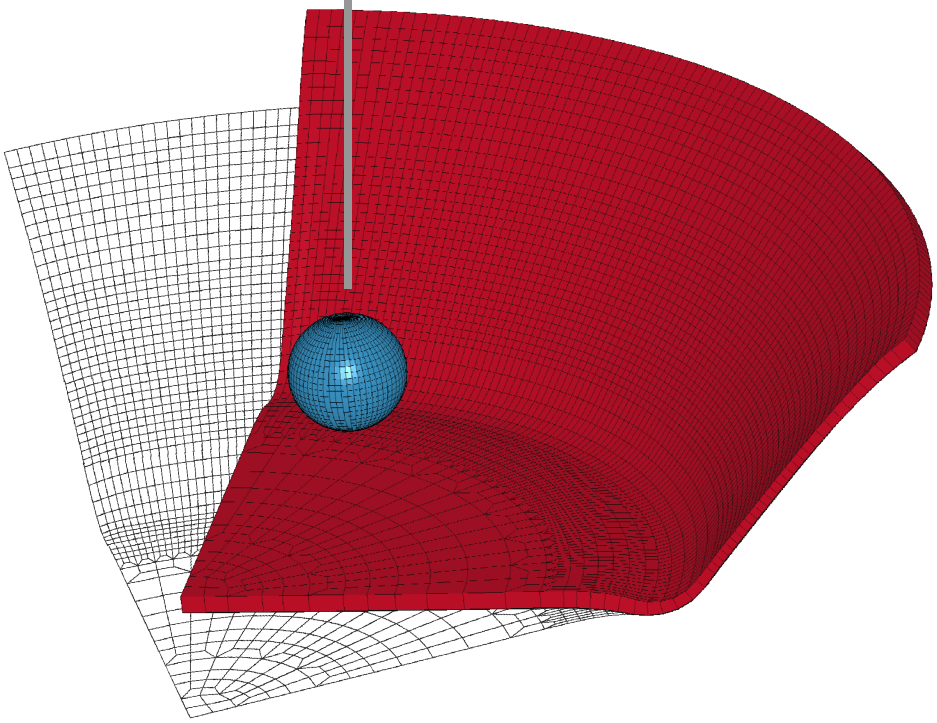
Numerical Simulations of the Single Point Incremental Forming Process

PhD thesis

Christophe Henrard

November 2008

Université de Liège



Contact Information

Author's coordinates:

Mr. Christophe Henrard

University of Liège
ArGEnCo Department, MS^{2}F division, building B52/3
1 Chemin des Chevreuils
4000 Liege 1
Belgium

Office phone: +32(0)4 366 95 23
Secretary phone: +32(0)4 366 92 60
Fax: +32(0)4 366 91 92
Email: Christophe.Henrard@ulg.ac.be
Website: http://www.argenco.ulg.ac.be

Supervisor's coordinates:

Dr. Anne Marie Habraken

Research Director (FNRS, Belgium)
University of Liège
ArGEnCo Department, MS^{2}F division, building B52/3
1 Chemin des Chevreuils
4000 Liege 1
Belgium

Office phone: +32(0)4 366 94 30
Secretary phone: +32(0)4 366 92 60
Fax: +32(0)4 366 91 92
Email: Anne.Habraken@ulg.ac.be
Website: http://www.argenco.ulg.ac.be

Jury Members

Prof. S. Cescotto, president of the jury
ArGEnCo Department, University of Liège, Belgium
Serge.Cescotto@ulg.ac.be

Dr. A.M. Habraken, supervisor of the thesis
ArGEnCo Department, University of Liège, Belgium
Anne.Habraken@ulg.ac.be

Prof. J.-P. Ponthot
ArGEnCo Department, University of Liège, Belgium
JP.Ponthot@ulg.ac.be

Prof. A.H. van den Boogaard
Faculty of Engineering Technology, University of Twente, The Netherlands
A.H.vandenBoogaard@utwente.nl

