TAMPERE UNIVERSITY OF TECHNOLOGY

PEKKA SIPPOLA

FINITE ELEMENT SIMULATION OF THICK STEEL PLATE STAMPING

Master of Science Thesis

Examiners: Arto Lehtovaara and Sami Pajunen
Examiner and topic approved by the Faculty council of The Faculty of Automation, Mechanical and Materials Engineering on 9 November 2011

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY
Master's Degree Programme in Mechanical Engineering
SIPPOLA, PEKKA: Finite element simulation of thick steel plate stamping
Master of Science Thesis, 98 pages, 12 Appendix pages
December 2011
Major: Applied Mechanics
Examiner: Professor Arto Lehtovaara and Associate Professor Sami Pajunen
Keywords: finite element method, stamping, simulation, quasistatic

Finite element simulation is a convenient tool for optimising a stamping process. By using it in the design process, the quality of the product can be improved without the expense of constructing a physical model for experimental testing. This thesis was made for the purpose of studying the theory of finite element stamping simulation so that it could be implemented to study a simulation of a specific stamping process performed on a 30 millimeter thick steel plate. The thesis is focused on the numerical finite element simulation of stamping, not on the practical applications of performing these forming processes.

The literature study part of this thesis is mainly focused on the sources of nonlinearity and the solution methods for integrating the nonlinear dynamic equations. The nonlinearity is caused by metal plasticity, large displacements/strains and changing contact conditions. The practical simulation part is performed with Abaqus finite element analysis software suite using both Abaqus/Standard and Abaqus/Explicit codes. A simplified two-dimensional (2D) plane strain model is compared to a more costly, but more thorough, three-dimensional (3D) model. The explicit and implicit solution methods are compared with the different models. Also, a parametrical study on the material properties is performed with the 2D model.

It was found that the studied stamping process is unlikely to succeed in practice without major improvements on the geometry design of the tooling. The 2D model misses important details when compared to the 3D model although offering significant reduction in computational time. Therefore, it is recommended for use only in the initial design process for optimisation purposes. The 3D model is recommended for use in an almost complete design process to verify the results and to further improve the design. Complicated contact conditions caused the simulations with the shell element model to break down so that the simulation had to be performed with a solid continuum element model. The advantages of the explicit solution in the 3D model and the advantages of the implicit solution in the 2D model were recognized. The implicit dynamic solution offers advantages over the static implicit procedure by improved convergence of the iterations when hard contact is modelled. The choice of material model is also an important aspect in the simulation.

TIIVISTELMÄ

TAMPEREEN TEKNILLINEN YLIOPISTO
Konetekniikan koulutusohjelma
SIPPOLA, PEKKA: Finite element simulation of thick steel plate stamping
Diplomityö, 98 sivua, 12 liitesivua
Joulukuu 2011
Pääaine: Teknillinen mekaniikka
Tarkastajat: Professori Arto Lehtovaara ja Yliopistonlehtori Sami Pajunen
Avainsanat: Elementtimenetelmä, simulointi, kylmämuovaus

Elementtimenetelmäsimulointi on hyödyllinen apuväline metallilevyn kylmämuovausprosessin suunnittelussa. Simuloinnin avulla muovausprosessin optimoinnista on mahdollista saada halpa verrattuna siihen, että muovauskomponenteista valmistettaisiin fyysiset mallit koeperäiseen suunnitteluun. Tämä työ on tehty metallilevyn kylmämuovausprosessin elementtimenetelmäsimuloinnin teorian ymmärtämiseksi ja teorian soveltamiseksi tietyn taivutusvaltaisen levymuovausprosessin simuloinnin tutkimiseen. Työ keskittyy levymuovauksen numeeriseen simulointiin, ei siihen kuinka muovaus käytännössä suoritetaan.

Kirjallisuustutkimusosassa esitetään muovausprosessin elementtimalliin liittyviä epälineaarisuuksia ja ne huomioon ottavia ratkaisumenetelmiä. Näitä epälineaarisuuksia ovat metallien plastinen käyttäytyminen, vaihtuvat kontaktiolosuhteet sekä suuret siirtymät ja venymät. Käytännön simultointiosuus suoritetaan Abaqus-ohjelmistolla käyttäen sekä Abaqus/Standard- että Abaqus/Explicit-koodia. Yksinkertaistettua kaksiulotteista (2D) tasovenymämallia verrataan laskennallisesti kalliimpaan, mutta perusteellisempaan kolmiulotteiseen (3D) malliin. Eksplisiittistä ja implisiittistä ratkaisumenetelmää verrataan keskenään jo mainituilla eri malleilla. Myös parametritutkimus suoritetaan materiaaliominaisuuksille tasovenymämallissa.

Työssä huomattiin, että tutkittua muovausprosessia ei todennäköisesti voida suorittaa ilman suuria parannuksia työkalujen geometriaan. 2D-malli ei huomaa kaikkia tärkeitä yksityiskohtia verrattaessa 3D-malliin. Toisaalta 2D-malli on laskennallisesti huomattavasti tehokkaampi, ja siksi sitä suositellaan vain alustavaan suunnitteluun muovausprosessin optimoimiseksi. 3D-mallia suositellaan käytettäväksi lähes valmiissa suunnitteluprosessissa tulosten varmistamiseksi ja suunnitelman parantamiseksi entisestään. Monimutkaiset kontaktiolosuhteet aiheuttivat kuorielementtimallille ongelmia, joten simulointi täytyi suorittaa kontinuumielementeillä. Eksplisiittisen ratkaisumenetelmän edut 3D-mallissa ja implisiittisen ratkaisumenetelmän edut 2D-mallissa huomattiin. Implisiittinen dynaaminen ratkaisumenetelmä tarjoaa etuja implisiittiseen staattiseen ratkaisumenetelmään nähden parantamalla iteroinnin konvergointia, kun kontakti mallinnetaan kovana. Materiaalimallin valinta on myös tärkeä yksityiskohta simuloinnissa.

PREFACE

This thesis was performed during the summer and autumn of 2011 mainly at the Gothenburg office of FS Dynamics Sweden as well as the Tampere office of FS Dynamics Finland.

I would like to express my gratitude to Arttu Kalliovalkama and Carl-Fredrik Stein for providing me the opportunity to perform this thesis and giving me valuable help when it was needed, Robert Lillbacka for his expert supervising at the Gothenburg office, Per Heintz for providing me information on this interesting subject in the beginning of the thesis process, and Lassi Syvänen as well as Rickard Juntikka for providing me expert structural analysis advice at the offices. In addition I would like to thank my thesis supervisors at the Tampere University of Technology, Arto Lehtovaara and Sami Pajunen, and every other person who has helped me in any way during the writing and simulation processes of this thesis.

Tampere, November 18, 2011

TABLE OF CONTENTS

TERMS AND SYMBOLS

$[\mathbf{k}]$	element stiffness matrix
$[\mathbf{B}]$	element strain-displacement matrix
$[\mathbf{C}]$	global damping matrix
$[\mathbf{C^c}]$	contact constraint contribution matrix
$[\mathbf{C_T^c}]$	tangential constraint contribution matrix
$[\mathbf{D}]$	element material matrix
$[\mathbf{D}^*]$	matrix for the plastic stress calculation
$[\mathbf{D_{impl}}]$	consistent material matrix in plastic stress calculation
$[\mathbf{E}]$	Lagrangian strain tensor
$[\mathbf{F}]$	material point deformation gradient in matrix form
$[\mathbf{G^c}]$	contact residual matrix
$[\mathbf{H}]$	logarithmic strain tensor
$[\mathbf{I}]$	identity matrix
$[\mathbf{J}]$	Jacobian matrix for isoparametric mapping
$[\mathbf{K}]$	global stiffness matrix
$[\mathbf{K_t^c}]$	matrix for frictionless Lagrange constraint enforcement
$[\mathbf{K_t^{cP}}]$	matrix for frictionless implicit penalty constraint enforcement
$[\mathbf{K_t^{cP}}]$	matrix for implicit penalty constraint enforcement including friction
$[\mathbf{K^{LM}}]$	stiffness matrix for Lagrange multiplier constraint enforcement
$[\mathbf{K^P}]$	global stiffness matrix for frictionless penalty constraint enforcement
$[\mathbf{K_{impl}}]$	invertable matrix in implicit dynamic direct integration
$[\mathbf{K_t}]$	global tangent stiffness matrix
$[\mathbf{M}]$	global mass matrix
$[\mathbf{N}]$	shape function matrix
$[\mathbf{R}]$	rigid body rotation matrix
$[\mathbf{U}]$	right stretch matrix
$[\mathbf{V}]$	left stretch matrix
$\boldsymbol{\alpha}$	coordinate vector of the center of the yield surface in stress space
$\boldsymbol{\varepsilon}$	vector of total strain state components
$\boldsymbol{\varepsilon}^e$	vector of elastic strain state components
$\boldsymbol{\varepsilon}^p$	vector of plastic strain state components
$\boldsymbol{\sigma}$	vector of stress state components
$\boldsymbol{\Lambda}$	vector of Lagrangian multipliers
$\mathbf{g_T}$	tangential total gap
$\dot{\mathbf{g}}_\mathbf{T}$	rate of tangential total gap
$\mathbf{g_T^{sl}}$	vector of tangential slip
$\mathbf{g_T^{st}}$	vector of tangential stick

$\bar{\mathbf{g}}$	residual vector for strain in the plastic stress iteration
$\mathbf{n}$	direction vector
$\mathbf{r}^{\mathbf{int}}$	vector of local element internal nodal forces
$\mathbf{t_T}$	tangential stress vector
$\mathbf{u}$	vector of material point displacements
$\mathbf{w}$	global vector of displacements and Lagrangian multipliers
$\mathbf{x}$	vector of spatial/Eulerian coordinates
$\mathbf{F}$	vector resulting from linearization of tangential stresses
$\bar{\mathbf{G}}$	residual vector for the structural dynamic equation
$\bar{\mathbf{G}}^{\mathbf{LM}}$	global residual vector for Lagrange multiplier constraint enforcement
$\bar{\mathbf{G}}^{\mathbf{P}}$	global residual vector for penalty constraint enforcement
$\mathbf{R}^{\mathbf{ext}}$	global vector of external nodal forces
$\mathbf{R}^{\mathbf{int}}$	global vector of internal nodal forces
$\mathbf{U}$	global vector of nodal displacements
$\dot{\mathbf{U}}$	global vector of nodal velocities
$\ddot{\mathbf{U}}$	global vector of nodal accelerations
$\mathbf{X}$	vector of material/Lagrangian coordinates
γ	shear strain
ε	total normal strain
ε^f	fracture strain
ε^e	uniaxial elastic strain
ε^p	uniaxial plastic strain
ε^u	necking strain
ε_e	engineering strain
ε_p	equivalent plastic strain
$\hat{\varepsilon_p}$	plastic strain magnitude
ε^m	elongation at break
ε_{log}	uniaxial logarithmic strain
ζ	reference element coordinate
η	reference element coordinate
ι	scalar defining the magnitude of plastic strain
λ	Lagrange multiplier
λ^s	stretch ratio
$\tilde{\lambda}$	Lamé constant
μ	Coulomb friction coefficient
$\tilde{\mu}$	Lamé constant, the shear modulus
ν	Poisson ratio
ξ	reference element coordinate
ρ	density

ρ_r	curvature of a longitudinal grid line at the neck of a tensile specimen
σ	normal stress
σ^u	ultimate strength / ultimate stress
σ^y	yield limit / yield stress
σ_{av}	average axial true stress at the current minimum cross-sectional area of a tensile sp
σ_e	engineering stress
σ_{true}	true stress
σ_{vm}	von Mises equivalent stress
τ	shear stress
ϕ	eigenvector of the right stretch matrix
a	radius of the smallest cross section at the neck of a tensile specimen
b	material parameter for yield determination
c	speed of sound
c_d	dilatational wave speed
c_t	stick constant
f	yield function
g	gravitational acceleration
h	distance
k	spring stiffness
k_p	penalty stiffness
l	length in spatial/Eulerian coordinates
l_0	initial length of a tensile specimen
l_n	current length of a tensile specimen
m	mass
p_N	normal pressure
r	radius of the actual cross section of a tensile specimen
t	time
u	displacement
A	area
A_0	initial cross-sectional area of a tensile specimen
J	determinant of the Jacobian matrix for isoparametric mapping
L	length in material/Lagrangian coordinates
L_{min}	smallest characteristic element length
E	Young's modulus
E_{kin}	kinetic energy
E_{int}	internal energy
E_p	plastic modulus
E_t	tangent modulus
G	shear modulus

K	bulk modulus
P	uniaxial force
R	radius of curvature at the neck of a tensile specimen
Q	plastic potential function
V	volume
2D	two-dimensional
3D	three-dimensional
ALLAE	artificial strain energy for the whole model
ALLIE	total internal strain energy for the whole model
ALLKE	total kinetic energy for the whole model
C3D8R	3D first-order reduced integration solid continuum element
C3D20R	3D second-order reduced integration solid continuum element
CPE4R	first-order reduced integration plane strain solid continuum element
CPE8R	second-order reduced integration plane strain solid continuum element
CPU	central processing unit
ELASE	artificial strain energy magnitude in the element for the whole element
FEA	finite element analysis
LE	logarithmic strain
PEEQ	equivalent plastic strain
PEMAG	plastic strain magnitude
R2D2	rigid link element
R3D4	rigid shell element
S4	first-order fully integrated general-purpose shell element
S4R	first-order reduced integration general-purpose shell element
SI	international system of units
tt	through thickness

1. INTRODUCTION

1.1 Stamping process

Stamping refers to a variety of sheet metal forming processes, e.g. blanking, embossing, bending, flanging and coining [1, p. 393]. In the context of this thesis, stamping refers to a bending-dominated cold forming process where a blank is formed into a specific shape using the tooling: a punch and a die. Stamping is usually performed on sheet metal pieces especially in the automotive industry, see e.g. [2]. The term sheet metal refers to pieces that are less than 6 millimeters in thickness, a thicker piece is considered plate [1, p. 320]. This thesis is focused on a bending-dominated stamping process of a steel plate with a thickness of 30 mm. The plate to be formed is referred to as blank throughout the majority of this thesis. The goal of this thesis is to study the theoretical background of metal cold forming and to use the theoretical knowledge to study this specific forming process and compare the solution methods and modelling considerations.

Illustrative figure demonstrating the stamping process in the context of this thesis is presented in figure 1.1. On the left side of the figure the undeformed blank is placed on top of the die with the punch above both before the forming process. On the right side of the figure the punch has been moved down and the blank has been formed by the punch by forcing it into the die cavity.

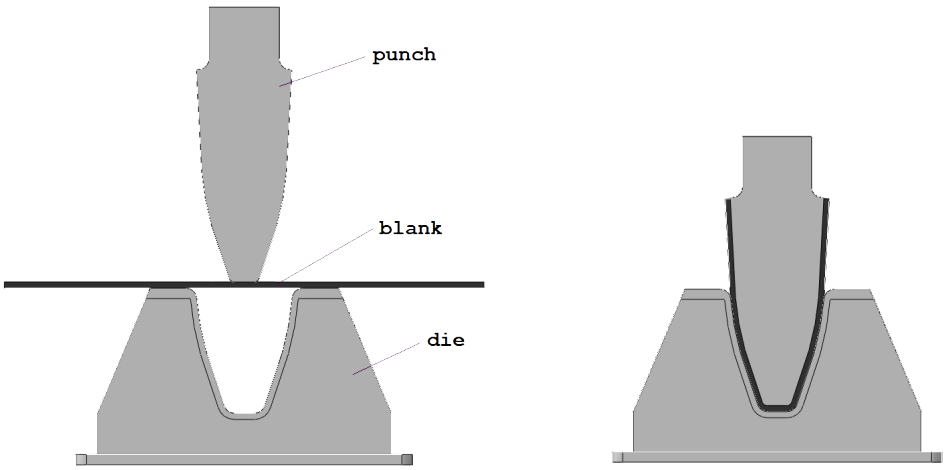


Figure 1.1: Stamping process

This problem is dominated by bending deformation. Sheet metal forming pro-

cesses usually involve stretching and bending deformation of the blank. However, the stretching part is not easily applicable to this forming process as it would require extremely high forces to stretch a plate 30 mm thick. The stretching deformation of the blank during the forming process is usually applied by blank holders in a deep drawing process, see e.g. [2]. No blank holders are present in this thesis.

If the geometry of the die cavity would present the desired shape of the final product, problems may be introduced because of the elastic properties of the material. After the blank is formed and the tooling is removed, the stresses present in the blank, caused by the tools forming it, will cause the blank to deform from the desired geometry to a state in which the internal stresses are in static equilibrium. This undesired deformation caused by the relaxation of the stresses after the removal of tools is called the springback effect. The springback has to be accounted for in the geometry design of the tooling and process parameter selection to optimise the shape of the final product.

1.2 Stamping simulation

A stamping process can be simulated using finite element analysis software to calculate the deformation of the blank and to study the effect of changing forming parameters. This is called stamping simulation. It provides an economic alternative for optimising the stamping process without the expense of manufacturing an actual physical tool. The finite element model used in the simulation has to be accurate in describing the actual physical phenomena involved in a stamping process for obtaining reliable simulation results.

The stamping process involves geometrically large and materially plastic deformations as well as discontinuous contacts between the tools and the work piece. Therefore, the problem involves high nonlinearity and it has to be solved by using a nonlinear solution method. The solution for the finite element analysis problem has to be obtained incrementally in a large number of time steps. This introduces significant amount of computational effort into the simulation. The methods for obtaining the simulation results are discussed in the theory part of this thesis.

High tooling velocities cause effects that are more difficult to control in the forming process. Such effects are, for instance, dynamic impact forces and rate-dependent plasticity. Therefore, the stamping process is usually performed at low enough tool velocities so that the dynamic and inertial effects are neglible. These kind of low velocity processes should be simulated as a quasistatic process, in which the velocity and acceleration terms are not of importance. The simulation can then be performed with a truly static solution procedure although the dynamic solution methods may provide some advantages in the simulation as will be discussed in this thesis.

To minimize unwanted surface wearage and surface traction, lubrication between

the tooling and the blank will have to be used. The lubrication also transfers heat caused by friction and plastic dissipation away from the blank. For quasistatic processes, the heat formed in a stamping process is usually assumed to not have a significant effect on the material properties of the tooling and the blank. For this reason, this thesis does not discuss any thermomechanical effects on the simulation. A small exception to this is the study on the preheating of the blank, where the temperature effects are only included in the material parameters, to see the effects on the forming parameters.

1.3 Notes on the thesis structure

This thesis was performed for the purpose of studying finite element simulation of a specific thick steel plate stamping process. The study is performed by trying out different modelling considerations and solution procedures and comparing their efficiency and accuracy. The thesis can roughly be divided into two parts, the theory part and the practical simulation part.

The chapters 2 to 4, after this introduction chapter, is a literature study on the governing theory of metal cold forming finite element simulation. This includes the nonlinearities involved in the finite element model, to which metal plasticity and contact modelling are devoted their own chapters, and the nonlinear solution methods including the explicit and implicit procedures. It also involves a short discussion on the choice of the elements.

The 5th chapter introduces the simulation model for the practical simulation part which is performed with the Abaqus FEA (finite element analysis) software suite. The implicit Abaqus/Standard code and the explicit Abaqus/Explicit code require different modelling considerations and both of them will be discussed in this chapter. The preprocessing of the model and postprocessing of the results were performed with Abaqus/CAE (CAE = Complete Abaqus Environment) version 6.10-1.

The 6th chapter follows the study on the actual simulation part. The results of the simulation performed with the model introduced in chapter 5 are presented. Some adjustments that the simulation results suggested are applied to the simulation model and a parametrical study on the material properties at different temperatures is performed.

The results will be further analysed in chapter 7. This includes the comparison of the efficiency and accuracy of the solution methods and the simplified models. Some of the initial simulation results with different material model will also be compared to the final simulation results.

Chapter 8 includes the conclusions based on the chapters 6 and 7 and presents the true outcome of this thesis in a summary form.

2. FINITE ELEMENT METHOD

2.1 Dynamic equilibrium equation

The displacement-based finite element method is a numerical method used for mechanical structural analysis. It is based on a mesh discretization of the structure to be analysed by dividing it into a series of smaller regions called elements. These elements are connected to each other at points called nodes. The displacement field of the element-filled domain is then approximately solved using the stiffness properties involved. It is assumed that the reader is familiar with the basics of the finite element method in mechanical analysis and this section is not meant to be a thorough introduction to it. More about the governing theory can be read from the source literature, such as [3] or [4].

The governing equation for structural dynamics, derivation can be read from [3, p. 375], is

$$[\mathbf{M}]\ddot{\mathbf{U}} + [\mathbf{C}]\dot{\mathbf{U}} + [\mathbf{K}]\mathbf{U} = \mathbf{R}^{\mathbf{ext}} \tag{2.1}$$

$$[\mathbf{M}]\ddot{\mathbf{U}} + [\mathbf{C}]\dot{\mathbf{U}} + \mathbf{R}^{\mathbf{int}} = \mathbf{R}^{\mathbf{ext}} \tag{2.2}$$

where $[\mathbf{M}]$ is the mass matrix, $[\mathbf{C}]$ the damping matrix and $[\mathbf{K}]$ the stiffness matrix of the structure. $\mathbf{R}^{\mathbf{ext}}$ is the vector of external nodal forces and $\mathbf{U}$ is the vector of nodal displacements with its corresponding time derivatives nodal velocity $\dot{\mathbf{U}}$ and nodal acceleration $\ddot{\mathbf{U}}$. $\mathbf{R}^{\mathbf{int}}$ in the latter equation is the vector of internal nodal forces defined by the equation

$$\mathbf{R}^{\mathbf{int}} = \sum_e \mathbf{r}^{\mathbf{int}} = \sum_e \int_{V^e} [\mathbf{B}]^T \boldsymbol{\sigma} dV \tag{2.3}$$

in which $[\mathbf{B}]$ is the strain-displacement matrix of the used element, $\boldsymbol{\sigma}$ is the true (Cauchy) stress tensor of a material point in Voigt notation, see e.g. [5, p. 615], and V^e is the element volume. $\mathbf{r}^{\mathbf{int}}$ is the element internal nodal force vector and e below the summation sign implies the summation of the forces over all elements in the model. The latter form of the equation (2.2) is convenient with certain solution procedures considering nonlinear problems where $[\mathbf{K}]$ changes between time steps. The different solution procedures will be discussed shortly in greater detail.

The global stiffness matrix [**K**] is calculated by summation of the individual element stiffness matrices [**k**] as defined by the equation

$$[\mathbf{K}] = \sum_e [\mathbf{k}] = \sum_e \int_{V^e} [\mathbf{B}][\mathbf{D}][\mathbf{B}] dV \tag{2.4}$$

where [**D**] is the material matrix which relates the stress and the strain, it will be discussed further in the section considering elastoplastic material. The material matrix accounts for the material nonlinearities. The integration of the equation (2.4) is performed by means of numerical integration, see e.g. [4, pp. 274].

The stress displacement matrix [**B**] used in equations (2.4) and (2.3) is constant in traditional small-displacement analysis but varies between the increments of the nonlinear solution when the deformations and displacements are large.

The global mass matrix and global damping matrix are assumed to remain constant in this thesis. The damping matrix is often dropped from the equation because of the difficulties in quantifying actual physical damping.

2.2 Direct integration of the equation of motion

Direct integration methods can be used to time integrate the governing dynamic equation to determine the structures dynamic response. In direct integration, the dynamic response history of the structure is determined by dividing the time period of interest into multiple small increments and advancing step-by-step in time evaluating the response at each step. Two methods for the integration will be used in this thesis, one is an explicit central-difference method, and the other is an implicit Euler backward method. The number of the time step, also called time increment, will be denoted with the subscript n throughout this thesis.

2.2.1 Explicit direct integration

Explicit integration methods use only variables known from the current increment n to determine the kinematic state of the system at the next increment $n+1$. The time step that is used when advancing from n to $n+1$ is denoted here as Δt_{n+1}.

An explicit central-difference integration rule is often used in solving the equation of motion (2.1). With the velocity terms lagging by half a time increment, the displacements for $n+1$ are solved from the equations

$$\dot{\mathbf{U}}_{n+\frac{1}{2}} = \dot{\mathbf{U}}_{n-\frac{1}{2}} + \frac{\Delta t_{n+1} + \Delta t_n}{2} \ddot{\mathbf{U}}_n \tag{2.5}$$

$$\mathbf{U}_{n+1} = \mathbf{U}_n + \Delta t_{n+1} \, \dot{\mathbf{U}}_{n+\frac{1}{2}} \tag{2.6}$$

and the initial condition (n=0) can be taken as

$$\dot{\mathbf{U}}_{-\frac{1}{2}} = \dot{\mathbf{U}}_0 - \frac{\Delta t_1}{2}\ddot{\mathbf{U}}_0$$

The nodal acceleration vector is solved from the governing dynamic equation

$$\ddot{\mathbf{U}}_n = [\mathbf{M}]^{-1}\left(\mathbf{R^{ext}}_n - [\mathbf{C}]\dot{\mathbf{U}}_{n-\frac{1}{2}} - \mathbf{R^{int}}_n\right) \tag{2.7}$$

where $[\mathbf{M}]$ is usually taken as the lumped mass matrix of the structure. The equation (2.7) is easy to solve because the lumped mass matrix is a diagonal matrix, and therefore, it is trivial to determine its inverse. The use of the lumped mass matrix reduces the computational cost significantly. See [3, pp. 380-383] for more information on lumping the mass matrix of an element.

Some computational cost involved in this method is introduced by the element-wise evaluation of $\mathbf{r^{int}}$ from (2.3) because of the nonlinear plastic stress-strain relationship involved in some parts of the model in a metal forming simulation. The computational procedure for determining the stress state in elements exhibiting plastic behaviour will be discussed later in the section concerning elastoplastic material.

The same order of quadrature in the element stiffness matrix integration is needed for the evaluation of $\mathbf{r^{int}}$ also. Therefore, reduced integration elements are often used in the method. For a fully integrated linear quadrilateral element with 4 integration points, reduced integration reduces the number of integration points to 1. In this case, the calculation time is reduced to 1/4 compared to the full integration. There is some problems involved with the use of reduced integration elements, these are discussed later on when considering the element selection for the stamping simulation.

Stable time step size

The explicit central-difference integration is stable only when sufficiently small time increments are used. The maximum stable time increment size is estimated by the Courant criterion as

$$\Delta t_{max} = \frac{L_{min}}{c} \tag{2.8}$$

in which L_{min} is the smallest characteristic element length and c is the speed of sound in the material. The speed of sound in the material can be calculated from the Young's Modulus E and the density ρ of the material, $c = \sqrt{\frac{E}{\rho}}$. The criterion is based on the assumption that the time increment must be smaller than the time it takes for information to travel between adjacent nodes in the finite element mesh [3, p. 413]. This criterion implies that the approximate size of the maximum step in sheet forming for most metallic materials is on the order of $10ns$-$2\mu s$ [6, p. 141].

Because of this extremely small time increment, it is not computationally efficient to model the forming process in its natural time period.

Increasing the tool velocity or scaling the mass of the blank to a larger value can be used to increase the stable time step size in the simulation. Both of these techniques also increase the inertia forces of the blank, which might not correspond the physical nature of the stamping process. Therefore, for a quasistatic simulation, a check has to be made that the kinetic energy E_{kin} of the blank does not exceed no more than a small percentage of the blank's internal energy E_{int} throughout the majority of the simulation process. The energies are defined by the equations

$$E_{kin} = \frac{1}{2}\dot{\mathbf{U}}^T[\mathbf{M}]\dot{\mathbf{U}} \qquad E_{int} = \sum_e \int_{V^e} \boldsymbol{\sigma}^T \boldsymbol{\varepsilon} dV^e$$

where internal energy includes the applied elastic strain energy and the energy dissipated by plastic behaviour.

2.2.2 Implicit direct integration

Implicit methods require iterations for equilibrium after each time increment. Compared to a time increment calculated by an explicit method, the calculation time for an implicit increment is usually much longer. On the other hand, most implicit methods are unconditionally stable. This means that the time increment is not limited by numerical stability, only accuracy of the solution introduces limits to the increment size: some details on the loading path might be missed if the size of the used increment is too large. The iteration procedure also ensures that the internal and external forces are in balance after each increment. The subscript $n+1$ is dropped here from the time increment Δt to make the presentation more simple.

A backward Euler scheme can be used for solving the acceleration vector at the end of the step from the equation (2.1). This requires the forming of the global mass matrix for determining the corrected displacements at each iteration. The backward Euler operator yields approximations for the displacements and velocities as

$$\dot{\mathbf{U}}_{n+1} = \dot{\mathbf{U}}_n + \Delta t \ddot{\mathbf{U}}_{n+1} \tag{2.9}$$

$$\mathbf{U}_{n+1} = \mathbf{U}_n + \Delta t \dot{\mathbf{U}}_{n+1} \tag{2.10}$$

Solving for the nodal velocities and accelerations from these approximations as func-

tions of the displacements yields

$$\dot{\mathbf{U}}_{n+1} = \frac{1}{\Delta t}(\mathbf{U}_{n+1} - \mathbf{U}_n) \tag{2.11}$$

$$\ddot{\mathbf{U}}_{n+1} = \frac{1}{\Delta t^2}(\mathbf{U}_{n+1} - \mathbf{U}_n) - \frac{1}{\Delta t}\dot{\mathbf{U}}_n \tag{2.12}$$

Placing these approximations to the governing dynamic equation (2.1), requiring the equation to be satisfied at $n+1$ and denoting $\mathbf{\Delta U}_{n+1} = \mathbf{U}_{n+1} - \mathbf{U}_n$ yields

$$[\mathbf{M}](\frac{1}{\Delta t^2}\mathbf{\Delta U}_{n+1} - \frac{1}{\Delta t}\dot{\mathbf{U}}_n) + [\mathbf{C}](\frac{1}{\Delta t}\mathbf{\Delta U}_{n+1}) + \mathbf{R^{int}}_{n+1} - \mathbf{R^{ext}}_{n+1} = \mathbf{0} \tag{2.13}$$

Here the nodal force vectors $\mathbf{R^{int}}$ and $\mathbf{R^{ext}}$ are dependent of the displacements at $n+1$. Let us denote the residual of this equation (2.13) as $\bar{\mathbf{G}}$ and linearize it with respect to the displacement with the introduction of the global tangent stiffness matrix

$$[\mathbf{K_t}]^i = \frac{\partial \mathbf{R^{int}}}{\partial \mathbf{U}_{n+1}}|_{\mathbf{U}^i_{n+1}} \tag{2.14}$$

a global Newton-Raphson iterative scheme for the displacements at $n+1$ is then obtained as

$$\left(\frac{1}{\Delta t^2}[\mathbf{M}] + \frac{1}{\Delta t}[\mathbf{C}] + [\mathbf{K_t}]^i\right)\mathbf{\Delta U}^{i+1}_{n+1} = \bar{\mathbf{G}}^i_{n+1} \tag{2.15}$$

$$\mathbf{U}^{i+1}_{n+1} = \mathbf{U}^i_{n+1} + \mathbf{\Delta U}^{i+1}_{n+1} \tag{2.16}$$

where i refers to the iteration step. Initial conditions are obtained from the values at step n and the residual is defined by the equation

$$\bar{\mathbf{G}}^i_{n+1} = -[\mathbf{M}]\left(\frac{1}{\Delta t^2}\mathbf{\Delta U}^i_{n+1} - \frac{1}{\Delta t}\dot{\mathbf{U}}_n\right) - [\mathbf{C}]\left(\frac{1}{\Delta t}\mathbf{\Delta U}^i_{n+1}\right) - \left(\mathbf{R^{int}}\right)^i_{n+1} + \left(\mathbf{R^{ext}}\right)^i_{n+1}$$

Note that the global tangent stiffness matrix $[\mathbf{K_t}]$ depends on the displacements $\mathbf{U}^i_{n+1}$ and has to be compiled at each iteration step from the equation (2.4) with the current corrected displacement values. The iteration procedure is carried out until the residual or the change in the displacement is smaller than a specified tolerance. The accelerations and velocities can then be solved from (2.11) with the use of the iterated displacements at n and the displacements from step $n+1$.

No contact conditions is assumed in (2.15). Let us compile the matrix that has to be inverted in (2.15) into a single matrix as

$$[\mathbf{K_{impl}}]^i = \frac{1}{\Delta t^2}[\mathbf{M}] + \frac{1}{\Delta t}[\mathbf{C}] + [\mathbf{K_t}]^i \tag{2.17}$$

In implicit methods, no real advantage is gained when using the lumped mass matrix

because of the summation of the mass matrix to the damping matrix and the tangent stiffness matrix in (2.17). The matrix obtained by this summation has to be inverted to gain the iterative corrections to the displacements. Even though the damping matrix could be diagonal, the stiffness matrix is not. Therefore, the consistent mass matrix is used. It is defined by the equation

$$[\mathbf{M}] = \sum_{e} \int_{V^e} \rho [\mathbf{N}]^T [\mathbf{N}] dV \tag{2.18}$$

where $[\mathbf{N}]$ is a matrix consisting of the shape functions that interpolate the displacements inside the element. The consistent mass matrix is more beneficial to accuracy than the lumped mass matrix when used with implicit methods [3, p. 425].

The backward Euler operator is mainly intended for quasistatic simulations in which an essentially static solution is desired [7, sect. 6.3.2]. In transient dynamic simulations, the Hilbert-Hughes-Taylor α-method should be used [8]. It is a generalization of the Newmark method [9, p. 29].

2.2.3 Selection of the direct integration method

Stability and economy of the solution are important aspects when choosing between the direct integration methods. When the explicit method is used, the number of increments needed for the solution is significantly larger than that of the implicit method. On the other hand, the calculation time for one increment is significantly smaller when using the explicit method. Implicit method also requires much more computer storage space than the explicit method because the global stiffness matrix has to be formed at each iteration.

Contact algorithm failure or convergence issues may arise when using the implicit method, especially when the problem involves a large number of equations to be solved. Smaller increments would be beneficial from the convergence point of view. On the other hand, smaller increments reduce computational efficiency. Also, for sheet metal and plate forming problems, the stiffness matrix can become ill-conditioned because the blank has much lower stiffness in the thickness direction than in other directions of the plate [6, p. 141]. The computational cost of the solution in the implicit method increases more than linearly with the problem size.

The explicit dynamic method is well suitable for dynamic impact problems with relatively small time periods. It can also be used for problems that can be modelled as quasistatic. For quasistatic problems, it is not computationally efficient to model the process in its natural time period as already mentioned. Also, when the forming process is modelled with the explicit method in a large time period, round-off errors may arise. Therefore, increase of the tool speed and mass scaling are important

techniques in quasistatic problems. An advantage of the explicit method in large problems is that the computational cost increases only linearly with the problem size.

A problem considering the springback calculation with the explicit method is the fact that the blank will not be in equilibrium state after the final increment. After removal of the tool contacts, the blank will be oscillating dynamically. It would take a long time for the oscillation to damp out if the springback would be calculated by means of the explicit method. The plastic strains are not affected by this oscillation so that the path to the final stress state after springback will not be of interest. Therefore, it is convenient to model the springback with a true static implicit method in which the acceleration and velocity are not taken into account. This is also the case for the implicit dynamic method. The solution procedure for the implicit static method is similar to that of the implicit dynamic method and is obtained by dropping the terms involving [**M**] and [**C**] from the equations (2.13)-(2.15) including the equation for the residual.

2.3 Element selection

The displacement field in an element is interpolated by the shape functions of each node and the nodal displacements. The shape function of an element node is formulated in such way that it has a value of 1 at its corresponding node and zero value at other nodes of the element.

2.3.1 Isoparametric formulation

Isoparametric elements use same shape functions to interpolate the nodal coordinates and the displacements. The formulation is performed in the local reference coordinates ξ,η and ζ of every element. These coordinates map the physical element into a reference element which has a shape of a square for rectangular quadrilateral elements and a cube for hexahedral elements. This allows for the physical elements to have more flexible shapes. The coordinate transformation to the actual physical coordinates is performed with the use of a Jacobian matrix [**J**]. Thus, the element stiffness matrix is calculated by means of numerical integration from

$$[\mathbf{k}] = \int_{V^e} [\mathbf{B}][\mathbf{D}][\mathbf{B}]dV = \int_{-1}^{-1}\int_{-1}^{-1}\int_{-1}^{-1} [\mathbf{B}][\mathbf{J}]^{-1}[\mathbf{D}][\mathbf{B}][\mathbf{J}]^{-1}Jd\xi d\eta d\zeta \tag{2.19}$$

where J is the determinant of the Jacobian matrix. See e.g. [3, p. 205-219] or [10, p. 104-109] for more information on the isoparametric shape functions and the Jacobian matrix.

2.3.2 Order of interpolation and numerical integration

Fully integrated elements

First-order elements use linear interpolation and only have nodes at the corners of the element. Second-order elements use quadratic interpolation and have nodes at the corners as well as nodes on the midsides. The fully integrated versions of solid continuum elements as planar cases can be seen in figure 2.1 where ξ and η refer to the reference coordinates in two dimensions. For the three-dimensional case, the parallel projection along each of the axis of the master element should look as the one seen in figure 2.1, with the addition of a third reference coordinate ζ.

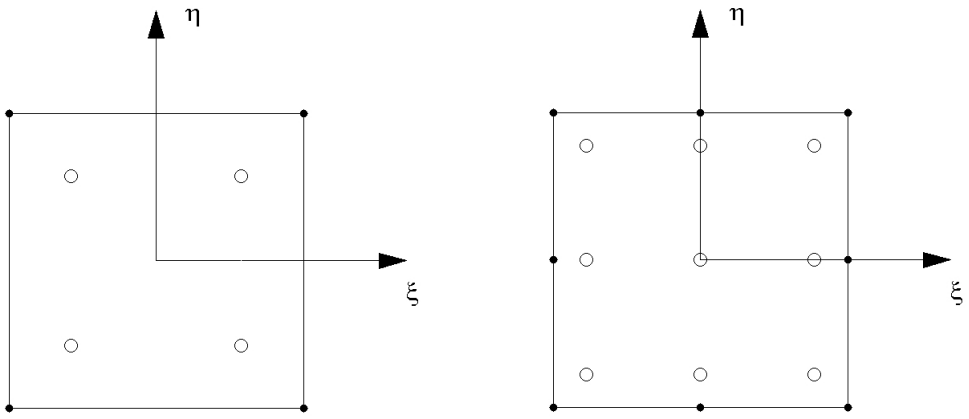


Figure 2.1: Fully integrated first-order (left) and second-order (right) elements in 2D

The Gauss integration point locations are illustrated as the circles inside the element and the node locations are illustrated as the points connected by the lines illustrating the element edges in figure 2.1. Full integration means that the order of numerical integration is sufficient to integrate the stiffness of the element exactly for an undistorted element [3, p. 223]. Thus, when the element is fully integrated, order 2 Gauss rule is used for the first-order elements and order 3 Gauss rule for the second-order elements. The number of nodes for a first-order element in two dimensions and three dimensions are 4 and 8, respectively. The second-order element has 8 nodes in two dimensions and 20 nodes in three dimensions.

