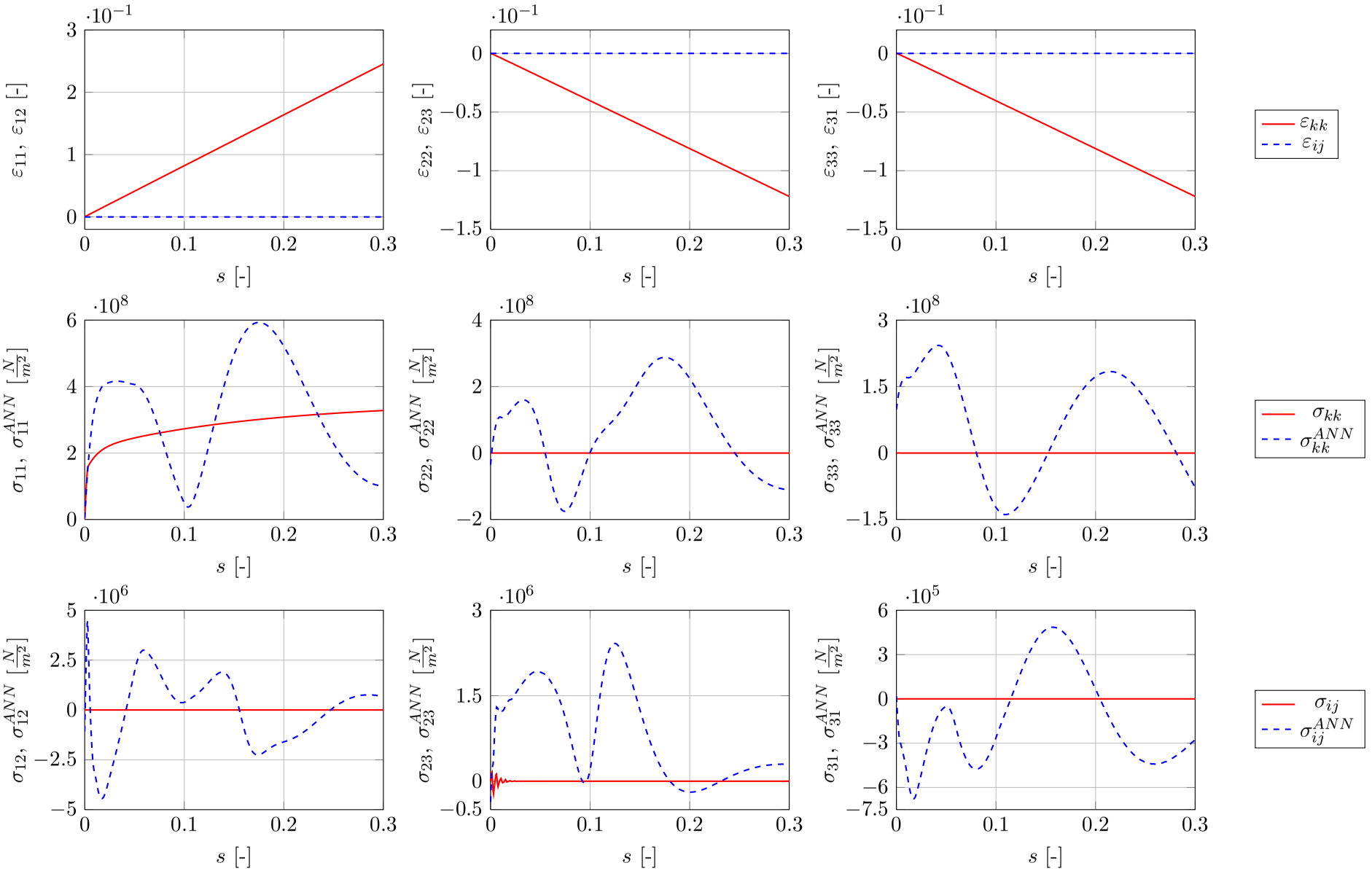

Fig. 3.13: *Uniaxial tensile strain load case in 1-direction over pseudo-time s; first row: solid red lines = strain components in normal direction ε_{kk}, dashed blue lines = strain components in shear direction ε_{ij}; second and third rows: solid red lines = stress components of deterministic material model σ_{kk}, σ_{ij}, dashed blue lines = stress components of ANN material model σ_{kk}, σ_{ij}^{ANN} $(i,j,k = 1,2,3;\ i \neq j)$*