Prof. J. Duflou
PMA Department, Catholic University of Leuven, Belgium
Joost.Duflou@mech.kuleuven.be

Prof. A. Van Bael
MTM Department, Catholic University of Leuven, Belgium
Albert.VanBael@mtm.kuleuven.be

Prof. H. Sol
MEMC Department, Free University of Brussels, Belgium
hugos@vub.ac.be

Acknowledgements

The author of the present work would like to thank:

Dr. Anne Marie Habraken for her invaluable help during these five years of fruitful collaboration;

Prof. Serge Cescotto for his enlightened advice and productive discussions;

Dr. Frédéric Pascon, Dr. Laurent Duchêne, Chantal Bouffioux, and Cédric Lequesne for their scientific and personal help during this thesis;

All the members of the SeMPeR project for their interesting ideas, useful discussions and cooperation during the project, in particular Prof. Joost Duflou (SeMPeR's coordinator), Prof. Albert Van Bael, Prof. Hugo Sol, Philip Eyckens, Johan Verbert, Bachir Belkassem, Jun Gu, and Richard Aerens;

The jury members, for the time spent in reading this document;

All the other professors and colleagues of the ArGEnCo Department, especially, Renée Schwartz, Anne-Françoise Gerday, Barbara Rossi, Dr. Paulo Flores, Dr. Christian Perez, Dr. Dan Pintea, Léo Studer, and Dr. Lam Ly for their support;

My wife, Ellen Harry, for her time spent in proofreading this document and many more articles, and for her loving patience and support;

My friends and family for their encouragement.

In addition, this work could not have been achieved without the financial support of the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Belgian Federal Science Policy Office (Contract P6/24), and the University of Liège (teaching assistant contract).

Contents

List of Figures

List of Tables

Abbreviations and Symbols

Abbreviations

AA3003-O	Aluminum alloy used in all the simulations and experimental tests (more information can be found in Appendix C)
CAD	Computer-Aided Design
CNC	Computer Numerically Controlled
CPU	Central Processing Unit
DICT	Digital Image Correlation Technique, a data analysis method which uses a mathematical correlation method to analyze digital image data taken while samples are subjected to mechanical strains
DKT	Discrete Kirchhoff Theory
DOF	Degree of Freedom
DOI	Digital Object Identifier, a unique permanent identifier representing a digital entity; in the bibliography of this thesis, whenever the DOI of an article is mentioned, go to the web page http://dx.doi.org and type in the DOI to find the online version of the paper
FE	Finite Element
FLD	Forming Limit Diagram
GUI	Graphical User Interface
IP	Integration Point
KUL	Catholic University of Leuven (*Katholieke Universiteit Leuven*)
MK	Marciniak-Kuczynski
RPM	Rotations per minute
SeMPeR	Sheet Metal oriented Prototyping and Rapid manufacturing, an SBO-project financed by IWT
SPIF	Single Point Incremental Forming

TPIF	Two Point Incremental Forming
ULg	University of Liège (*Université de Liège*)
VUB	Free University of Brussels (*Vrije Universiteit Brussel*)
2D	Two-dimensional
3D	Three-dimensional