Actually the second-order element introduced here is a serendipity element. An alternate Lagrange element in two dimensions would have internal nodes also [3, p. 97], and internal nodes as well as surface nodes in three dimensions. An advantage of the serendipity elements is that the size of the element matrices become smaller while the internal nodes of second-order Lagrange elements would not contribute to the element connectivity.

Reduced integration elements

Reduced integration means that the order of numerical integration is one order less than that of the full integration. It can be advantageous to use this integration order because of the displacement formulation resulting in an overestimation of the system stiffness [4, p. 282]. Also, there are some problems involved with the use of fully integrated elements that will be discussed later on in this thesis. See figure 2.2 for an illustration of the reduced integration quadrilateral elements in a two-dimensional case.

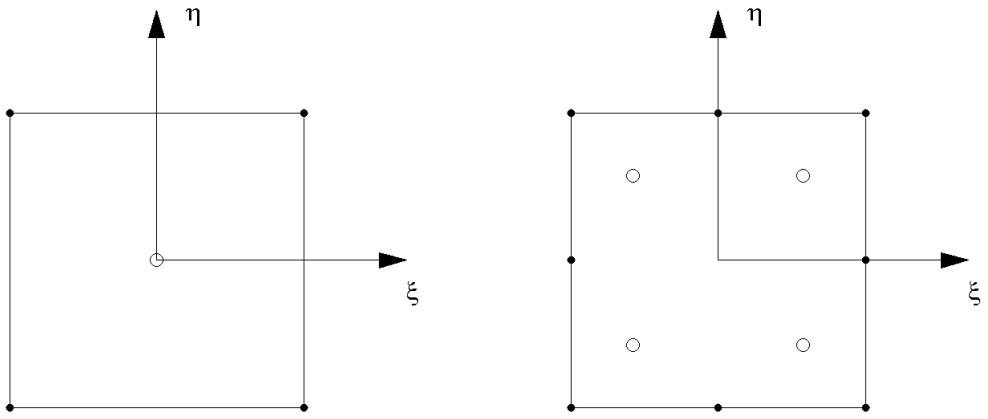


Figure 2.2: Reduced integration first-order (left) and second-order (right) elements in 2D

With the first-order element, the number of integration points has decreased to 1 from 4 and 8 in the two-dimensional and three-dimensional cases, respectively, when compared to the fully integrated version. With the second-order element, the number of integration points has decreased to 4 from 9 in the two-dimensional case and 8 from 27 in the three-dimensional case.

2.3.3 Element families

Two kinds of element families will be used in this thesis. The first one is the family of solid continuum elements which is the most used element family in this thesis. It has displacement degrees of freedom only and is intended for modelling a material continuum.

The other family is the family of shell elements. These structural elements may be used for modelling a structure with one dimension significantly smaller than the others. They use plane stress formulation but differ from the plane stress solid continuum elements in the way that they have rotational degrees of freedom in addition to the displacement degrees of freedom to model out-of-plane bending. Thus, out-of-plane loading is accounted for in their formulation also. The directions on the shell surface coinciding with the plane stress directions are referred to as

membrane directions. For more information on these elements, see for example [4, p. 251] or [3, p. 561-588].

2.3.4 Locking and spurious modes

Modelling bending with solid continuum elements

If bending-related problems are to be modelled with solid continuum elements, special care has to be taken on the selection of the elements. The problems involved are related to phenomena called shear locking and hourglassing.

First-order fully integrated solid continuum elements exhibit shear locking when used in a bending-related problem. Shear locking means that the model behaves overly stiff when compared to the physical nature of the problem. This is because of the element formulation: the element detects nonphysical shear stresses at integration points so that the energy that should be used for bending the element is gone to shear deformation. Therefore, first-order fully integrated solid elements should not be used in regions of the model that are subjected to bending. Shear locking can be avoided using reduced integration elements, although they have another problem involved in bending-related problems called hourglassing.

First-order reduced integration elements exhibit hourglassing in a bending-dominated problem if the element mesh is too coarse. The element does not detect bending strain because only one integration point is used in the element. If only one element through the thickness of the structure is used, the integration point lies on the neutral axis of the bending strain and will not detect bending strain at all. This is a zero-energy deformation mode, also called a spurious mode, and it would lead to an overly flexible behaviour of the structure. Hourglassing can be compensated by improving mesh density: multiple elements through the thickness of a bending-dominated region will give more accurate results related to the bending strains. Methods called hourglass control is often used in first-order reduced integration element formulations. Some of them include hourglass shape/base vectors that are used to define a set of hourglass-resisting forces that try to control the hourglass modes, see for example [11]. For a more detailed demonstration of these spurious modes, see e.g. [3, p. 223-227].

Modelling incompressible materials with solid continuum elements

Fully integrated elements may also suffer from volumetric locking when modelled with a nearly incompressible material, such as rubber or a metal experiencing large plastic strains. This is because the interpolation functions are not properly able to approximate a strain field that preserves the volume of the element. The volumetric strain that might occur at an integration point causes a very high contribution to

virtual power. This problem can not be avoided by refining the mesh but it can be avoided by using first-order reduced integration elements with one integration point so that the incompressibility constraints can be met.

2.4 Finite strain

The commonly known infinitesimal strain theory bases its formulations on the assumption of small displacements and strains. In infinitesimal strain theory, the difference between the initial and current configuration is minimal and the respective coordinates need not to be distinguished. However, when the finite element problem involves large displacements and strains, the finite strain theory should be used. This calls for the use of two different coordinates, the material coordinates and the spatial coordinates, to ensure a clear distinction between the undeformed and deformed configurations.

This section is based on continuum mechanics theory but it is included in the finite element method chapter as the theory is applicable to the finite element discretization also. In finite element applications the measures presented here have to be treated incrementally.

2.4.1 Material and spatial coordinates

The material point (referred here also to as particle) deformation gradient $[\mathbf{F}]$ in matrix form is defined by the equation

$$[\mathbf{F}] = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} \tag{2.20}$$

where $\mathbf{X}$ and $\mathbf{x}$ are the coordinate vectors of the material point at the reference position and at the current position, respectively. The reference position coordinates are Lagrangian coordinates, also called material coordinates, of the particle and the current position coordinates are the particle's Eulerian coordinates, also called spatial coordinates. The coordinate system of Lagrangian coordinates moves with the particle during deformation while Eulerian coordinates measure the current position of the particle with the coordinate system staying fixed in space.

The history of the current location of the particle can be written in equation form as

$$\mathbf{x} = \mathbf{x}(\mathbf{X}, t) \tag{2.21}$$

The current displacement of the particle can then be defined as $\mathbf{u} = \mathbf{x}(\mathbf{X}, t) - \mathbf{X}$. The initial reference coordinates can be taken as the spatial coordinates at $t = 0$, mathematically written as $\mathbf{x}(\mathbf{X}, 0) = \mathbf{X}$.

2.4.2 Stretch ratio

Denoting an infinitesimal gauge length of a material fiber in an arbitrary direction at the initial position as $d\mathbf{X}$, the infinitesimal reference length of the fiber dL and its current length dl are defined by the equations

$$dL = \sqrt{d\mathbf{X}^T d\mathbf{X}} \quad \text{and} \quad dl = \sqrt{d\mathbf{x}^T d\mathbf{x}}$$

By using the mapping (2.21), we can write

$$d\mathbf{x} = [\mathbf{F}]d\mathbf{X} \tag{2.22}$$

A stretch ratio λ^s for the infinitesimal gauge length can then be defined by the equation

$$\lambda^s = \frac{dl}{dL} = \sqrt{\frac{d\mathbf{X}^T[\mathbf{F}]^T[\mathbf{F}]d\mathbf{X}}{d\mathbf{X}^T d\mathbf{X}}} \tag{2.23}$$

where the connection in equation (2.22) was used for dl.

2.4.3 Polar decomposition of the deformation gradient

According to the polar decomposition theorem [12, p. 463], the deformation gradient (2.20) can be composed into a symmetric pure stretching part and an orthogonal rigid body rotation part as

$$[\mathbf{F}] = [\mathbf{R}][\mathbf{U}] = [\mathbf{V}][\mathbf{R}] \tag{2.24}$$

where $[\mathbf{R}]$ is the pure rigid body rotation matrix, $[\mathbf{U}]$ is the right stretch matrix and $[\mathbf{V}]$ the left strain matrix. The two forms of the equation exist because every homogeneous deformation can be decomposed into a stretch followed by a rotation, or into a rotation followed by a stretch. $[\mathbf{U}]$ is used when pure stretching precedes the rotation and $[\mathbf{V}]$ is used when pure stretching follows the rotation. The stretch matrices have the same eigenvalues λ_i^s but the eigenvectors differ: If we denote the eigenvectors of $[\mathbf{U}]$ as $\boldsymbol{\phi}_i$, the eigenvectors for $[\mathbf{V}]$ are obtained by using the rotation matrix as $[\mathbf{R}]\boldsymbol{\phi}_i$. The equation (2.24) distuingishes the straining part of the motion, described by $[\mathbf{U}]$ or $[\mathbf{V}]$, from the rigid body rotation part of the motion described by $[\mathbf{R}]$. The rigid body translation is not important in this context since the relative motion of adjacent material points, which is the deformation of the material, is only of interest when linking the kinematics of the motion to the constitutive behaviour of the material. The constitutive behaviour of an elastoplastic material is discussed in chapter 3 of this thesis.

The following relations [13, p. 52] exist in the polar decomposition theorem:

$$[\mathbf{U}]^2 = [\mathbf{F}]^T[\mathbf{F}], \quad [\mathbf{V}]^2 = [\mathbf{F}][\mathbf{F}]^T, \quad [\mathbf{V}] = [\mathbf{R}][\mathbf{U}][\mathbf{R}]^T \quad \text{and} \quad [\mathbf{V}]^2 = [\mathbf{R}][\mathbf{U}]^2[\mathbf{R}]^T$$

$[\mathbf{U}]^2$ and $[\mathbf{V}]^2$ are called the right and the left Cauchy-Green deformation tensors, respectively. The eigenvalues for $[\mathbf{U}]^2$ and $[\mathbf{V}]^2$ are squares of the principal stretches $(\lambda_i^s)^2$ associated with the principal directions $\boldsymbol{\phi}_i$ or $[\mathbf{R}]\boldsymbol{\phi}_i$, respectively. The tensorial square roots of these tensors are obtained by means of spectral decomposition as

$$[\mathbf{U}] = \sum_{i=1}^{3} \lambda_i^s \boldsymbol{\phi}_i \boldsymbol{\phi}_i^T \quad \text{and} \quad [\mathbf{V}] = \sum_{i=1}^{3} \lambda_i^s \left([\mathbf{R}]\boldsymbol{\phi}_i\right)\left([\mathbf{R}]\boldsymbol{\phi}_i\right)^T \tag{2.25}$$

This requires the solving of the squares of the principal stretches with the associated principal directions as the (right or left) Cauchy-Green deformation tensor eigenvalues from

$$det([\mathbf{F}]^T[\mathbf{F}] - (\lambda^s)^2[\mathbf{I}]) = 0 \quad \text{or} \quad det([\mathbf{F}][\mathbf{F}]^T - (\lambda^s)^2[\mathbf{I}]) = 0 \tag{2.26}$$

and the eigenvectors from

$$[\mathbf{F}]^T[\mathbf{F}]\boldsymbol{\phi} = (\lambda^s)^2\boldsymbol{\phi} \quad \text{or} \quad [\mathbf{F}][\mathbf{F}]^T([\mathbf{R}]\boldsymbol{\phi}) = (\lambda^s)^2([\mathbf{R}]\boldsymbol{\phi}) \tag{2.27}$$

The rotation matrix can be obtained from (2.24) as

$$[\mathbf{R}] = [\mathbf{F}][\mathbf{U}]^{-1} = [\mathbf{V}]^{-1}[\mathbf{F}] \tag{2.28}$$

The determination of the inverses of the stretch matrices is trivial because the stretch matrices are constructed from their eigenvalues and eigenvectors (2.25). The inverses are obtained by replacing λ_i^s with $(\lambda_i^s)^{-1}$ in equations (2.25).

The strain state of the material point can be determined from the stretch matrix by attaching it into a coordinate system. Different formulations for strain tensors exist, some of them will be discussed next. The Lagrangian description (in reference coordinates) with the right stretch matrix $[\mathbf{U}]$ will be used.

2.4.4 Strain tensors

A general formula for Lagrangian strain tensors $[\mathbf{E}]_{(m)}$ can be defined by the equation

$$[\mathbf{E}]_{(m)} = \frac{1}{2m}\left([\mathbf{U}]^{2m} - [\mathbf{I}]\right)$$

where $[\mathbf{I}]$ is the identity matrix. For $m = 1$ this is the Green-Lagrangian strain tensor, and for $m = \frac{1}{2}$ this is the Biot strain tensor. A particular case of interest in

problems involving material nonlinearity is the limit case when $m = 0$:

$$[\mathbf{H}] = \lim_{m \to 0} \frac{1}{2m} \left([\mathbf{U}]^{2m} - [\mathbf{I}] \right) = \ln [\mathbf{U}] \tag{2.29}$$

where $[\mathbf{H}]$ is called the logarithmic strain tensor (also natural/true/Hencky strain). This tensor reserves the tension/compression-symmetry, volumetric-deviatoric decomposition is additive with it, and two subsequent transformations are additive when the principal stretch directions are the same [12, p. 466].

The Green-Lagrangian strain tensor is computationally more efficient than the logarithmic strain tensor because it can be computed directly from the deformation gradient without the need of the polar decomposition solution for the principal stretches and their directions [14, p. 35]. However, the logarithmic strain measure is more suitable for metal plasticity and the Green-Lagrangian strain measure should only be used when the strains are small (rotations can be large).

By using the principal stretches, the principal logarithmic strains are obtained from the equation

$$\varepsilon_i = \ln \lambda_i^s \tag{2.30}$$

and the corresponding principal directions are ϕ_i. This defines the strain state of the material point completely.

3. ELASTOPLASTIC MATERIAL

3.1 Uniaxial behavior

A uniaxial stress-strain curve describing the behavior of structural steel is presented in figure 3.1.

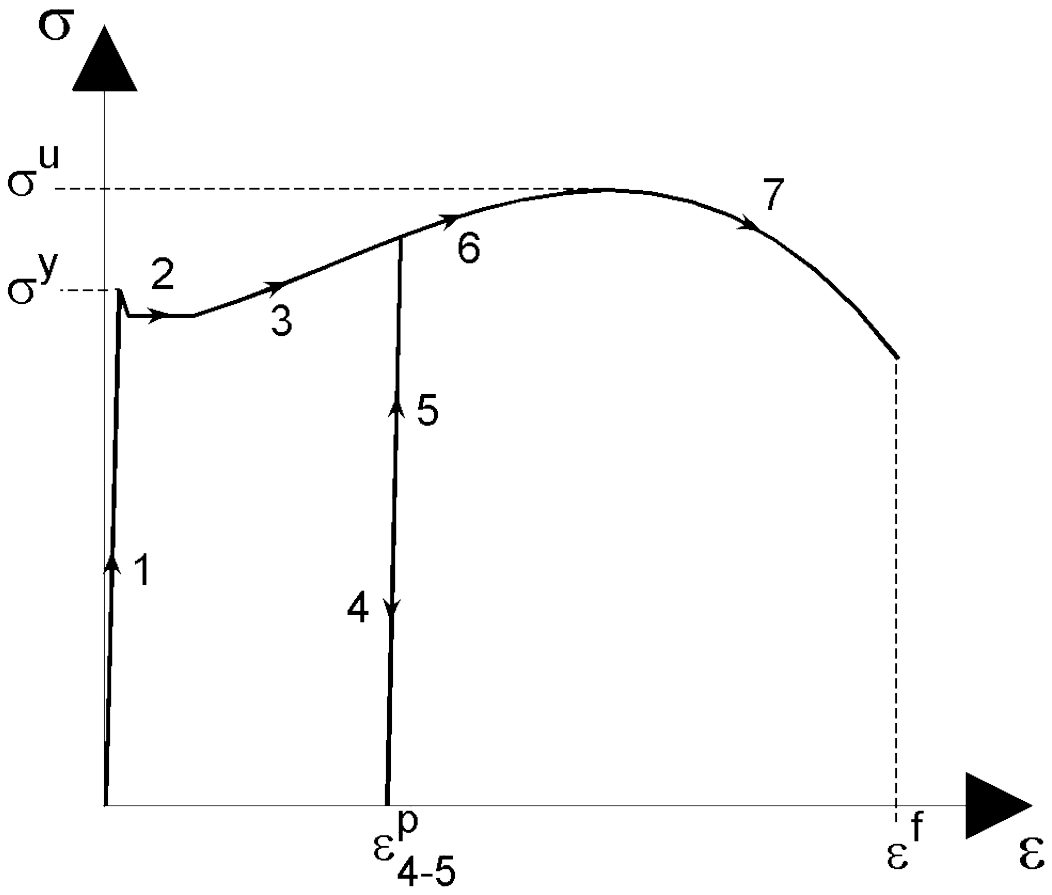


Figure 3.1: Typical $\sigma\epsilon$ curve for structural steel

A specific curve for a given material can be obtained by means of a tensile test. The test is performed by slowly extending the material specimen and measuring the tensile force and the specimen length. More about these tests can be read from [15].

The material exhibits linear elastic behavior when the value of stress σ is less than the yield limit σ^y of the material. This linear elastic behaviour can be seen in figure 3.1 as phase 1. Elastic behaviour means that if the load would be removed and the stress would decrease to zero value, the total strain would also return to zero

value, in other words, the strain is fully reversible. The linear elastic stress-strain relationship is known as Hooke's law and it can be expressed as a function

$$\sigma = E\varepsilon \tag{3.1}$$

in which the slope of the curve E is the Young's modulus of the material and ε is the total uniaxial strain. The yield limit is usually preceded by a proportional limit beyond which the response is still elastic but not linear. The values of the yield limit and the proportional limit do not usually differ much [16, p. 8], and therefore, the proportional limit is not presented in figure 3.1.

Mild steel also exhibits a lower yield limit seen as the drop in the value of σ after the upper yield limit has been reached. Lower yield limit can be used as a conservative value for the yield limit of the material if it is not supposed to yield.

When the stress reaches the yield strength, the material yields and irreversible deformations occur. When the material has yielded, the total strain consists of an irreversible plastic part ε^p and a reversible elastic part ε^e, incrementally written as

$$d\varepsilon = d\varepsilon^e + d\varepsilon^p \tag{3.2}$$

In phase 2 of the curve, the strain grows without any increase in stress. This behaviour is referred to as plastic flow [16, p. 8]. The equation (3.2) could also be written in rate form but the incremental presentation is used here as the plasticity in this thesis is assumed to be rate-independent.

The plastic flow is followed by phase 3 in which the material exhibits work hardening. This means that the value of stress increases as a function of strain during yielding.

If the load is removed (phase 4), the stress as well as the elastic strain will decrease to zero, but the irreversible plastic strain ε^p_{4-5} will remain. Upon reloading in phase 5, the material exhibits linear elastic behavior until the stress reaches the point between phases 3-4, which is the new yield limit. The value of the yield strength has increased because of work hardening.

The unloading curve during phase 4 is only approximately linear. Therefore, a closed hysteresis loop remains between the curves of phase 4 and the linear reloading curve of phase 5. The area of the loop is related to the plastic dissipation energy lost in the process. This phenomenon is important only in cyclic loading that involves plastic behaviour and is not discussed in this context further.

After the new yield limit has been reached, further plastic straining coupled with work hardening of the material occurs as seen in phase 6 of figure 3.1. This phase continues until the stress reaches the ultimate strength of the material σ^u and a neck begins to form in the tensile specimen. This is followed by an instable decrease in

the cross-sectional area of the tensile specimen. The necking can be seen as phase 7 in figure 3.1. The necking stage is followed by the fracture of the tensile specimen at which the strain is specified by the fracture strain ε^f.

The previously introduced measures of stress and strain are related to the initial geometry of the tensile specimen and do not take the changes in the cross-sectional area nor the stress/strain localisation into account. Alternative measures for stress and strain and a short introduction to a method for obtaining the true stress at the necking phase are discussed next.

3.1.1 Strain measures in tensile tests

The engineering strain ε_e corresponds to the engineering stress σ_e and these measures are defined by the equations

$$\varepsilon_e = \frac{l_n - l_0}{l_0} \quad \text{and} \quad \sigma_e = \frac{P}{A_0} \tag{3.3}$$

where A_0 is the initial stress-free cross-sectional area, l_n the current length and l_0 the initial stress-free length of the tensile specimen. P is the axial force acting on the tensile specimen. These are the strain and stress measures used in figure 3.1.

The true stress is defined by the current area A of the tensile specimen by the equation

$$\sigma_{true} = \frac{P}{A} \tag{3.4}$$

The tensile specimen will exhibit reduction in its cross-sectional area already at the elastic stage of the test through the Poisson effect. This reduction is not as drastic as the reduction in the specimen cross-sectional area when the material yields. Therefore, the engineering stress could be used within the linear elastic region without significant error for metals, but if the material yields, the true stress measure should be used.

The plastic deformation of metallic materials is usually assumed not to change the volume of the sample. By taking this assumption into account and assuming that the reduction of the cross-sectional area caused by the elastic strain is negligible, we can write a connection $A_0 l_0 = A l_n$ which leads to the equation connecting the engineering stress and the true stress

$$\sigma_{true} = \sigma_e(1 + \varepsilon_e) \tag{3.5}$$

This equation holds until the neck forms in the tensile specimen when the stress/strain has not localized at the necking area. In the necking stage one would need more accurate measurement of the localized deformation at the neck. This could be achieved by means of optical strain measurement and digital image analysis, see e.g. [17, p.

78].

A strain measure often used in conjuction with the true stress is the logarithmic strain ε_{log}, also called the true strain. It takes the incremental strain as the incremental increase in the length of the tensile specimen dl_n divided by the current length of the specimen

$$\varepsilon_{log} = \int_{l_0}^{l_n} \frac{1}{l_n} dl_n = \ln(\frac{l_n}{l_0}) \tag{3.6}$$

An equation connecting the logarithmic strain and the engineering strain can be obtained by comparing the equations (3.3) and (3.6), and noting that $\Delta l = l_n - l_0$, as

$$\varepsilon_{log} = \ln(\frac{l_n}{l_0}) = \ln(\frac{l_0 + \Delta l}{l_0}) = \ln(1 + \varepsilon_e) \tag{3.7}$$

This connection together with equation (3.5) can be used for obtaining a true stress / true strain relation when the tensile test results are reported in terms of the engineering stress and engineering strain. The engineering and logarithmic strain measures are almost equal at small strains.

The presented stress measures were assumed to be distributed uniformly in the cross sectional cut of the tensile specimen. In a nonuniform case, the theoretical value of the stress at a material point is defined as

$$\sigma = \lim_{\Delta A \to 0} \frac{\Delta P}{\Delta A} \tag{3.8}$$

This is taken only as a theoretical definition as it is very difficult to measure ΔP and ΔA independently. Only the average stress at the cross-sectional area cut can be determined experimentally by means of a traditional tensile test.

At the necking phase, more accurate measurement of the local cross-sectional area reduction is needed because of the localisation of the stress and the strain. Also, the stress state is not uniaxial in the formed neck anymore. A correction method for handling the stress multiaxiality in the necking phase is discussed next.

3.1.2 Bridgman correction method

In the necking stage, the state of stress changes from the simple uniaxial stress state to a more complex triaxial or biaxial stress state. This complex state of stress depends on the geometry of the tensile specimen. For the necking stage of the tensile test, neither of the simple uniaxial stress/strain measures are accurate. Bridgman's correction method [18] is commonly used to obtain a correction in the uniaxial stress state for a rod-shaped tensile specimen in the necking stage.

The Bridgman correction method assumes a uniform strain distribution in the

minimum cross-sectional area and that a longitudinal grid line on the tensile specimen is assumed to deform into a curve at the neck with its curvature ρ_r defined by the function

$$\frac{1}{\rho_r} = \frac{r}{aR}$$

where r is the radius of the actual cross section (not at the neck), a is the radius of the smallest cross section (at the neck) and R is the radius of curvature of at the neck on the surface of the tensile specimen. Also, the ratio of principal stresses are assumed to remain constant during the loading.

By using these assumptions, the radial stress σ_r and the axial stress σ_a in the neck of the tensile specimen can then be defined by the equations

$$\sigma_r = \frac{\sigma_{av}}{\left(1 + \frac{2R}{a}\right)} \left[\frac{\ln\left(\frac{a^2+2aR-r^2}{2aR}\right)}{\ln\left(1 + \frac{a}{2}R\right)} \right]$$

$$\sigma_a = \frac{\sigma_{av}}{\left(1 + \frac{2R}{a}\right)} \left[\frac{1 + \ln\left(\frac{a^2+2aR-r^2}{2aR}\right)}{\ln\left(1 + \frac{a}{2R}\right)} \right]$$

where σ_{av} is the average axial true stress defined by the current minimum cross-sectional area of the tensile specimen, assuming the stress to be uniformly distributed in the cross-sectional area. The shear stresses disappear at the smallest cross section and an equivalent uniaxial von Mises stress can then be calculated from the stress components as

$$\sigma_{true} = \left[\left(1 + \frac{2R}{a}\right) \ln\left(1 + \frac{a}{2R}\right) \right]^{-1} \sigma_{av} \tag{3.9}$$

This method requires a series of tests involving different loadings to determine the measures R and a. These measures are difficult to measure with sufficient accuracy. Therefore, the method is quite complicated to use in practice.

This correction method should only be applied to round tensile specimens, see e.g. [19] for information for the case of flat tensile bars. For flat tensile bars, two types of necking must be considered. The other one is diffuse necking, which is similar to the necking of round tensile specimens, and the other one is called localized necking where the neck is a narrow band at an angle to the specimen axis at the diffused neck. The localized neck often follows the diffused neck and it makes the thickness along the necking band shrink rapidly. See [20] for an illustrative presentation on this subject. It is theoretically possible, but very difficult and expensive in practice, to obtain a correction method for the flat tensile bars also.

3.2 Yield function

The uniaxial yield criterion has to be generalized for a multiaxial case also. This generalization is handled in the form of a yield function.

The theoretical yielding of a material can be determined from a set of material parameters and the stress state of the material point of interest. Therefore, the yield criterion can be expressed with a mathematical function f called the yield function as

$$f(\boldsymbol{\sigma}, b_1, b_2, ...) \leq 0 \tag{3.10}$$

in which $\boldsymbol{\sigma}$ is a vector consisting of all stress components of a material point in a chosen coordinate system and b_i are material parameters. The criterion implies that yielding is present in the material point if $f > 0$ and it is in an elastic state if $f \leq 0$.

3.2.1 von Mises yield function for isotropic material

The stress state of a single material point can be expressed with principal stresses σ_i and their corresponding principal direction vectors $\mathbf{n}_i$, $i = 1, 2, 3$ for a 3D stress case. For an isotropic material, the material properties are identical in all directions. Therefore, no direction vectors are needed to determine the yielding of an isotropic material. A common yield function used for modelling of isotropic metallic materials is the von Mises yield function which can be expressed with the principal stresses and a single material parameter as

$$f = (\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_1 - \sigma_3)^2 - 6b^2 \tag{3.11}$$

The von Mises yield criterion assumes that the yielding of the material is independent of the hydrostatic stress component of the stress state of a material point [16, p. 72]. The criterion is based on the assumption that distortional energy has reached a critical value when yielding is occurring.

The parameter b can be determined using the yield strength σ^y from uniaxial test results of the material. When the material yields in a uniaxial tensile test, the values of the stress components are $\sigma_1 = \sigma^y$ and $\sigma_2 = \sigma_3 = 0$. Substituting the corresponding stress values into the yield function (3.11) and solving for b, the material parameter is obtained as $b = \frac{1}{\sqrt{3}}\sigma^y$. The von Mises yield criterion can now be expressed as a function of the principal stresses and the uniaxial yield strength of the material

$$f = (\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_1 - \sigma_3)^2 - 2(\sigma^y)^2 \tag{3.12}$$

The von Mises yield function forms a convex yield surface when $f = 0$ is plotted

in the principal stress space. The surface can be described as an infinite length cylinder with its axis on the line $\sigma_1 = \sigma_2 = \sigma_3$. A cut of the surface on the plane stress ($\sigma_3 = 0$) plane can be seen in figure 3.2a and 3.2b is the parallel projection of the surface along the $\sigma_1 = \sigma_2 = \sigma_3$-axis (also known as π plane projection).

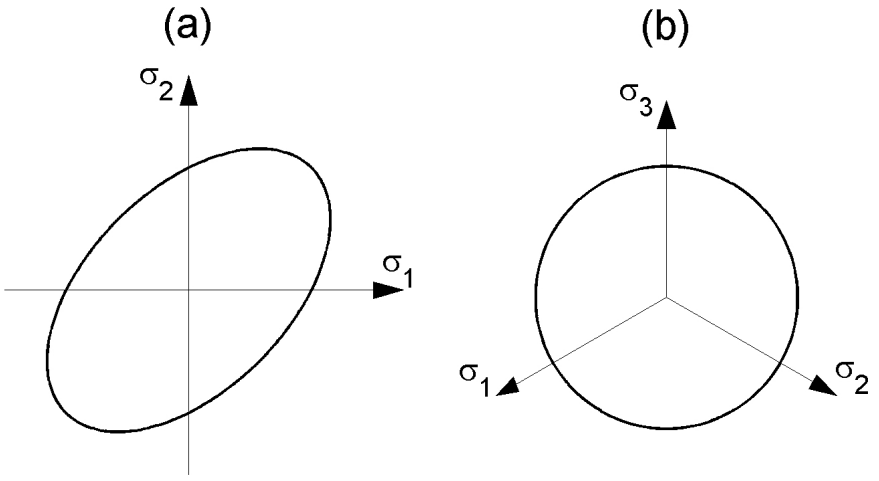


Figure 3.2: von Mises yield surface

For easier comparison between the stress state and the yield strength, a von Mises equivalent stress σ_{vm} is used. It can be expressed as a function of the stress state as

$$\sigma_{vm} = \sqrt{\frac{1}{2}(\sigma_1 - \sigma_2)^2 + \frac{1}{2}(\sigma_2 - \sigma_3)^2 + \frac{1}{2}(\sigma_1 - \sigma_3)^2} \tag{3.13}$$

By incorporating it into the yield function $f = \sigma_{vm} - \sigma^y$ and comparing this to the yield criterion (3.10), it can be seen that the material is assumed to yield when $\sigma_{vm} > \sigma^y$.

3.2.2 Quadratic Hill yield function for anisotropic material

The von Mises yield criterion assumes isotropic material properties. Often the manufacturing process of a sheet metal or a plate involves methods, such as cold rolling or hot rolling, that produce anisotropic material properties to the blank. In these cases the use of an anisotropic yield function for determining the onset of yielding should be considered. Hill [21] proposed an anisotropic yield function which can be written for an orthotropic material with three mutually orthogonal planes of symmetry in the form

$$f = F(\sigma_{22}-\sigma_{33})^2+G(\sigma_{33}-\sigma_{11})^2+H(\sigma_{11}-\sigma_{22})^2+2L\tau_{23}^2+2M\tau_{31}^2+2N\tau_{12}^2-1 \tag{3.14}$$

where σ_{11},σ_{22} and σ_{33} are the normal stresses coinciding the axes of anisotropy, not to be confused with the principal stresses, and the shear stresses τ_{23}, τ_{31} and τ_{12} are

taken with respect to these axes as well. F,G,H,L,M and N are constants defined by the equations

$$\begin{aligned} F &= \frac{1}{2}\left[\frac{1}{(\sigma_{22}^y)^2}+\frac{1}{(\sigma_{33}^y)^2}-\frac{1}{(\sigma_{11}^y)^2}\right] \quad , \quad L = \frac{1}{2(\tau_{23}^y)^2}, \\ G &= \frac{1}{2}\left[\frac{1}{(\sigma_{33}^y)^2}+\frac{1}{(\sigma_{11}^y)^2}-\frac{1}{(\sigma_{22}^y)^2}\right] \quad , \quad M = \frac{1}{2(\tau_{31}^y)^2}, \\ H &= \frac{1}{2}\left[\frac{1}{(\sigma_{11}^y)^2}+\frac{1}{(\sigma_{22}^y)^2}-\frac{1}{(\sigma_{33}^y)^2}\right] \quad \text{and} \quad N = \frac{1}{2(\tau_{12}^y)^2} \end{aligned}$$

where σ_{ii}^y and τ_{ij}^y are experimentally determinable yield strengths in the corresponding directions. This requires for the determination of six material parameters.

The Hill yield criterion is pressure independent and predicts the same yield strength in tension and compression. It reduces to the von Mises yield criterion when $\sigma_{11}^y = \sigma_{22}^y = \sigma_{33}^y = \sqrt{3}\tau^y$ and $\tau_{23}^y = \tau_{31}^y = \tau_{12}^y = \tau^y$, then $F = G = H = 0.5$ and $L = M = N = 1.5$.

More general yield functions for orthotropic anisotropy are the Hoffman criterion and Tsai-Wu criterion [12, p. 352].

3.3 Flow rule

The direction of the increment of plastic strain $d\varepsilon^p$ in the stress space must also be defined for a multiaxial case. This definition is made in the form of a flow rule which can be expressed in equation form as

$$\boldsymbol{d\varepsilon^p} = d\iota \frac{\partial Q}{\partial \boldsymbol{\sigma}} \tag{3.15}$$

Where $d\iota$ is the increment of a scalar that defines the size of the plastic strain and Q is the plastic potential function.

An associative flow rule often used in conjuction with the von Mises yield surface defines the direction of plastic straining to be normal to the yield surface, which is the case when the yield function is used as the plastic potential function, $Q = f$ in (3.15). This associates the flow rule to the yield function.

The condition for plastic flow can be defined as $d\iota > 0$ while $f = 0$. No plastic straining occurs ($d\iota = 0$) if $f \leq 0$.

3.4 Equivalent plastic strain

A scalar measure equivalent to the plastic strain state of a material point can be obtained, see e.g. [16, p. 258], incrementally as

$$d\varepsilon_p = \sqrt{\frac{2}{3}(\boldsymbol{d\varepsilon^p})^T \boldsymbol{d\varepsilon^p}} \tag{3.16}$$

where ε_p is called equivalent plastic strain. When this is integrated through the loading path, we get the form

$$\varepsilon_p = \int_0^t \sqrt{\frac{2}{3}(\boldsymbol{d\varepsilon^p})^T \boldsymbol{d\varepsilon^p}} \quad dt \tag{3.17}$$

A measure called plastic strain magnitude is defined as

$$\hat{\varepsilon}_p = \sqrt{\frac{2}{3}(\boldsymbol{\varepsilon^p})^T \boldsymbol{\varepsilon^p}} \tag{3.18}$$

The equivalent plastic strain (3.17) is a measure that considers the path of the plastic straining by integrating it through the loading path. The plastic strain magnitude (3.18) depends only on the current plastic strain state. For monotonous plastic straining these measures are equal.

3.5 Hardening Laws

Work hardening in a uniaxial problem was introduced earlier in the thesis. The effect of the work hardening on the yield surface must be specified also. The hardening law specifies the changes in the configuration and size of the yield surface during yielding.

Isotropic hardening law is simple and easy-to-use. It is well suitable for cases with monotonous loading but it should not be used in situations in which the loading directions vary. The effect of isotropic hardening on von Mises yield surface can be seen in figure 3.3a in the plane stress cut of the yield surface. The radius of the von Mises yield surface cylinder with a center axis of $\sigma_1 = \sigma_2 = \sigma_3$ expands uniformly during yielding. The hardening law can be incorporated into the yield function as

$$f\left(\boldsymbol{\sigma}, \sigma^y\right) = 0 \tag{3.19}$$

where the value of the current yield strength σ^y depends on the value of equivalent

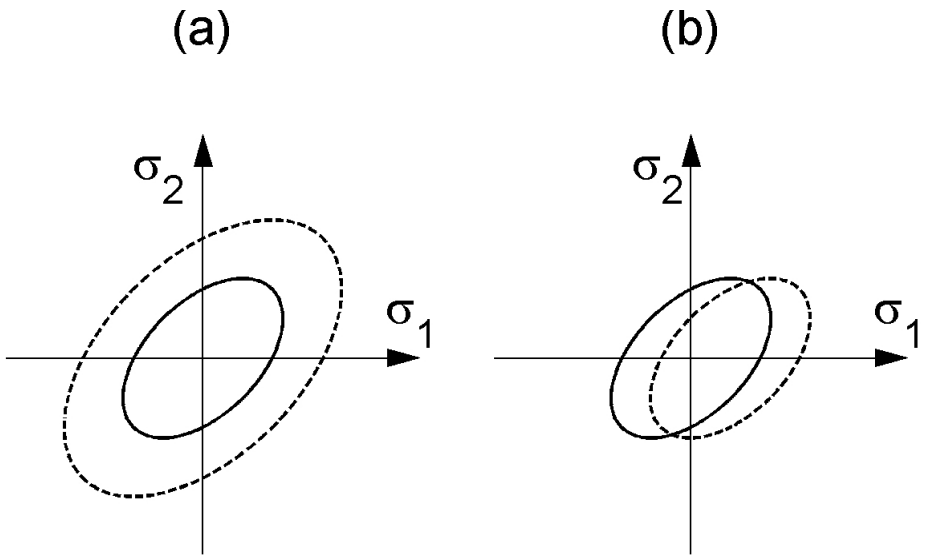


Figure 3.3: Hardening laws

plastic strain ε_p (3.17), written in equation form as

$$\sigma^y = \sigma_{vm}(\varepsilon_p) \tag{3.20}$$

This can be obtained from the uniaxial $\sigma\varepsilon^p$ material test curve by setting $\sigma = \sigma_{vm}$ and $\varepsilon^p = \varepsilon_p$. The $\sigma\varepsilon^p$ curve is obtained from the $\sigma\varepsilon$ curve using the additive strain decomposition (3.2) with the Hooke's law (3.1) connection for the elastic part of strain.

When the loading directions vary, the use of a kinematic hardening law should be considered. It accounts for the Bauschinger effect, which means the reduction of the absolute value of the compressive yield stress when the tensile yield stress is increased by work hardening. Figure 3.3b demonstrates the effect of the kinematic hardening law on the von Mises yield surface. It accounts for the Bauschinger effect ideally as the yield surface translates as a rigid body in the stress space during yielding. The kinematic hardening law can be incorporated to the yield function with the use of a vector $\boldsymbol{\alpha}$ called back-stress, which consists of the coordinates of the yield surface center, as

$$f(\boldsymbol{\sigma} - \boldsymbol{\alpha}) = 0 \tag{3.21}$$

where value of $\boldsymbol{\alpha}$ depends on the plastic strain. Ziegler's hardening rule [16, p. 248] can be used for simple linear work hardening. It defines the increment of back-stress in the direction of the reduced stress vector $\boldsymbol{\sigma} - \boldsymbol{\alpha}$ by the function

$$d\boldsymbol{\alpha} = E_p \frac{1}{\sigma^y}(\boldsymbol{\sigma} - \boldsymbol{\alpha}) d\epsilon_p \tag{3.22}$$

where E_p is the constant plastic modulus (from the uniaxial $\sigma\epsilon^p$ curve) for a given

material. σ^y is the yield strength of the material.