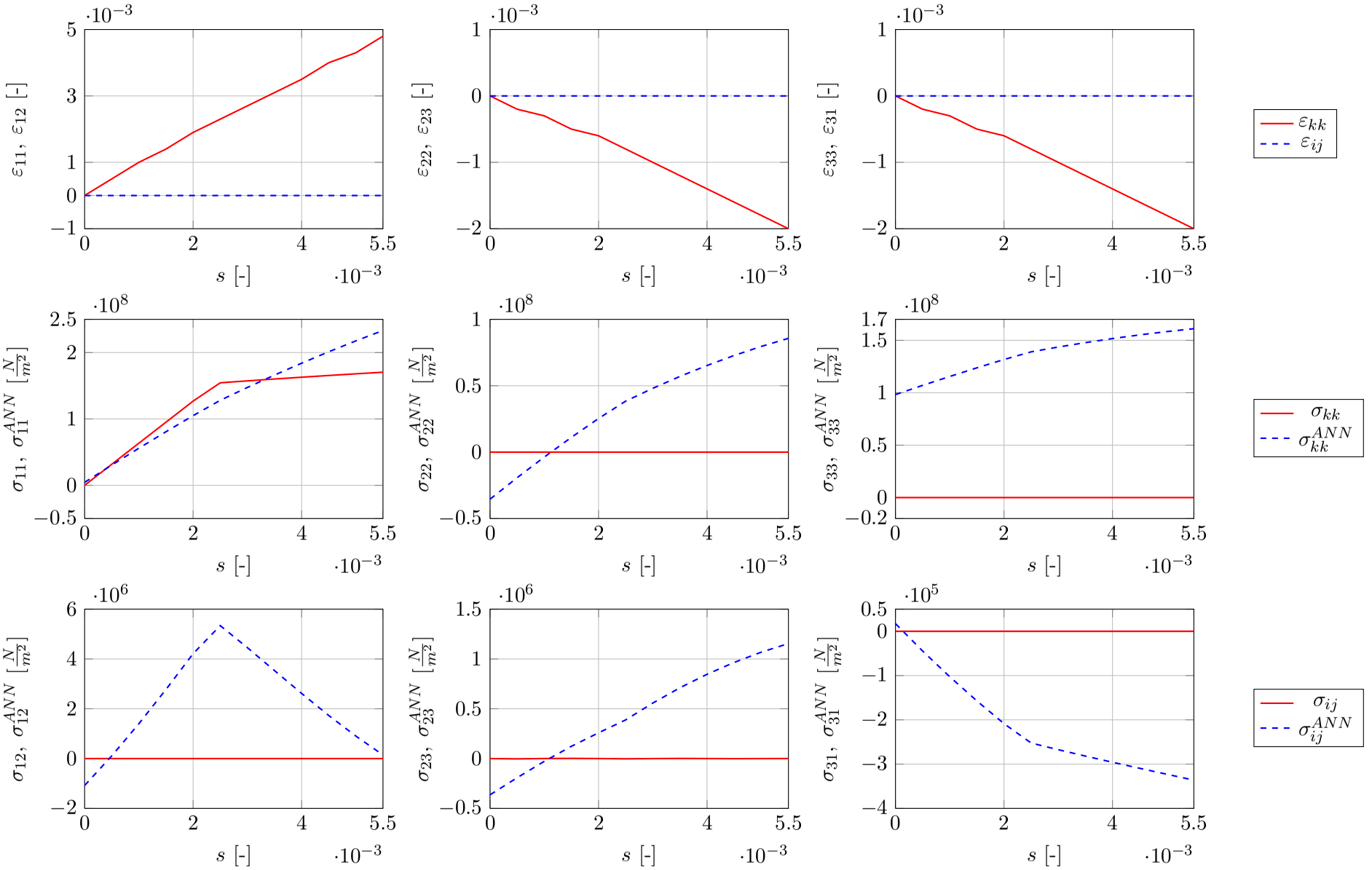

Fig. 3.14: *Uniaxial tensile strain load case in 1-direction over pseudo-time s with good performance in the σ_{11} stress component; first row: solid red lines = strain components in normal direction ε_{kk}, dashed blue lines = strain components in shear direction ε_{ij}; second and third rows: solid red lines = stress components of deterministic material model σ_{kk}, σ_{ij}, dashed blue lines = stress components of ANN material model σ_{kk}, σ_{ij}^{ANN} $(i,j,k = 1,2,3;\ i \neq j)$*

3.4 Implementation of the ANN in a VUMAT

Finally the ANN material model was implemented in a user defined material subroutine for ABAQUS/Explicit, a so called VUMAT, which must be written in FORTRAN. This subroutine is called once per time step and per integration point. Its parameters used in the implementation of a VUMAT with an ANN based material model are depicted in fig. 3.15. Essentially, ABAQUS/Explicit provides strain increments and thus the strain state and the VUMAT has to evaluate the stress increment from these strain increments, using the old stress state and suitable state variables. Although the ANN can be exported by MATLAB as FORTRAN code, suitable interfaces between the variables provided and required by ABAQUS and the data used by the ANN must be provided. For the present work this was done in a prototype implementation that uses static structures for processing and storing state variables such as the window integrals defined in eqn. (3.15).