General Symbols

$\bullet$	Scalar
$\underline{\bullet}$	Vector
$\underline{\underline{\bullet}}$	Tensor
$\bullet_i$	Component i of vector $\underline{\bullet}$
$\bullet_{ij}$	Component ij of tensor $\underline{\underline{\bullet}}$
$\dot{\underline{\bullet}}$	First derivative of vector $\underline{\bullet}$ with respect to time
$\ddot{\underline{\bullet}}$	Second derivative of vector $\underline{\bullet}$ with respect to time
$\overset{\nabla}{\underline{\underline{\bullet}}}$	Jaumann's objective derivative of tensor $\underline{\underline{\bullet}}$
$\underline{\delta\bullet}$	Virtual variation of vector $\underline{\bullet}$
$\underline{\Delta\bullet}$	Variation of vector $\underline{\bullet}$
$\underline{\widehat{\bullet}}$	Estimate of the vector $\underline{\bullet}$
$\underline{\bullet}^*$	Vector not including contact forces
$\underline{\underline{\bullet}}^{-1}$	Inverse of tensor $\underline{\underline{\bullet}}$
$\underline{\bullet}^T$, $\underline{\underline{\bullet}}^T$	Transposed vector or tensor
$\langle \bullet_1 \cdots \bullet_N \rangle$	N components of the vector $\underline{\bullet}$ stored in line
$\mathrm{tr}(\underline{\underline{\bullet}})$	Trace of a tensor $\underline{\underline{\bullet}} = \bullet_{kk}$, i.e., the sum of its diagonal components
$\dfrac{\partial \bullet}{\partial \bullet}$	Partial derivative
$\underline{\bullet} \wedge \underline{\bullet}$	Cross product between two vectors, the result being a vector
$\underline{\bullet} \cdot \underline{\bullet}$	Scalar product between two vectors, the result being a scalar
$\underline{\underline{\bullet}} : \underline{\underline{\bullet}}$	Double tensor contraction, the result being a scalar
$(m \times n)$	Size of a matrix, vector or tensor (m lines, n columns)

Roman Letters

c	Proportionality constant

c_1	Damping parameter
c_2	Damping parameter
$\underline{\underline{C}}$	Damping matrix
$C^{(i)}$	i^{th} contact point on the tool surface
C_A	Material parameter of Ziegler's mixed hardening
C_X	Material parameter of Armstrong-Frederick's mixed hardening
$d^{(i)}$	Distance between the i^{th} contact point on the metal sheet and the tool center
D_i^J	Diagonal term of the mass matrix (i^{th} DOF of node J)
E	Young's modulus
F	Material parameter of Hill's yield locus
F_f	Norm of the friction force, $\sqrt{F_s^2 + F_t^2}$
$F_n^{(j)}$	Energetically-equivalent nodal forces due to an arbitrary contact force $\Delta P^{(j)}$ on the j^{th} contact point
$F_s^{(j)}$, $F_t^{(j)}$	Two tangential components of the contact force on the j^{th} contact point
$\underline{F}$	Force vector
$\underline{F}_{contact}$	Contact forces
$\underline{F}_{ext}$	Externally applied forces
$\underline{F}_f$	Friction force vector
$\underline{F}_{fixed}$	Supporting forces (reactions on fixed DOFs)
$\underline{F}_{int}$	Internal forces (integration of stresses)
$F(\ldots)$	Yield criterion, a function which is negative if the material is in the elastic domain and zero in the plastic domain
G	Material parameter of Hill's yield locus
$\underline{G}$	Center of gravity of an element
G_A	Material parameter of Ziegler's mixed hardening
H	Material parameter of Hill's yield locus
$\underline{\underline{H}}$	Two-dimensional tensor defining the anisotropy of the material using Doghri's notation
$\underline{\underline{\mathbf{H}}}$	Fourth-order tensor defining the anisotropy of the material
I	Inertia of the tool about its rotation axis
$\underline{\underline{I}}$	Unit tensor, represented by ones on the diagonal and zeros everywhere else