Mixed hardening [16, p. 249] combines these two hardening laws simultaneously.

3.6 Calculation of stresses

The stresses are usually calculated at the element integration points and can be extrapolated from these points to other points in the element as well. It has been suggested that the stresses may be most accurate at the integration points of an element [22].

In small-displacement problems the strain state at a material point of interest used for the stress calculations is obtained from the local element nodal displacements $\mathbf{u}$ with the use of the element stress-displacement matrix as

$$\boldsymbol{\varepsilon} = [\mathbf{B}]\mathbf{u} \tag{3.23}$$

In finite strain problems the incremental strain $\boldsymbol{\Delta\varepsilon}$ can be calculated from the polar decomposition of the incremental deformation gradient [23, sect. 1.4.3], and in some cases, making some approximations on the rotation of the principal axes of strain during the increment [23, sect. 3.2.2]. For finite strain problems involving large plastic strains as well as large elastic strains, a multiplicative decomposition of the elastic and the plastic parts of the deformation gradient (2.20) should be used, see e.g. [24, p. 300]. However, with small elastic strains characteristic for metal plasticity, the more simple additive decomposition (3.2) (holds also for multiaxial case) can be used with little or no effect on the numerical solution [25, p. 162] or [5, p. 248]. The stress-displacement matrix for each increment could be compiled for these cases also, then the material matrix would be a function of the shape functions and the current position of the material point $[\mathbf{B}] = [\mathbf{B}](\mathbf{x}, \mathbf{N})$.

If the yield function at an integration point (calculated from the current yield limit and the stress state of the material point) at the end of an increment is less than zero, $f \leq 0$, the material point is assumed to be in a linear elastic state and the stress state is obtained from the equations of the linear elasticity region. Elastoplastic region is entered when the stress state, calculated from the displacements/strains by using the linear elasticity equations, is situated outside the linear elastic region defined by the yield surface, this can be expressed with the use of the yield function as $f > 0$. Different methods for the stress state calculation have to be used for these two different cases. Both of them will be discussed next.

3.6.1 Linear elastic region

When the material point is in a linear elastic state, the stress state of the point can be calculated from the matrix/vector-equation

$$\boldsymbol{\sigma} = [\mathbf{D}]\boldsymbol{\varepsilon} \tag{3.24}$$

in which $[\mathbf{D}]$ is the elasticity matrix. This equation can be written for an isotropic material in a cartesian xyz-coordinate system as

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{xz} \end{bmatrix} = \begin{bmatrix} 2\tilde{\mu}+\tilde{\lambda} & \tilde{\lambda} & \tilde{\lambda} & 0 & 0 & 0 \\ \tilde{\lambda} & 2\tilde{\mu}+\tilde{\lambda} & \tilde{\lambda} & 0 & 0 & 0 \\ \tilde{\lambda} & \tilde{\lambda} & 2\tilde{\mu}+\tilde{\lambda} & 0 & 0 & 0 \\ 0 & 0 & 0 & \tilde{\mu} & 0 & 0 \\ 0 & 0 & 0 & 0 & \tilde{\mu} & 0 \\ 0 & 0 & 0 & 0 & 0 & \tilde{\mu} \end{bmatrix} \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{xz} \end{bmatrix} \tag{3.25}$$

where $\tilde{\lambda}$, $\tilde{\mu}$ are Lamé constants defined by the equations

$$\tilde{\lambda} = K - \frac{2G}{3} = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad \text{and} \quad \tilde{\mu} = G = \frac{E}{2(1+\nu)}$$

where ν is the Poisson ratio, K is the bulk modulus and G is the shear modulus of the material. This generalizes the Hooke's law (3.1) for a multiaxial case.

3.6.2 Elastoplastic region

When yielding occurs, the nonlinear plastic stress-strain relationship behaviour makes the stress calculation more complicated. The stress state is obtained by a local Newton-Raphson iteration procedure making sure that the nonlinear $\sigma\epsilon$-relation obtained from material tests is satisfied at the material point. An implicit Euler method for determining the plastic stress state will be introduced here. The stress indicating that yielding has occurred $f > 0$ obtained from the linear elastic region equations is referred here to as trial stress.

Approximation for the change in plastic strain and a requirement for the stress not to leave the yield surface at the end of a time step result in the equations

$$\boldsymbol{\Delta\varepsilon}_{n+1} - [\mathbf{D}]^{-1}\boldsymbol{\Delta\sigma}_{n+1} - \Delta\iota\frac{\partial f}{\partial \boldsymbol{\sigma}}|_{n+1} = 0 \tag{3.26}$$

$$f_{n+1} = 0 \tag{3.27}$$

where an associative flow rule is used in the approximation of the increase in plastic strain. $\boldsymbol{\Delta\varepsilon}_{n+1}$ and $\boldsymbol{\Delta\sigma}_{n+1}$ are changes in the total strain vector and stress vector

when advancing from step n to $n+1$, respectively. The first equation can be regarded as a strain balance equation as $\boldsymbol{\Delta\varepsilon}_{n+1} = \boldsymbol{\Delta\varepsilon}^e_{n+1} + \boldsymbol{\Delta\varepsilon}^p_{n+1}$, see equation (3.2). The second equation requires for the stress to lie on the yield surface at the start of the step $n+1$.

By differentiation of the equations (3.26) and (3.27) keeping in mind that $\boldsymbol{\Delta\varepsilon}_n$ is a known constant, a Newton-Rhapson iteration formula is obtained in matrix form as

$$\begin{bmatrix} [\mathbf{D}]^{-1} + \Delta\iota\frac{\partial^2 f}{\partial\boldsymbol{\sigma}^2} & \frac{\partial f}{\partial\boldsymbol{\sigma}} \\ \frac{\partial f}{\partial\boldsymbol{\sigma}}^T & -E_p \end{bmatrix}^k_{n+1} \begin{bmatrix} \boldsymbol{d\sigma}^k \\ d\iota^k \end{bmatrix} = - \begin{bmatrix} \bar{\mathbf{g}}^k \\ f^k \end{bmatrix}$$

where $\bar{\mathbf{g}}$ is the residual of the equation (3.26) calculated at each iteration step from

$$\bar{\mathbf{g}} = -\boldsymbol{\Delta\varepsilon}^k_{n+1} + [\mathbf{D}]^{-1}\boldsymbol{\Delta\sigma}^k_{n+1} + \Delta\iota\frac{\partial f}{\partial\boldsymbol{\sigma}}|^k_{n+1} \tag{3.28}$$

and k is the index for the iteration step. The lower right element of the coefficient matrix is obtained by assuming that the yield function is dependent on hardening and E_p is the plastic modulus corresponding to linear kinematic hardening or isotropic hardening. For nonlinear isotropic hardening with a non-constant plastic modulus, this could be taken as the slope of the $\sigma\varepsilon^p$ curve with the current equivalent stress/strain values. The hardening behaviour has to be included in the calculation of the yield function f also.

Let us drop the iteration index k from the equations (3.29)-(3.30) except for the iterative corrections $d\iota$ and $\boldsymbol{d\sigma}$ to clarify the presentation. By denoting $[\mathbf{D}^*] = [[\mathbf{D}]^{-1} + \Delta\iota^k\frac{\partial^2 f}{\partial\boldsymbol{\sigma}^2}]^{-1}$ and solving for $d\iota^k$ and $\boldsymbol{d\sigma}^k$ from the pair of equations above, the iterative corrections for the scalar and the stress are obtained as

$$d\iota^k = \frac{f - \frac{\partial f}{\partial\boldsymbol{\sigma}}^T[\mathbf{D}^*]\bar{\mathbf{g}}}{\frac{\partial f}{\partial\boldsymbol{\sigma}}^T[\mathbf{D}^*]\frac{\partial f}{\partial\boldsymbol{\sigma}} + E_p} \tag{3.29}$$

and

$$\boldsymbol{d\sigma}^k = -[\mathbf{D}^*]\bar{\mathbf{g}} - [\mathbf{D}^*]\frac{\partial f}{\partial\boldsymbol{\sigma}}d\iota^k \tag{3.30}$$

The update formulas and the initial conditions for the iteration are

$$\begin{aligned} \Delta\iota^{k+1} &= \Delta\iota^k + d\iota^k \quad , \Delta\iota^0 = 0 \\ \boldsymbol{\Delta\sigma}^{k+1}_{n+1} &= \boldsymbol{\Delta\sigma}^k_{n+1} + \boldsymbol{d\sigma}^k \quad , \boldsymbol{\Delta\sigma}^0_{n+1} = [\mathbf{D}]\boldsymbol{\Delta\varepsilon}^0_{n+1} \end{aligned}$$

The yield function, the stress gradients of the yield function and the residual vector $\bar{\mathbf{g}}$ can then be calculated with these updated values for the next iteration step. The iteration is continued until the yield function f and the strain residual $\bar{\mathbf{g}}$ are sufficiently small.

When the iteration finishes, the updated values for the stress and the plastic strain increase are obtained from the equations

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \boldsymbol{\Delta\sigma}_{n+1} \tag{3.31}$$

$$\boldsymbol{\varepsilon}^{\boldsymbol{p}}_{n+1} = \boldsymbol{\varepsilon}^{\boldsymbol{p}}_n + [\mathbf{D}]^{-1}\boldsymbol{\Delta\sigma}_{n+1} \tag{3.32}$$

When using the implicit method for the direct integration of equation (2.1), this local plastic stress iteration procedure is carried out at each global iteration step for the element integration points with stresses that exceed the yield limit. When the full Newton-Raphson iteration is used for the global iteration, the global stiffness matrix has to be updated for every iteration step. A material matrix that is consistent with the implicit Euler algorithm can be obtained as [26, p. 233]

$$[\mathbf{D_{impl}}] = [\mathbf{D}^*] - \frac{[\mathbf{D}^*]\frac{\partial f}{\partial \boldsymbol{\sigma}}\frac{\partial f}{\partial \boldsymbol{\sigma}}^T[\mathbf{D}^*]}{\frac{\partial f}{\partial \boldsymbol{\sigma}}^T[\mathbf{D}^*]\frac{\partial f}{\partial \boldsymbol{\sigma}} - E_p} \tag{3.33}$$

It is used for the calculation of $[\mathbf{K}]$ from equation (2.4). When the approximation made in this method approaches zero, $\Delta\iota_n \to 0$ in $[\mathbf{D}^*]$, then $[\mathbf{D}^*] = [\mathbf{D}]$ and thus (3.33) is the equation for the differential elastoplastic stress-strain matrix.

The algorithm presented here is called a closest-point projection method [12, p. 410]. It projects the stress into a point on the yield surface that is closest to the trial stress. It also has to be noted that some modifications for the stress return algorithm may have to be made if it is to be used in a plane stress case, see e.g. [24, p. 126].

4. CONTACT

4.1 Contact detection

Master-slave (node-to-segment) algorithm is widely used for contact detection in finite element analysis. It is based on the specification of regions, surfaces, that are likely to interact with each other during the simulation. Dividing the model into these specified surfaces is computationally efficient, only the parts of the model likely to establish contact have to be monitored throughout the analysis for contact detection.

The principal idea of the master-slave algorithm is to specify two surfaces, a master surface and a slave surface. The slave surface will be subordinated to the master surface. If the master surface is element-based, it is defined by the facets connecting the nodes of the elements. The slave surface is merely defined as a set of nodes when node-to-surface discretization is used. Contact is detected only when the slave surface nodes penetrate the master surface facets. Master surface nodes can penetrate the imaginary slave surface facets without contact being detected as illustrated in figure 4.1 where a master surface node has penetrated the slave surface facets but not the slave surface nodes. The slave surface facets are imaginary in the sense that the slave surface is defined merely as a set of nodes, and therefore, no facets exist on the slave surface.

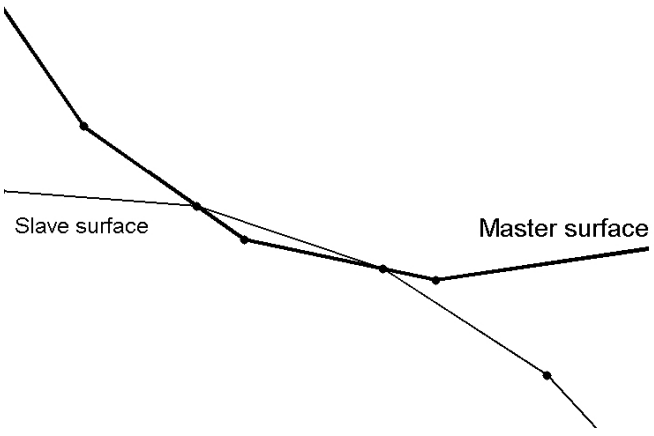


Figure 4.1: Pure master-slave node-to-surface contact

Some guidelines for selecting the slave and the master surface are as follows:

- Rigid surface must be the master surface
- Node-based surface has to be the slave surface
- Slave surfaces must be attached to deformable bodies
- Smaller surface should be the slave surface
- The surface with stiffer behaviour (materially as well as geometrically) should be the master surface
- The slave surface should have a finer mesh than the master surface

The previously introduced basic idea of the master-slave algorithm was based on the use of pure master-slave weighting with node-to-surface discretization for demonstration purposes. The presented guidelines are especially important for this case. More accurate and sophisticated techniques are also available in FEA softwares, such as Abaqus [7]. The selection of the discretization method, contact surface weighting, constraint enforcement method and sliding formulation all have influence on the accuracy of the results and the computational efficiency of the simulation. These will be discussed later on in this section. Alternate techniques concerning surface discretization and surface weighting will decrease the importance of the presented guidelines.

4.2 Contact weighting

The previously introduced pure master-slave contact (one pass) calculates the contact condition only once, and therefore, the master surface nodes can penetrate the slave surface without any opposing force if the mesh used in the model is too coarse. There is also an alternate method called balanced master-slave contact (two pass) weighting. It calculates the contact condition twice, switching the master-slave configuration opposite for the second calculation. The final contact configuration is then determined by applying a weighted average of these two results. The balanced contact weighting minimizes the possibility of contact penetration but is computationally more expensive because the contact calculations have to be performed twice.

The balanced master-slave weighting is well suitable for an explicit finite element code when two deformable surfaces contact each other. Possible overconstraint issues in an implicit analysis typically make the choice of pure master-slave weighting more suitable [7, sect. 32.1.1].

4.3 Contact discretization

The previously introduced node-to-surface discretization is the formulation used by Abaqus/Explicit.

In addition to the node-to-surface discretization, Abaqus/Standard offers a surface-to-surface contact discretization that considers also the slave surface shape. The contact conditions are calculated with an averaging integral over regions near the slave nodes. The averaging regions for each slave node is approximately centered at the slave node so that the adjacent slave nodes are also considered in the contact constraint calculation. Contact direction is defined by an average normal of the slave surface in the surrounding region of the slave node.

Surface-to-surface discretization provides smoother and better results in contact pressure and stresses compared to the more discrete node-to-surface discretization. It also minimizes the possibility of large penetrations of the master surface nodes into the slave surface. When mesh densities are similar, surface-to-surface contact is not as sensitive on the choice of master and slave surface as node-to-surface discretization. For a fine mesh, the results between these two discretization methods will probably not differ much, but for a coarse mesh the difference is evident.

4.4 Contact constraint enforcement

The detected contact conditions add constraint terms into the equilibrium equations. The configuration of the surfaces and the forces between the surfaces are then determined by enforcing these contact constraints. Two kinds of principal ideas for contact constraint enforcement are commonly used in finite element analysis problems. These are the Lagrange multiplier method and the penalty method.

4.4.1 Principal idea of contact constraint enforcement

A demonstration of the contact constraint enforcement with the Lagrange multiplier and penalty methods in a simple problem is illustrated in figure 4.2a where a point mass m is supported by a spring with a stiffness k attached to a rigid support at the top. A gravity field with a downward acceleration of g is present. The gap between the rigid surface at the bottom and the point mass is denoted here as h. If $h > 0$, contact is not present and if $h \leq 0$, contact is present. The solution for the force equilibrium is trivial for $h > 0$ but boundary nonlinearity arises when $h \leq 0$. In the following, let us assume that the contact constraint is active, in other words, contact has been detected: $h \leq 0$, $mg \geq kh$.

In case of the Lagrange multiplier method in figure 4.2b, the equilibrium equation

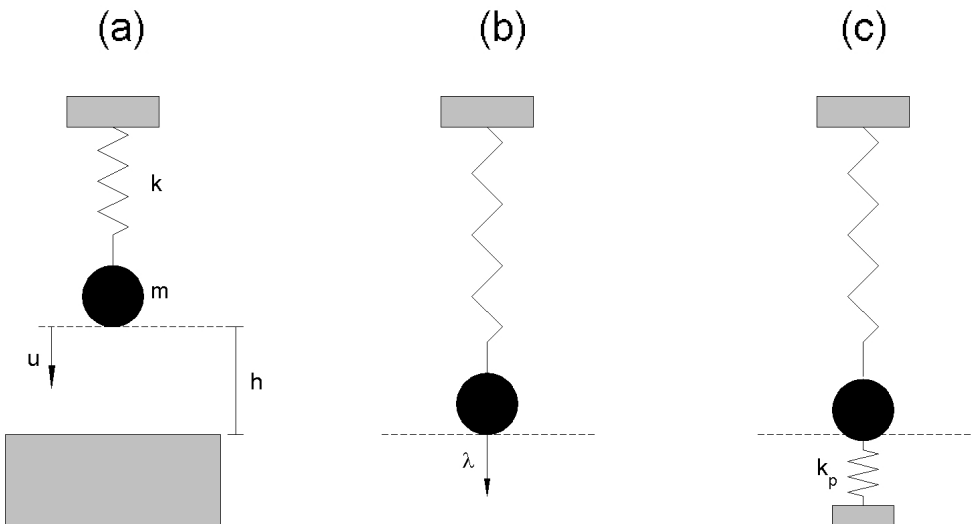


Figure 4.2: A single degree of freedom contact gap problem

is written as

$$\begin{bmatrix} k & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ \lambda \end{bmatrix} = \begin{bmatrix} mg \\ h \end{bmatrix} \tag{4.1}$$

where λ is the Lagrange multiplier that has to be solved for. In this simple case it is simply the contact reaction force easily solved from $P = \lambda = kh - mg$. This method adds additional variables to the equilibrium equations and is therefore computationally more costly but it enforces the contact condition exactly.

The solution with the penalty method in figure 4.2c can be described as adding of a spring between the point mass and the rigid surface at the bottom. The equilibrium equation is then written as

$$(k + k_p)u = mg \tag{4.2}$$

where k_p is the penalty stiffness of the spring which defines how strictly the constraint is enforced. Overall, the constraint is enforced in an approximate fashion. If the value of k_p is too large, it may produce convergence difficulties, and if it is too small, overclosure between the point mass and the bottom surface will occur as seen in 4.2c.

A constraint enforcement method called the augmented Lagrange method is also in use. It is similar to the Lagrange method but penalty terms are added to it. In general, this results in improved convergence rates when compared to the Lagrange method but is computationally more costly than the simple penalty method. See for example [14, p. 126] for more information on the augmented Lagrange method.

The constraint enforcement in this simple demonstrative case was easy to calculate. However, this is not the case when the contact involves multi-dimensional deformable bodies and multiple contact constraints to be enforced during an increment. Demonstration on the difference of the explicit and implicit method concern-

ing contact constraint enforcement is presented next.

4.4.2 Finite element method implementation

Algorithms for frictionless contact in finite element simulations are introduced here. Some discussion on the inclusion of friction will follow later on. Also, this thesis is focused on quasistatic finite element simulation so that the impact phenomena is not considered here.

Let us assume in the following that the active set of contact constraints is determined for the increment or iteration step. The contact problem can then be stated in the form

$$[\mathbf{M}]\ddot{\mathbf{U}} + [\mathbf{C}]\dot{\mathbf{U}} + \mathbf{R}^{\mathbf{int}} - \mathbf{R}^{\mathbf{ext}} = 0 \tag{4.3}$$

$$\text{and} \qquad [\mathbf{G}^{\mathbf{c}}] \geq 0 \tag{4.4}$$

where the single contributions of nodes/facets are combined in the contact residual matrix $[\mathbf{G}^{\mathbf{c}}]$ which is dependent of the displacement field. In the case of the simple problem with one degree of freedom introduced in the previous subsection, $[\mathbf{G}^{\mathbf{c}}]$ would simply be $G^c(u) = u - h \geq 0$. However, in a more general context the matrix is nonlinear with respect to the displacement field. It relates the kinematic gap variables to the global finite element solution.

The nonlinear equation system derived from a minimum of potential energy principle [14, p. 330] for the contact problem residuals results for the Lagrange multiplier method as

$$\bar{\mathbf{G}} + [\mathbf{C}^{\mathbf{c}}]^T \mathbf{\Lambda} = \mathbf{0} \tag{4.5}$$

$$[\mathbf{G}^{\mathbf{c}}] = [\mathbf{0}] \tag{4.6}$$

and for the penalty method as

$$\bar{\mathbf{G}} + k_p [\mathbf{C}^{\mathbf{c}}]^T [\mathbf{G}^{\mathbf{c}}] = \mathbf{0} \tag{4.7}$$

where $\bar{\mathbf{G}}$ is the residual introduced in section 2.2.2 and $[\mathbf{C}^{\mathbf{c}}]$ is the constraint contribution matrix that can be defined as a partial derivative of the contact residual matrix as

$$[\mathbf{C}^{\mathbf{c}}] = \frac{\partial [\mathbf{G}^{\mathbf{c}}]}{\partial \mathbf{U}} \tag{4.8}$$

see [14, ch. 9] for more information on how to compile it for different discretizations in large displacement problems. The constraint contribution matrix is also dependent of the nodal displacements. $\mathbf{\Lambda}$ is a vector of Lagrange multipliers that are added to the system of equations for every constraint degree of freedom.

Contact constraint enforcement in explicit method

The penalty method can fairly easily be implemented to the explicit method. Only the contact contribution terms of the penalty method have to be added to the equilibrium equation at the start of each step which yields, for the equation (2.7), the form

$$\ddot{\mathbf{U}}_n = [\mathbf{M}]^{-1}\left(\mathbf{R^{ext}}_n - k_p[\mathbf{C^c}]^T[\mathbf{G^c}] - [\mathbf{C}]\dot{\mathbf{U}}_{n-\frac{1}{2}} - \mathbf{R^{int}}_n\right) \tag{4.9}$$

where the penalty stiffness k_p is assumed to be the same for all constraints.

The Lagrange multiplier method can not be directly applied to the explicit method because there is no mass associated with the Lagrange multipliers. However, different predictor/corrector-type algorithms can be constructed to enforce the contact constraints exactly. These kind of algorithms use, at first, an unconstraint predictor step after which the corrector step is performed so that the constraint is enforced exactly at the end of the time step.

One of this type of predictor/corrector-algorithm can be constructed by fulfilling the rate of the constraint $\dot{g}_N = 0$, see [14, p. 353]. This idea leads to an additional system of equations

$$\frac{\Delta t_{n+1} + \Delta t_n}{2}[\mathbf{C^c}][\mathbf{M}]^{-1}[\mathbf{C^c}]^T\mathbf{\Lambda} = [\mathbf{C^c}]\left[\dot{\mathbf{U}}_{n-\frac{1}{2}} + \frac{\Delta t_{n+1} + \Delta t_n}{2}[\mathbf{M}]^{-1}\left(\mathbf{R^{ext}} - \mathbf{R^{int}}\right)_n\right]$$

from where the vector of Lagrangian multipliers $\mathbf{\Lambda}$ can be solved iteratively, see e.g. [27]. The coefficient matrix on the left side is not generally diagonal. The assumption made in this method is that the gap and gap rate lead to the same contact constraint matrix since velocities and displacements use the same shape functions for the interpolation.

Contact constraint enforcement in implicit method

It is assumed here that the update of the active set of contact constraints is performed within each step of the global Newton-Raphson iteration.

The Lagrange multiplier method requires additional variables $\mathbf{\Lambda}$ for the iteration procedure. Let us compile a vector $\mathbf{w} = [\mathbf{U}^T\mathbf{\Lambda}^T]^T$ consisting of the nodal displacements and the lagrange multipliers to make the presentation more compact and drop the subscript for the time increment $n+1$ from the following. The Lagrange multiplier method leads to the following iterative scheme:

$$[\mathbf{K^{LM}}](\mathbf{w}^i)\mathbf{\Delta w}^{i+1} = \bar{\mathbf{G}}^{\mathbf{LM}}(\mathbf{w}^i) \tag{4.10}$$

$$\mathbf{w}^{i+1} = \mathbf{w}^i + \mathbf{\Delta w}^{i+1} \tag{4.11}$$

where

$$[\mathbf{K^{LM}}] = \begin{bmatrix} [\mathbf{K_{impl}}]^i + [\mathbf{K_t^c}]^i & [\mathbf{C^c}]^{iT} \\ [\mathbf{C^c}]^i & [\mathbf{0}] \end{bmatrix} \tag{4.12}$$

and

$$\bar{\mathbf{G}}^{LM} = \begin{bmatrix} \bar{\mathbf{G}}^i - [\mathbf{C^c}]^{iT}\mathbf{\Lambda}^i \\ -[\mathbf{G^c}]^i \end{bmatrix} \tag{4.13}$$

where $[\mathbf{K_{impl}}]$ and $\bar{\mathbf{G}}$ are presented in section 2.2.2 where no contact conditions were assumed to be present. The matrix $[\mathbf{K_t^c}]$ is obtained by linearization of the product of the constraint matrix and the vector of Lagrange multipliers with respect to the displacement field [14, p. 331] as

$$[\mathbf{K_t^c}] = \frac{\partial [\mathbf{C^c}]}{\partial \mathbf{U}} \mathbf{\Lambda} \tag{4.14}$$

and it disappears for a linear problem. All the other matrices in (4.12)-(4.13) depend on the displacements.

The iterative scheme for the penalty method is

$$[\mathbf{K^P}](\mathbf{U}^i)\mathbf{\Delta U}^{i+1} = \bar{\mathbf{G}}^{\mathbf{P}}(\mathbf{U}^i) \tag{4.15}$$

$$\mathbf{U}^{i+1} = \mathbf{U}^i + \mathbf{\Delta U}^{i+1} \tag{4.16}$$

where the tangent matrix obtained from the linearization of the residual function is

$$[\mathbf{K^P}] = [\mathbf{K_{impl}}]^i + k_p \left([\mathbf{K_t^{cP}}]^i + [\mathbf{C^c}]^{iT} [\mathbf{C^c}]^i \right) \tag{4.17}$$

and the residual function is

$$\bar{\mathbf{G}}^{\mathbf{P}} = \bar{\mathbf{G}}^i - [\mathbf{C^c}]^{iT} [\mathbf{G^c}]^i \tag{4.18}$$

Here the matrix $[\mathbf{K_t^{cP}}]$ obtained from the linearization of the variational potential energy function [14, p. 331] as

$$[\mathbf{K_t^{cP}}] = \frac{\partial [\mathbf{C^c}]}{\partial \mathbf{U}} [\mathbf{G^c}] \tag{4.19}$$

also disappears for a linear problem. Here only the displacements which are the primary variables enter the formulation and no additional Lagrange multipliers is needed.

4.5 Finite sliding formulation

Finite sliding is a general, but a computationally expensive formulation. In finite sliding formulation, the node can contact any of the facets on the other surface. The formulation allows for arbitrary separation, sliding, and rotation of the surfaces. However, it is assumed that the relative tangential motion of the two surfaces in one increment does not significantly exceed the facet size of the master surface. This usually is not a problem when the explicit direct integration method, characterized by small increments, is used. For the implicit method, this has to be accounted for. This formulation can be used in a geometrically nonlinear analysis as well as a geometrically linear analysis. The latter one being in situations with finite sliding between two stiff bodies not undergoing large rotations.

If node-to-surface discretization with finite-sliding formulation is used, slave nodes might get stuck at sharp corners of the master surface. At the corner points, the master surface normal is discontinuous. Therefore, for attaining realistic sliding conditions, smoothing has to performed for the master surface corners. Some techniques for surface smoothing, such as Hermite, Bézier and spline polynomials, can be read from [14, p. 279-302].

Other computationally more simple formulations for sliding between the contact surfaces are for example small sliding and infinitesimal sliding & rotation in Abaqus [7, sect. 33.2.2]. In the small sliding formulation, every slave node acts with its respective tangent plane on the master surface. The tangent plane orientation is defined by shape functions at an anchor point and the normals of the master surface nodes. These master surface nodal normals are calculated as an average of the adjacent element face normals. This means that the tangent planes for contact are usually only an approximation of the mesh geometry. Therefore, it has to be noted that for attaining good results, the tangent planes must approximate the mesh geometry well throughout the analysis. Nonlinear geometric effects are taken into account in the small sliding formulation in the way that the tangent plane orientation is updated during the analysis. The tangent plane rotates fixed to the master surface.

The infinitesimal sliding formulation ignores geometrically nonlinear effects, and therefore, only small displacements are allowed when it is used. It differs from the small sliding formulation only in the manner that the local tangent plane orientations are not updated during analysis. Both absolute motions and relative motions of the surfaces should remain small in the model when this formulation is used.

4.6 Contact tracking algorithm

The active set of contact constraints has to be determined at each increment in a finite sliding formulation. This can be done by measuring the minimum distance gaps for each slave node and the master surface by means of contact tracking. This works well for quasistatic problems.

When the slave surface nodes can contact any one of the master surface facets, a large proportion of the computational cost involved in a finite sliding simulation is introduced by the contact tracking algorithm which tracks the motion of the surfaces. The tracking algorithm is usually divided into global and local contact searches for improved computational efficiency. The global search is responsible for most of the computational cost involved in a contact tracking algorithm, even though a bucket sorting algorithm or other spatial search algorithms [14, p. 315], which sorts the potential master surface facets for each slave node can be used to minimize the computational cost of the global search.

The tracking algorithms used in small sliding/deformation problems is computationally less expensive. In these formulations, the global search can be performed only once at the beginning of the simulation to determine the nearest master surface facets for the slave surface nodes.

For demonstration purposes, let us introduce the Abaqus/Explicit finite-sliding tracking algorithm as presented in [7, sect. 33.2.2]:

A global contact search for the nearest master surface facet of each slave node is performed first. The algorithm can then track the master surface node which is the node on the master surface facet that is nearest to the slave node. A local contact search is then performed in subsequent increments until the next global search. It locally searches only the facets attached to the previously tracked master surface node for the nearest master surface facet. Then the nearest master surface node is tracked and updated if necessary. If the new tracked master surface node is not the same as the previously determined, a new local search will be performed. The local search will stop only when the new tracked master surface node is the same as the previously tracked node. The global search is performed once in one hundred increments by default in Abaqus/Explicit. The frequency of the global search can also be manually adjusted, although reducing the increment size will significantly increase the computational cost of the simulation.

4.7 Friction

Because of the difficulties in predicting actual friction forces accurately in a metal forming simulation, a constant Coulomb friction coefficient is often used in modelling the frictional behaviour. The Coulomb friction law for a slip case defines

the magnitude of the tangential traction forces between two surfaces caused by the relative tangential motion of these surfaces as

$$\boldsymbol{t_T} = -\mu|p_N|\frac{\dot{\mathbf{g}}_\mathbf{T}}{||\dot{\mathbf{g}}_\mathbf{T}||} \tag{4.20}$$

where μ is the friction coefficient and $|p_N|$ the magnitude of contact pressure acting normal and into the surface. $\dot{\mathbf{g}}_\mathbf{T}$ is the rate of the tangential gap vector which defines the direction of relative motion. The shear stress caused by friction between the contacting two surfaces is in the opposite direction of the motion.

A stick condition has to be distuingished here. This is applied when two surfaces do not slide with respect to each other. This is the case when the contact pressure has not exceeded a certain limit. The stick condition for the tangential motion of the surfaces can be expressed in equation form as

$$\dot{\mathbf{g}}_\mathbf{T} = 0 \quad \text{and} \quad \mathbf{g}_\mathbf{T} = 0 \tag{4.21}$$

where $\mathbf{g}_\mathbf{T}$ is the tangential gap in vector form. Equation (4.21) defines the rate of the tangential gap to be zero.

The inclusion of friction complicates the calculation of the contact conditions. For a frictional case in the implicit method, the tangential constraints have to be included in the iteration also. This leads [14, p. 357] in the penalty method described in 4.4.2 to a tangent matrix and a residual vector as follows:

$$[\mathbf{K^P}] = [\mathbf{K_{impl}}] + [\mathbf{K_T^{cp}}]^T + k_p[\mathbf{C^c}]^T[\mathbf{C^c}] + \mathbf{F}[\mathbf{C_T^c}] \tag{4.22}$$

$$\mathbf{G^P} = \bar{\mathbf{G}} + k_p[\mathbf{C^c}]^T[\mathbf{G^c}] + \mathbf{t_T}^T[\mathbf{C_T^c}] \tag{4.23}$$

where $[\mathbf{C_T^c}]$ is the tangential constraint contribution matrix and $[\mathbf{K_T^{cp}}]$ contains the contributions from the linearization of both of the terms including $[\mathbf{C_T^c}]$ and $[\mathbf{C^c}]$ in the residual vector (4.23). The vector $\mathbf{F}$ results from the linearization of the tangential stresses $\mathbf{t_T}$ with respect to the displacement field. The tangential stresses depend on the state of stick and slip as will be discussed next.

The friction force vector $\mathbf{t_T}$ in (4.23) has to be compiled from single contributions of the frictional forces where the stick and slip state at each node have to be distuingished. A technique analogous to the return mapping algorithm of non-associative plastic stress calculations can be read from [14, p. 360]. The tangential total slip $\mathbf{g_T}$ is divided here into a stick (elastic) $\mathbf{g_T^{st}}$ and a slip (plastic) $\mathbf{g_T^{sl}}$ part

$$\mathbf{g_T} = \mathbf{g_T^{st}} + \mathbf{g_T^{sl}} \tag{4.24}$$

where the tangential traction depends on the stick part. The stick condition can be

chosen to be fulfilled only approximately to smooth the nonlinearity involved, this leads to the regularization of Coulomb law [14, p. 78]. And thus, for an isotropic friction case, we have the connection

$$\mathbf{t_T} = c_T \mathbf{g_T^{st}} \tag{4.25}$$

where c_T is the stick (elastic) constant which relates the traction force and the stick part of the tangential gap. For the Coulomb law, the local integration of the frictional interface law can be written explicitly [14, p. 362]

$$\mathbf{t_T}_{n+1} = \mu p_{Nn+1} \mathbf{n_T^{tr}}_{n+1} \tag{4.26}$$

$$\mathbf{g_T^{sl}}_{n+1} = \mathbf{g_T^{sl}}_{n} + \frac{1}{c_T} \left(||\mathbf{t_T^{tr}}_{n+1}|| - \mu p_{Nn+1} \right) \mathbf{n_T^{tr}}_{n+1} \tag{4.27}$$

$$\tag{4.28}$$

where $||\mathbf{t_T^{tr}}_{n+1}||$ is the norm of the trial tangential traction vector defined by the equation

$$\mathbf{t_T^{tr}}_{n+1} = \mathbf{t_T}_{n} + c_T \mathbf{\Delta g_T}_{n+1} \tag{4.29}$$

where it was assumed that the whole step would be in stick state. $\mathbf{n_T^{tr}}$ is the direction of the trial traction.

The implementation to the Lagrange multiplier method will not be discussed here. It is often formulated in a way that the Lagrange multiplier method is formulated in the normal direction only and the tangential direction is handled by means of the penalty method introduced here [14, p. 357].

5. SIMULATION MODEL

5.1 Model geometry

5.1.1 Full geometry of the tools and the blank

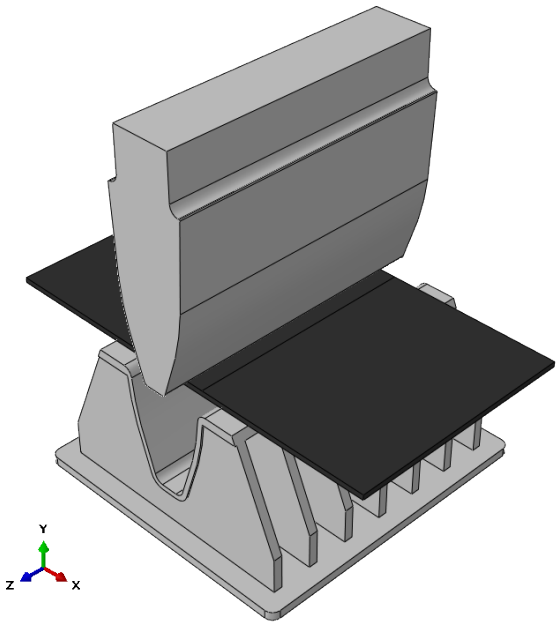


Figure 5.1: Full model geometry

The complete geometry of the tools and the blank can be seen in figure 5.1. The part geometries were imported to Abaqus/CAE from a step (.step) file. The initial geometry of the blank in xyz-coordinates is $x = 2282$mm, $y = 30$mm and $z = 1300$mm, see figure 5.1 for the coordinate axes.

The dimensions of the die and the punch can be seen in figure 5.2. All of the dimensions are in millimeters. The curved side wall geometry with changing radius was a nurb which are not supported for dimensioning in Abaqus/CAE. Therefore, its accurate dimensions can not be shown in the figure and only its shape seen in the figure is taken as a sufficient presentation of its geometry. Furthermore, only the dimensions directly related to the shape of the final product and the tool stresses of interest are presented. The other dimensions are not of importance at this stage of a simulation design process. The angle measure shown in the die dimensions

is the angle between a tangent of the arc of the smallest bend radius curve and a horizontally aligned line. It is the same for the smallest punch bend radius curve.