As before the resulting strain components are sampled over a pseudo-time and the necessary window integrals are calculated. Two buffers storing the strain components and the window integrals over the pseudo-time are necessary for calculating the window integrals and deducing the previous window integral values. The processed strain signals act as inputs for the ANN to produce the resulting stress responses, which are written into the output file by ABAQUS (cp. fig. 3.16).
In the course of the implementation several problems occurred. According to [1] the stable time increment Δt_{stable} for ABAQUS explicit analyses is estimated with the equation

$$\Delta t < \Delta t_{\text{stable}} = \frac{L_{\min}}{c_{\text{d}}} \tag{3.18}$$

$$c_{\text{d}} = \sqrt{\frac{E}{\rho}\frac{1-\nu}{(1+\nu)(1-2\nu)}} \quad , \tag{3.19}$$

where $L_{\min}$ is the smallest linear element dimension in the mesh, c_{d} the dilatation wave speed, calculated with the Young's modulus E, the material density ρ and the Poisson ratio ν. The initial stable time increment, estimated over the linear elastic material properties, equals $\Delta t_{\text{stable}} = 8.3 \times 10^{-5}$ s for the single element test and might decrease during the calculation at, e.g., spurious peaks in the material behaviour. The used time increment Δt should be smaller than the stable time increment Δt_{stable}, to make sure that a correct result is found.