K	Material parameter of Swift's isotropic hardening
$\underline{\underline{K}}^e$	Elastic stiffness matrix
K_p	Penalty coefficient
L	Material parameter of Hill's yield locus
L_0	Initial length of an element
L_f	Final length of an element
L_{min}	Characteristic length of the smallest element of the mesh
M	Material parameter of Hill's yield locus
M	Mass-scaling factor
MR	Additional mass-scaling factor for the rotational DOFs
$\underline{M}_d$	Vector containing the diagonal terms of the mass matrix
$\underline{\underline{M}}$	Mass matrix
$\underline{\underline{M}}^{IJ}$	Submatrix IJ of the mass matrix of one element
n	Material parameter of Swift's isotropic hardening
$\hat{n}^{(i)}$	Normal unit vector located at the i^{th} contact point
N	Material parameter of Hill's yield locus
$P^{(j)}$	Normal component of the contact force on the j^{th} contact point
$\Delta P^{(j)}$	Arbitrary normal contact force on the j^{th} contact point
$P^{I+4}(\xi, \eta)$	First function used to simplify the notations in the mass matrix of the COQJ4 shell element
$Q^{I+4}(\xi, \eta)$	Second function used to simplify the notations in the mass matrix of the COQJ4 shell element
r, θ, z	Cylindrical coordinates (radial, circumferential, z)
$\underline{r}$	First vector of the local coordinate system defined at a contact point
r_0, r_{45}, r_{90}	Lankford's coefficients at 0, 45, and 90 degrees from the rolling direction
r_L	Lankford's coefficient
R	Tool radius
$\underline{R}$, $\underline{S}$, $\underline{T}$	Local axes of the COQJ4 shell element
$\underline{\underline{R}}$	Rotation matrix
s	Coefficient used to adjust the size of the neighborhood in the MST method
$\underline{s}$	Second vector of the local coordinate system defined at a contact point

$\underline{\underline{s}}$	Deviatoric part of the tensor $\underline{\underline{\sigma}} = \underline{\underline{\sigma}} - 1/3\,\mathrm{tr}(\sigma)\underline{\underline{I}}$
t	Thickness of a shell element
$\underline{t}$	Third vector of the local coordinate system defined at a contact point
Δt	Time step
Δt_c	Time step, critical to explicit stability
t_0	Initial thickness
t_f	Final thickness
u_1, u_2	Translational DOFs of the COQJ4 shell element
V	Finite element's volume
$v_s^{(i)}$, $v_t^{(i)}$	Two components of the relative velocity between the i^{th} contact point on the metal sheet and on the tool surface
$\underline{V}$	Relative velocity vector (between a contact point and the tool surface)
w	Translational DOF of the COQJ4 shell element
w^{I+4}	Relative displacement of the mid-side node $(I+4)$ for a COQJ4 element
x, y, z	Cartesian coordinates
$\underline{x}$	Position vector
$\underline{x}^I$, $\underline{x}^J$	Position vector of node I or J of one finite element
$\underline{X}^I$, $\underline{X}^J$	Position vector of node I or J of one finite element in its initial configuration
$\left[\underline{x}^N\right]$	(12×1) vector containing the position vectors of all the nodes of one finite element
$\underline{\underline{X}}$	Back-stress tensor
X_{sat}	Material parameter of Armstrong-Frederick's mixed hardening

Greek Letters

α	Wall angle
γ	Pie-mesh angle
$\underline{\underline{\epsilon}}$	Strain tensor
$\underline{\underline{\epsilon}}^p$	Plastic strain tensor
ϵ_0	Material parameter of Swift's isotropic hardening
ϵ_{eq}	Equivalent strain