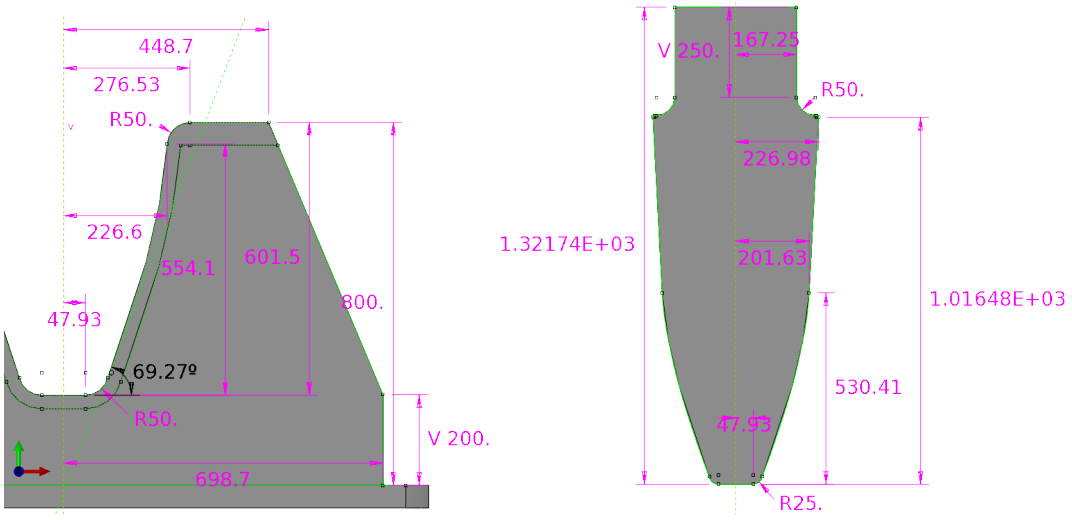


Figure 5.2: Die (left) and punch (right) dimensions in millimeters

The z-dimension of the punch is 1600mm and the z-dimension of the die (the bottom plate of the die is not included) is 1500 mm. The die geometry was not the same on all xy-plane cuts as seen in figure 5.1 but this is not of importance as the z-direction variation of the geometry is not taken into account in any of the simulations, see the simulation models that will be introduced next and the section introducing the meshes used in the simulations for more information.

The desired shape of the blank after forming is taken to coincide with the die cavity shape.

5.1.2 3D model

As the geometry as well as the punch movement are assumed to be symmetric throughout the process with respect to two planes, the xy-plane and the yz-plane, only one fourth of the geometry has to be modelled, see left side of the figure 5.3 for an illustration of this model. This simplification requires symmetry boundary conditions on the symmetry planes. Also, the punch pressing force has to be multiplied by a factor of 4 as only fourth of the plate stiffness is resisting the force.

This model is used for 3D simulations with rigid tools. Therefore, the only regions of the tools, which are of importance in this model, are the tool surfaces that will establish contact with the blank.

5.1.3 Plane strain model

The geometry was also modelled as a planar geometry. A symmetry plane exists also for this model so that only half of the planar geometry has to be modelled, see

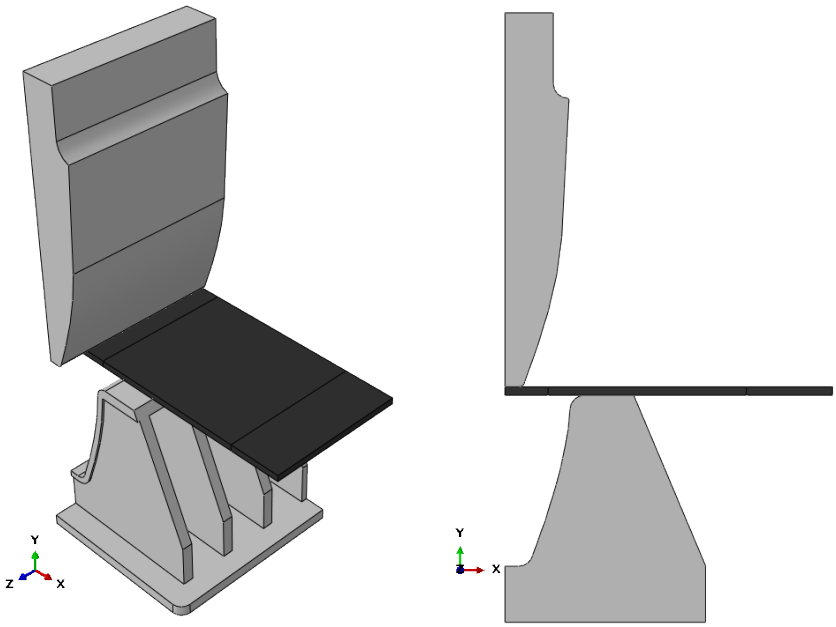


Figure 5.3: Models

right side of figure 5.3. The planar geometry was taken from the middle cut of the geometry on the xy-plane. Plane strain conditions $\varepsilon_z = \gamma_{yz} = \gamma_{xz} = 0$ were assumed for the planar model as the z-direction dimension of the plate is large compared to the thickness direction. This assumption holds quite well for the middle cut but might miss some details on the edges that are parallel to the xy-plane. This model is used for most of the simulations and parametrical studies because of its superior computational efficiency when compared to the 3D model.

The blank and the punch geometries were the same on all xy-plane cuts but the die geometry differed, see figure 5.3. Therefore, it has to be kept in mind that this planar geometry results in a more stiff response from the die when modelled as deformable than that of the 3D model geometry would. The tools were modelled as deformable only with this plane strain model.

The plane strain thickness was assigned in the section properties as 650 mm corresponding to the z-direction thickness of the quarter symmetry model for easier comparison purposes between the models. This requires for the same multiplication of a factor of 4 on the pressing force as with the quarter-symmetry 3D model.

5.2 Analysis types

5.2.1 Explicit dynamic analysis for forming

The Abaqus/Explicit code uses the central difference-operator, which was introduced in the theory part of the thesis, to calculate the dynamic response of the structure.

Abaqus/Explicit will automatically calculate the minimum time step size with

a procedure based on the Courant criterion (2.8). The code uses dilatational wave speed c_d in place of the speed of sound in the material c introduced in the theory chapter. Dilatational wave speed is defined for an isotropic material by the equation

$$c_d = \sqrt{\frac{\tilde{\lambda}_e + 2\tilde{\mu}_e}{\rho}} \tag{5.1}$$

where $\tilde{\lambda}_e$ and $\tilde{\mu}_e$ are effective Lamé constants, see [23, sect. 2.4.5]. An analytical upper bound expression for the maximum eigenvalue of the element defines the characteristic length for an element type. For example, a 4-node reduced integration quadrilateral with the uniform strain formulation has a characteristic length of

$$L_e = \frac{A}{\sqrt{B_i B_i}} \tag{5.2}$$

where A is the element area and B_i is the element gradient operator [23, sect. 3.2.4].

Damping related with the volumetric straining is introduced by means of bulk viscosity in Abaqus/Explicit [23, sect. 2.4.5]. It improves the modeling of high-speed dynamic events and is not part of the constitutive response of the material. Therefore, it will not be of much importance in this context and the default bulk viscosity coefficients are used in the simulations.

Computational efficiency of the simulation is improved by adjusting the time scale of the process to as small value as possible without introducing any notable inertia effects to the simulation. This is obtained by performing the simulation with different time scales with a relatively coarse mesh and checking the ratio of the kinetic energy of the blank with respect to the internal energy of the blank. The Abaqus convention for the total internal energy is ALLIE and for the total kinetic energy it is ALLKE. In the rigid tool simulations these energies are completely defined by the blank as no mass is associated with the rigid tools. The ALLKE/ALLIE ratio is not allowed to exceed a large percentage, typically less than 10% [7, sect. 6.3.3], throughout the majority of the simulation. A time period of 0.5 seconds was found to be appropriate based on this guideline.

5.2.2 Implicit dynamic analysis for forming

A quasi-static application option for the implicit solution was selected. It uses the backward Euler operator, which was introduced in the theory chapter, to obtain an implicit solution to the dynamic equibrium equations at each increment. This application option is intended for quasistatic simulations, includes high numerical dissipation and seems to be more rapid in convergence when compared to the other application options that use the Hilbert-Hughes-Taylor α method. The quasi-static

application option may require considerable numerical dissipation at some parts of the simulation to obtain convergence.

The time increment size is controlled automatically with the chosen application type in Abaqus/Standard by reducing the size when the iteration is diverging or suffering from low convergence rate and increasing the size fairly aggressively when the iteration has converged rapidly in previous increments. A total of 5 cutbacks on the increment size at each increment is allowed, otherwise the simulation will break down. The automatic time increment size control is based on the half-increment residuals [23, sect. 2.2.1].

5.2.3 Static analysis for springback

The springback was solved with the general static analysis type after the forming process. A static general step can be created after the implicit dynamic analysis step as both analysis types use the implicit Abaqus/Standard code.

When the explicit method is used for the forming part of the simulation, a static general step can not be created after the explicit dynamic step. This is because the explicit dynamic analysis uses Abaqus/Explicit code and the static general analysis uses Abaqus/Standard code. The material state has to be imported to another Abaqus/CAE model with the same blank geometry as an initial state from the end of the explicit dynamic analysis. A static analysis is then performed with the initial state defined as a predefined field for the blank.

5.3 Material model

The deformable tools were modelled as linear elastic material with the following properties:

$$E = 200GPa \quad \nu = 0.3 \quad \rho = 7800\frac{kg}{m^3}$$

The plate material was modelled with the same density and linear elastic properties as the tools, but in this case, the plasticity model needed to be included as well. The following material parameters from a tensile test are given:

$$\sigma^y = 426MPa \qquad \sigma^u = 565MPa \quad \varepsilon^m = 0.25 \quad \varepsilon^u = 0.15$$

where ε^m is the engineering strain at break measured from the unstressed length of the tensile specimen after the test, and ε^u is the necking strain, which is the value of strain corresponding to the ultimate stress.

The chosen plasticity model uses associative flow rule with von Mises yield criterion and isotropic hardening unless otherwise mentioned. Abaqus always assumes

the logarithmic elastic strain to be small so that the additive strain rate decomposition can be used [23, sect. 1.4.4]. In the Abaqus plasticity models, the total strain for solid continuum elements is defined by an integral over the rate of deformation. This integral is calculated approximately by using a central-difference scheme and by approximating the rigid body rotations during the increment, for further information see [23, sect. 3.2.2]. The material matrix is approximated also. The approximations made in the formulation provide improved convergence but may cause some problems with anisotropic plastic behavior, such as kinematic hardening, when large strains and rotations are present [23, sect. 3.2.2].

The uniaxial plasticity data has to be converted to the true stress and logarithmic plastic strain measures. The formulas for conversion are

$$\sigma_{true} = \sigma_e(1 + \varepsilon_e) \quad \text{and} \quad \varepsilon^p_{log} = ln(1 + \varepsilon_e) - \frac{\sigma_{true}}{E} \tag{5.3}$$

The engineering strain value of interest is the necking strain that corresponds to the ultimate stress value, and therefore, $\varepsilon_e = \varepsilon^u$ is used in (5.3). The strain value is used to obtain an approximate linear work hardening curve for the material data input as no further information of the stress-strain curve is available. Thus, the values obtained from the given material parameters for data input are

$$\sigma^y = 427MPa \quad \text{at} \quad \varepsilon^p_{log} = 0 \quad \text{and} \quad \sigma^u = 650MPa \quad \text{at} \quad \varepsilon^p_{log} = 0.14$$

This corresponds to a plastic modulus of approximately $E_p = 1.6GPa$. With this input in Abaqus, the uniaxial plasticity curve input would be ideally plastic (no work hardening, $E_p = 0$) after the strain exceeds 0.14. This is not the case in reality (see section 3.1) as the localization of stresses at the necking stage is not properly measured with the given material parameters. Therefore, an approximation of the behaviour after $\varepsilon^p_{log} = 0.14$ is obtained by keeping the plastic modulus constant until the strain exceeds a value of 1. Thus, the final material data for the simulation is input as

$$\sigma^y = 427MPa \quad \text{at} \quad \varepsilon^p_{log} = 0 \quad \text{and} \quad \sigma^u = 2061MPa \quad \text{at} \quad \varepsilon^p_{log} = 1$$

The material parameters are only an approximation of the true material behavior. The properties of the steel blanks can vary significantly as they are often sold with a high probability guarantee on the strength of the material. The material properties are to be modified to perform some parametrical studies on them. This is also an important aspect if one is to choose methods such as preheating of the plate to decrease the stiffness and the yield strength of the blank during the forming process.

5.4 Boundary conditions

5.4.1 Rigid tool model

The die was fixed in space with all degrees of freedom constrained at the rigid body reference point.

A displacement boundary condition corresponding to the punch movement along the negative direction of the y-axis of 601.5mm was set for the punch. All other degrees of freedom were constrained at the rigid body reference point. This displacement boundary condition advanced the kinematic state of the punch incrementally causing contact conditions between the punch and the top surface of the blank. These contact conditions together with the contact conditions between the blank bottom surface and the die forced the blank to be formed. A smooth step amplitude curve was applied for the movement of the punch to reduce the propagation of stress waves at the start and the end of the simulation. The smooth step amplitude curve is a fifth-order polynomial fit with zero velocity in the start and in the end, see figure 5.4 for the punch displacement as a function of time.

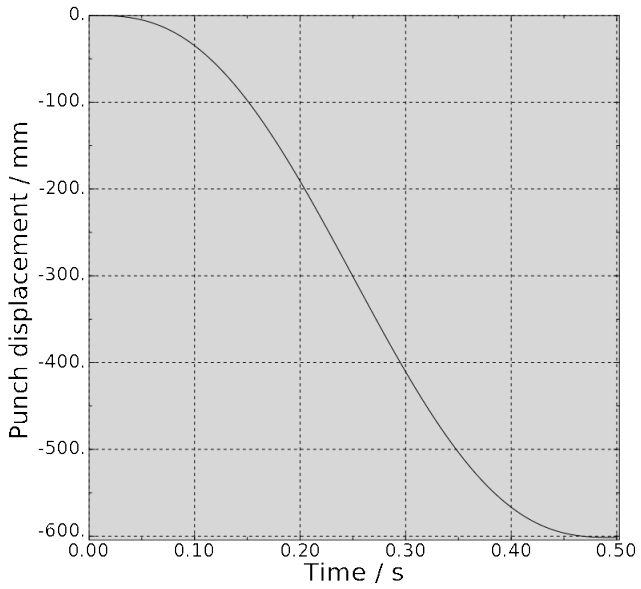


Figure 5.4: Smooth step amplitude curve

The springback step included the symmetry boundary conditions on the blank and one node fixed in the y-direction at the cut of the symmetry planes on the top surface of the blank. The tools were either fixed at their positions or removed from the assembly for the springback step.

5.4.2 Deformable tool model

The symmetry boundary conditions on the symmetry planes had to be set for all of the part instances: the punch, the die and the blank. The symmetry boundary conditions smooth the master surface at the symmetry plane by constructing a parabolic curve segment between the end segment and the reflection of the end segment about the symmetry plane [7, sect. 34.1.1]. This smoothing procedure is especially important in node-to-surface discretization.

The displacement boundary condition corresponding to the punch movement was set in similar way as in the rigid tool model. This time it was set for all of the nodes on the punch top surface as no rigid body reference point existed.

The die was fixed on the bottom edge with all degrees of freedom constrained.

The springback was achieved similarly to that of the rigid tool model.

5.5 Meshing and elements

5.5.1 Blank partitioning and mesh

The blank was partitioned into three regions as shown in figure 5.5.

Figure 5.5: Blank partitioning

This was done because the blank mesh needed different fineness in different regions to obtain accurate results. The region to the left, adjacent to the symmetry plane, is the region of most interest. It is subjected to most plastic straining in the model. The region in the middle requires some mesh fineness while the region to the right is not subjected to significant stress nor straining and requires only a reasonably coarse mesh.

The mesh of the blank in the plane strain model will be referred throughout this thesis to as: element edge length at the region to the left / element edge length at the region to the right. The middle region is a transition region between these two regions with quad-dominated free mesh technique. A demonstration for a 5/10-mesh on the plane strain model blank can be seen in figure 5.6.

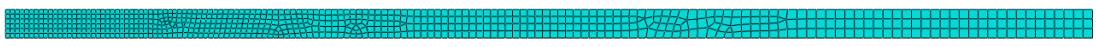


Figure 5.6: 5/10-mesh on the blank

5.5.2 Explicit method elements

Solid continuum elements

It is only possible to use first-order quadrilateral and hexahedral elements in the explicit dynamic procedure based on the Abaqus/Explicit code. For this bending-dominated forming problem, it is clear that first-order fully integrated solid continuum elements should not be used because of the overly stiff behaviour these elements exhibit due to shear locking. As stated in the theory chapter, first-order reduced integration elements avoid this problem but care has to be taken when using these elements because of the hourglass modes. With a sufficiently large number of elements through the thickness of the plate this will not be a problem. An enhanced hourglass control option is selected for this simulation to ensure a consistent calculation of the forces when importing the material state between Abaqus/Explicit and Abaqus/Standard for the springback analysis. It has to be checked after the analysis that the artificial strain energy caused by the forces used in hourglass control does not exceed 1% of the total internal energy of the plate [28, A2.23].

The first-order reduced integration solid continuum elements in Abaqus have a uniform strain formulation rather than calculating the strain at the Gauss point in the center of the element. This is better for accuracy when the elements are skewed [23, sect. 3.2.4].

Shell elements

The blank was also modelled with shell elements with the 3D model in explicit analysis.

The shell elements used are doubly-curved S4R 4-node first-order reduced integration conventional shell elements and its fully integrated version S4. These are general-purpose shell elements that are suitable for thin and thick shell problems. The thickness change is allowed in their formulation as a function of in-plane deformation through an effective section Poisson's ratio [23, sect. 3.6.1]. The membrane kinematics of the fully integrated version S4 is based on an assumed strain formulation [23, sect. 3.6.5] which should provide accurate solutions for in-plane bending behaviour. No hourglass control is needed in the bending nor membrane response of the S4 element. These elements are suitable for large-strain analysis by accounting for finite membrane strains and arbitrary large rotations.

The default settings for the section Poisson's ratio was used. To capture the nonlinear material behavior appropriately, the section had to be integrated during the analysis and the Simpson integration rule with 7 through-thickness integration points is selected.

The conventional shell elements discretize a reference mid-surface only so that

the plate thickness has to be input in the section properties.

5.5.3 Implicit method elements

In the implicit dynamic analysis with the Abaqus/Standard code, it is possible to use second-order solid continuum quadrilateral/hexahedral elements. These elements do not exhibit hourglass modes, and therefore, a smaller number of elements is needed through the plate thickness. The larger number of element integration points offer also strain/stress values that can be regarded as more accurate with use of fewer elements.

The second-order elements in Abaqus are always integrated by the Gauss quadrature [23, sect. 3.2.4]. The reduced integration second-order elements CPE8R and C3D20R are chosen for this simulation as second-order fully integrated elements start to develop volumetric locking when the plastic strains are on the order of the elastic strains. The reduced integration second-order elements suffer from volumetric locking only after large amounts of plastic strain has occurred and this is often seen as an hourglass-like mode in the mesh. This can be avoided by refining the mesh in regions subjected to large plastic straining. The presence of volumetric locking can be checked by a quilt-style contour plot of the pressure stress which shows a checkerboard pattern with significant changes in the values of adjacent integration points if volumetric locking is occurring [7, sect. 25.1.1].

5.5.4 Tool meshes

The tools were modelled as discrete rigid with R2D2 2-node rigid link elements in the plane strain model. The mesh in the round corners was fined with a total of ten R2D2 elements over the radius to capture the round geometry sufficiently well. The rigid elements in the 3D model were R3D4 four-node bilinear quadrilaterals as the tool surfaces in the 3D model needed shell-like shape from the geometry. The shell element models and the implicit solid continuum model used 8 R3D4 elements and the explicit solid continuum used 10 R3D4 elements over the radius of the tools.

The tools were modelled as deformable only in the implicit analysis with the plane strain model. The elements used for the tool meshes were the same CPE8R elements as was used for the blank. The tool meshes used for the simulation can be seen in figure 5.7. Minimum element side length is 4 mm. Number of elements in the die mesh is 3521 and in the punch mesh 1837.

5.6 Contact modelling

In figure 5.8 the contact surface definitions on the punch bottom surface and the die top surface can be seen as the red lines on the corresponding edges.

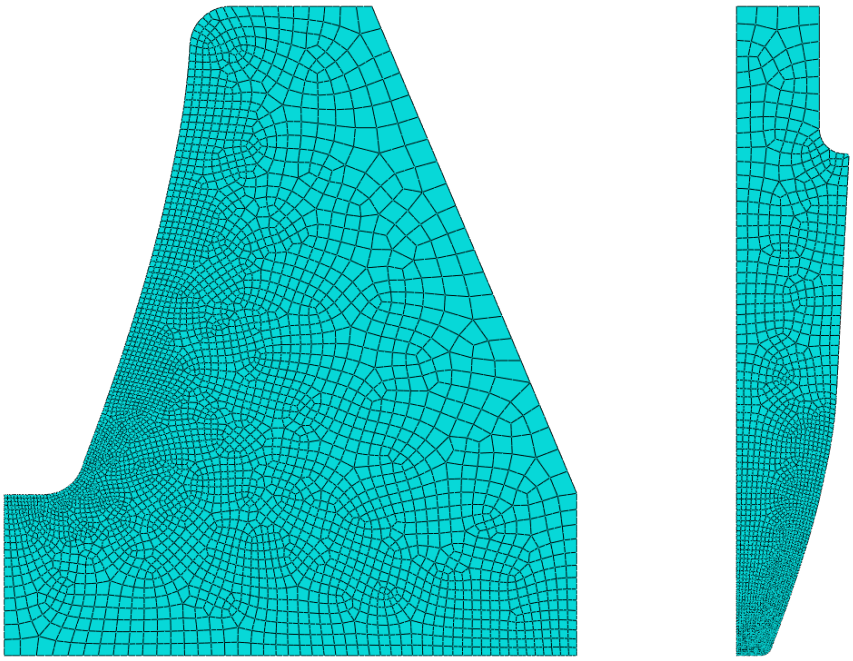


Figure 5.7: The tool meshes for the initial results with implicit method

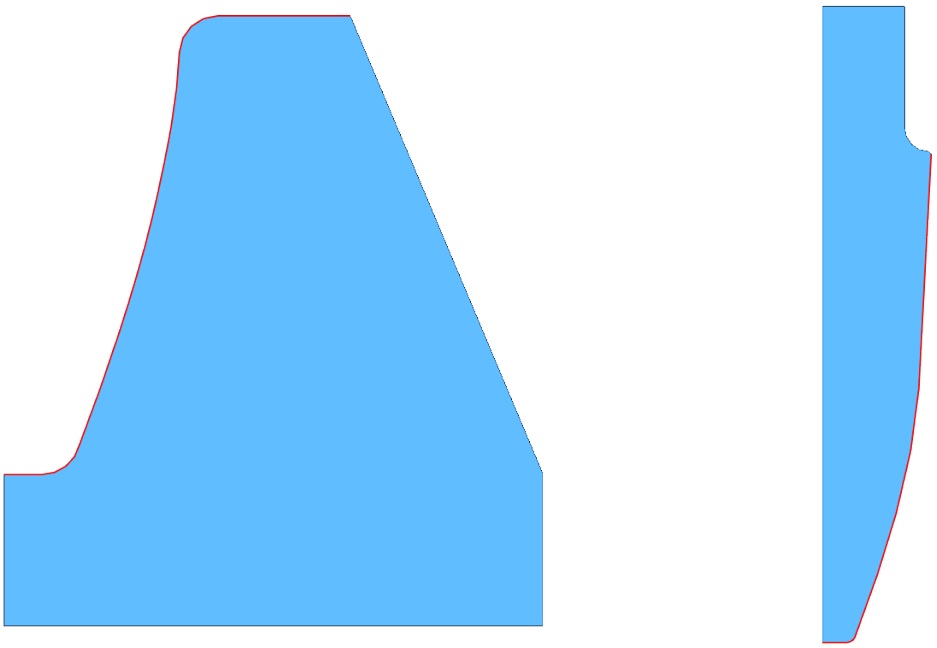


Figure 5.8: Surface definitions on the tools

The plate had two contact surfaces: one on the top of the plate and one on the bottom of the plate as seen in figure 5.9.

For the 3D model, the surface definitions are obtained by extending these lines in the direction of z-axis. It was especially important for the 3D-model not to define the surfaces in the areas that were not likely to establish contact in the simulation because of the reduced computational cost in contact tracking.

Figure 5.9: Surface definitions on the blank

Two contact pairs are defined in the model. The punch surface and the top surface of the plate form one, and the other is formed by the die surface and the bottom surface of the plate. The tool surfaces were defined as master surfaces and the plate surfaces were defined as slave surfaces. The contact pairs were active only for the forming part of the simulation and inactive for the springback step.

Finite sliding option was chosen for the simulation as the plate surfaces experience significant sliding with respect to the tool master surfaces during the simulation.

Unless otherwise mentioned, the constraint enforcement methods for the simulations were selected in the way that they would capture the shape of the final product accurately. In Abaqus/Standard this was accomplished by choosing the augmented Lagrange method option, which seemed to yield as accurate results with improved convergence when compared to the more strict direct method. In Abaqus/Explicit the kinematic method option was chosen. Penalty methods seemed to allow too large penetrations between the surfaces. The pressure-overclosure relationship was chosen as hard contact in Abaqus/Standard.

The contact discretization option was chosen as surface-to-surface contact in Abaqus/Standard and pure node-to-surface contact in Abaqus/Explicit.

The tangential friction behavior was modelled as isotropic with a constant slip rate independent Coulomb friction coefficient. The frictional constraints were imposed by means of the penalty stiffness method in Abaqus/Standard. This allows a small amount of relative motion between the surfaces when the surfaces should be sticking. This is to improve the convergence of the iterations [7, sect. 33.1.5]. The default value of 0.005 for slip tolerance was used. In Abaqus/Explicit the frictional constraints were imposed by means of the kinematic method, which is the default method for contact pairs.

5.7 Consistent units

The Abaqus software does not have a built-in unit system, except for rotation degrees of freedom and angles. Therefore, the units have to be input into the software ensuring that they use a consistent unit system. The unit system used in these simulations is SI(mm), see the table 5.1.

All data is input in these units to the software and these are also the units for the results unless otherwise mentioned. However, some of the units are converted to the

Table 5.1: Consistent units

Quantity	SI(mm)
Length	mm
Force	N
Mass	$tonne$ ($10^3 kg$)
Time	s
Stress	MPa (N/mm^2)
Energy	mJ ($10^{-3} J$)
Density	$tonne/mm^3$

standard SI units in the written part of the thesis because of their more common and well-known usage.

6. SIMULATION RESULTS

6.1 Results of the initial model

The initially provided geometry was designed in the way that the punch would thin the plate at the sides in the end. In other words, the punch geometry had an overlap of 5 mm with the plate at the sides of the punch if the punch would be assigned the whole punch depth. This is illustrated in figure 6.1 where half of the tools (cut on the symmetry plane) is illustrated as the gray areas and the profile of the desired shape of the formed plate is illustrated by the red lines. Pay attention to the overlap between the punch (upper tool) and the right side of the desired shape.

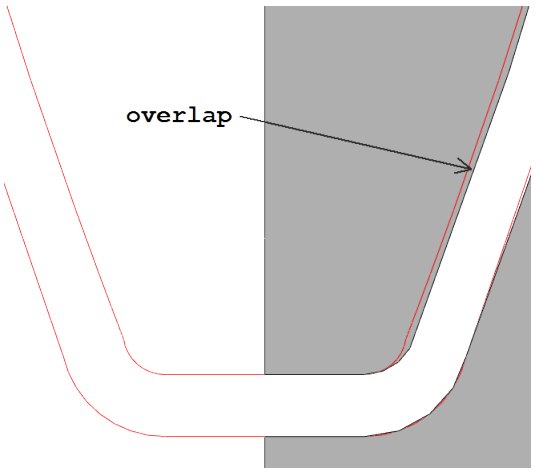


Figure 6.1: Overlap between the punch and the desired shape

6.1.1 Explicit analysis precision and efficiency

At an early stage of the simulations it turned out that the single precision accuracy used in analysis resulted in inaccurate displacements for the punch. This is seen as an extreme situation in figure 6.2 where the nodal displacements are also calculated with insufficient precision resulting in element distortion.

When double precision was used in the analysis, this problem disappeared. Abaqus manual mentions that double precision should be used in an analysis with more than 300 000 increments or with time increment size less than $10^{-6}s$ [7, sect. 6.1.1]. The time increment in the 1/5-mesh (see section 5.5.1) model was approximately $10^{-7}s$ and the number of increments it took to complete the analysis was 5 243 109. Even

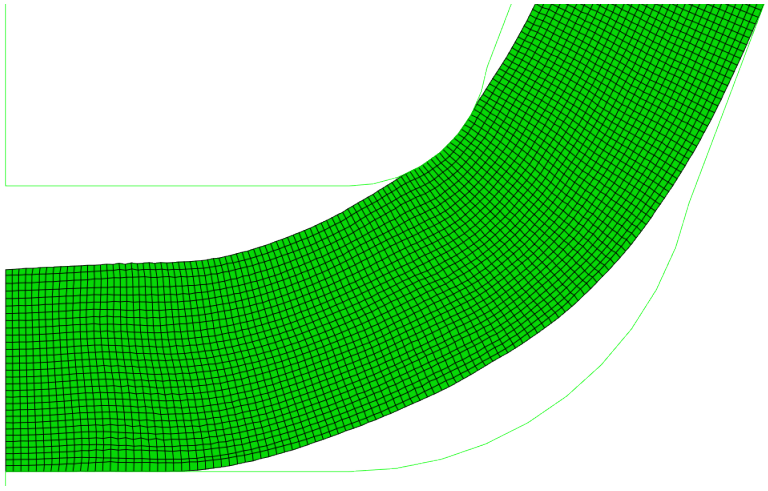


Figure 6.2: Single precision explicit analysis results with 1/5 mesh

with 10 elements through the thickness (3/5-mesh) the number of increments was over a million (1 445 332) and the time increment approximately $3.5 * 10^{-7}$. It was quite clear that double precision should be used to obtain accurate results in this simulation.

The small time increment also led to lengthy computation times so that the implicit analysis was considered to be more suitable for studying the initial results. The implicit analysis with the Abaqus/Standard code offered second-order elements. A comparison between the computational times of the explicit model with a 1.5/5-mesh of CPE4R-elements and the implicit model with a 3/5-mesh of CPE8R-elements performed with the rigid tool model showed that the implicit analysis was approximately one third faster than the explicit analysis. The implicit model also ensured that the force equilibrium was in balance after each step. Some techniques for accelerating the explicit analysis solution and comparison between the implicit analysis results will be discussed later.

6.1.2 Implicit analysis initial results

For the reasons stated, the results for the initial model are obtained by means of the implicit analysis performed with the Abaqus/Standard code. The 3/5-mesh of CPE8R-elements is used in these studies.

When the tools were modelled as rigid, the previously mentioned overlap of the initial geometry caused the plate to get compressed between the punch and the blank at the sides in the end so that its thickness reduced 5 mm. This required, even with frictionless contact, high pressing force of $360000kN$ from the punch.

When the tools were modelled as deformable, it was quite clear that the tools will have trouble withstanding the stresses in the process when the initial geometry of the model is used, as seen in the figure below where the gray area refers to a von

Mises equivalent stress value larger than 800 MPa.

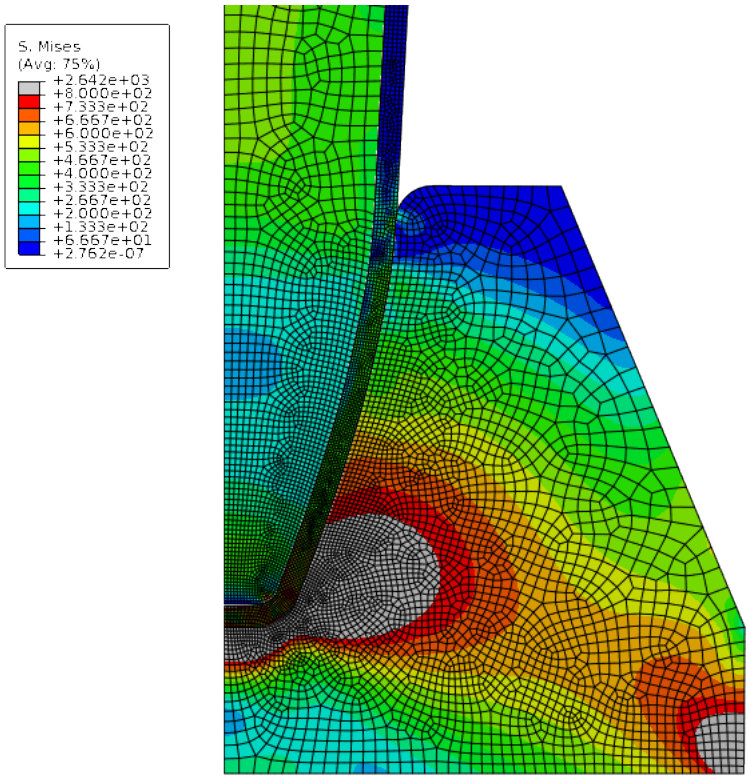


Figure 6.3: von Mises equivalent stresses in the tools with the initial geometry

The tools also deformed because of the high forces so that the whole punch depth at the bottom of the punch was not achieved. This resulted in an insufficient deformation considering the desired shape of the blank as seen in figure 6.4

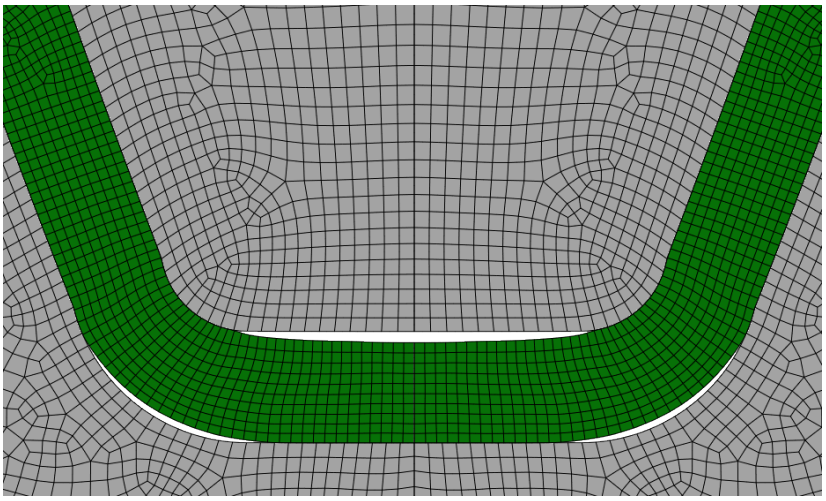


Figure 6.4: Blank configuration after forming step with the initial model geometry when the tools were modelled as deformable

When the plate was thinned on the sides, the normal forces acting between the contact surfaces got very high which resulted in high friction forces when a friction

coefficient of 0.1 was used in the model. When the tools were modeled as rigid, this was unrealistic and produced extremely high friction forces acting on the contact surfaces. Deformable tools gave the plate more space between the tools so that the friction forces were lowered. However, this implied that some ratio of the input energy put into the process had gone into the strain energy of the tools. The deformation of tools is not desired as almost all of the input energy should be used to deform the blank.

The needed pressing forces for each of the studied three cases are presented in figure 6.5. The rigid tool model with a friction coefficient of 0.1 resulted in an extremely high pressing force.

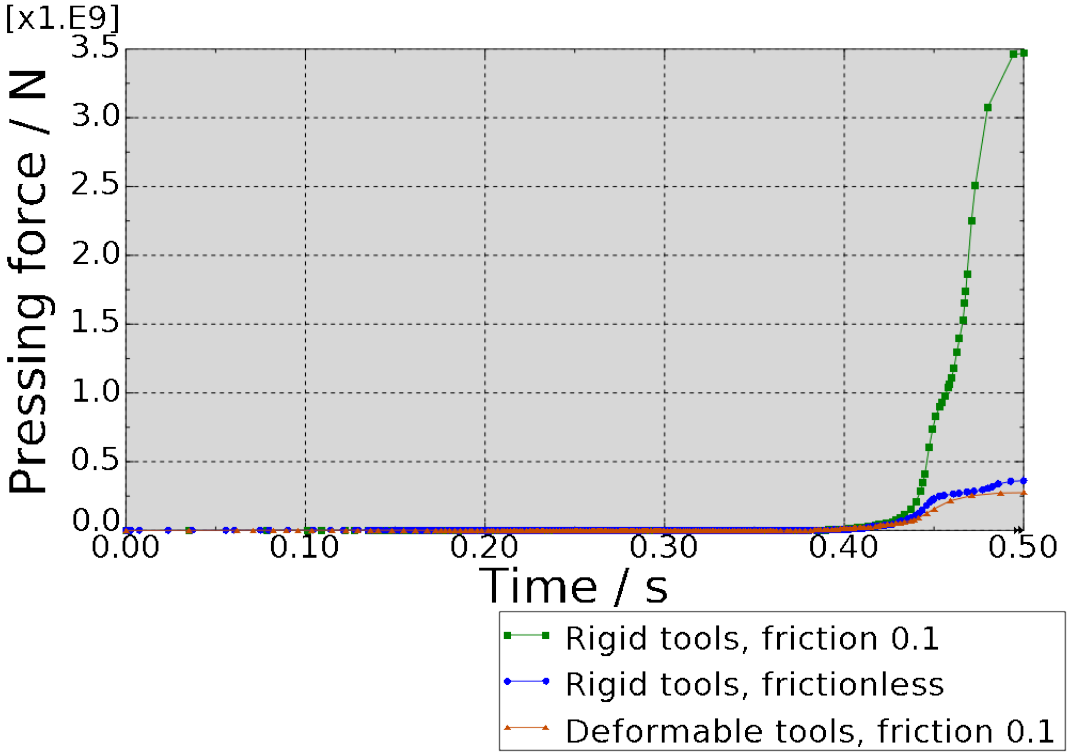


Figure 6.5: Needed pressing force for the initial model with rigid tools

The previously mentioned problems indicate that the initial geometry is unlikely to succeed. A decision was made not to study this geometry any further.

6.2 Modified punch geometry

An obvious way to improve the geometry was to remove the overlap from the punch geometry. This was done by offsetting the side geometry of the punch 5 mm inwards as seen as the dimensioned sketch on the bottom region of the punch in figure 6.6. Only the dimensions that have a significant effect on the simulation results are shown.

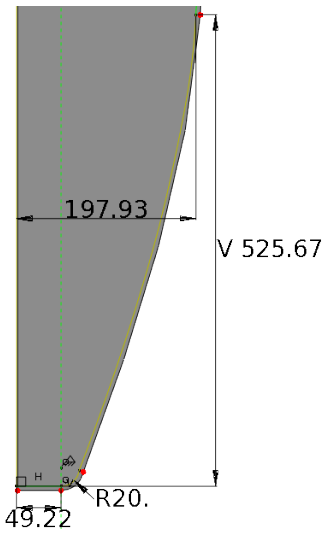


Figure 6.6: Modified punch geometry

6.2.1 Implicit analysis results with no overlap

When the simulation was performed with the deformable tool model, a coefficient of friction of 0.1 and this modified punch geometry, the results showed that the needed pressing force was lowered to approximately $114000kN$. The pressing forces for each of the three studied cases can be seen in figure 6.7.