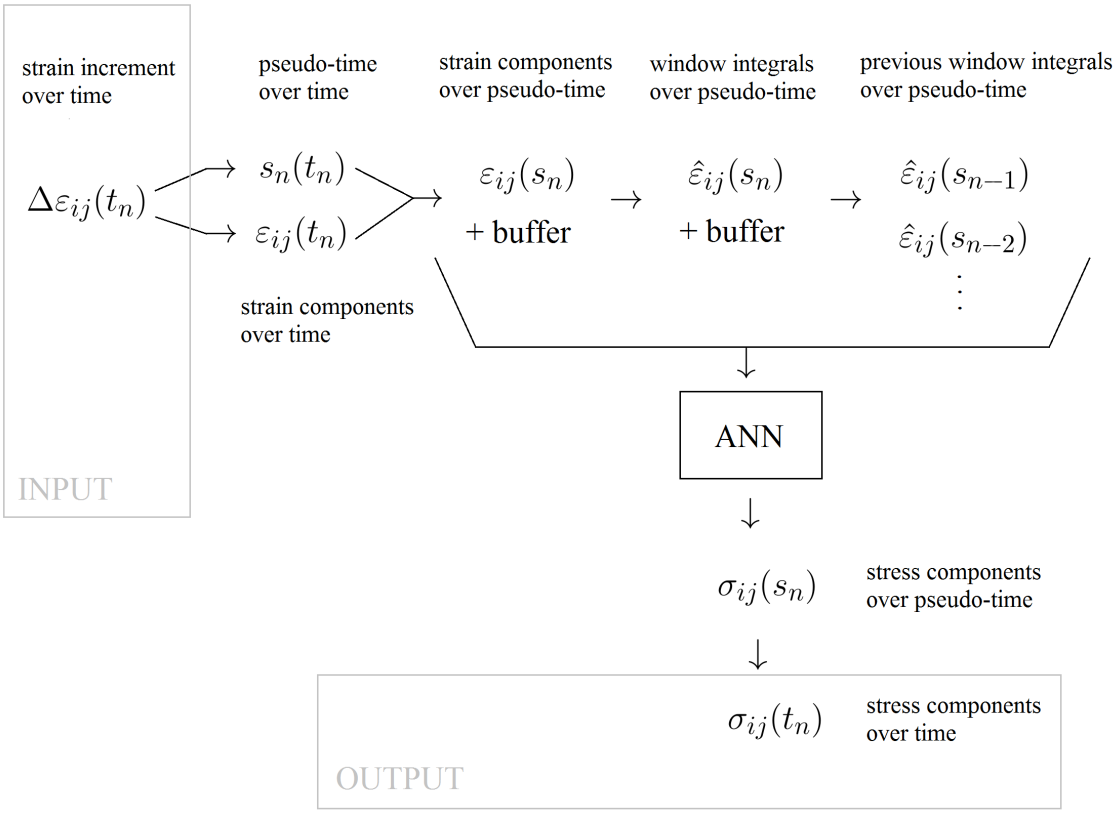

Fig. 3.15: VUMAT implementation of the ANN based material model.

The maximum window length of the window integrals was chosen to be $l_w = 0.1$ and the previous window integrals were deduced at a pseudo-time of $n - 0.01$, if n stands for the current state. These two parameters and the stable time increment influence the required size of the buffers. The buffer containing the window integrals has a size of 7000 points and the buffer containing the strain components a size of 20000 points for each stored component, due to the small stable time increment and the large window length. Since the buffers have to be updated during every step of an analysis their sizes directly influence the calculation time.

It should be noted that even if the time increment is constant over the whole calculation, the pseudo-time steps are not, since they are norms of the total strain components and the material behaviour is nonlinear. Moreover if, due to an instability or other errors, one or several strain components show an unwanted spike or noisy behaviour over time, the pseudo-time would increase strongly leading to a severe alteration of the input signals.

In addition an interface problem occurred. In the ABAQUS manual [1] it is stated that *"[...] the Green-Naghdi stress rate is used in VUMAT. However, the stress*

rate used for built-in material models [...] used with solid (coninuum) elements in Abaqus/Explicit employ the Jaumann stress rate.". The stress measure conjugates to a certain strain measure. This leads to the effect that the strain components deduced from the strain increments in the VUMAT differ from the strain components written into the ABAQUS output file if non-linear geometry calculation is activated (cp. fig. 3.16). Since the ANN were trained with inputs deduced from the strain components evaluated with a built in material model (Jaumann stress), this training is not fully appropriate for the behaviour they are supposed to reproduce (Green-Naghdi stress). Accordingly, it would be necessary to employ the Green-Naghdi stress during the generation of the training data.

If displacements were not applied in all directions, the stress response in the free direction would naturally be zero. However, the high error of the ANN for zero-stress responses caused either wrong results or the termination of the calculation due to element distortion.