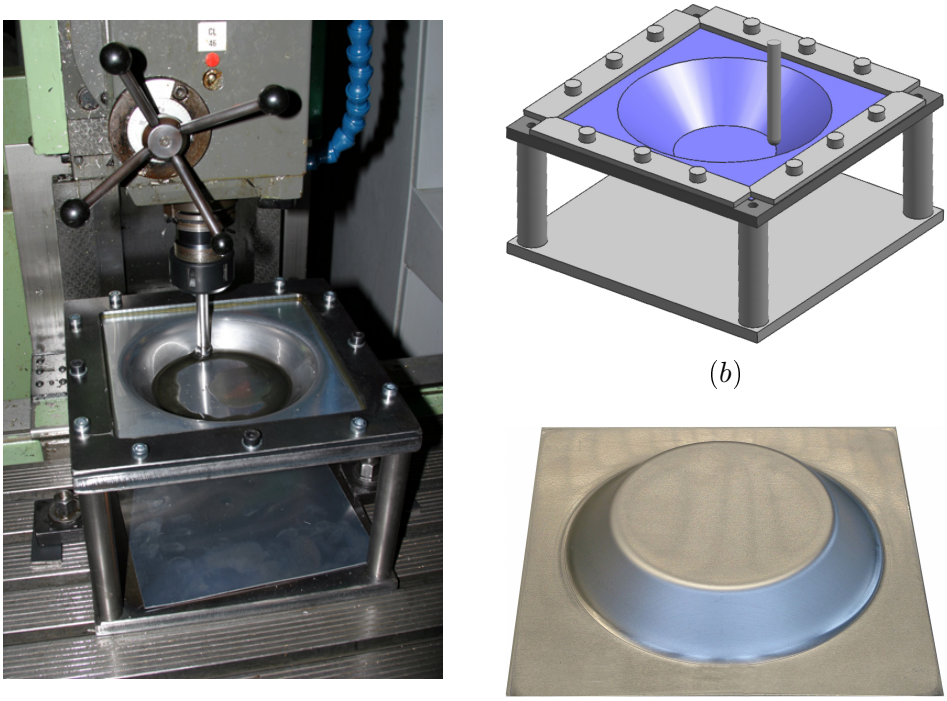
ϵ^p_{eq}	Equivalent plastic strain
η	Second local coordinate in the reference element
$\eta^{(j)}$	Second local coordinate (in the reference element) of the j^{th} contact point
θ	Circumferential coordinate
θ^I_1, θ^I_1, θ^I_1	Rotational DOFs of node I of a COQJ4 element (using the shell element convention)
$\underline{\theta}^I$	Vector containing the rotational DOFs of node I of a COQJ4 element
λ	Elongation ratio of an element
μ	Friction coefficient
ν	Poisson's ratio
ξ	First local coordinate in the reference element
$\xi^{(j)}$	First local coordinate (in the reference element) of the j^{th} contact point
Φ	Friction coefficient
φ_1, φ_2, φ_z	Rotational DOFs of the COQJ4 shell element (positive in the clockwise direction)
$\varphi^{(i)}$	Latitude of the i^{th} contact point on the tool surface
ρ	Specific mass, in kg/mm^3
$\underline{\sigma}$	Stress tensor written as a vector using Doghri's notation
$\underline{\underline{\sigma}}$	Stress tensor
σ_{eq}	Equivalent stress
$\sigma_F(\bullet)$	Yield stress, a scalar function representing the size of the yield surface
σ^0_F	Material parameter of Swift's isotropic hardening
$\Phi^I(\xi,\eta)$	I^{th} first-order interpolation function of an element, which is equal to 1 at node I and 0 at the other nodes
$\Phi^{I+4}(\xi,\eta)$	I^{th} second-order interpolation function of an element, which is equal to 1 at mid-side node $I+4$ and 0 at the other nodes
ω	Rotational speed of the tool about its axis
ω_n	Normalized cutoff frequency of Butterworth's filter

Chapter 1

Introduction

1.1 Scope of the Study

In the modern engineering world, technological advancements drive the product design process. Increasingly powerful CAD programs make more complex product designs possible, which in turn boost the demand for more complex prototypes. At the same time, fast-moving competitive markets require frequent design changes, shorter lead times, and tighter budgets. In short, prototyping must be faster, better, and less expensive.



(*a*) (*b*) (*c*)

Figure 1.1: Sheet metal prototyping with the incremental forming process; (*a*) milling machine forming a conical part, (*b*) CAD surface, blank-holder and tool, (*c*) final part; pictures courtesy of Johan Verbert (KUL)

Within this context, rapid prototyping in sheet metal is highly desirable be-

cause the manufacturing of functional prototypes speeds up the time to market. While the market is well developed when it comes to rapid prototyping for plastic parts, the options for prototyping geometrically complicated sheet metal components are more limited and extremely expensive, because all the methods available require expensive tooling, machinery or manual labor.

Unlike many other sheet metal forming processes, incremental forming does not require any dedicated dies or punches to form a complex shape. Instead, the process uses a standard smooth-end tool, the diameter of which is far smaller than the part being made, mounted on a three-axis CNC milling machine.