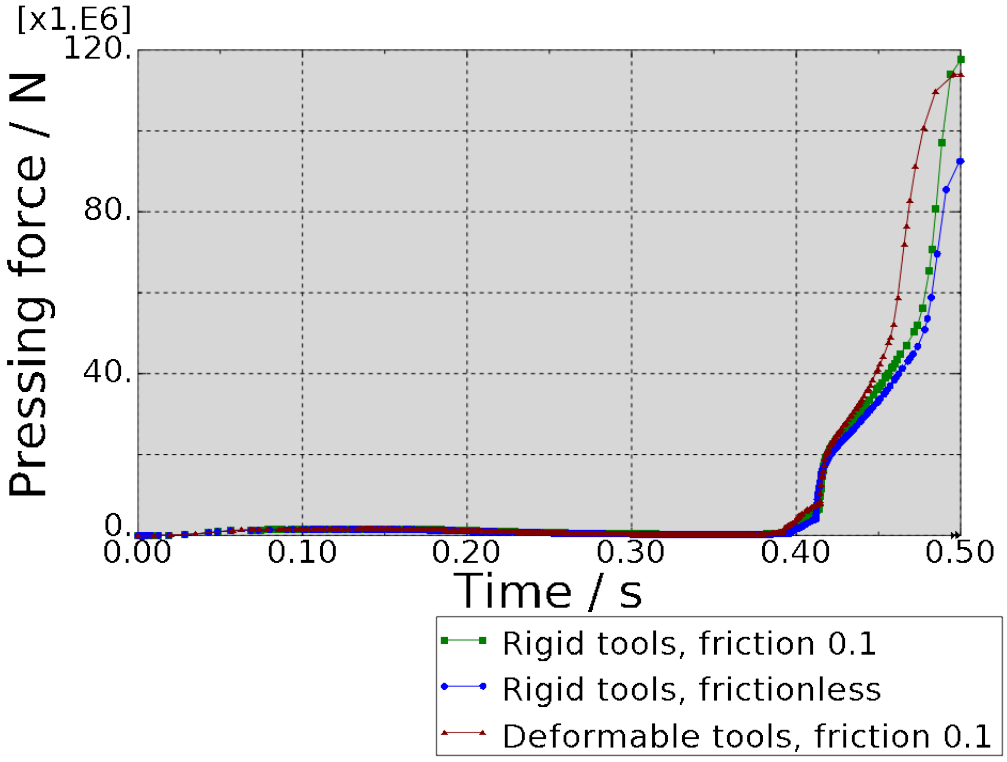


Figure 6.7: Needed pressing force for the modified punch geometry

However, the stresses in the tools were most likely still too high for them to withstand, see figure 6.8.

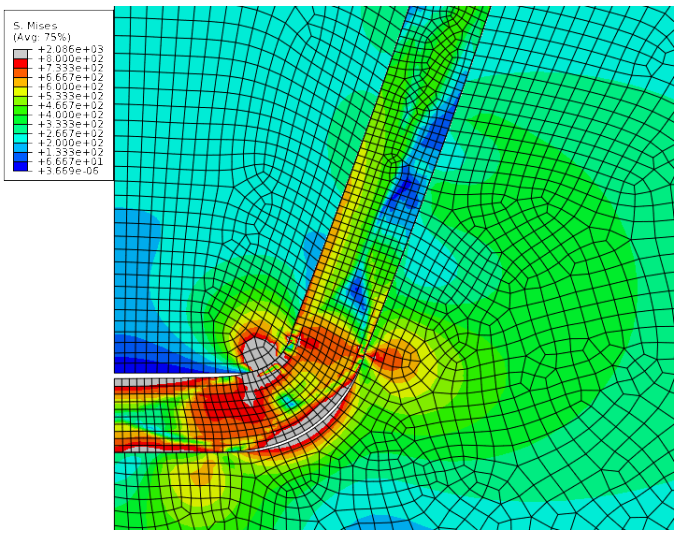


Figure 6.8: von Mises equivalent stresses in the tools with the modified punch geometry

The reason for the radical increase of the needed pressing force in the end of the forming process is the short lever arm that arises when the blank gets in contact with the die bottom, see figure 6.9. The lever arm between the die side wall and the die bottom gets shorter and shorter towards the end.

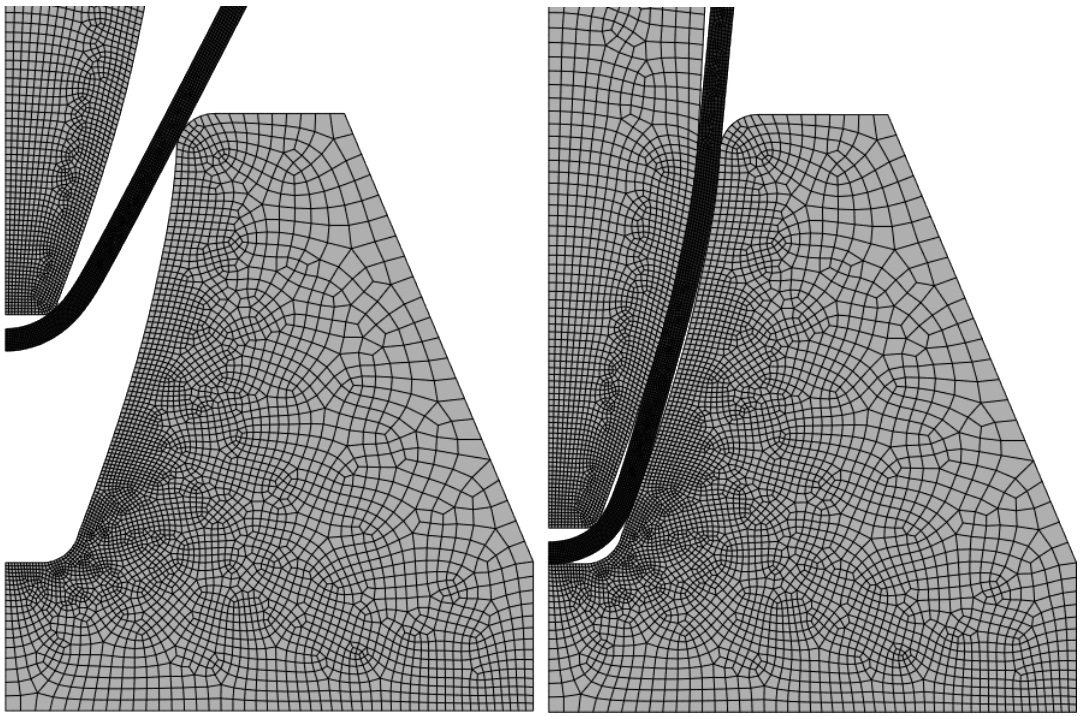


Figure 6.9: Blank configurations at $t = 0.25s$ (right) and $t = 0.45s$ (left)

6.2.2 Mesh density for parametrical studies

The model was tested for the effect of mesh density variations on the springback and the needed pressing force.

Only the meshes of the left side and the middle side were modified in these studies, the mesh density of the right side was kept constant. This was because the maximum amount of springback will be because of the bottom curve of the part near the symmetry plane that magnifies the effect to the upper part.

The model was first simulated with no friction between the contact interfaces with meshes of 1/5, 2/5, 3/5, 4/5 and 5/5. The springback displacements along the path of the bottom of the plate for different meshes are plotted in figure 6.10.

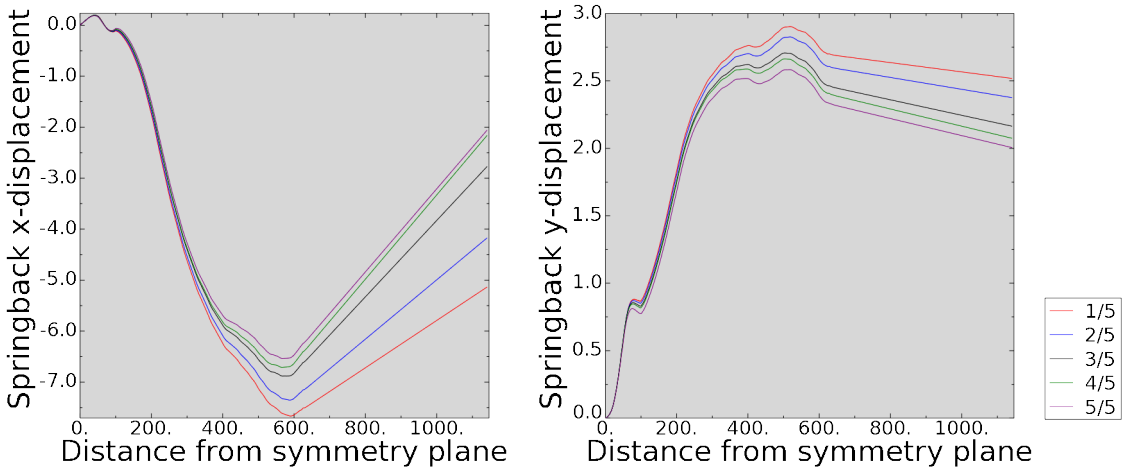


Figure 6.10: Springback displacements for the bottom edge of the plate with different meshes, frictionless model and millimeters as units

Considering springback, the thing that varied most between these meshes was the springback of the free end. The springback of the bending deformations get magnified in it. All the results after springback with different meshes were less than 3 millimeters from each other. The shape of the curves is pretty much the same, only amplitude changes with different mesh densities.

The simulation with the finest mesh of 1/5 took 12793 seconds of user time. User time is the CPU time spent on the execution of an Abaqus process and excludes the CPU time spent by the operating system working for the Abaqus process. Therefore, it is a good measure of the spent CPU time for the calculations of the finite element method solution. The user times percentually to that of the simulation with 1/5 mesh and the maximum pressing forces with different meshes are combined in the table 6.1.

The pressing force for the mesh of 5/5 is higher than some of the finer meshes. This is assumed to be because of an insufficient accuracy of the mesh to follow the loading path and the hardening behaviour involved.

Table 6.1: CPU user times and pressing forces for different mesh densities, frictionless model

mesh	User time / %	pressing force / MN
1/5	100	101
2/5	29	96
3/5	20	93
4/5	19	92
5/5	6	97

See figure 6.11 for a demonstration of the complete configuration of the 1/5 mesh model before and after springback. The half-symmetry blank is rotated $-90°$ about the z-axis for space-saving purposes.

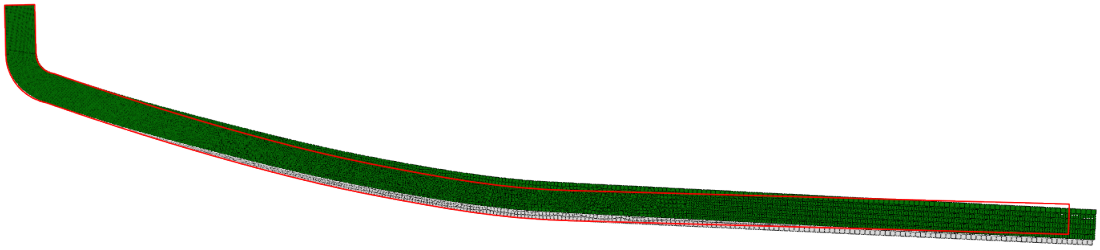


Figure 6.11: Blank configuration after springback (green), before springback (white) and the desired shape (red profile)

The 2/5 implicit CPE8R model is found to be appropriate for the parametrical studies because of its assumed sufficient accuracy and computational efficiency. The user time for the simulation with the selected 2/5-mesh model was 3708 seconds. The needed pressing force varied between the chosen 2/5 model and the finest 1/5 model but this was not of considerable importance as the parameter modifications that were to be applied to the process are used only to see the effect of these changes on the springback and pressing force. The 2/5-mesh is assumed to capture the effects of these parameter changes sufficiently well.

The meshes were tested also with a Coulomb friction coefficient of 0.1 only to see that the friction did not affect the springback significantly, see figure 6.12. In fact, the springback displacement values decreased a bit when the simulation was performed with the friction model.

The user times percentually to that of the 1/5-mesh frictionless model and the computed pressing forces for three of the finest meshes are combined in table 6.2.

The user time for the chosen 2/5-mesh model increased 33% to 4946 seconds when penalty friction was added to the contact behaviour. The needed pressing force increased 23%.

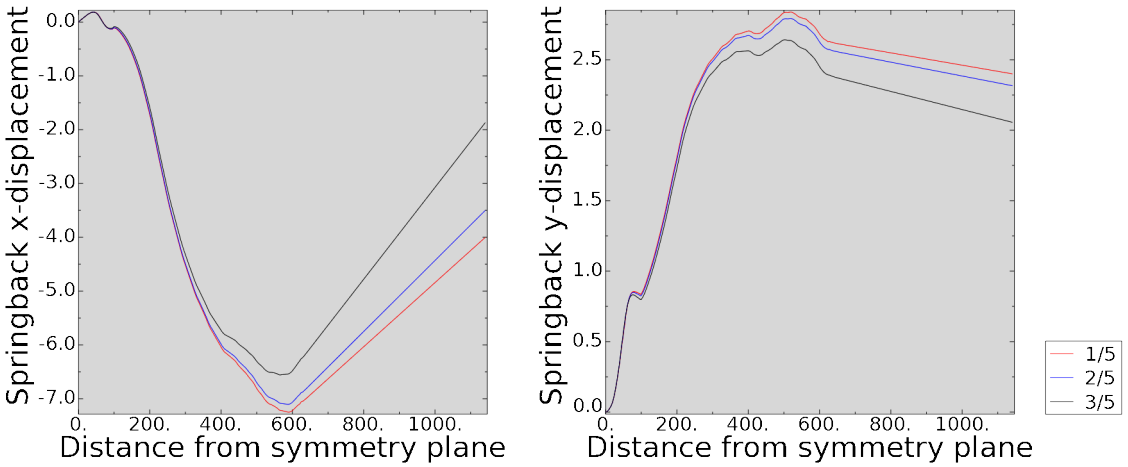


Figure 6.12: Springback displacements for the bottom edge of the plate with different meshes, friction coefficient of 0.1 between the contact surfaces and millimeters as units

Table 6.2: CPU user times and pressing forces for different mesh densities, model with penalty friction of 0.1

mesh	User time / %	pressing force / MN
1/5	173	132
2/5	39	124
3/5	25	118

6.2.3 Explicit analysis mass scaling

The computational disadvantages of the explicit analysis were evident. The decision for the time scale of 0.5 seconds was based on the guideline that the kinetic energy should not exceed 10 % of the total internal energy throughout the majority of the simulation. The ratio did exceed this guideline in the beginning but was reduced quite quickly to approximately 3 % at t=0.1s. The kinetic energy in the beginning was caused by the rigid body motion of the large region on the right side of the modelled plate. This region was not subjected to any plastic straining nor significant stresses at that time, and therefore, might not be of much interest when considering an acceptable quasistatic solution. See figure 6.13 for the configuration of the model at t=0.1s.

The time period of 0.5 seconds most probably yielded results accurate enough concerning the quasistatic solution but this time period with the true density of the material were computationally inefficient. Therefore, a study was made on the possibilities of mass scaling in increasing the computational efficiency of the solution without loosing the accuracy on the model. The study was made with the 1/5-mesh of CPE4R elements.

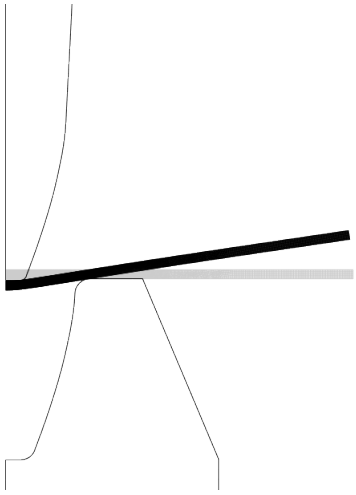


Figure 6.13: Configuration at t=0.1s with a total time scale of 0.5s

Mass scaling and increasing the tool velocity have the same effects on the solution time unless the model includes rate-dependent materials or damping, which is not the case in the current simulation model. From the computational efficiency point-of-view, scaling the mass by a factor of s_m corresponds to a time scaling factor of $\frac{1}{\sqrt{s_m}}$ in a simulation with unscaled density of the material. This corresponds to a factor of $\sqrt{s_m}$ on the punch speed. The mass scaling is performed by modifying the material density. The equivalent plastic strain keeps track of the history of the plastic straining (see equation (3.17)), and therefore, it is assumed to be a good measure of the deformation throughout the loading history. The influences of the scaling on the amount of springback and needed pressing force are also of interest.

The mass scaling factors s_m were set to 2, 4, 9 and 16, these would correspond to a speed-up of the tool velocity of $\sqrt{2}$, 2, 3 and 4, respectively. See table 6.3 for the corresponding densities.

Table 6.3: Densities for different mass scaling factors

s_m	1	2	4	9	16
ρ / $\frac{kg}{m^3}$	7800	15600	31200	70200	124800

The results for the kinetic energy and the ALLKE/ALLIE-ratio can be seen in figure 6.14. The needed pressing force increased with the increase in the mass of the blank as seen in figure 6.15 where the pressing forces for each different mass scaling factors are plotted at $t = 0.375 - 0.5s$ where the forces differed the most.

The equivalent plastic strains (PEEQ in Abaqus convention) at the bottom part of the plate were found to not differ much with different mass scaling factors. The maximum values were near the bend radius and differed $0.521 - 0.539$. No notable

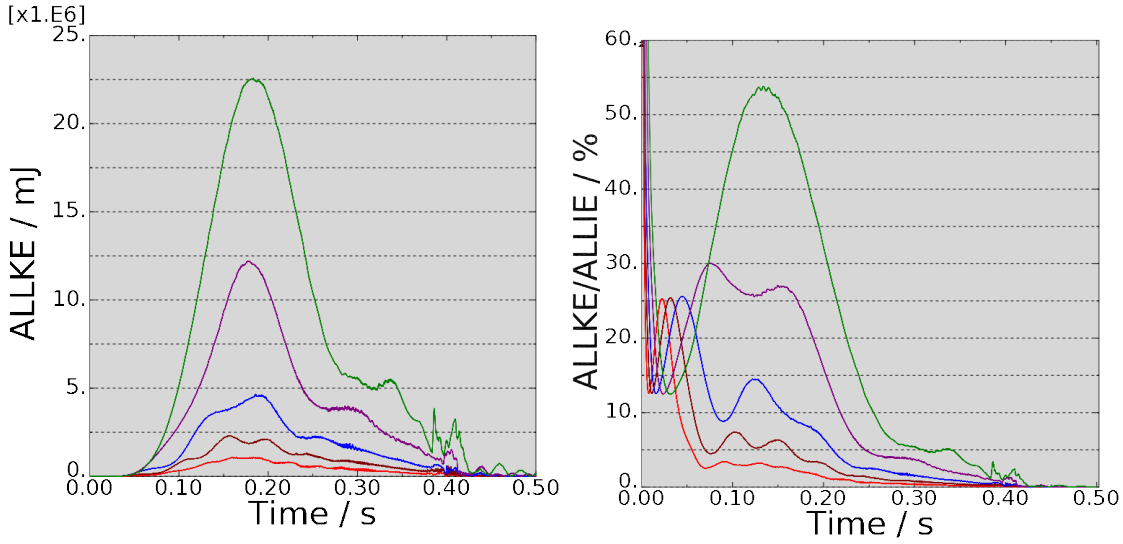


Figure 6.14: The kinetic energy and the ratio of kinetic energy / total internal energy with different mass scaling factors, color codes in the next figure

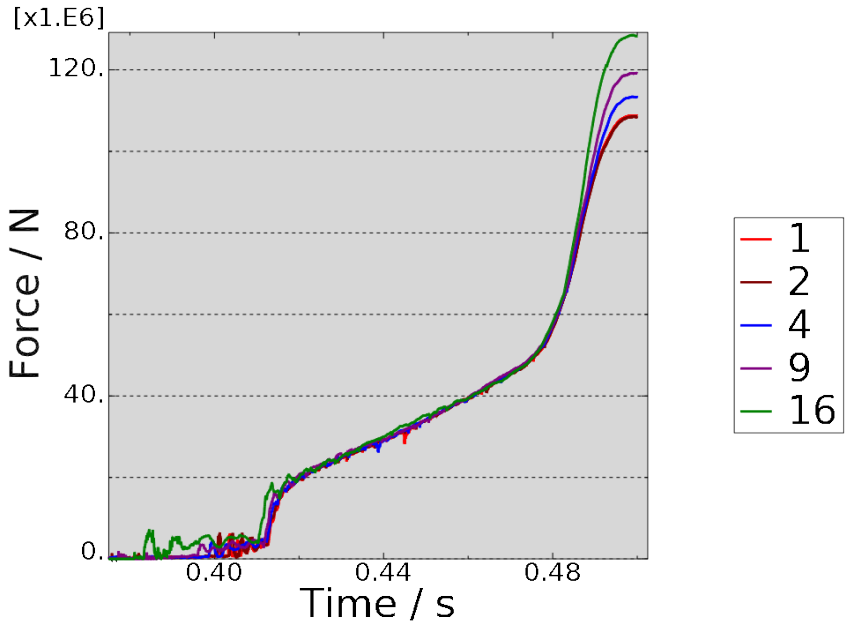


Figure 6.15: Needed pressing forces with different mass scaling factors at t=0.375-0.5s

difference could be seen in the PEEQ contour plots between the results obtained with different scaling factors. Therefore, the PEEQ contour plot could be unsuitable to study the effects of mass scaling.

See figure 6.16 the springback nodal displacements on the bottom of the plate. This time the path definition is the true distance of the bottom edge nodes in the deformed configuration, which results in some distortion when compared to those of the section 6.2.2. However, the shape of the curves should be quite similar. These springback displacements did not completely define the blank configuration after the springback because the configurations before springback differed. See figure 6.17 for

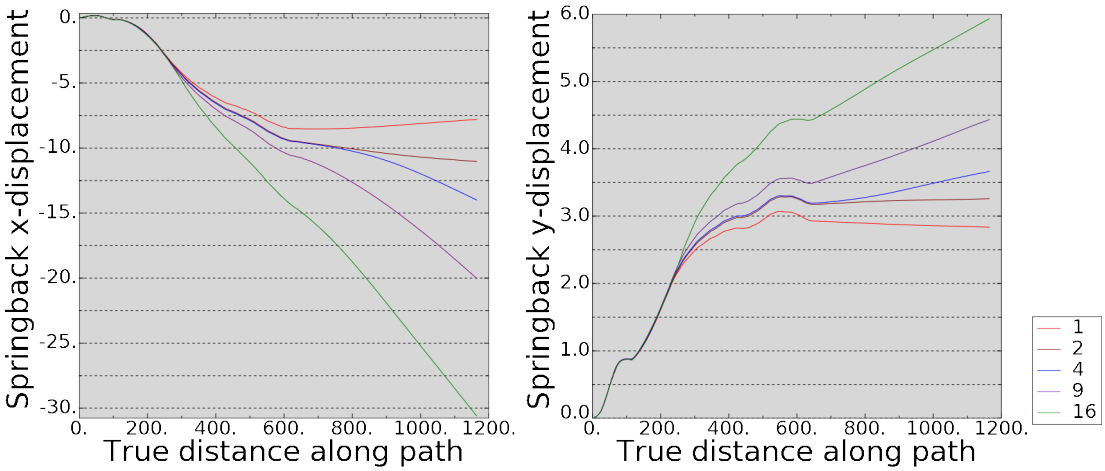


Figure 6.16: Springback displacements for different mass scaling factors

differences in the blank configuration before springback with the same color codes as in figures 6.14 and 6.15.

Figure 6.17: Blank configurations before springback with different mass scaling factors

The distance at the free end of the plate before springback was about $14mm$ in magnitude when the model with no mass scaling was compared to the one with most mass scaled. The distance with the mass scaling factor of 2 was less than $1mm$, $4mm$ with mass scaling factor of 4, and $7mm$ for the mass scaling factor of 9.

Based on these results, the mass scaling factor of 2 was found to be suitable for the speed-up of the computation in further studies. This is assumed to be of sufficient accuracy, although the springback differed about $2mm$ in the x-direction, the difference in pressing force was negliglible. The possibilities of mass scaling could further be broadened by only scaling the mass of the regions which include the smallest elements, because the step size of the explicit procedure is defined by the properties of the smallest element in the mesh, but the scaling done here is taken to be of sufficient computational efficiency for the purposes of this thesis.

The springback displacements seemed to differ from those obtained with the second-order elements CPE8R in the implicit mesh density study. Some further studying on the reasons for this is discussed next.

6.2.4 Difference in springback between explicit and implicit procedures

The simulation was ran as an implicit procedure with first-order CPE4R elements for comparison purposes. The results for different meshes are seen in figure 6.18.

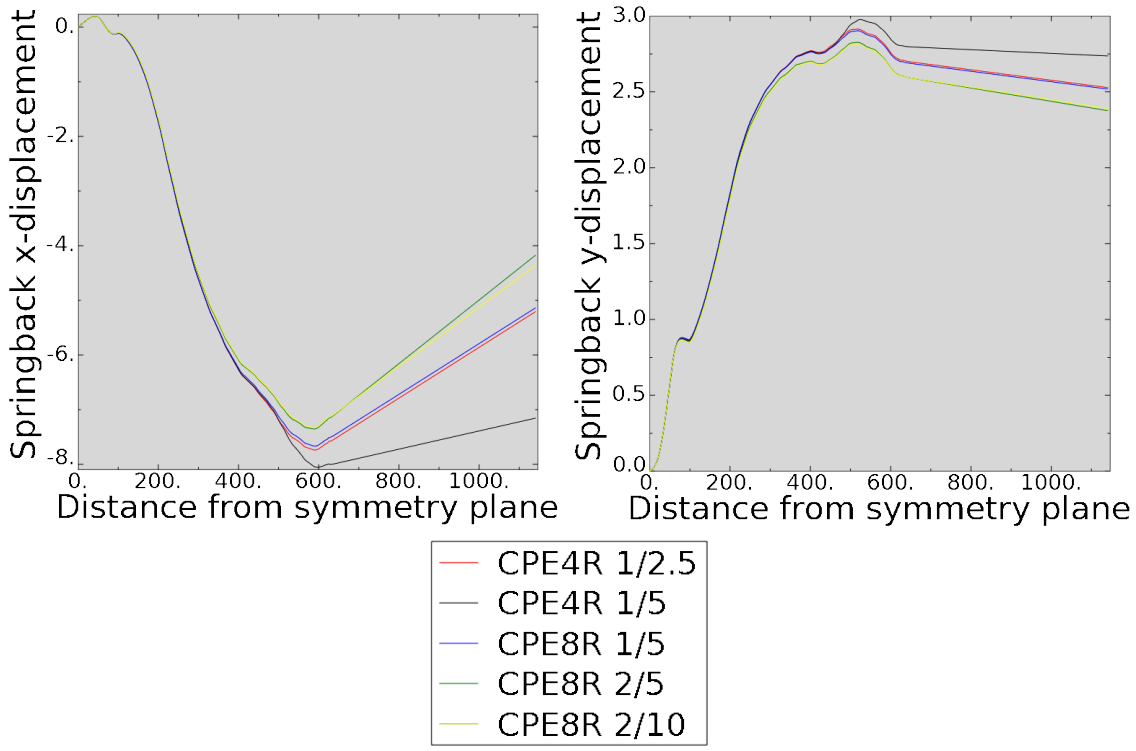


Figure 6.18: Implicit method springback comparison with different meshes of first-order and second-order reduced integration elements

The results with CPE4R 1/5-mesh are similar with the explicit and implicit procedures. It seems that the CPE8R results model the bending stresses at the middle region better than the CPE4R elements, because of their higher order interpolation, and this results in more accurate springback displacements. This conclusion was based on the fact that both of the elements seemed to converge to the same springback displacements, see the red and blue curves in figure 6.18.

It was also noticed that the CPE8R 2/10-mesh yielded similar results with the CPE8R 2/5-mesh concerning the springback displacements. The pressing force between these meshes did not differ no more than $600kN$. Therefore, the 2/10 mesh was chosen for the parametrical studies.

6.2.5 Explicit analysis hourglass control option comparison

Abaqus analysis manual mentions that the used enhanced hourglass control option can cause overly stiff behavior in bending problems that involve plastic yielding [7,

sect. 22.1.4] which is exactly the case in the current simulation. Therefore, the enhanced hourglass control option was compared to other hourglass control options available in Abaqus/Explicit. This study was done with a coarse mesh with only 5 elements through the thickness of the plate as seen in figure 6.19.

Figure 6.19: Coarse mesh for hourglass control method comparison

It seemed that the enhanced hourglass control was the method that differed the most from all of these as seen in figure 6.20. The forces are plotted only at $t = 0.375 - 0.5$ seconds where the most of the difference is observed. The use of the

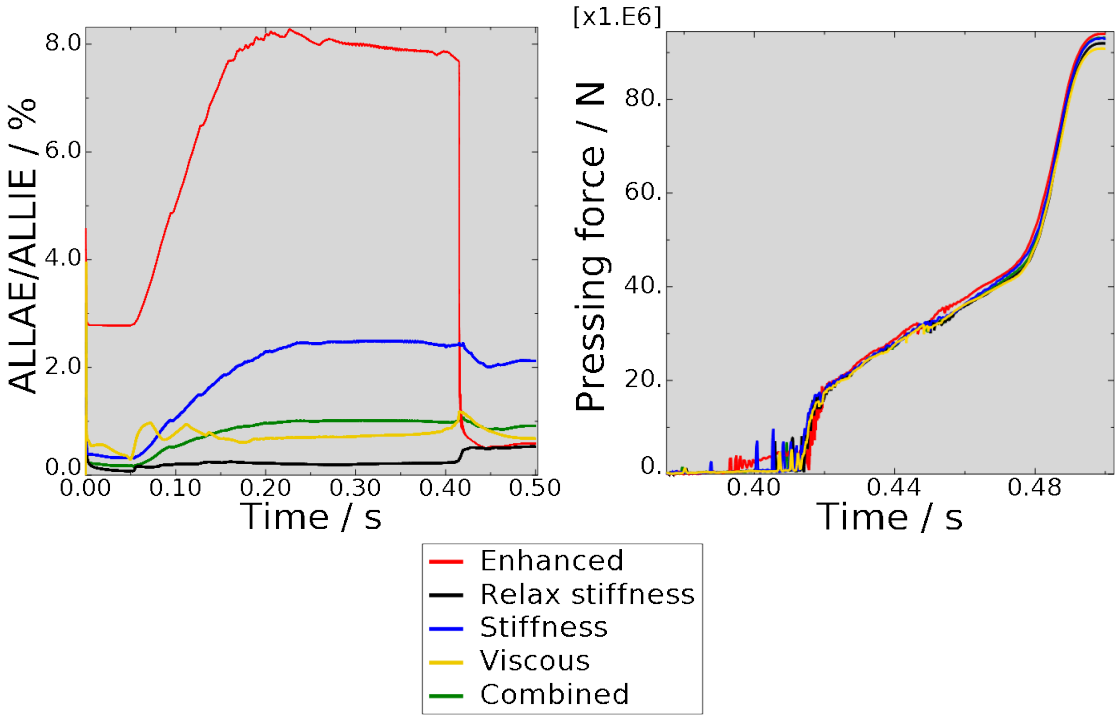


Figure 6.20: Hourglass control option comparison with coarse mesh

enhanced hourglass control yielded a bit more stiff response than the other hourglass control methods when considering the pressing force. Also, the enhanced hourglass control option created the most artificial strain energy to control the hourglass modes.

The viscous hourglass control option is the most computationally efficient option but it is not recommended for low frequency dynamic or quasistatic problems because the static-like loading causes the hourglass modes to excessively deform due to the lack of nominal stiffness [7, sect. 24.1.4]. The combined hourglass control option is a weighted method including both viscous and stiffness method and thus

is not of much interest in this context. Therefore, these options were dropped and a further study was made with a mesh of 1/5 with the enhanced, relax stiffness and the stiffness hourglass control options to see if the difference would be clear even with a finer mesh. The relax stiffness option is the default option for the used element and material in Abaqus/Explicit and it is based on the integral viscoelastic approach [7, sect. 24.1.4]. Stiffness option uses artificial stiffness coefficients to calculate the hourglass resisting forces and is recommended for quasistatic as well as transient simulations. The enhanced option is a refinement of the stiffness method in which the artificial stiffness coefficients are obtained from a three-field variational principle. It is based on the enhanced assumed strain and physical hourglass control methods proposed in [29, 30, 31, see [23, sect. 3.2.4]].

The results for the comparison between these three methods with the finer mesh can be seen in figure 6.21. The color codes are the same as in figure 6.20. The

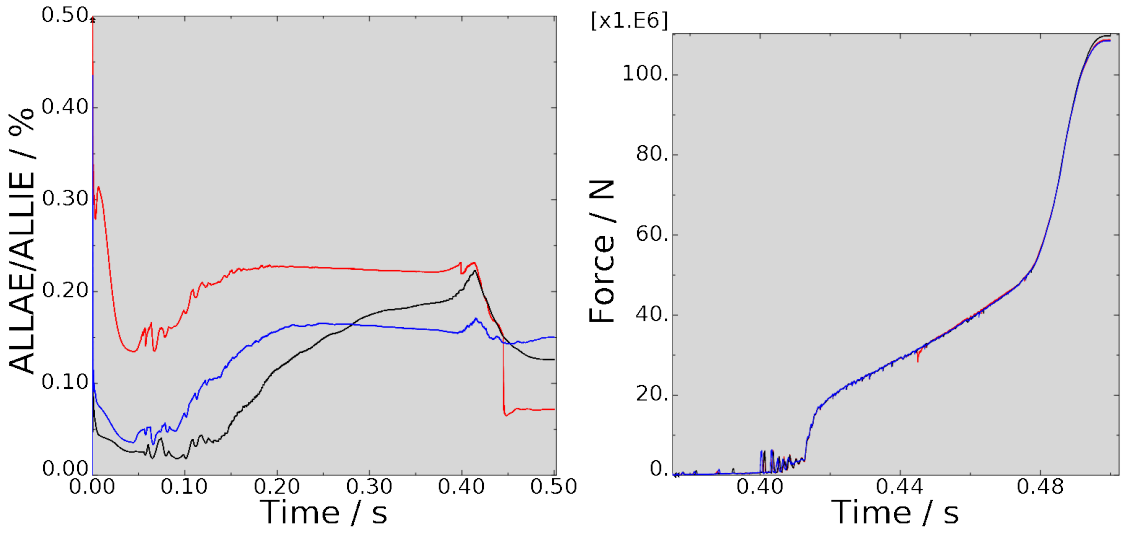


Figure 6.21: Enhanced, relax stiffness and stiffness option comparison with 1/5 mesh

ALLAE/ALLIE-ratio was decreased to a neglible value with every option so that the pressing forces as well were technically the same. Only relax stiffness gave a bit higher pressing force of $1000kN$ in the end. For a mesh this fine, the hourglass control option selection does not have any significant effect on the results. The possible effect on the results can be viewed by the ALLAE/ALLIE-ratio: as it becomes small, so does the effect of the hourglass control option choice on the results.

The ALLAE/ALLIE-ratios for different number of elements through the plate thickness with the enhanced hourglass control option can be seen in figure 6.22. The corresponding meshes are, from the most fine to the most coarse, 1/5, 1.5/5, 2/5, 3/5 and the coarsest is the one seen in figure 6.19. One can see that the ratio reduces to less than 1 % with about 15 CPE4R elements through the thickness of the plate at the region of most interest. However, this is not the case for a more

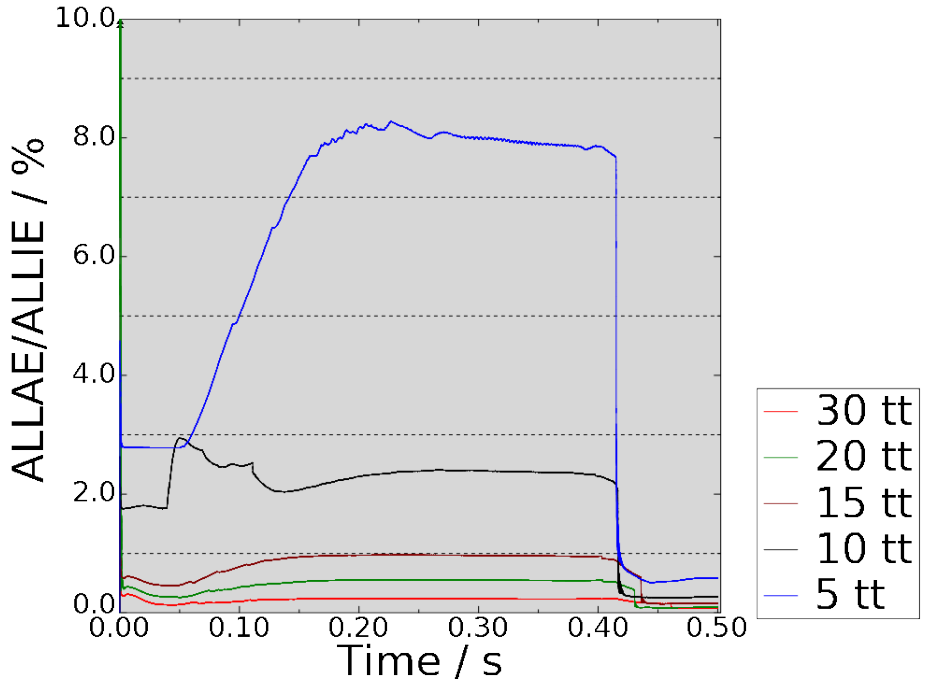


Figure 6.22: The ratio of artificial strain energy to total strain energy with different number of elements through the thickness of the plate (tt) in the region of most interest

flexible material model as will be discussed in the result analysis chapter.

6.3 3D model results

6.3.1 Shell element model

The simulation was performed with a shell element model in Abaqus/Explicit because of the computational advantage of their plane stress formulation when compared to the 3D stress elements. Also, shell elements model bending accurately with only one element through the thickness.