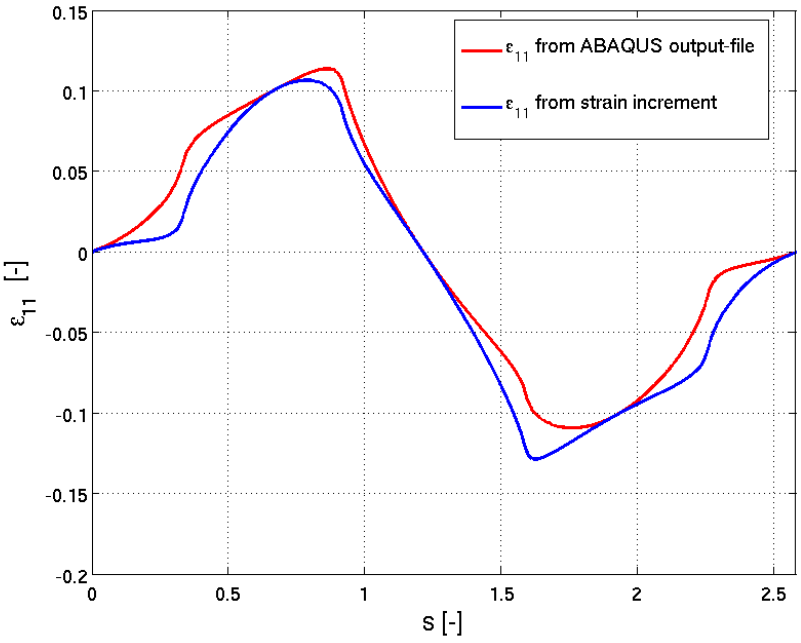

Fig. 3.16: *Comparison of ε_{11} from the ABAQUS output-file and from the strain increment directly in the VUMAT*

4 Discussion and Conclusion

The goal of the present master's thesis was to develop an ANN which can reproduce elasto-plastic material behaviour with mixed hardening properties, ductile damage and finally failure and is capable of handling finite strain. The necessary data for training, validation and testing were obtained by FEM-calculations with ABAQUS/Explicit using a single three dimensional solid element, capturing a wide spread of material responses in six-dimensional stress space.

The search for a proper ANN architecture and suitable parameters lead to six separate ANN with current strain components and current and previous window integrals of strains as the 42 inputs. The inputs over the pseudo-time s are transformed with a linear transformation function onto the interval $[-0.5, 0.5]$ and processed through four hidden layers with 25 neurons each, using the tan-sigmoid function as transfer function. The output layer uses a purely linear transfer function and produces, after the retransformation, the stress response of one stress component. The learning algorithm in MATLAB was chosen to be the back propagation algorithm, combined with the mean squared error and the Levenberg-Marquardt algorithm. From the 1544 load cases 287 and 279 load cases were used for the training of the normal and shear component ANN, respectively, with 70% of the samples used for training.

If we look at the results we can see that cyclic and random behaviour was captured best, whereas simple tension and pressure load cases show larger errors. Moreover zero-stress responses could not be reproduced well, with oscillations of considerable magnitude occurring. Typically, the results of the shear stress components show a lower error and often a better generalization behaviour than the normal stress components. Since the stress components were deduced from separate ANN their performance also varies for the same load case.

Since the goal was to represent the material behaviour as fully as possible, ranging from uniaxial tensile strain to complex butterfly behaviour, a mayor question was with which load cases the ANN are to be trained. After testing several different approaches a basic set of all load cases extended with load cases showing bad performance was chosen. Since shakedown behaviour was represented quite well,

without being in the training set, it was not included. However, the question remains, if certain material behaviour might be over- or under-presented. Since, especially, cyclic material behaviour was captured well, the number of load cases in the training set could probably be reduced, while, e.g., an increase in uniaxial tensile load cases could lead to better performance in the corresponding responses. It is also possible to add weight factors to ensure that certain "important" load cases are actually learned by the ANN.

However, a more detailed choice of training load cases is also limited by other influencing factors. One problem was the high calculation time for training an ANN. In the final stage the calculation time was approximately 20 hours for one ANN, which could be reduced to a few hours by setting a threshold for the target value of the error gradient.