The shell elements first tried in the simulation were S4R first-order reduced integration shell elements. The simulation broke down at $t = 0.4647s$. This was the case even when the contact constraint enforcement method was changed to the penalty method to allow some penetration between the tools and the blank. The fully integrated S4 shell elements were also tried only to see the same kind of breakdown in the simulation at $t = 0.48s$. The region which made the simulation break down is seen in figure 6.23 where the most distorted elements are located at the bend radius. The edges visible in the figure are at the symmetry planes. The contour plot in these figures is of equivalent plastic strain on the top surface of the shell. The meshes used in the simulations of figure 6.23 included 18280 elements for the S4R case and 10835 elements for the S4 case. The minimum element size at the bend radius is 2x4 for the S4R case and 3x6 for the S4 case with the smaller dimension

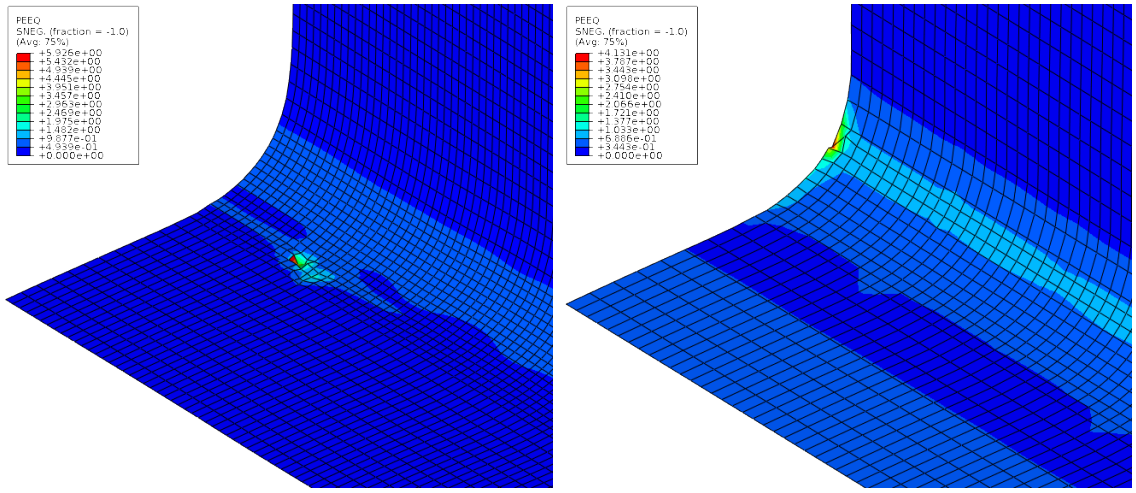


Figure 6.23: Overly distorted elements with the conventional shell model with S4R (left) and S4 (right) elements

coinciding, in the initial configuration, with the x-axis which can be seen in figure 5.1.

The problems in this conventional shell model are assumed to be because of the complex two-sided contact conditions that arise in the end of the simulation.

6.3.2 Solid continuum 3D stress elements

Some more modifications to the mesh of the 3D model had to be done to make the model computationally reasonable when the plate was modelled with solid continuum elements. The meshes for the 3D models can be seen in appendix A.1 for the implicit procedure with 32266 C3D20R elements and 146632 nodes and appendix A.2 for the explicit procedure with 216389 C3D8R elements and 238293 nodes. A relatively small number of wedge elements are included in both meshes to smooth the mesh transition: 6 second-order wedge elements for the implicit mesh and 533 first-order reduced integration wedge elements for the explicit mesh. The explicit mesh has at least 10 elements through the thickness of the plate in regions where the bending deformation happens, and thus, is assumed to more accurately capture the springback behaviour than the mesh used in section 6.2.4 for the plane strain model.

Implicit procedure

The complex contact conditions in the end of the forming process made the 3D implicit model converge very slowly. The penalty contact constraint enforcement method was selected to improve the convergence, and the punch displacement was set to be ramped up to ensure that the amplitude curve did not affect the built-in automatic increment size control of Abaqus/Standard.

The needed pressing force in this analysis was $123000kN$ with the frictionless model. The pressing force got higher than that of the plane strain model results. The reason for this can be seen in figure 6.24 where the free edge (parallel to xy-plane) of the plate is seen in the boxed regions. The bending deformation around the axis parallel to z-axis caused compression stress in the x-direction on the top surface of the plate and tension stress in the x-direction on the bottom surface of the plate. This caused, through the Poisson effect, bending at the free edge of the plate around an axis parallel to x-axis. See the boxed regions in the figure 6.24 for the deformation caused by this bending. The fact that also the new bending deformation had to be bent straight added some extra forces to resist the punch movement in the end of the forming process.

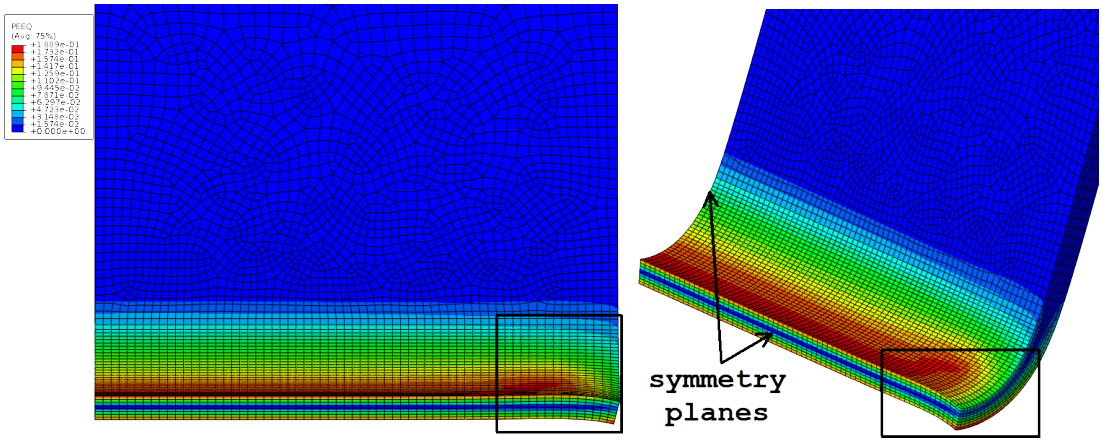


Figure 6.24: Undesired bending deformation at the free edge of the plate bottom part before it gets in contact with the die cavity bottom at $t = 0.85s$ (the total time scale for forming is $1s$ here)

This implicit 3D model took several days to complete even when the simulation was ran on multiple processors. The springback step failed with no increments taken. More discussion on the results with the implicit analysis can be found in the result analysis chapter section concerning the comparison between the dynamic and static implicit procedures.

Explicit procedure

The explicit procedure was performed with a mass scaling factor of 2 on the whole blank to make the model computationally efficient without loosing significant accuracy, see section 6.2.3.

The 3D model with C3D8R elements resulted in a need for the pressing force as high as $132000kN$ with no friction and $184000kN$ with a friction coefficient of 0.1 in the contact pairs. The mesh seemed to be sufficiently fine with the ALLAE/ALLIE ratio lower than 1 % throughout the analysis. However, it was found that the

kinematic contact constraint enforcement method coupled with the the small contact area between the top surface of the plate and the punch radius caused hourglass-like patterns in the mesh when the plate got in contact with the die bottom, see figure 6.25 where the contour plot of artificial strain magnitude in the element for the whole element (ELASE) is plotted on the left side and contact pressure (CPRESS) contour plot on the surface nodes on the right side. The hourglass patterns are in the bend curve triggered by point loads caused by the small area of contact between the punch and the blank. The ELASE contour plot does not completely describe the regions where the most artificial strain energy is created as the value is integrated over an element and the element sizes differ. The hourglass patterns were more clearly seen in the model with no friction.

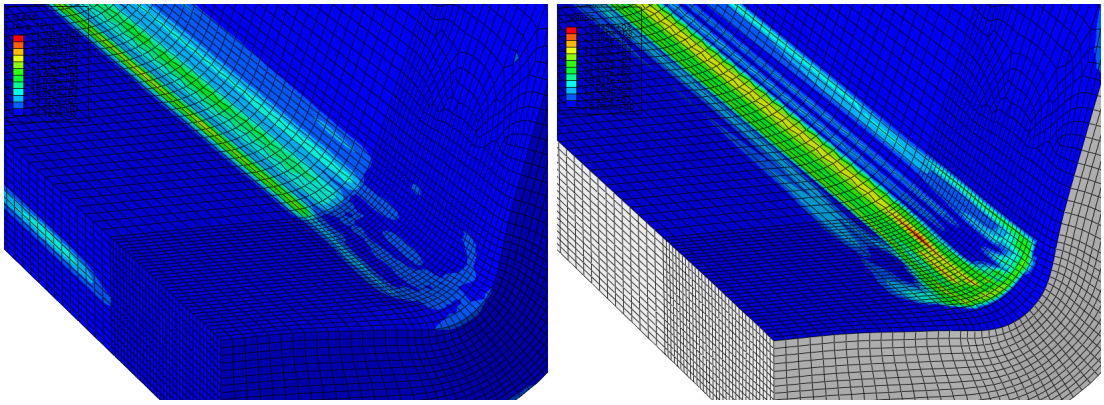


Figure 6.25: Hourglass patterns in the minimum radius bend curve region with the explicit 3D model and kinematic contact constraint enforcement, contours of ELASE (left) and CPRESS (right)

The hourglass patterns reduced when the penalty contact constraint enforcement method was chosen to allow a small amount of penetration between the punch and the blank top contact surfaces. This distributes the contact forces between the blank and the punch to a larger area on the blank mesh. Some hourglass patterns could still be seen in the mesh although greatly reduced from the solution with the kinematic contact constraint enforcement method, see appendix A.3 for close-ups at the bend curve region adjacent to the symmetry plane. This resulted, with the friction model, in a pressing force of approximately $164000kN$. For the frictionless model, the pressing force with the penalty method was $121000kN$.

The forces obtained from each of the performed simulations with penalty/kinematic contact constraint enforcement and frictionless/friction models at $t = 0.375 - 0.5s$ can be seen in figure 6.26.

Even though the penalty method reduces the hourglass patterns at the inside bend radius, it does not capture the shape of the bend accurately and results in a lower need for the pressing force when compared to the kinematic constraint enforcement

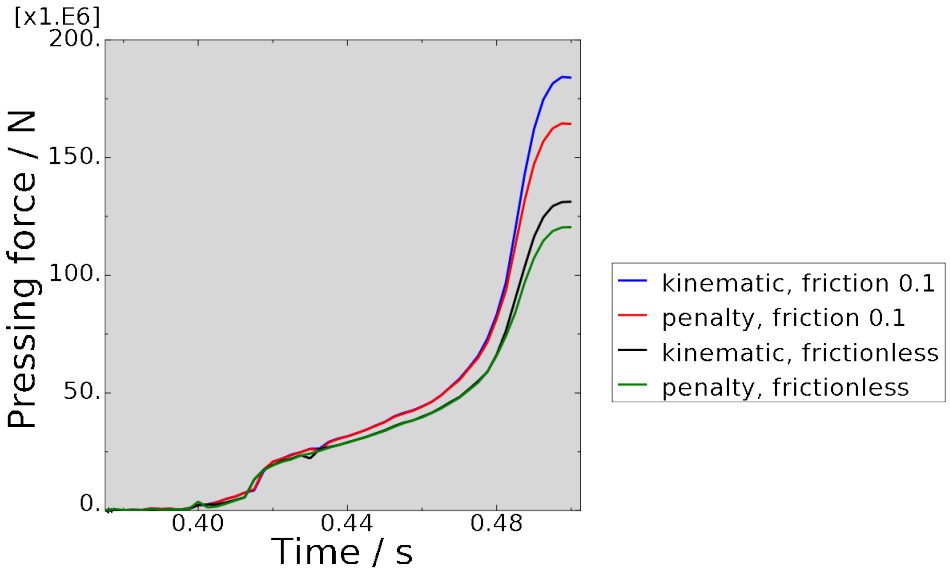


Figure 6.26: Explicit 3D model pressing forces with different contact constraint enforcement method and friction models

method. The hourglassing phenomena should lead to an overly flexible behavior of the structure, and thus, it is debatable which of these methods should be used for the simulation.

The springback displacements of the model with penalty contact constraint enforcement were similar to those of the kinematic, only slightly lower in value. This is because of the small penetration between the surfaces that the penalty method allowed at the bend curve. The springback displacements at the bottom surface of the plate for the frictionless/friction models with kinematic contact constraint enforcement can be seen in appendices A.4 and A.5. The springback contour plots for the penalty constraint enforcement can be seen in appendices A.6 and A.7. The z-direction springback was negligible with a maximum value of $-0.36mm$. The friction model springback was again quite similar to the frictionless model but were a little lower in value with the maximum magnitude value differing about $1.5mm$ with both models in x-direction. The x-direction springback seemed to be similarly distributed in the xy-plane cuts of the plate quarter.

6.4 Material parameter modifications

6.4.1 Preheating the plate

The parametrical studies were performed using the implicit dynamic procedure with the CPE8R 2/10-mesh seen in figure 6.27.

The reduction factors for the material properties at different temperatures are obtained from source [32, p. 282]. These are the test results for structural steel

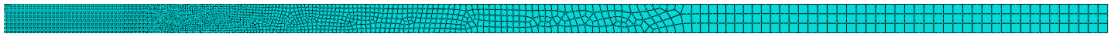


Figure 6.27: CPE8R 2/10-mesh for the parametrical studies

S355J2H.

As no data for the shape of the hardening curve is provided, the hardening is again modelled as linear by using the same reduction factor on the plastic modulus E_p as is used for the elastic modulus E. Although the proportional limit also varies in the test results of the article, it is not taken into account so that the stress/strain curve input is still bilinear.

The material property modifications for different temperatures T are compiled in the following table:

Table 6.4: Assumed material properties at different temperatures

T / °C	E / GPa	σ_y / MPa	E_p / MPa	σ at $\varepsilon^p = 1$ / MPa
20	200	427	1634	2061
200	180	414	1471	1885
300	160	406	1307	1713
400	140	363	1144	1507
500	120	278	980	1258
600	62	137	507	644

The needed pressing forces obtained from the studies are compiled in the table 6.5. The needed pressing force decreases with increasing temperature as the yield strength and the plastic modulus decrease.

Table 6.5: Needed pressing force at different temperatures with frictionless/friction models

T / °C	$\mu = 0$	$\mu = 0.15$
20	96 000 kN	137 000 kN
200	92 000 kN	128 000 kN
300	85 000 kN	120 000 kN
400	76 000 kN	110 000 kN
500	62 000 kN	89 000 kN
600	34 000 kN	48 000 kN

The springback displacements increase as the Young's modulus decreases and the springback decreases as the yield strength decreases. This is because the elastic part of the strain, which tries to relax after tool removal, is directly related to the

Young's modulus and the stress value in the form of Hooke's law, see equations (3.1) or (3.24). See figure 6.28 for the x- and y-components of springback displacements at the bottom edge of the plate. The results are from the simulations without friction. The springback behaved similarly with the friction model.

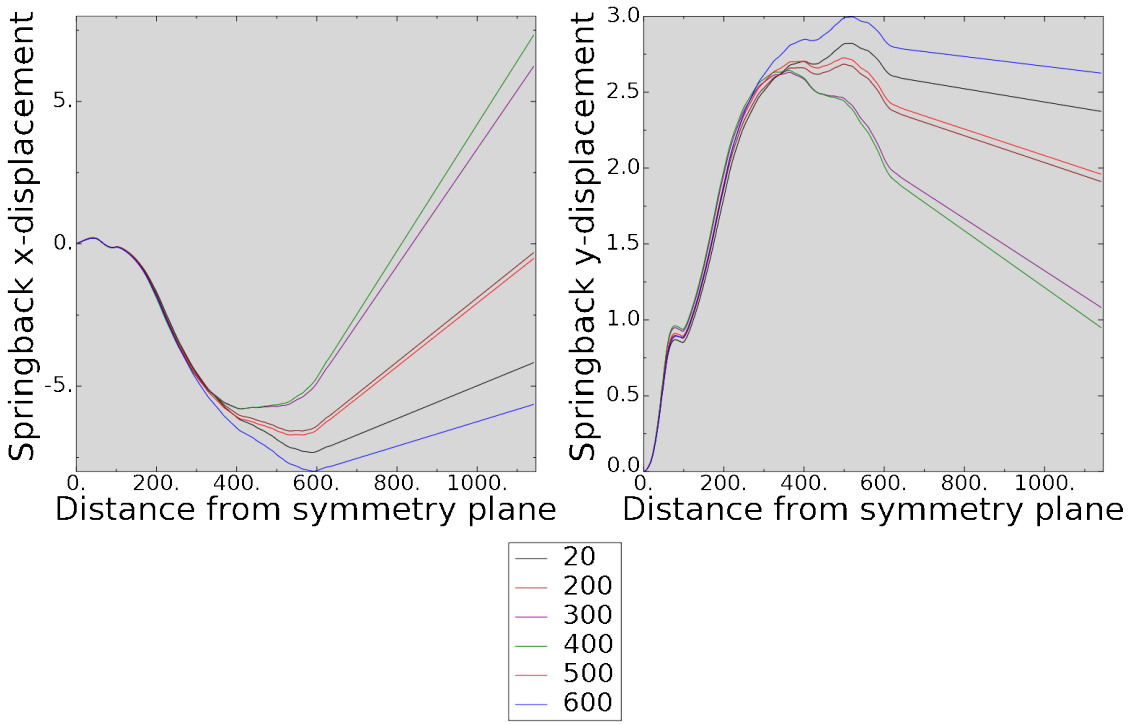


Figure 6.28: Springback displacements along the path of the bottom of the plate at different temperatures

The elastic relaxation is dependent on the ratio of Young's modulus to the yield strength. It increases with the $\frac{\sigma_y}{E}$ ratio which is directly related to the amount of the elastic strain. The springback is directly related to this ratio near the symmetry plane. However, the relaxations at the different bend curves get magnified into other regions and result in different configurations of the blank. Therefore, it has to be kept in mind that the geometry of the blank affects the springback also. This explains why the springback of the material properties at $T = 600°C$ yielded the largest magnitude of springback displacement although the $\frac{\sigma_y}{E}$ ratio is larger with the material properties at $T = 20°C$.

6.4.2 Plate anisotropy

The bending of sheet steel is usually performed in the way that the bending axis is perpendicular to the rolling direction. This is because the formability (fracture strain ε^f, see figure 3.1) in the rolling direction is assumed to be higher than in the direction perpendicular to the rolling direction. The yield strength ratios at different directions with respect to the rolling direction were obtained from source [33] as

the test results for microalloyed S355N steel. The direction 11 coincides with the rolling direction, 22 with the transverse direction and 33 with the through thickness direction. The ratios are compiled in table 6.6 where σ^0 is the yield strength of the uniaxial material data that is input to Abaqus as the uniaxial stress/strain curve, which is taken here as the value of the yield strength in direction 11.

Table 6.6: Yield ratios for anisotropy

direction	σ^y_{11}/σ^0	σ^y_{22}/σ^0	σ^y_{33}/σ^0	τ^y_{12}/τ^0	τ^y_{23}/τ^0	τ^y_{31}/τ^0
ratio	1	1.10	1.08	1	1	1

As no test results were available for the shear yield strength ratios, the ratios are input here with a value of 1. This is the quadratic Hill yield criteria with an associative flow rule and an isotropic hardening law. The elastic part of the behavior is assumed to be isotropic.

An explicit procedure with a 3D model with no friction and the penalty contact constraint enforcement resulted in a pressing force of $120000kN$. The pressing force value was practically the same as with the isotropic model. One would have thought that the slightly higher yield stresses in the transverse as well as the thickness direction would have increased the need for the pressing force. However, most of the deformation in the simulation happens in the material 1 direction (x-axis direction in initial configuration) which is perpendicular to the axis of the majority of the bending. The slightly lower result for the pressing force in the anisotropic model $120000kN$ when compared to that of the isotropic model $121000kN$ is assumed to be because of the differences in the behavior of the penalty stiffnesses of the contact constraint enforcement.

The springback values did not differ significantly from those obtained with the isotropic material model although the x-direction springback is not as evenly distributed on the xy-plane cuts of the blank as with the other models, see appendix A.8 for the x- and y-direction springbacks on the bottom surface of the blank. z-direction springback was again negligible with a maximum value of $-0.37mm$.

7. RESULT ANALYSIS

7.1 Implicit static vs. dynamic analysis for forming

The simulations for the forming process in Abaqus/Standard were performed with the implicit dynamic procedure with quasi-static application option which uses the backward Euler time integrator. The reason for this was that, with the 2D model, the implicit dynamic analysis seemed to converge better in the global Newton-Raphson iterations than the truly static analysis.

7.1.1 Forming step

The static frictionless plane strain analysis with 2/5-mesh of 3208 CPE8R elements broke down at $t = 0.3945$ where the time increment size had to be cut down 5 times with no convergence achieved. The blank configuration at this time can be seen in figure 7.1. It can be seen that some complexity in the contact conditions arise at

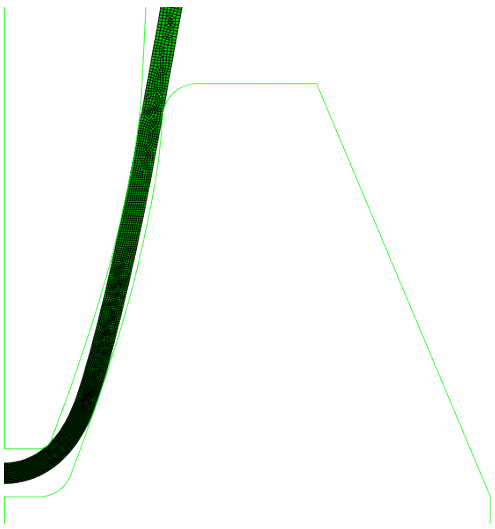


Figure 7.1: Blank configuration at the breakdown of the static analysis, $t = 0.3945$

this time of the simulation.

The friction model, however, completed with both analysis types. A comparison between the analysis types for the plane strain model with a friction coefficient of 0.1 is presented in the table 7.1. The frictionless 3D model results with penalty contact constraint method is also compiled into this table.

Table 7.1: Implicit static and dynamic analysis comparison

analysis type	user time / s	increments	iterations
static 2D friction 0.1	6679	1091	6093
dynamic 2D friction 0.1	4946	1134	5880
static 3D frictionless	$1.763 * 10^6$	873	5232
dynamic 3D frictionless	$1.891 * 10^6$	902	5155

In the 2D model, the number of increments taken with the dynamic analysis is a bit larger than that of the static analysis, but the number of iterations is about 200 lower. The reason for the more rapid convergence in the dynamic plane strain analysis is assumed to be because of the numerical damping that the inertia terms provide in the global iterations when hard contact is being modelled.

However, the static procedure took less user time to complete with the 3D model. This is assumed to be because of the penalty contact constraint enforcement method used with the 3D model. A plane strain frictionless model with a 2/5-mesh and penalty contact constraint enforcement backs up this assumption: both analysis types complete, and the user time with the static procedure is $433s$ and with the dynamic procedure it is $468s$.

The reason for the more rapid convergence and stability of the dynamic method with the augmented Lagrange method can be demonstrated by comparing the matrices which have to be inverted in the global Newton-Raphson iteration of the implicit procedure:

$$[\mathbf{K^{dyn}}] = \frac{1}{\Delta t^2}[\mathbf{M}] + [\mathbf{K_t}] \tag{7.1}$$

$$[\mathbf{K^{sta}}] = [\mathbf{K_t}] \tag{7.2}$$

where $[\mathbf{K^{dyn}}]$ is the same matrix as the one introduced in equation (2.15) in the theory chapter, excluding the damping term as no external damping was included in the simulations. These are the matrices that have to be summed into the linearizations of the contact contribution matrices introduced in section 4.4.2. Some modes that are singular for the matrix $[\mathbf{K^{sta}}]$ used in the static procedure are not singular for the matrix used in the dynamic procedure $[\mathbf{K^{dyn}}]$ which includes the inertia term. The contribution of inertia increases in $[\mathbf{K^{dyn}}]$ when the time increment size reduces because of the coefficient $\frac{1}{\Delta t^2}$ in the mass matrix term. This further stabilizes the iteration when a cutback on the increment occurs. The relaxed tolerances of the penalty contact constraint enforcement lower the need of numerical stabilization, and thus, converge better with the static procedure. The inclusion of the inertia

term also stabilizes rigid body modes that can occur in the contacting bodies.

The penalty constraint enforcement method is more rapid in convergence than the augmented Lagrange method in both procedures but fails to accurately capture the shape at the bend radius as seen in figure 7.2. The default penalty stiffness could

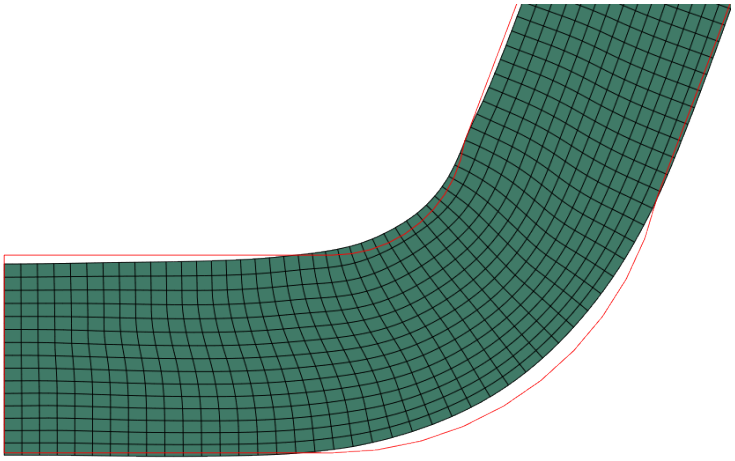


Figure 7.2: Blank configuration after forming at the steepest bend radius with penalty contact constraint enforcement

be manually modified to a larger value to capture the shape at the bend radius more accurately, although this would require some additional time to find out a suitable combination of computational efficiency and solution accuracy.

7.1.2 Springback step

The springback step in the dynamic implicit procedure broke down with no increments taken after five cutbacks on the automatic incrementation. The springback step did complete with the static analysis but yielded unaccurate results, see figure 7.3 for the configuration of the blank after springback. This springback is not physically reasonable nor consistent with other simulation results.

However, the springback step did complete with both analysis types, with the same initial increment size, when a new analysis was performed where the initial state of the blank was input as a predefined field imported from the end of the forming step of the implicit analyses. This is exactly the same technique as the one used for springback analysis when the forming is performed with the explicit procedure. This import analysis yielded physically reasonable results for the springback displacements. The springback obtained with this procedure after the static implicit forming procedure is similar to that after the dynamic implicit forming procedure with a maximum difference of only $0.9mm$ in the x-direction. See appendix A.9 for the springback displacements on the bottom surface of the plate obtained when the forming was performed with the implicit dynamic procedure with penalty contact

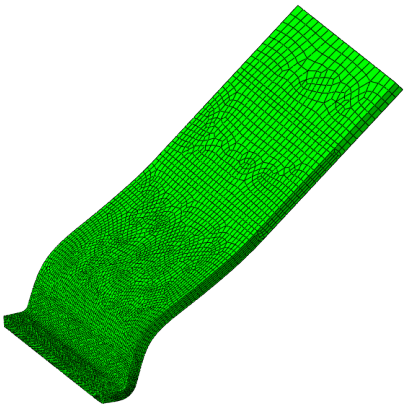


Figure 7.3: The erroneous configuration after springback with the completely static solution procedure

constraint enforcement. The z-direction springback was negligible with a maximum value of $-0.35mm$.

When the springback step is included in the same analysis as the forming step, the springback convergence issues are assumed to be because of the ramp amplitude for the punch displacement over the forming step, which might cause problems in subsequent steps when the boundary conditions are removed [7, sect. 6.1.1]. Thus, the smooth step amplitude curve for forming is recommended for the implicit analysis also if the springback is to be included as a step after the forming step although coarse meshes with the plane strain model seemed to converge even when the ramp step was used.

7.2 A note on the hardening law

The isotropic hardening law was chosen for the simulations. This meant that the yield surface at each material calculation point would uniformly expand in every direction in the stress space during yielding, see section 3.5.

However, the direction of the plastic straining changes in some parts of the plate in the end when the plate gets in contact with the die bottom. This is demonstrated in figure 7.4 where the contours of equivalent plastic strain (PEEQ) and plastic strain magnitude (PEMAG in Abaqus convention) are plotted in the end of the forming step. PEMAG is calculated from the current state of plastic strain and PEEQ is calculated by following the path of the plastic straining so that it increases monotonously even if the direction of the plastic straining changes. Both are scalar measures of the plastic strain, see section 3.4. The values are the same up to approximately $t = 0.42s$.

Because of the changes in the direction of plastic straining, a linear kinematic hardening law was tested also. The simulation with CPE8R 1/5-mesh frictionless model resulted in a need for the pressing force of approximately 14 % lower than

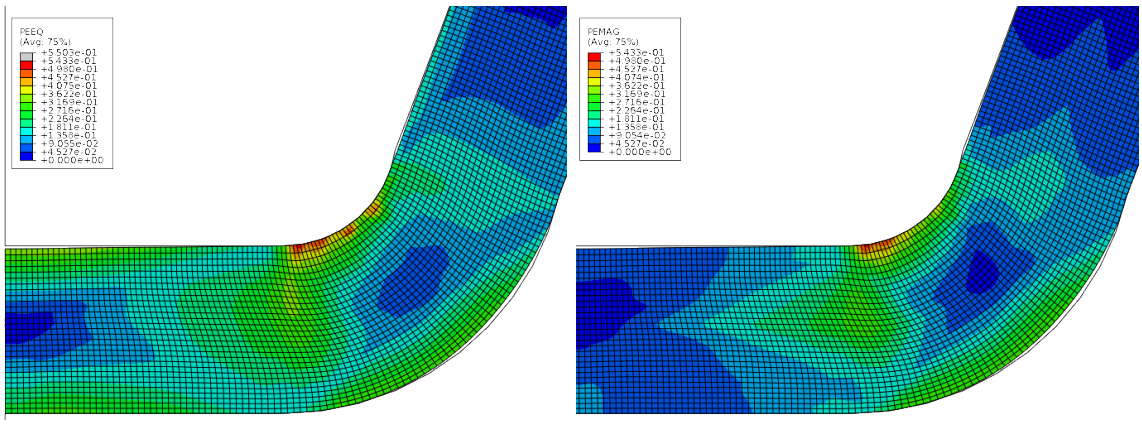


Figure 7.4: The change in the direction of plastic straining illustrated in plane strain model solution with PEEQ and PEMAG contours in the end of the forming step

that of the simulation with isotropic hardening. This means that the pressing force was lowered from $100500kN$ to $86500kN$ by changing the isotropic hardening law to linear kinematic hardening. The different hardening law also resulted in a whole different kind of springback as seen in figure 7.5. The configurations before springback did not differ.

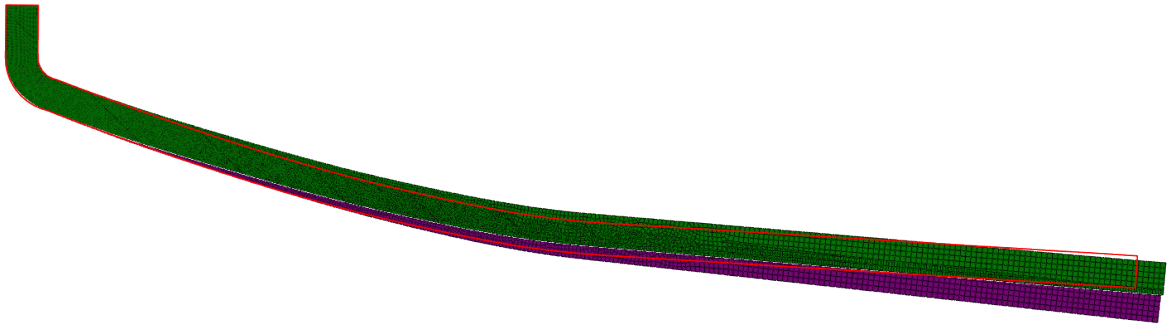


Figure 7.5: The blank configurations after springback with isotropic hardening (green) and kinematic hardening (violet) and the desired shape (red profile)

The kinematic hardening law was included here merely as a demonstration for a different hardening model. The used linear kinematic hardening law in Abaqus is physically reasonable only for small strains of approximately less than 5 % [7, sect. 20.2.2]. Also, the approximations made in the solid continuum element formulation used in Abaqus is not suitable for large strains with the kinematic hardening model [23, sect. 3.2.2]. According to [12, p. 324], the hardening behavior of most materials seems to be a combination of kinematic and isotropic hardening. Sometimes the yield surface seems to change shape as well, see [12, p. 510].

The most affected regions by the change in the direction of the plastic strain is on the outer edge of the plate at the yz-plane of symmetry and on the bottom surface. This fact should be taken into account when considering the material strength properties after the forming is performed. See figure 7.6 for the contour

plots of PEEQ and PEMAG on the 3D model with the implicit procedure.

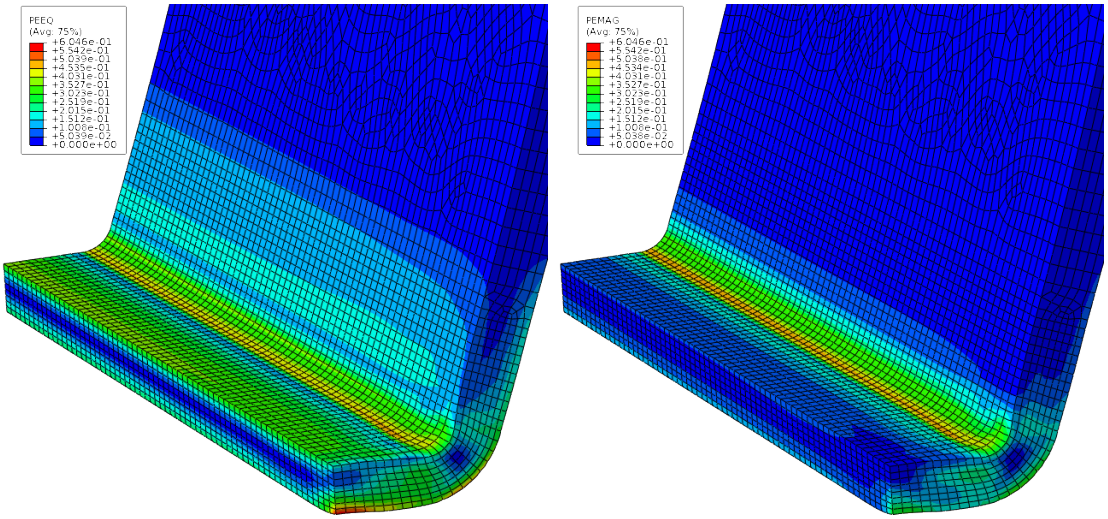


Figure 7.6: PEEQ (left) and PEMAG (right) contour plots on the implicit 3D model

A way of getting rid of the undesired plastic deformation near the symmetry plane would be to clamp the blank to stay in contact with the punch bottom surface throughout the forming process. However, this can be fairly challenging to implement in practice.

7.3 2D/3D model comparison

The results of the 3D model showed that the deformation was not the same on all xy-plane cuts. However, the 2D plane strain model was computationally superior when compared to the 3D model. Thus, the plane strain model results need to be compared to the 3D model more precisely.

7.3.1 Plane strain assumption

The z-direction strain contour plot of the friction 3D model explicit procedure with penalty contact constraint enforcement can be seen in figure 7.7 before and after the plate establishes contact with the die bottom. The corresponding contour plots of other models show the same behavior.
It is clear that the plane strain assumption does not hold well at the free edge of the plate. The strain components at the integration point of an element adjacent to both of the symmetry plane cuts and on the bottom surface of the plate are presented in the left side of the figure 7.8. Here the logarithmic strain components are presented in Abaqus convention (LE11=ε_x, LE22=ε_y, LE33=ε_z, LE12=γ_{xy}, LE23=γ_{yz}, LE13=γ_{xz}). The strain components at the integration point of an element adjacent to the xy-plane of symmetry, in the bend curve, and on the bottom surface of the

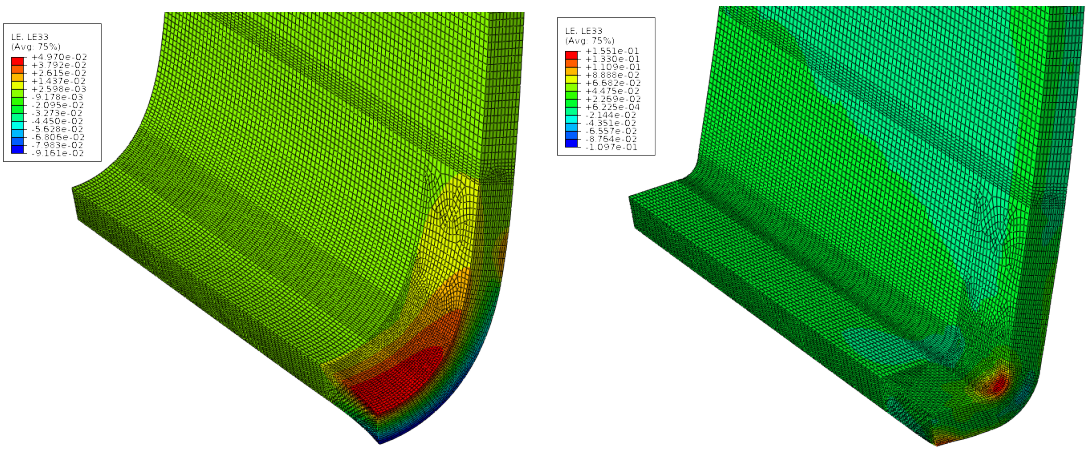


Figure 7.7: contour plots of z-direction logarithmic strain at $t = 0.4s$ (left) and $t = 0.5s$ (right)

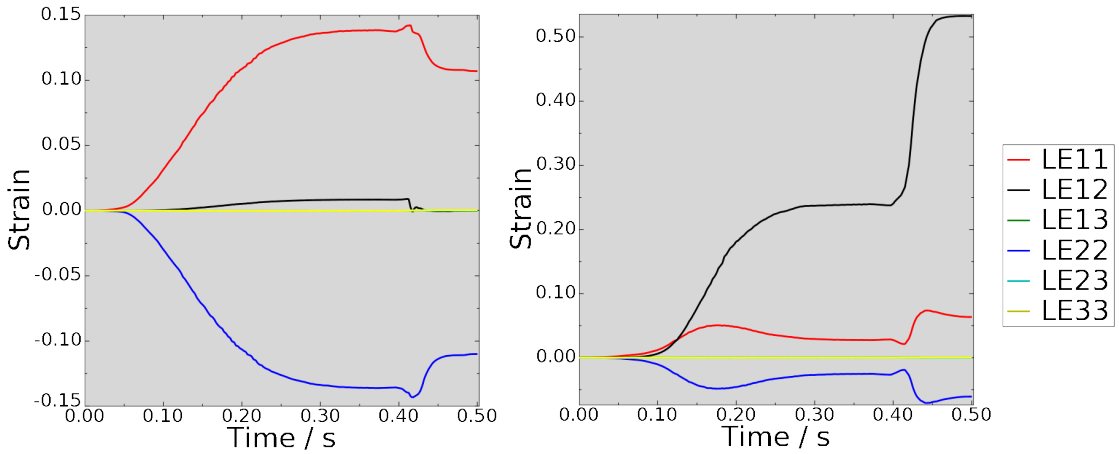


Figure 7.8: Strain components near the bottom surface of the plate adjacent to both symmetry plane cuts (left) and adjacent to xy-plane symmetry cut at the bending curve (right)

plate, is presented on the right side of the figure 7.8. It can be seen that the plane strain assumption holds well for these integration points. The only significant strain components are ε_x, ε_y and γ_{xy}. The results are quite similar for the top surface of the plate with the strain components only changing directions to the opposite. The case is the same for the majority of the blank xy-cuts, see appendix A.10 for contour plots of the other strain components at $t = 0.4s$ and $t = 0.5s$.

7.3.2 Springback and pressing force difference

The 3D model simulation results were compared to the plane strain model by performing plane strain model simulations with the same analysis parameters as the 3D model simulations. The mesh of the 2D model was taken as the xy-plane cut of the 3D model mesh.

Punch geometry difference between the models

The punch geometry modification, see figure 6.7, led to an inconsistency between the punch geometries of the 3D and 2D model by a user mistake. The x-direction width of the punch geometry used in the 3D model was larger than that of the plane strain model. The punch geometry for the 3D model is obtained by translating the 2D model punch side wall nurb, seen as the red line in figure 7.9, $0.00411mm$ to the positive x-direction. The rest of the punch side geometry is not of interest here as it does not establish contact with the blank. The difference between the geometries

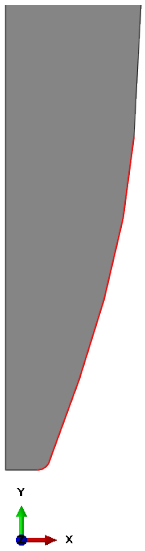


Figure 7.9: The punch side wall geometry

was minimal but it led to notable differences in the results.

The springback for both of these punch geometries simulated as frictionless plane strain models can be seen in figure 7.10. The word "exact" refers here to the simulation with the punch geometry used in the 3D simulations, and "old" refers to the punch geometry used in all of the previous plane strain simulations in this thesis, excluding the initial geometry.

One can see that the nature of the springback differs between these punch geometries quite much. The pressing forces were $22000kN$ lower in the friction (coefficient 0.1) model and $10000kN$ lower in the frictionless model with the old geometry. This was because the blank was not compressed as much as with the exact geometry.

The compression of the blank between the tools is an important aspect in the simulation when the tools are modelled as rigid and the punch movement is set as a displacement boundary condition. This can lead to unrealistic results when the blank gets compressed between the tools. In real life, the tools would most probably deform and give the blank more space between them. However, the "exact" punch geometry for the plane strain model will be used next in the comparison between

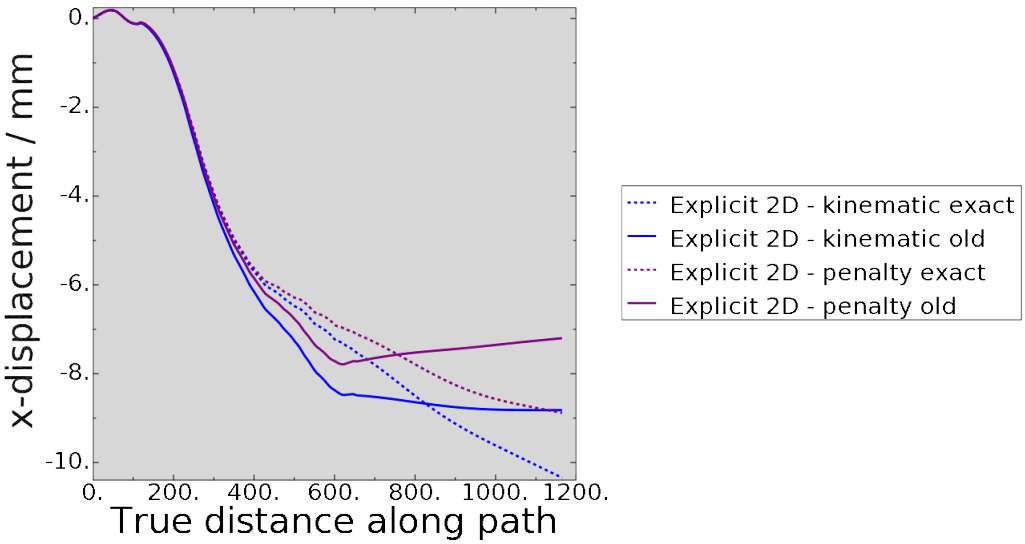


Figure 7.10: 2D model springback comparison between different punch geometries

the 3D and 2D models because it is consistent with the 3D model punch geometry.

Springback

As discussed earlier, the penalty contact constraint enforcement with the default penalty stiffness seemed to capture the shape of the bend curve inaccurately in the plane strain model, see figure 7.2. This made the springback lead to erroneous results. In fact, the maximum springback displacements with the penalty contact constraint enforcement in the 2D model was in different direction when compared to the other simulation results. For this reason, the implicit plane strain model with penalty contact constraint enforcement is not included in these comparisons. The penetration problem with penalty contact constraint enforcement did not seem to be as severe with the implicit 3D model.

See figure 7.11 for the springback displacements in x-direction for the 3D implicit models, the frictionless 3D explicit models and the explicit frictionless 2D models. The dashed black curve is the springback of the implicit 2D model with augmented Lagrange contact constraint enforcement for comparison purposes. The 3D model springbacks are taken from the blank bottom surface at a line coinciding with the xy-plane of symmetry. The displacements are plotted as a function of the true distance path of the nodes along the bottom of the plate starting from the node at the yz-plane of symmetry.

The springback curves after the explicit solution procedure at distance $600mm - 1200mm$ along the path are not straight lines, although the springback curves after implicit solution procedure are. This is assumed to be because of undesired inertia contribution of the explicit solution to the blank. The mass scaling factor might be too large after all. However, a compromise between the computational efficiency

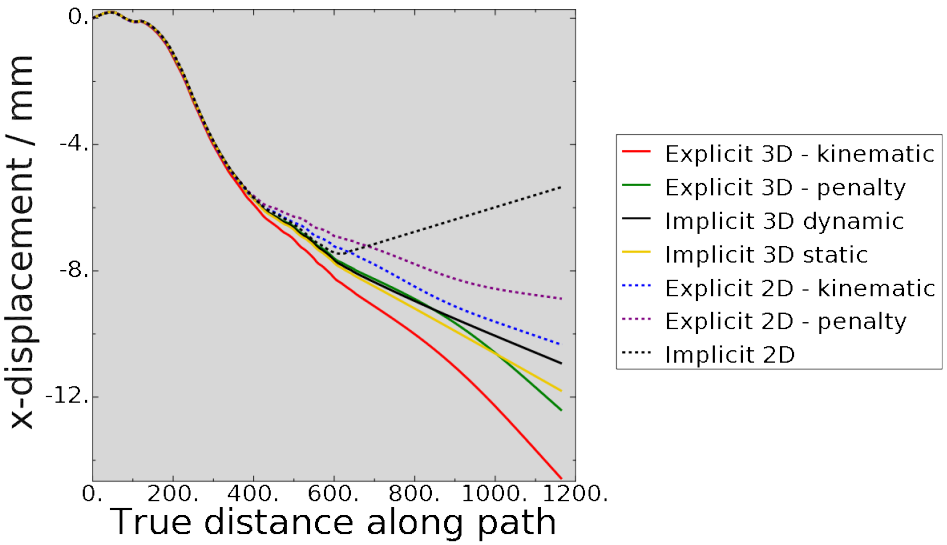


Figure 7.11: 2D/3D model springback comparison between frictionless models

and accuracy had to be made. See section 6.2.3 for the effect of mass increase in the springback.

The friction model springbacks are smaller than in the frictionless model because of the contributions of surface friction stresses resisting the blank movement. The area for the surface stresses is larger between the die and the blank than between the punch and the blank, and thus, the friction force contribution on the bottom of the blank is higher. These friction forces made the blank configurations differ about $1mm$ at most in the end of the forming step (but before springback step) between the frictionless and the friction models. The final configurations after springback are less than $1mm$ from each other for both constraint enforcement cases. Thus, the springback is even less dependent on the friction than the springback contour plots in the appendices imply. For this reason, the friction model springback is not included in figure 7.11.

Needed pressing force

The results showed that the need for the pressing force was $10000kN$ higher in the 3D model with the frictionless model and $42000kN$ higher with a friction coefficient of 0.1. The reason for this is the extra bending deformation that the 3D model was able to capture at the free edge of the plate, see figure 6.24. This extra deformation had to be bent straight also in the end of the forming step. Also, the isotropic hardening law had already work hardened the bottom of the outer edge before the plate got in contact with the die bottom so that the plastic deformation in the other direction needed high pressing forces. The pressing forces started to significantly differ at approximately $t = 0.48s$, see figure 7.12.

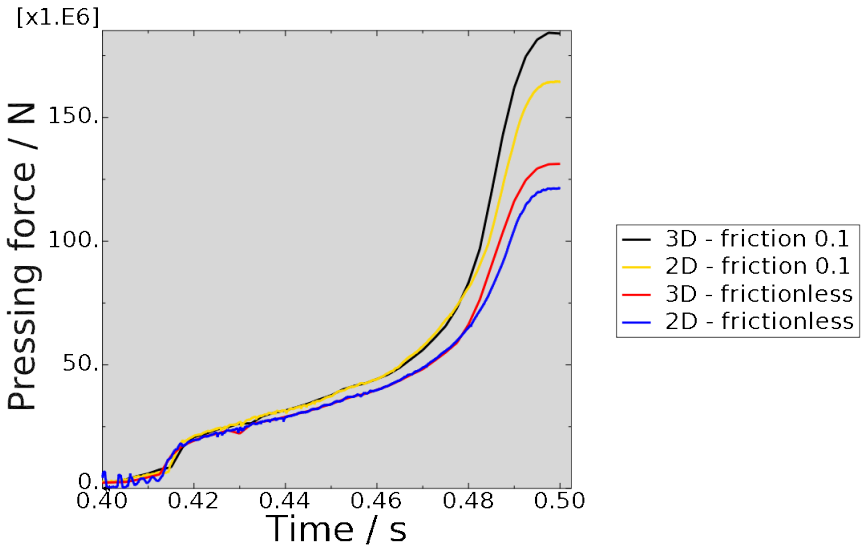


Figure 7.12: 2D/3D model pressing force comparison

The smoother response of the 3D pressing force is because of the lower frequency of history output data points for the punch pressing force.

7.4 Solution method efficiency comparison

The 2D model had a smaller number of equations to be solved and more simple 2D contact conditions so that the implicit procedure with the second-order elements seemed to be more efficient for this procedure. Also, the springback step could be completed within the same analysis with no additional import analysis. The small element side length, needed with the first-order reduced integration elements through the plate thickness to make the ALLAE/ALLIE ratio small enough, made the time increment size for the explicit procedure inefficient in the 2D model.

A 3D model comparison between the explicit procedure with kinematic contact constraint enforcement and the implicit static procedure with penalty contact constraint enforcement is presented in table 7.2. Both simulations did not include friction. The times are presented here only for the forming step as a successful springback analysis takes only a relatively small amount of time: about 15 minutes after the explicit procedure and about 10 minutes after the implicit procedure.

The element and node numbers include the rigid tool elements which do not contribute to the stiffness or mass matrices. See section 6.3.2 for the number of elements and nodes in the blank with both procedures.

In finite sliding contact involving a rigid body, Abaqus/Standard automatically creates contact elements by using the information provided in the contact pair definitions. These contact elements are excluded in table 7.2 as they only measure the normal and tangential gaps associated with the contact definitions at their integra-

Table 7.2: Implicit / Explicit procedure comparison for 3D model

analysis type	implicit static	explicit dynamic / kinematic
Number of elements	37699	222496
Number of nodes	152037	243841
total number of variables	679512	731532
CPU time	$489h$ / user time	$188h$ / CPU time

tion points [23, sect. 5.1.3] and are used for the contact calculations together with the Lagrange multipliers or penalty stiffnesses. The number of elements and nodes of these internal elements generated for contact are 74464 and 148928, respectively.

The CPU time for the explicit dynamic procedure is obtained by modification from the dat-file created by Abaqus and is approximately the wallclock time it took to complete the job multiplied by the number of CPUs (8) the job was ran with. This is not the complete truth of the computation time as the threads multiprocessing mode was used which decomposes the whole structure into 8 (number of CPUs more generally) separate domains which still share some nodes with each other so that the CPUs must communicate with each other during the simulation. The multiplication factor of 8 is thus questionable.

Even with these difficulties in comparing the solution procedures in mind, one can still state that the explicit procedure is more efficient in a large 3D problem, which involves some complexity on the contact conditions, than the implicit method. This advantage is because of the highly efficient solving of the global equations at each increment without the need to assemble the global stiffness matrix for iteration procedures. Also, the severe contact nonlinearity is more easily solvable with the small time increments characteristic for an explicit solution. It still has to be kept in mind that the Abaqus/Standard code provides a larger variety of usable elements including the second-order solid continuum elements and could provide better results with fewer elements than the Abaqus/Explicit code.

The CPU times, measured with similar technique as in 7.2, for the other 3D model simulations are as follows: frictionless implicit dynamic procedure and penalty contact constraint enforcement - $525h$, frictionless explicit procedure with penalty contact constraint enforcement - $182h$, explicit procedure with a friction coefficient of 0.1 and kinematic contact constraint enforcement - $228h$, explicit procedure with a friction coefficient of 0.1 and penalty contact constraint enforcement - $202h$.

The explicit procedure could be made even more efficient by scaling only the mass of the regions with the smallest elements in the mesh. Initial studies suggest that the inertia contribution of the coarse part of the mesh is significant to the pressing force and the springback.

7.5 Different material model

At an early stage of the simulation studies, the material model was input to Abaqus as

$$\sigma^y = 426MPa \quad \text{at} \quad \varepsilon^p_{log} = 0 \quad \text{and} \quad \sigma^u = 565MPa \quad \text{at} \quad \varepsilon^p_{log} = 0.22314$$

where the stresses were input as the engineering stresses from the test results and the plastic strain at σ^u was taken as the true strain measure of $\varepsilon^m = 0.25$ which is the engineering strain at break measured from the unstressed length of the tensile specimen after the tensile test has been performed. With this input, the plastic behaviour of the material was ideally plastic after the plastic strain had exceeded 0.22314. This together with a lower plastic modulus led to the fact that the material model used in the simulations of chapter 6 was a more stiff (and most likely more accurate as well) material model.

An interesting detail in the simulations performed with this material model was the large amount of artificial strain energy created, which is discussed next.

7.5.1 Hourglassing problems

When the simulation was performed with the explicit dynamic procedure with this material and the enhanced hourglass control option, hourglassing seemed to be a true problem. Different fineness was applied to the blank mesh to reduce the ALLAE/ALLIE ratio. The artificial strain energy magnitude in the element for the whole element, ELASE, was requested as a field output variable to see the regions of the blank where the most of the artificial strain energy was created. Obviously most of the energy was created in the area near the symmetry plane which was subjected to most of the bending deformation as seen in figure 7.13.

The ALLAE/ALLIE ratio of this model is plotted with different number of elements through the thickness of the plate in figure 7.14.

The lowest ratio was achieved when a new partition was made halving the area of most interest. 60 elements were assigned through the thickness in the new partition reaching 75 mm from the symmetry plane. This resulted in a ratio less than 1% throughout the majority of the forming process. However, the area subjected to most of bending changed when the plate got in contact with the die bottom and resulted in a peak of artificial strain energy in the area where only 30 elements through thickness was used. This made the ratio peak above 1 % for a short period at approximately $t = 0.4s$. The large number of elements and equations to be solved coupled with the small element side length and the double precision used in analysis resulted in computational inefficiency.

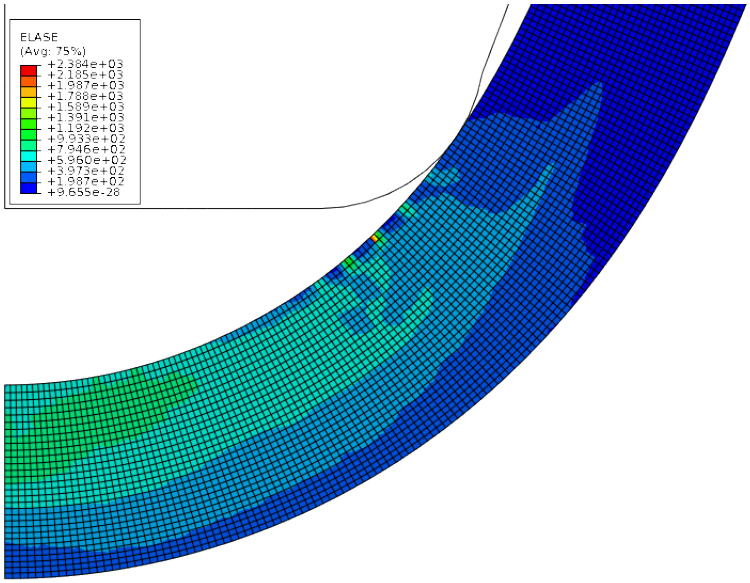


Figure 7.13: ELASE at the bottom part of the plate at t=0.325s with 30 elements through thickness in the area of most bending

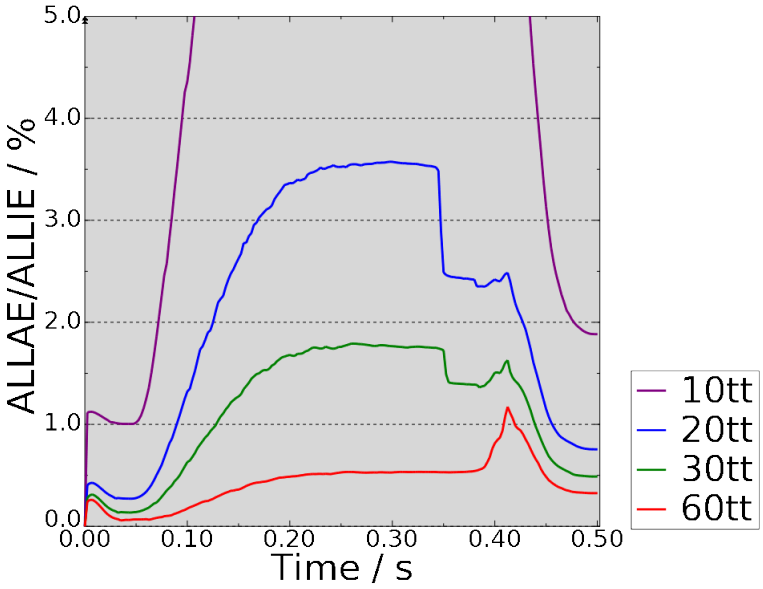


Figure 7.14: ALLAE/ALLIE-ratio with different number of elements through the plate thickness

When the artificial strain energies created in the simulations of the 3/5 mesh models were compared, it seemed that the amount of ALLAE created with the more flexible model was significantly higher than that of the model with more stiff material properties, see figure 7.15 for the comparison.

Similar tests comparing the hourglass control methods in Abaqus/Explicit with this material model as those performed in chapter 6 are presented in appendix A.11. The coarse mesh comparison is performed with the same coarse mesh as in chapter

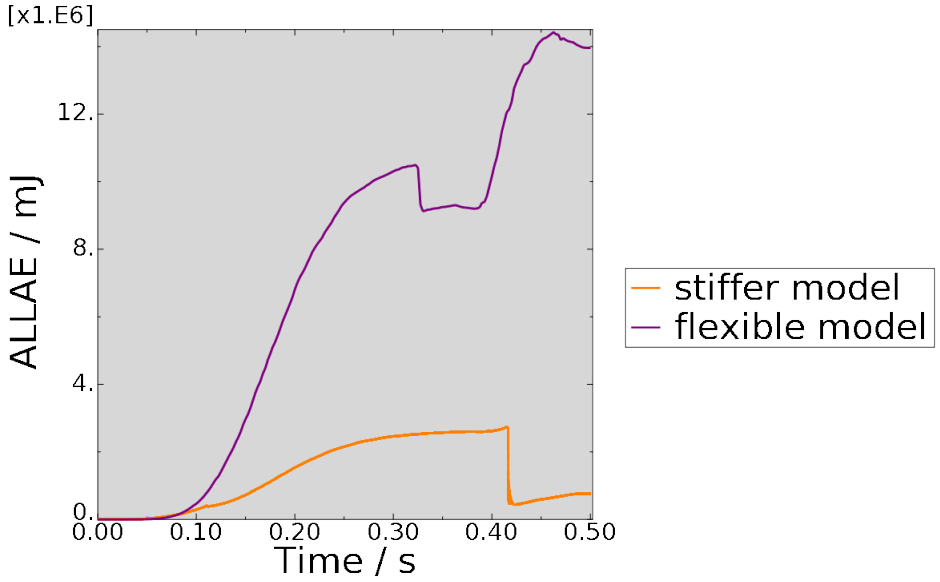


Figure 7.15: ALLAE with different material models, enhanced hourglass control option and 3/5 mesh

6, see figure 6.19. The enhanced option differs even more than with the results obtained in chapter 6.

It seems that the enhanced hourglass control creates more hourglass-controlling forces with flexible plastic behaviour. This should be taken into account when choosing the hourglass control option and considering the energy balance checks with a flexible plastic material model.

The enhanced hourglass control option is recommended for import analyses between Abaqus/Explicit and Abaqus/Standard for ensuring the correct calculations of the hourglass controlling forces. This is because the option is available for hourglass control in both codes. It seems, however, that the import analysis can be done without the simulation breaking down by using the other hourglass control options as well, although this might result in inaccuracy considering the calculations of the hourglass controlling forces between the analyses.

Further studying the theoretical background of the hourglass control methods could explain the differences between these results, see section 6.2.5 for a brief introduction on the methods. However, this study would be beyond the scope of this thesis.

8. CONCLUSIONS

The literature commonly available for stamping processes is focused on sheet metal forming with thinner blanks and more simple contact conditions. For sheet metal or plate bending, the common applications are more simple, e.g. bending along one axis only. The two-sided contact conditions and multiple bend curvature make the case simulated in this thesis more complicated.

Based on the simulation results, major improvements may have to be done to the forming process design for the process to become achievable. The needed pressing force and the stresses in tools should be lowered significantly. These improvements could be some radical changes to the tooling geometry or a new approach to the problem such as hot forming. It is left for future research to find the optimal design solution to this specific forming process as the main objective of this thesis was not to find this solution but rather to study the simulation of this case to compare the different solution and modelling considerations.

The most important conclusions based on the results of the simulations are as follows:

1. The implicit dynamic procedure is more efficient than the explicit procedure with the plane strain model of this thesis

2. The explicit dynamic procedure is more efficient than the implicit dynamic procedure in large 3D problems with complexity on the contact conditions because there is no need to form and invert the global tangent stiffness matrix for the global iterations and the contact calculations are simplified

3. The implicit dynamic procedure with the backward Euler operator offers an advantage over the implicit static procedure by offering improved convergence in the global iterations when hard contact is being modelled

4. It is important for forming simulations with rigid tools and two-sided contact conditions that the blank does not get compressed between the tools when the punch movement is set as a displacement boundary condition

5. The enhanced hourglass control option in Abaqus can cause problems with a flexible plasticity model

A guideline for future work would be to try to optimize the process with the plane strain model because of its superior computational efficiency when compared to the 3D model. It has to be kept in mind that the model may give lower need for the pressing force than the more thorough 3D model. However, this does not matter at this stage of the design process as the pressing force and the stresses in the tools should be lowered significantly.

It would be advantageous if the plastic deformation would be subjected only to the regions of the blank that are essential for the final product shape. This is not the case with the current design, see section 7.2. Also, the contact forces between the punch and the blank could be distributed to a larger area for easier control on the process and lower contact pressure stresses.

In the simulation part, it would be interesting to see the effect of increasing the number of rigid elements at the smallest punch radius to capture the actual shape of the tool even more accurately. The number of elements used in the simulations was based on the guideline that the slave surface should have a finer mesh. However, only the penetration of master nodes into the slave surface introduces a limit to this case.

The inclusion of fracture criteria for the material in the simulation could also be considered. In sheet metal or plate bending, the fracture of the blank usually starts at the outer minimum bend radius and the formability is usually expressed with a minimum bend radius value [34, p. 660]. However, the change in the plastic strain directions in this simulation make the prediction of formability complicated.

REFERENCES

[1] Kalpakjian, S. & Schmid, S.R. Manufacturing Engineering and Technology - Internal edition 4th Ed., Prentice Hall, Inc. 2001.

[2] Eggertsen, P-A. Prediction of Springback in Sheet Metal Forming Chalmers University of Technology, Doctoral thesis, 2011.

[3] Cook, R.D, Malkus, D.S & Plesha, M. E. Concepts and Applications of Finite Element Analysis, 4th Ed., John Wiley and Sons, 1989.

[4] Bathe, K.J. Finite Element Procedures in Engineering Analysis, Prentice-Hall, 1982.

[5] Belytschko, T., Liu, K.W. & Moran, B. Nonlinear Finite Elements for Continua and Structures John Wiley & Sons, 2000.

[6] Lenard, J.G. Metal Forming Science and Practive - A State-of-the-Art Volume in Honour of Professor J.A. Schey's 80th Birthday, Elsevier Science, 2002. pp. 135-181.

[7] Abaqus Analysis User's Manual Version 6.10 Simulia Inc

[8] Hilbert, H.M., Hughes, T.J.R. & Taylor R.L. Improved Numerical Dissipation for Time Integration Algorithms in Structural Dynamics, Earthquake Engineering and Structural Dynamics, Vol. 5, pp. 283-292.

[9] Géradin, M. & Cardona, A. Flexible Multibody Dynamics, A Finite Elelement Approach, John Wiley and Sons, 2001.

[10] Wriggers, P. Nonlinear Finite Element Methods. Springer-Verlag. 2008.

[11] Flanagan, D.P. & Belytschko, T. A Uniform Strain Hexahedron and Quadrilateral with Orthogonal Hourglass Control, International Journal for Numerical Methods in Engineering vol. 17, 1981. pp. 679-706.

[12] Jirásek, M. & Bažant, D.J. Inelastic Analysis of Structures. John Wiley & Sons, Ltd 2002. 734 p.

[13] Truesdell, C. & Noll, W. The Non-linear Field Theories of Mechanics edited by S.S. Antman, 3rd edition, Springer-Verlag, 1965.

[14] Wriggers, P. Computational Contact Mechanics. 2nd edition. Springer-Verlag Berlin Heidelberg 2006. 518 p.

[15] ASM Handbook Vol. 8 - Mechanical testing and evaluation ASM International, 2000. pp. 124-151

[16] Chen, W.F. & Han, D.J. Plasticity for Sructural Engineers, J. Ross Publishing, 2007.

[17] Arola, A-M. FEM-modelling of bendability of ultra-high-strength steel - Ultralujan teräksen särmättävyyden FEM-mallinnus (in finnish), Master's thesis - University of Oulu, Department of Mechanical Engineering, 2010.

[18] Bridgman, P.W. Studies in Large Plastic Flow and Fracture, McGraw-Hill, New York, 1952.

[19] Aronofsky, J. Evaluation of Stress Distribution in the Symmetrical Neck of Flat Tensile Bars, Journal of Applied Mechanics vol. 18, 1951. pp. 75-84.

[20] Ling, Y. Uniaxial True Stress-Strain after Necking. AMP Journal of Technology Vol. 5. 1996. pp. 37-48.

[21] Hill, R. A theory of the yielding and plastic flow of anisotropic metals. Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences vol. 193, 1948. pp. 281-297.

[22] Barlow, J. Optimal Stress Locations in Finite Element Models, International Journal for Numerical Methods in Engineering vol. 10, 1976. pp. 243-251.

[23] Abaqus Theory User's Manual. Version 6.10. Dassault Systèmes Simulia Inc.

[24] Simo, J.C. & Hughes, T.J.R Computational Inelasticity - Interdisciplinary Applied Mathematics Vol. 7, Springer, 1997.

[25] Mühlhaus, H.B. & Dyskin, A.V. Bifurcation and localisation theory in geomechanics, Taylor & Francis, 2001.

[26] Zienkiewicz, O.C. & Taylor, R.L. The Finite Element Method Vol. 2, 4th edition, McGraw-Hill Book Company Europe, 1991.

[27] Heinstein, M.W., Mello, F.J., Attaway, S.W. & Laursen T.A., Contact-impact modeling in explicit transient dynamics, Computer methods in applied mechanics and engineering, vol 187, pp. 621-640 , 2000.

[28] Introduction to Abaqus. Training Course Lecture Notes. Dassault Systèmes Simulia Inc.

[29] Engelmann, B.E. & Whirley, R.G. A New Explicit Shell Element Formulation for Impact Analysis (in Computational Aspects of Contact, Impact and Penetration), Ed. R.F. Kulak and L.E. Schwer, Elmepress International, 1990.

[30] Belytschko, T. & Bindeman, L.P. Assumed Strain Stabilization of the Eight Node Hexahedral Element, Computer Methods in Applied Mechanics and Engineering, vol. 105, pp.225-260, 1993.

[31] Puso, M.A. A Highly Efficient Enhanced Assumed Strain Physically Stabilized Hexahedral Element, International Journal for Numerical Methods in Engineering, vol. 49, pp. 1029-1064, 2000.

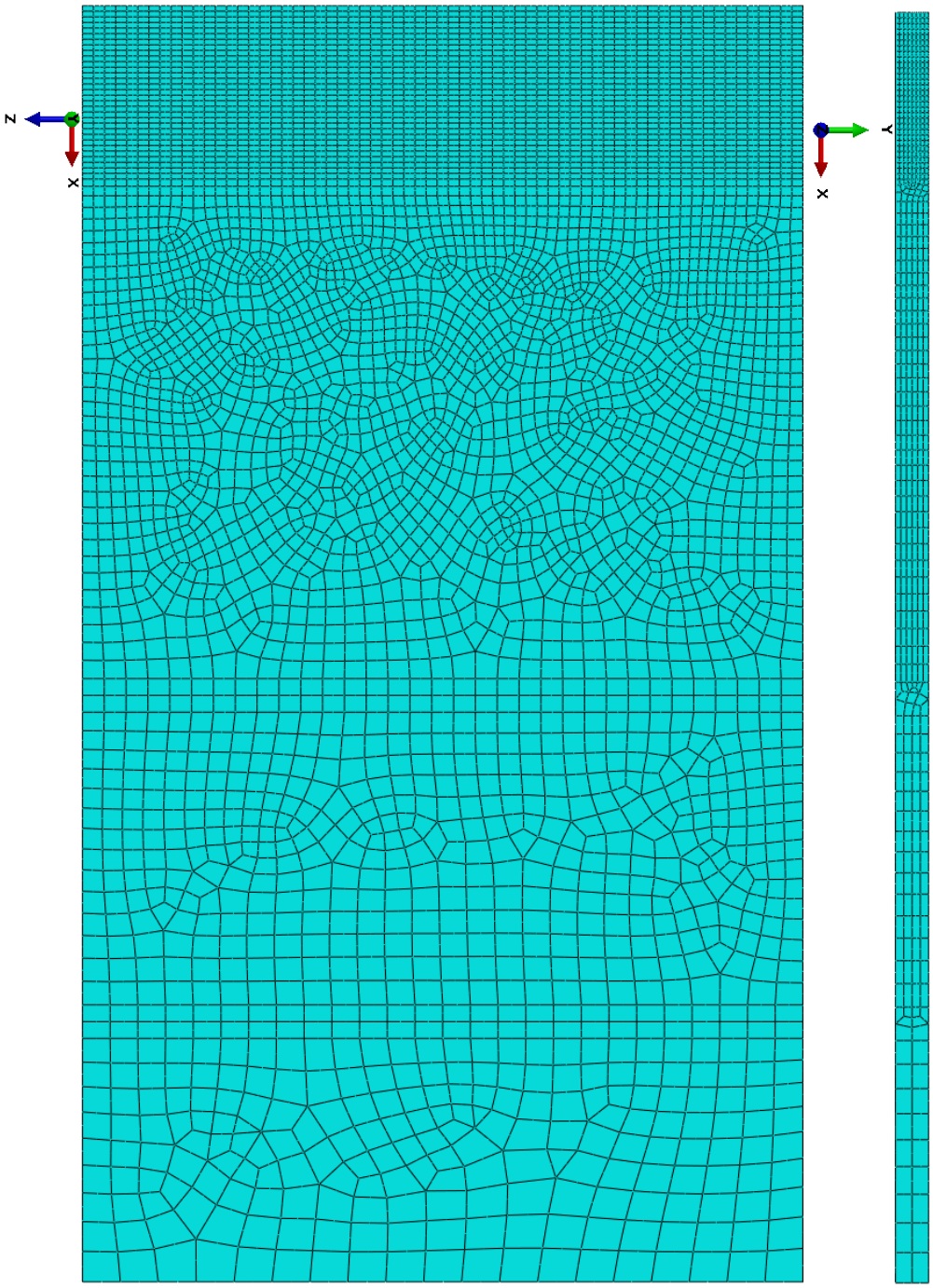
[32] Outinen, J. & Mäkeläinen, P. Mechanical Properties of Structural Steel at Elevated Temperatures and After Cooling Down, Fire and Materials - An International Journal, vol. 28, pp. 237-251, 2004.

[33] Ebrahimi, A.R., Abyazi, A. & Abbasi, S.M. Anisotropy in Microalloyed S355N Steel, International Journal of ISSI, Vol.5, No. 2, pp. 14-20, 2008.

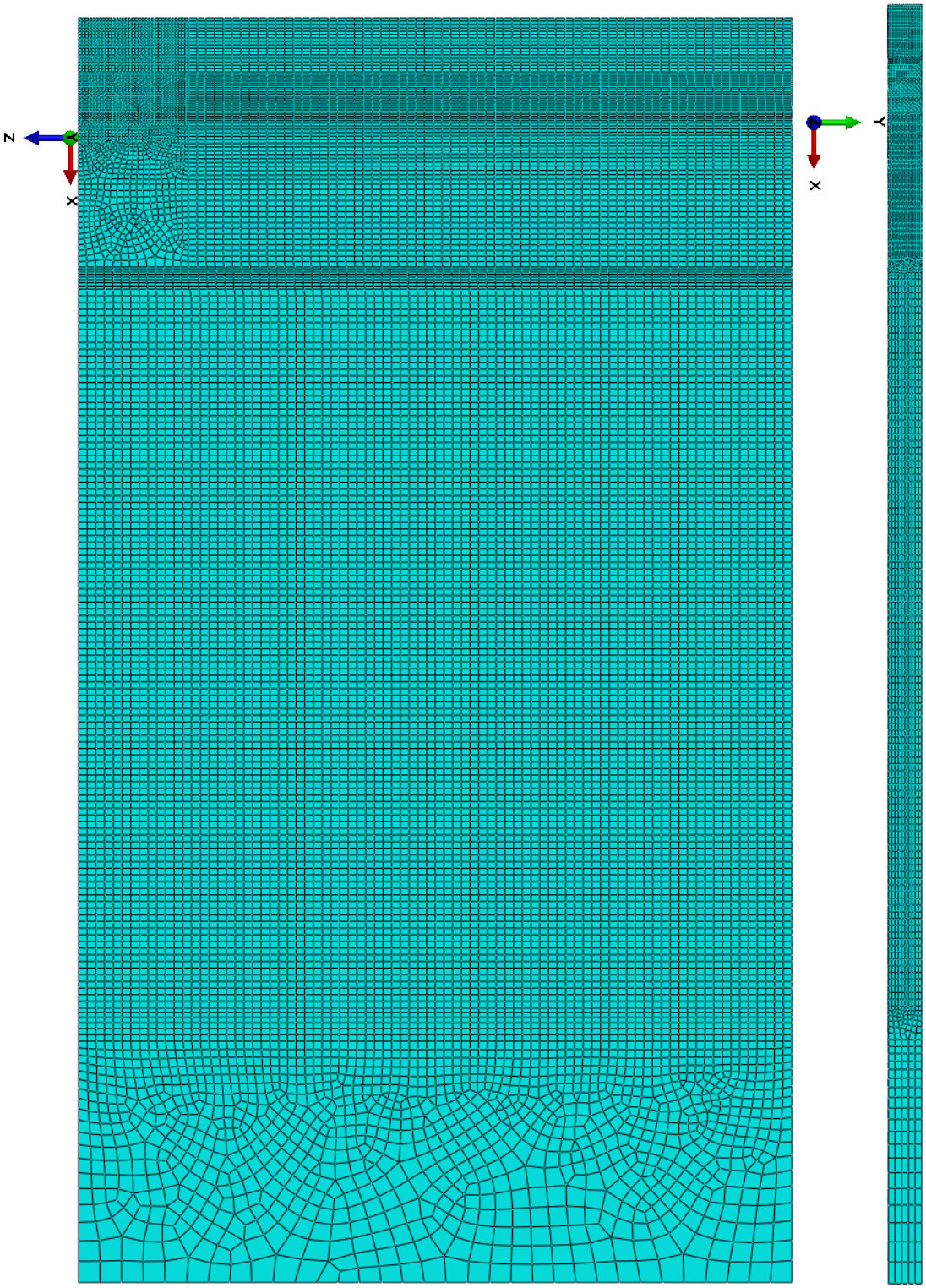
[34] Dieter, G.E. Mechanical Metallurgy, 3rd edition, McGraw-Hill Book Company, 1988.