Looking, e.g., at figs. 3.10 and 3.11 we can see that the stress responses of the ANN don't reliably follow the desired stress curves. This could be an indicator for underfitting. Conversely the oscillating stress responses during, e.g., tensile load cases or desired zero-stress responses indicate overfitting. Comparing the errors of cyclic load cases and tensile load cases from two different resulting ANN for the σ_{11} stress components with the same settings we get the following square roots of the mean errors for the cyclic load cases: 6.753×10^8, 2.533×10^9 and for the tensile load cases: 1.076×10^8, 7.048×10^7. We can see that the error of the cyclic load case is small, when the error of the tensile load case is large and the opposite way around. This leads to the conclusion that the responses, which should be reproduced, differ excessively from each other and that conflicts in the training data may exist.

To obtain ANN with better performance several steps may be taken. One step would be to perform a more thorough data mining, to select the training data in more detail, limiting the number of different sets of load cases which should be reproduced and or estimating data conflicts.

It may be possible to train different ANN for different load cases, but then the networks would not be applicable for general purpose application, since the load case must be known a priori or for every combination of load case, where linear combination of load cases is not an option, a particular ANN would be needed.

Another option is to combine different ANN representing different material behaviour. Suitable identifiers must be trained to switch between elastic, plastic and damage behaviour. Certain parameters like the accumulated plastic strain ($\bar{\varepsilon}^p = 0$ for elasticity, $\bar{\varepsilon}^p \neq 0$ for plasticity), the damage indicator ($w_D = 1$ for onset of damage, else $0 \leq w_D < 1$) or the damage variable ($D = 1$ for failure, else $0 \leq D < 1$) could

be used. An identifier trained to catch the onset of material failure would solve the problem of the zero-stress response after failure. Either the stress component could be set to a certain value or the element deletion option in ABAQUS could remove the element. However, for load cases with "free" displacement in certain direction leading to zero-stress responses the solution is not so easy. In a mesh used in an FEM calculation the displacement of a node is "free" in the sense that no displacement is directly applied. However, constraints and boundary conditions might lead to non-zero stress responses. Therefore the stress component of a "free" direction can not simply be set to zero.
In general it seems possible to train ANN reproducing complex material behaviour. However, several problems occurred, which could be solved by splitting up the ANN in smaller ANN reproducing only a certain part of the material behaviour, e.g., damage, performing closer data mining, excluding certain load cases from the ANN, which then unfortunately can't capture the full material behaviour or choosing the training data more carefully. These alterations can also make it necessary to adapt the architecture of the ANN. Another important issue is to reduce the computational time. To address this problem it might be useful to look further into different learning algorithms.

Bibliography

[1] *ABAQUS-Manual 6.13*, 3DS Simulia, Waltham, MA, 2013

[2] Abendroth, Martin; Kuna, Meinhard: *Identification of ductile damage and fracture parameters from the small punch test using neural networks.* In: Engineering Fracture Mechanics 73: 2006, p. 710-725

[3] Altenbach, Holm: *Kontinuumsmechanik. Einführung in die materialunabhängigen und materialabhängigen Gleichungen.* Berlin/Heidelberg: Springer-Verlag, 2012

[4] Böhm, Helmut J.: *ILSB Report 206. A Short Introduction to Basic Aspects of Continuum Micromechanics.* Institute of Lightweight Design and Structural Biomechanics, Vienna University of Technology, Vienna, Austria: 2010

[5] Bui, Tinh Quoc: *Studies on Modeling the Failure Behavior of Spot Welds.* Doctoral Thesis. Institute of Lightweight Design and Structural Biomechanics, Vienna University of Technology, Vienna, Austria: 2009

[6] Chen, W.F.; Han, D.J.: *Plasticity for Structural Engineers.* New York: Springer-Verlag, 1988

[7] Chinh, Pham Duc: *Shakedown kinematic theorem for elastic-perfectly plastic bodies.* In: International Journal of Plasticity 17: 2001, p. 773-780