A. APPENDICES

A.1 Implicit 3D model mesh



A.2 Explicit 3D model mesh



A.3 Hourglass patterns in explicit 3D mesh

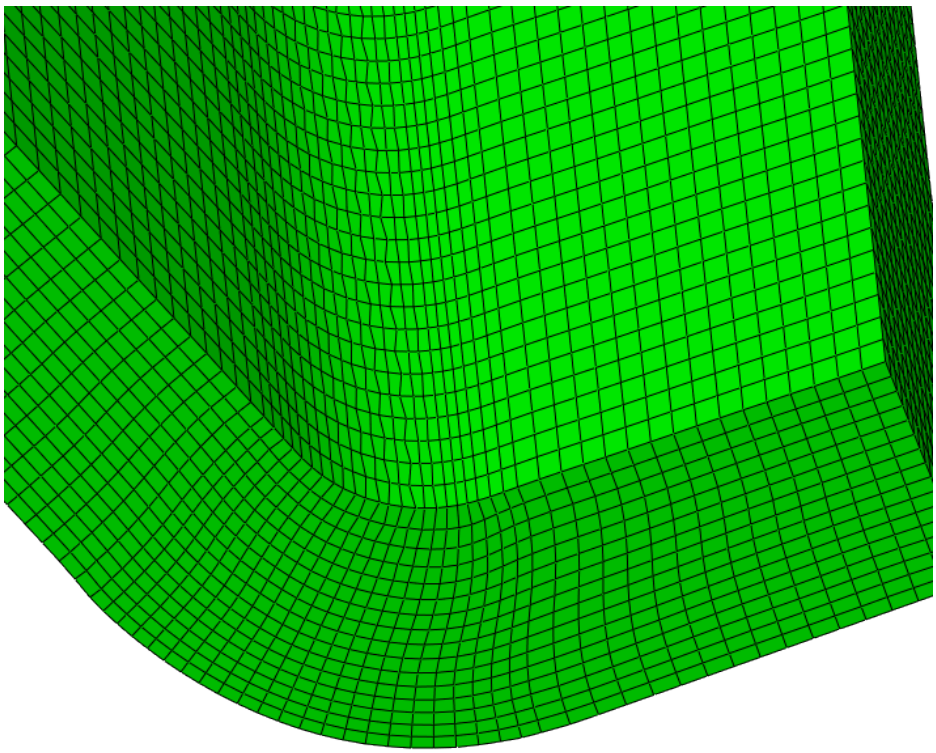


Figure A.1: kinematic contact constraint enforcement

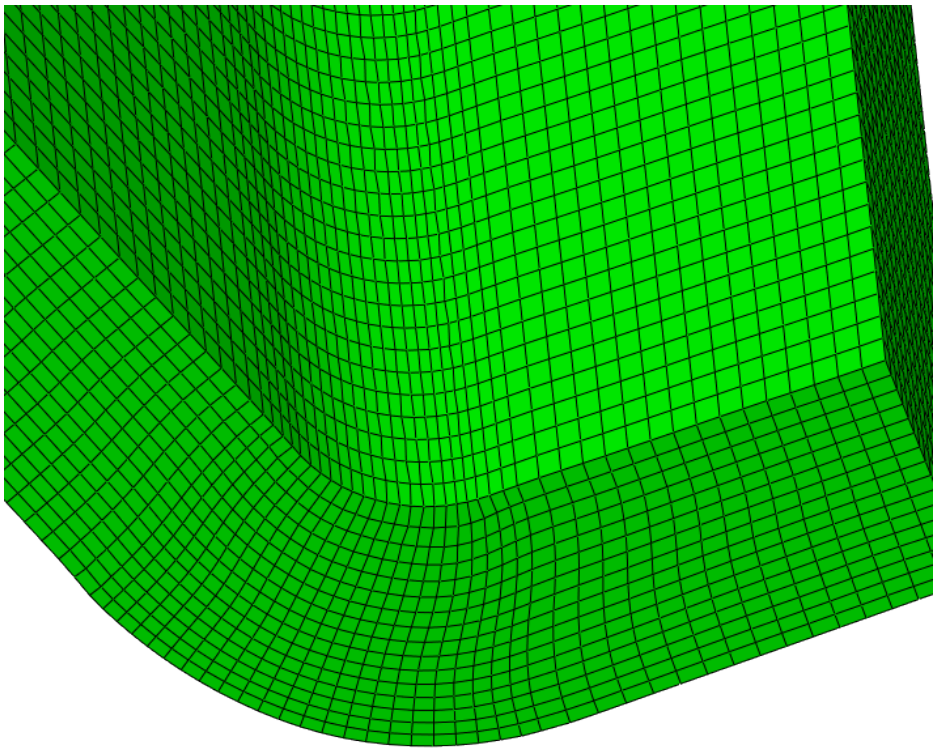
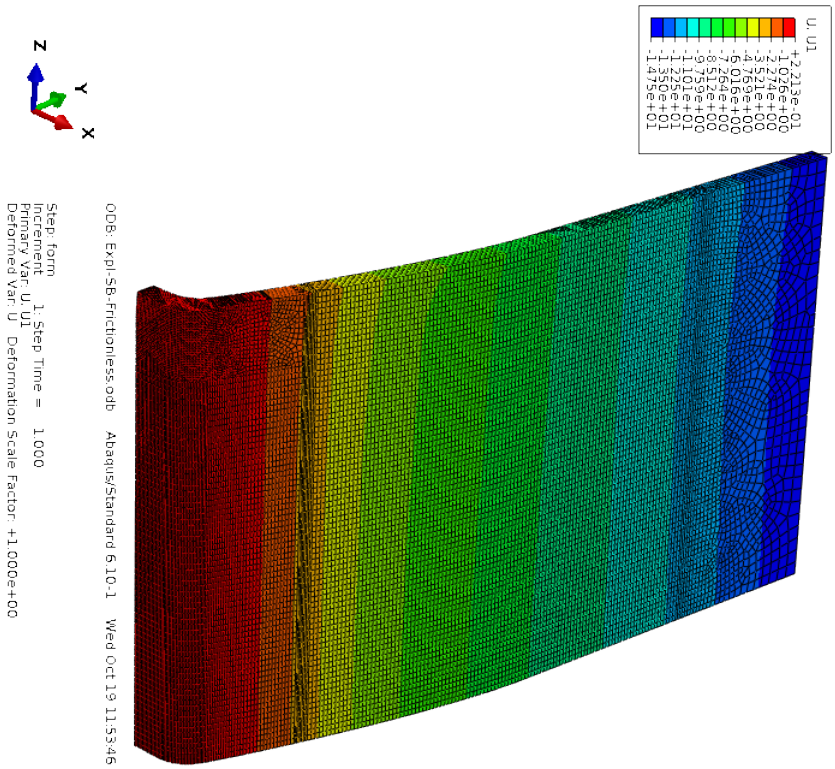
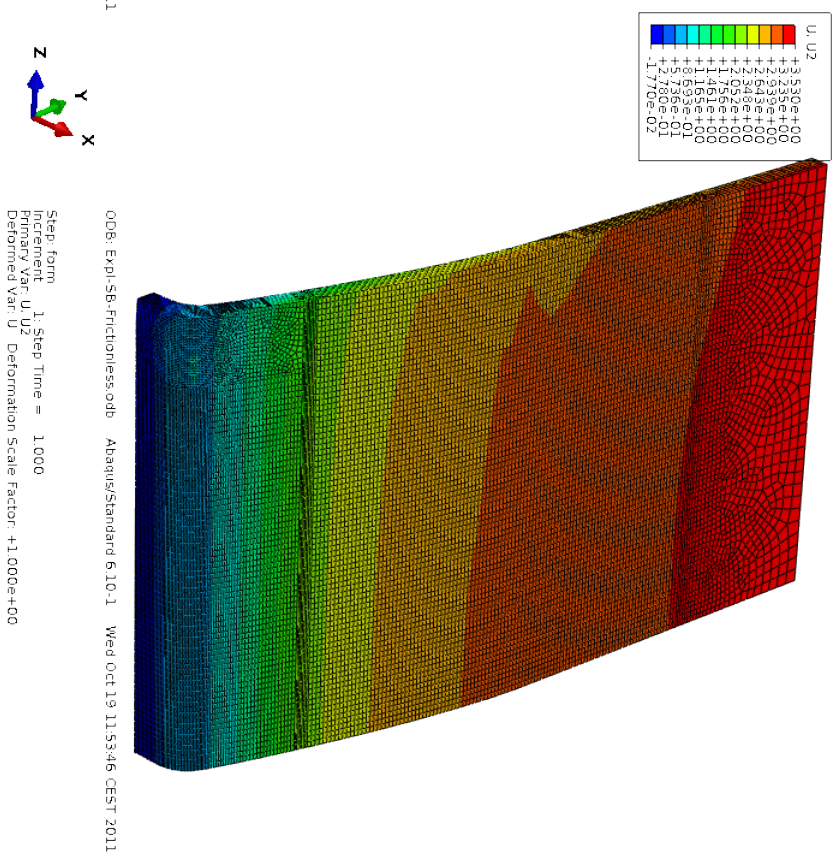


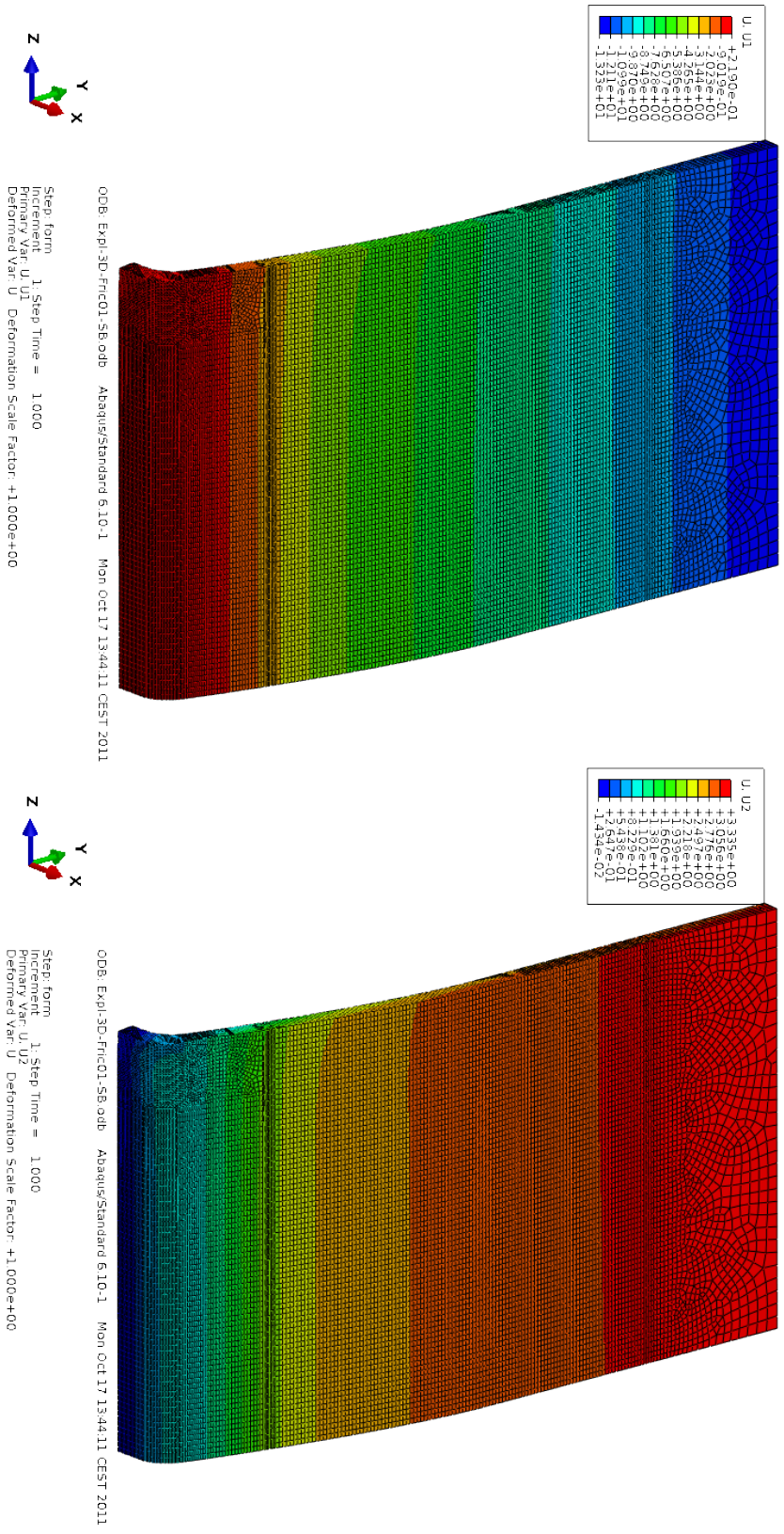
Figure A.2: penalty contact constraint enforcement

A.4 Explicit dynamic frictionless kinematic springback

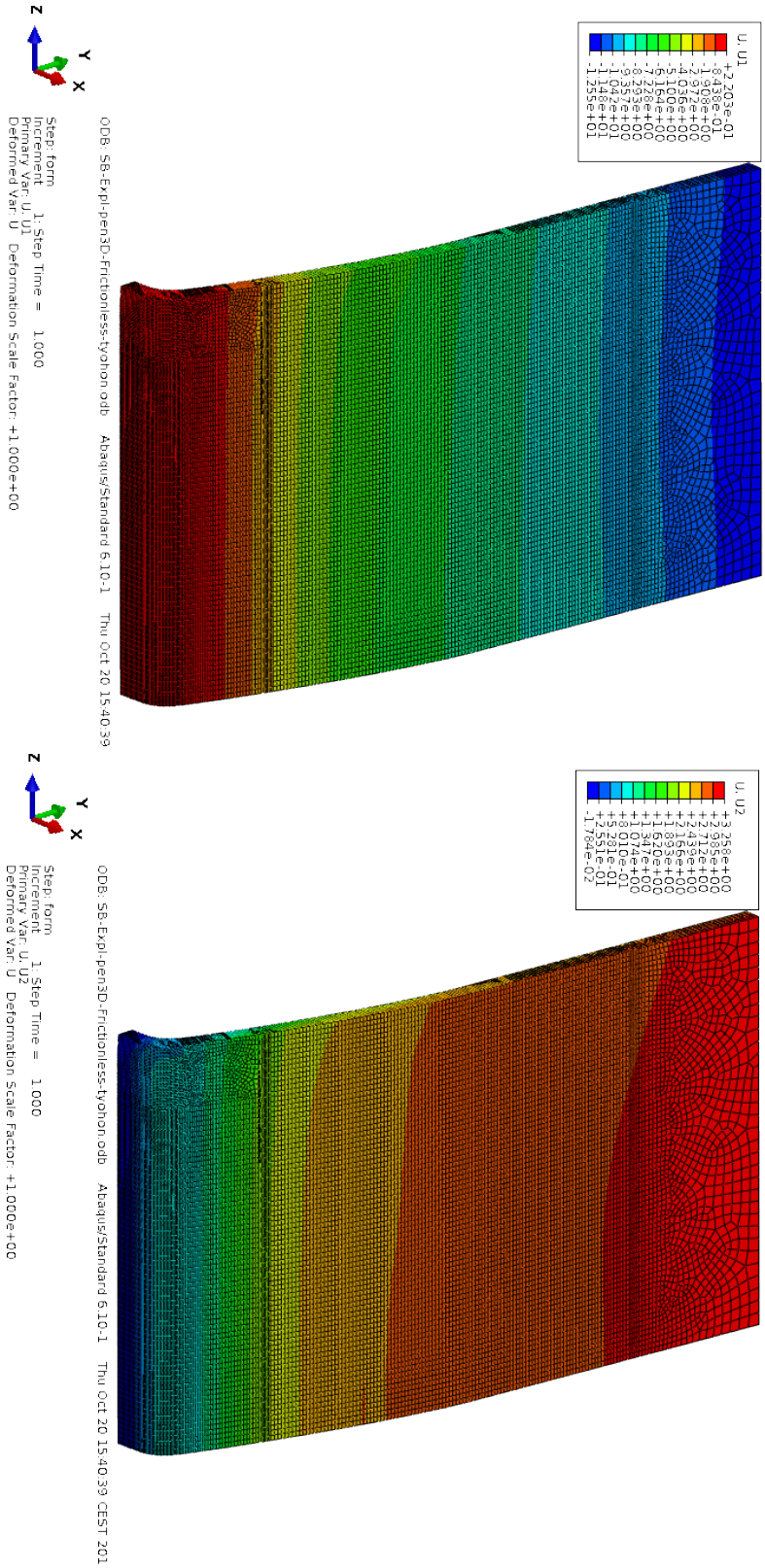




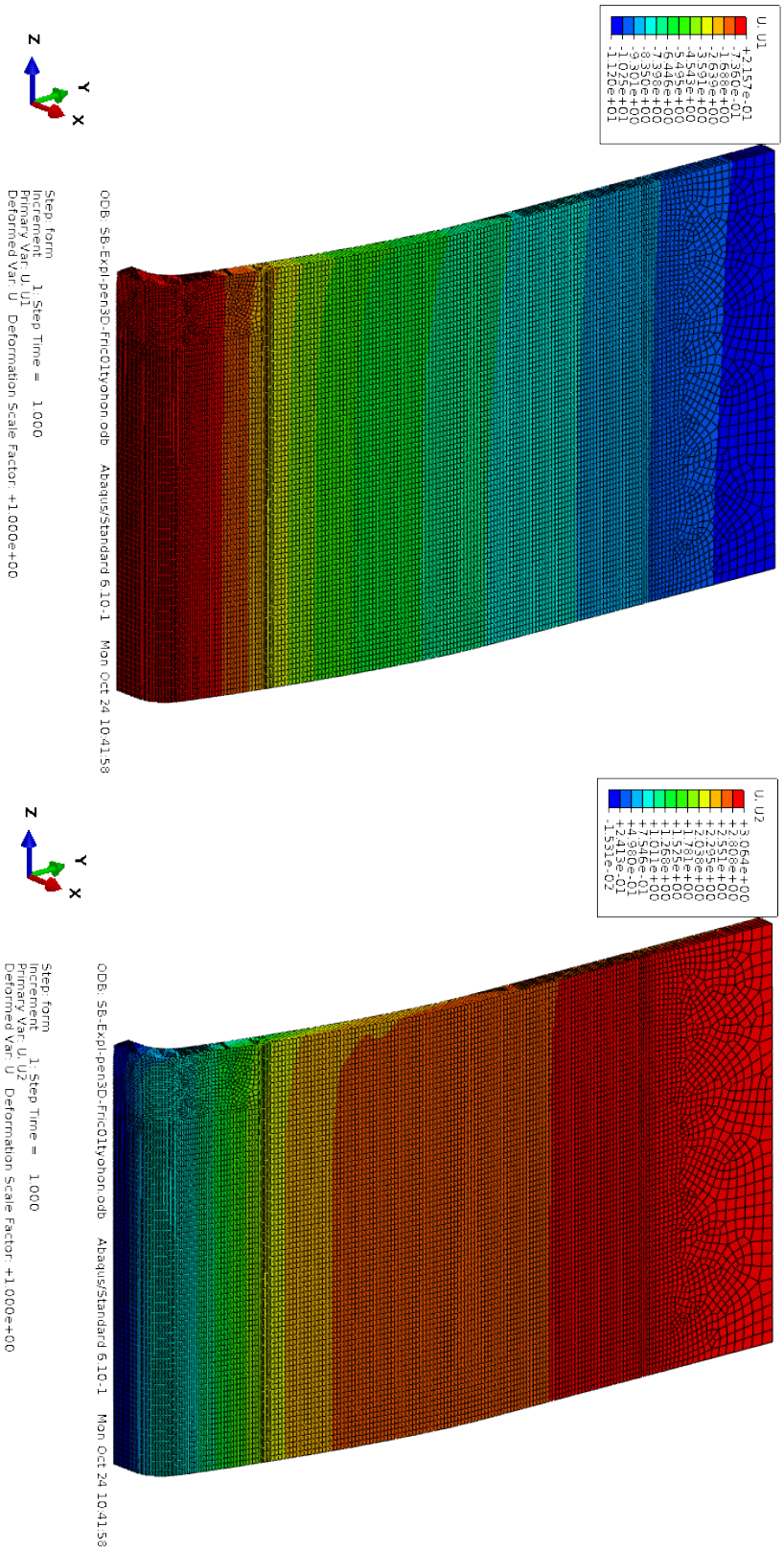
A.5 Explicit dynamic kinematic friction coefficient of 0.1 spring-back



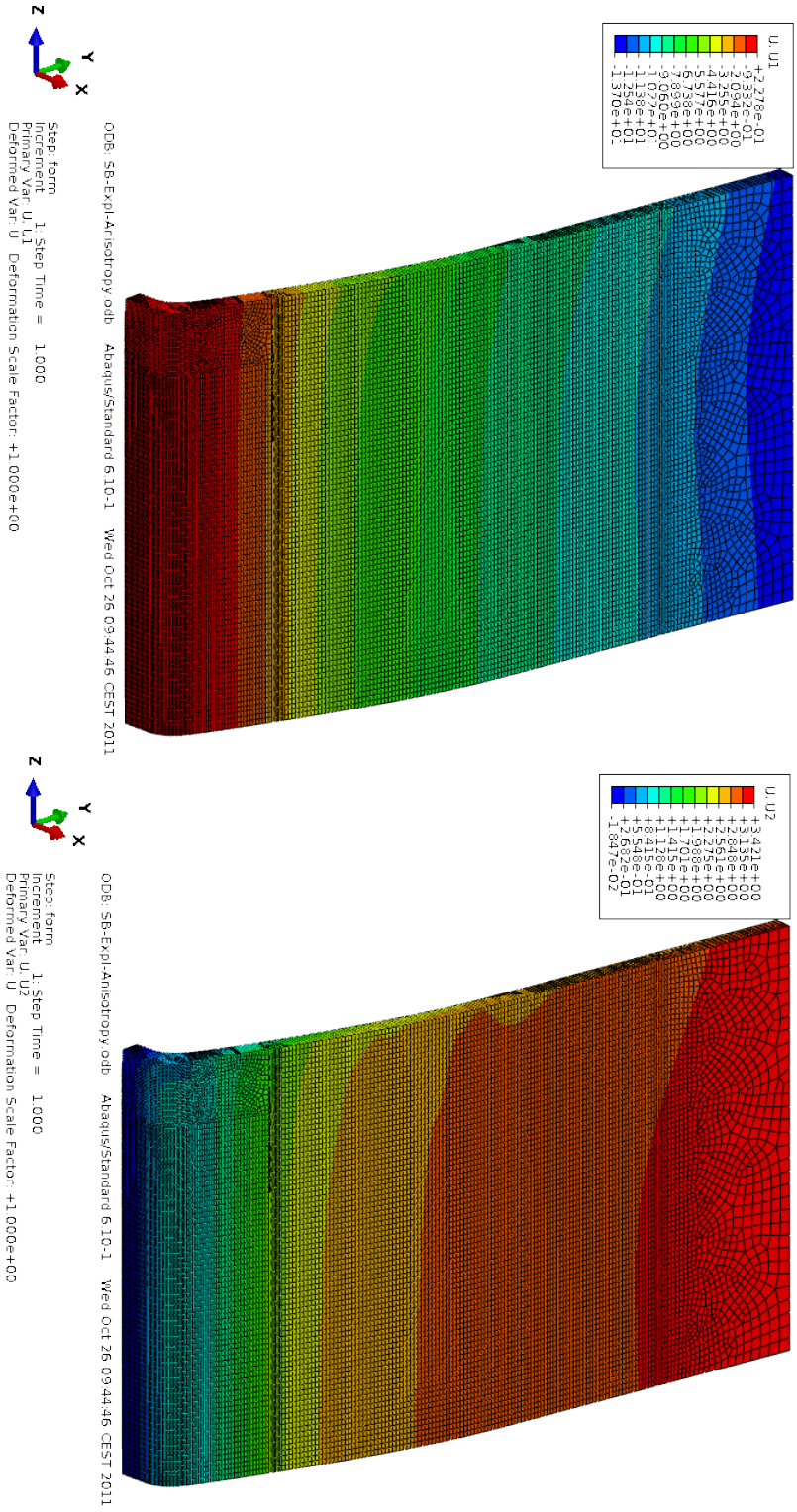
A.6 Explicit dynamic frictionless penalty springback



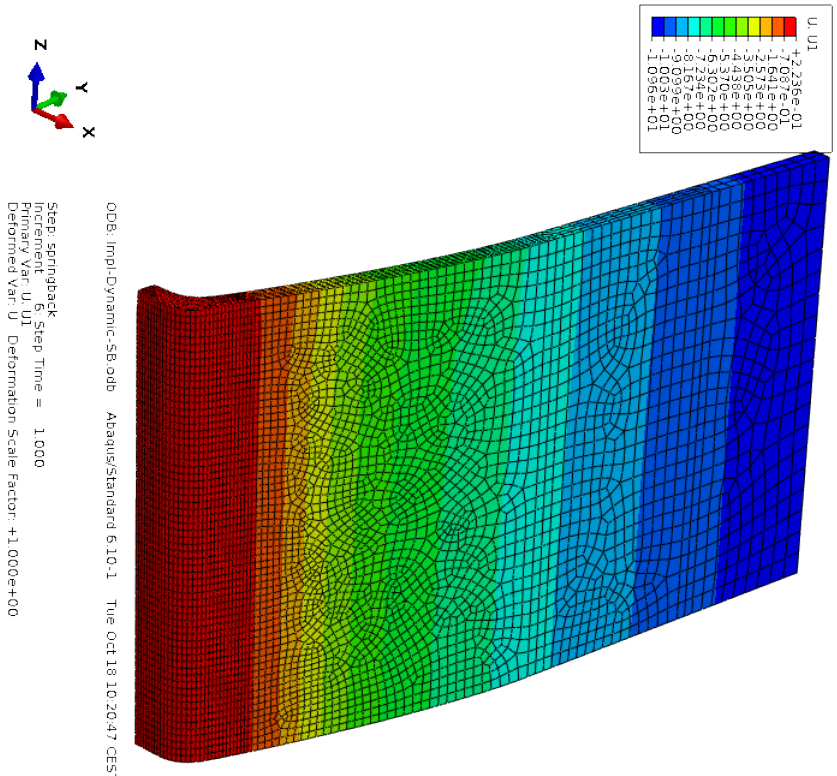
A.7 Explicit dynamic penalty friction coefficient of 0.1 spring-back

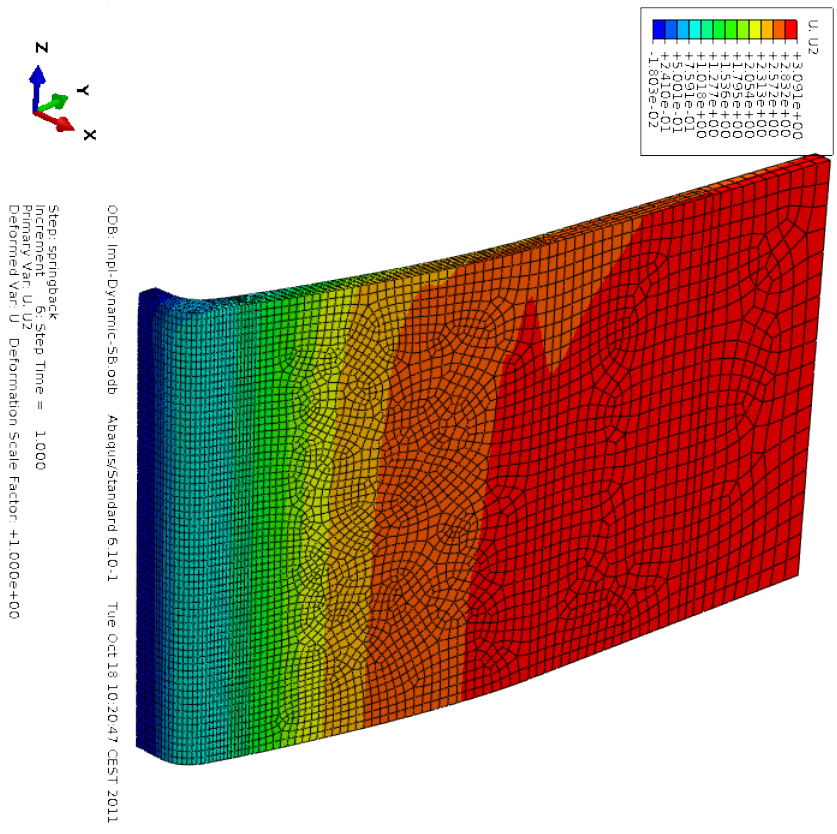


A.8 Explicit anisotropic model springback

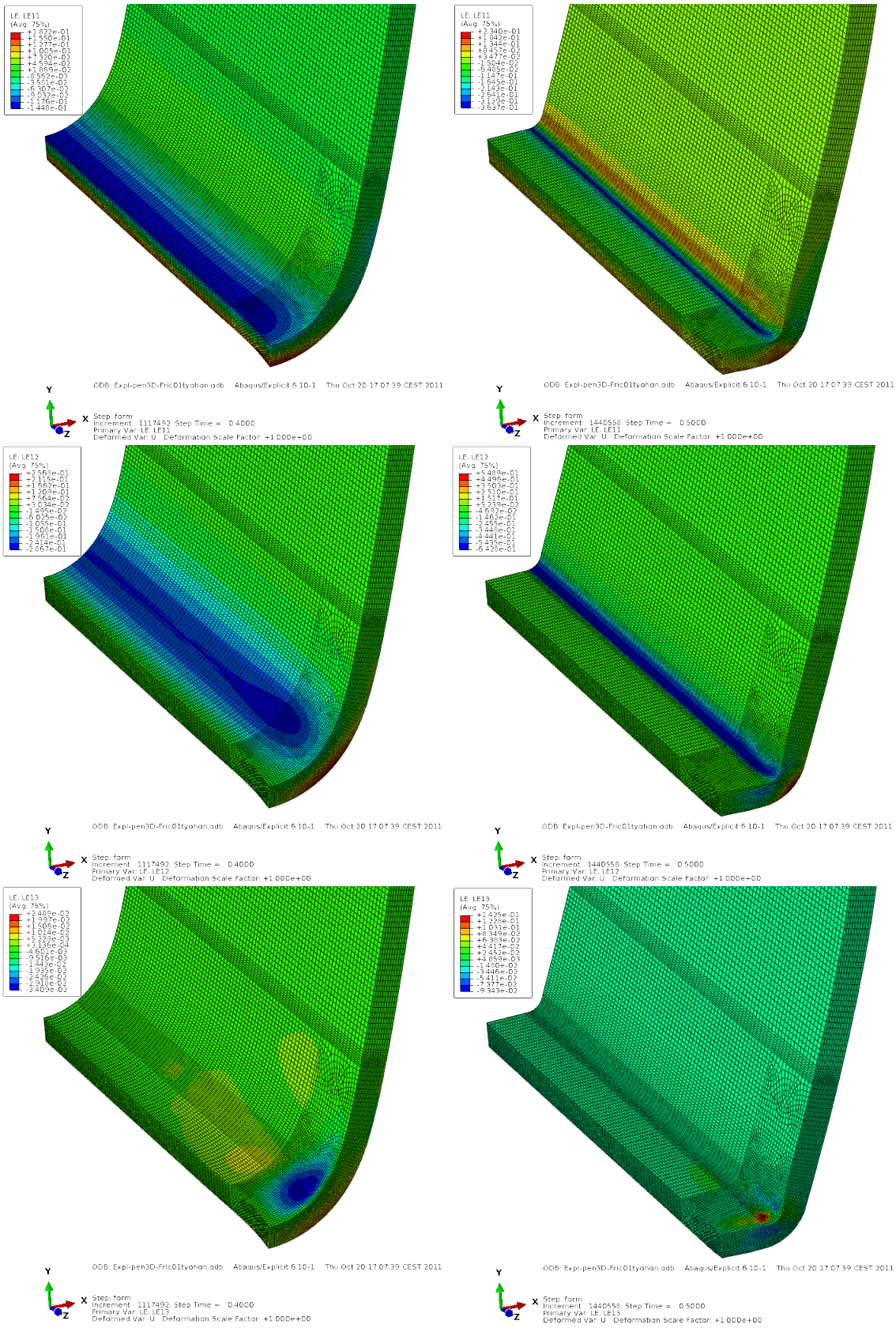


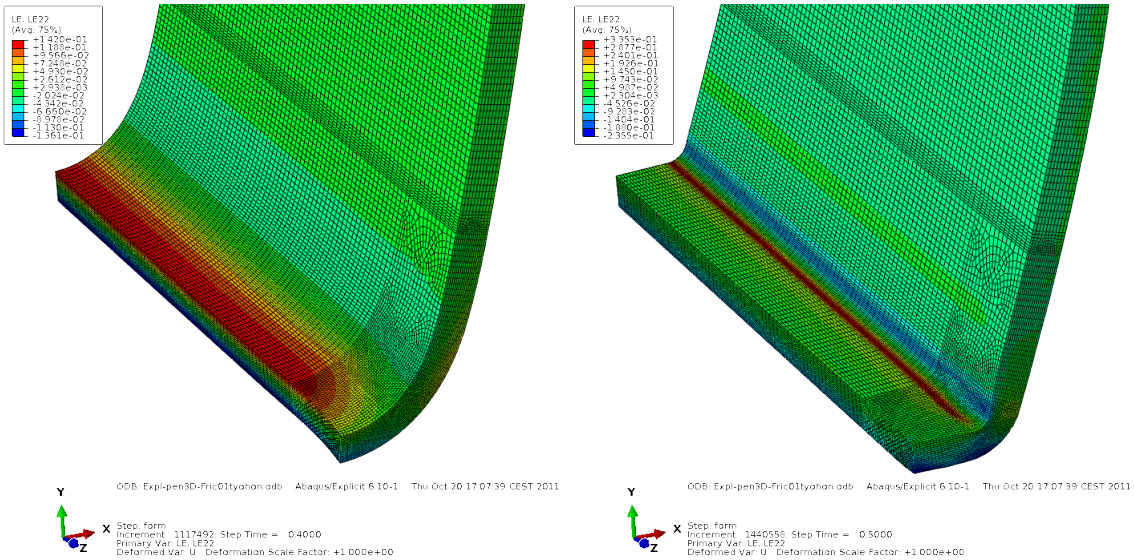
A.9 Implicit dynamic frictionless penalty springback





A.10 Other strain components on the 3D model




LE, LE22
(Avg: 75%)
+1.420e-01
+1.188e-01
+9.566e-02
+7.248e-02
+4.930e-02
+2.613e-02
+2.938e-03
-2.024e-02
-4.342e-02
-6.660e-02
-8.978e-02
-1.130e-01
-1.361e-01
ODB: Expl-pen3D-Fric01tyohon.odb Abaqus/Explicit 6.10-1 Thu Oct 20 17:07:39 CEST 2011
Step: form
Increment 1117492: Step Time = 0.4000
Primary Var: LE, LE22
Deformed Var: U Deformation Scale Factor: +1.000e+00
LE, LE22
(Avg: 75%)
+3.353e-01
+2.877e-01
+2.401e-01
+1.926e-01
+1.450e-01
+9.743e-02
+4.987e-02
+2.304e-03
-4.526e-02
-9.283e-02
-1.404e-01
-1.880e-01
-2.355e-01
ODB: Expl-pen3D-Fric01tyohon.odb Abaqus/Explicit 6.10-1 Thu Oct 20 17:07:39 CEST 2011
Step: form
Increment 1440558: Step Time = 0.5000
Primary Var: LE, LE22
Deformed Var: U Deformation Scale Factor: +1.000e+00
LE, LE23
(Avg: 75%)
+3.509e-02
+3.021e-02
+2.533e-02
+2.046e-02
+1.558e-02
+1.070e-02
+5.825e-03
+9.486e-04
-3.928e-03
-8.805e-03
-1.368e-02
-1.856e-02
-2.343e-02
ODB: Expl-pen3D-Fric01tyohon.odb Abaqus/Explicit 6.10-1 Thu Oct 20 17:07:39 CEST 2011
Step: form
Increment 1117492: Step Time = 0.4000
Primary Var: LE, LE23
Deformed Var: U Deformation Scale Factor: +1.000e+00
LE, LE23
(Avg: 75%)
+1.806e-01
+1.607e-01
+1.408e-01
+1.209e-01
+1.009e-01
+8.096e-02
+6.103e-02
+4.109e-02
+2.115e-02
+1.216e-03
-1.872e-02
-3.866e-02
-5.859e-02
ODB: Expl-pen3D-Fric01tyohon.odb Abaqus/Explicit 6.10-1 Thu Oct 20 17:07:39 CEST 2011
Step: form
Increment 1440558: Step Time = 0.5000
Primary Var: LE, LE23
Deformed Var: U Deformation Scale Factor: +1.000e+00

A.11 Hourglass control method comparison with a more flexible plasticity model

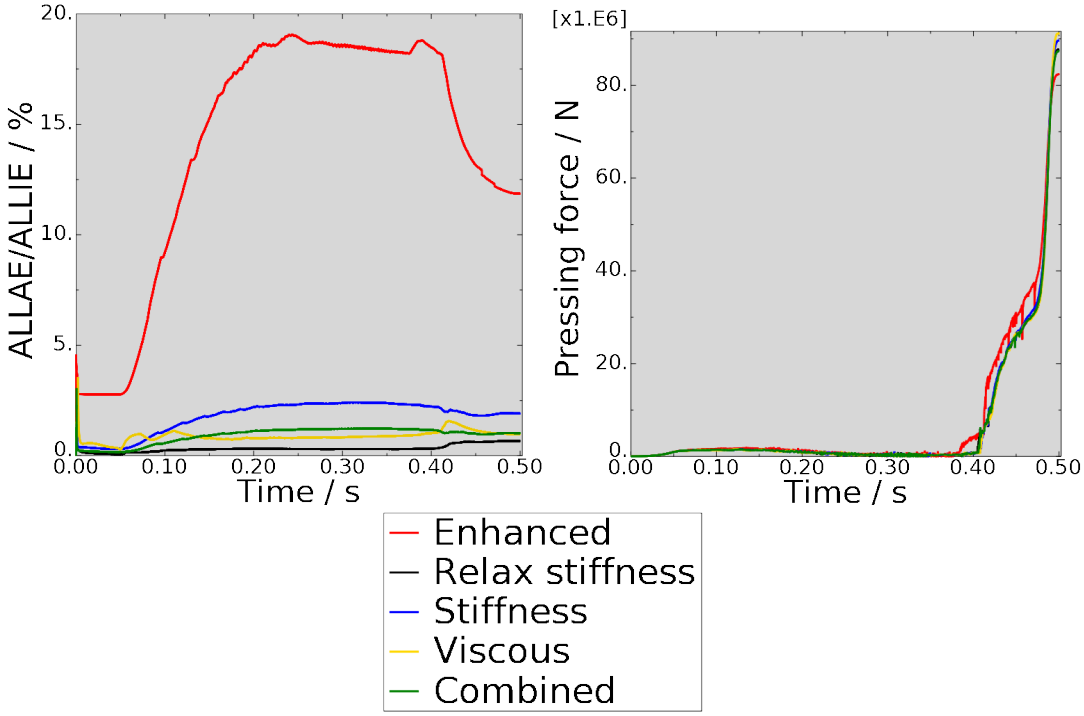


Figure A.3: Coarse mesh, see figure 6.19

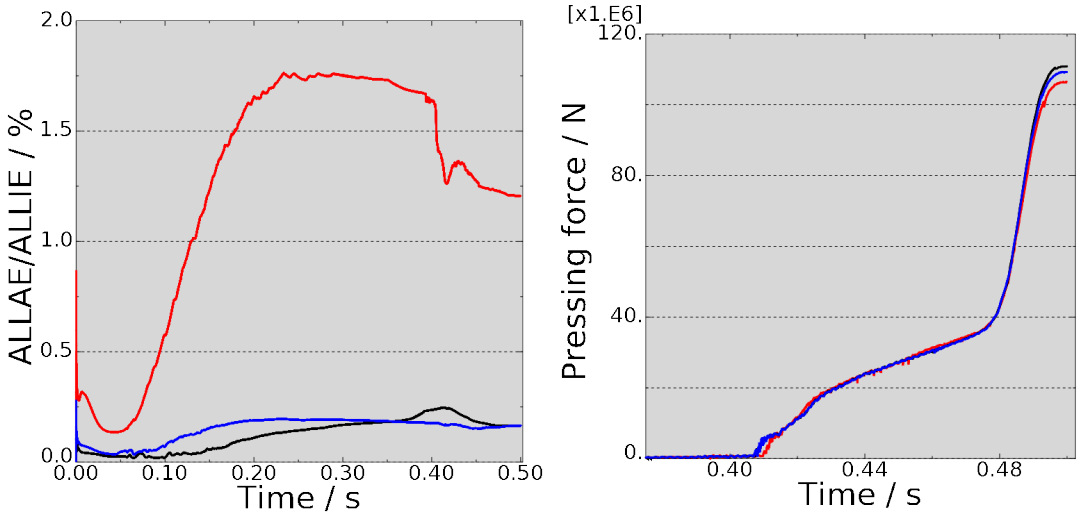


Figure A.4: 1/5 mesh with enhanced, relax stiffness and stiffness options