[8] de Saxcé, Géry; Oueslati, Abdelbacet; Charkaluk, Eric; Tritsch, Jean-Bernard: *Limit State of Materials and Structurs. Direct Methods 2.* Dordrecht Heidelberg New York London: SpringerScience + Business Media Dordrecht, 2013

[9] Dunne, Fionn; Petrinic Nick: *Introduction to Computational Plasticity.* New York: Oxford University Press, 2006

[10] Hashiguchi, Koichi; Yamakawa, Yuki: *Introduction to Finite Strain Theory for Continuum Elasto-Plasticity.* West Sussex: John Wiley & Sons Ltd Reprinted, 2013

[11] Haykin, Simon: *Neural Networks. A Comprehensive Foundation*. New Jersey: Prentice Hall Inc., 1999

[12] Johnson, W.; Mellor, P. B.: *Engineering Plasticity*. Letchworth, Hertfordshire: The Garden City Press Limited, 1973

[13] Koiter, Warner T.: *A new general theorem an shakedown of elastic-plastic structures.* Proceedings Koninklijke Nederlandse Akademje van Wetenschappen B59, 1956, S. 24-34

[14] König, Jan A.: *Shakedown of Elastic-Plastic Structures*. New York, Elsevier Science Publisher Co., Inc.; Warszawa: Ars Polona; Amsterdam: Elsevier Science Publishers, 1987

[15] Lemaitre, Jean: *A Course on Damage Mechanics*. Berlin/Heidelberg: Springer-Verlag, 1996

[16] Lemaitre, Jean; Rodrigue, Esmorat: *Engineering Damage Mechanics. Ductile, Creep, Fatigue and Brittle Failures*. Berlin/Heidelberg: Springer-Verlag, 2005

[17] Mang, Herbert; Hofstetter, Günter: *Festigkeitslehre*. Wien/New York: Springer-Verlag, 2008

[18] *MATLAB-Manual 6.12a*, The MathWorks Inc., Natick, MA, 2012

[19] Melan, Ernst: *Theorie statisch unbestimmter Systeme aus ideal plastischem Baustoff.* Sitzungsberichte der Akademie der Wissenschaften Wien, Serie IIa 145, 1936, S. 195-218

[20] Mouhtamid, Said: *Anwendung direkter Methoden zur industriellen Berechnung von Grenzlasten mechanischer Komponenten*. Dissertation, Fakultät für Maschinenwesen; Rheinisch-Westfälische Technische Hochschule Aachen: 2007

[21] Nguyen, Quoc-Son: *On shakedown analysis in hardening plasticity*. In: Journal of Mechanics and Physics of Solids 51: 2003, p. 101-125

[22] Palau, Toni; Kuhn, Andreas; Nogales, Sergio; Böhm, Helmut J.; Rauh, Andrea *A Neural Network Based Elasto-Plasticity Material Model*. In: ECCOMAS 2012, Vienna

[23] Reckling, Karl-August: *Plastizitätstheorie und ihre Anwendung auf Festigkeitsprobleme*. Berlin/Heidelberg/New York: Springer-Verlag, 1967

[24] Reed, Russell D.; Marks II, Robert J.: *Neural Smithing. Supervised Learning in Feedforward Artificial Neural Networks*. Cambridge, MA; London, England: The MIT Press, 1999

[25] Ripley, B. D.: *Pattern Recognition and Neural Networks*. Cambridge: Cambridge University Press, 1997

[26] Rojas, R.: *Neural Networks. A Systematic Introduction*. Berlin/Heidelberg: Springer-Verlag, 1996

[27] Zhao, K. M., Lee, J. K.: *Material Properties of Aluminium Alloy for Accurate Draw-Bend Simulation*. In: Journal of Engineering Materials and Technology, Vol. 123, p. 287 - 292: ASME: 2001

[28] Zhang, Wohua; Cai, Yuanqiang *Continuum Damage Mechanics and Numerical Applications*. Hangzhou: Zhejiang University Press; Heidelberg/Dordrecht/London/New York: Springer-Verlag, 2010