The sheet metal blank is clamped around its edges using a blank-holder. During the forming process, the tool moves along a succession of contours, which follow the final geometry of the part, and deforms the sheet into its desired shape incrementally.

The incremental forming process is illustrated in Figure 1.1.

1.2 Context of the Research

The work presented in this thesis was started in October 2003 in the framework of the SeMPeR project (Sheet Metal oriented Prototyping and Rapid manufacturing). This was a four-year-long project, whose purpose was to develop a research platform that would support an in-depth analysis of the incremental forming and laser forming processes. This platform supported experimental, numerical, and analytical research activities, the interaction between which was expected to lead to the design of new and improved process variants and the identification of effective process planning and control strategies.

Four research partners from three different universities were involved in the project, covering the various academic disciplines required. As project leader, the PMA Department of the Catholic University of Leuven (KUL) provided extensive background knowledge in numerically controlled sheet metal forming processes, as well as long-term experience of experimental hardware development and process planning. This department was in charge of the experimental study of the processes. The MTM Department from the same university studied the processes in detail using accurate finite element models. The MEMC Department of the Free University of Brussels (VUB) provided expertise in in-process strain and displacement measurement, and material characterization by means of inverse method techniques. Finally, the ArGEnCo Department of the University of Liège (ULg), to which the present author is affiliated, undertook the task of developing a finite element code adapted to the incremental forming process.

Because of its promising outcome, the project held wide industrial interest: several companies assisted in ensuring the ultimate industrial relevance of the research and provided logistical support in terms of hardware, materials, and specific data.

1.3 Objective of the Thesis

Although the SeMPeR project aimed at studying two rapid prototyping processes, the present work focused only on one of those: incremental forming.

The goal of the team at the University of Liège was to adapt a department-made finite element code, LAGAMINE, to the incremental forming process. In particular, the computation time had to be reduced as much as possible while maintaining a sufficient level of accuracy.

1.4 Outline of the Thesis

The body of the text is divided into three parts.

The first part contains two chapters. The first of these provides a literature review in the field of incremental forming. More specifically, it introduces the process, presents an overview of its practical implementation and experimental setup requirements, and shows its benefits and limitations. Then, the chapter focuses on the latest developments in terms of finite element modeling and analytical computations.

The second chapter presents the numerical tools used throughout this research. This consists mainly of the finite element code, the elements, and the constitutive laws. Then, this chapter gives an overview of the experimental setup and measuring devices used during the experimental tests performed in Leuven.

The second part focuses on dynamic explicit simulations of incremental forming and contains four chapters. The first justifies the use of a dynamic explicit strategy. The second presents the new features added to the finite element code in order to be able to model incremental forming with such a strategy. The third explains the computation of the mass matrix of the shell element used throughout this part of the thesis and justifies this computation. Finally, the fourth chapter analyzes the overall performance of the dynamic explicit simulations both in terms of accuracy and computation time.

The third part of this thesis contains an in-depth analysis of the incremental forming process using more classic implicit finite element simulations. This analysis is performed in two steps. In a first chapter, the influence of using a partial mesh for the simulations is evaluated in terms of accuracy and computation time. Then, in a second and final chapter, a detailed analysis of the deformation mechanism occurring during this forming process is carried out.

Finally, this thesis ends with the major conclusions drawn from the research and perspectives on possible means of further improving the simulation tool.

1.5 Original Contributions

Through this research, several major contributions were achieved.

First, a comprehensive literature review of the incremental forming process was carried out. In particular, the review focused on original articles concerning the limitations of the process and possible ways of bypassing them; on the most recent explanations for the increased formability observed during the process; and on the state of the art in finite element simulations of incremental forming. Understanding the concepts and difficulties inherent in these publications was

made possible particularly by the SeMPeR project thanks to the discussions held and the monthly follow-ups on research performed by its members.

Secondly, LAGAMINE's shell element was corrected and its mass matrix modified to enable its use with an explicit strategy. Following this, a new approach for modeling the contact between an element and the forming tool during simulations in a dynamic explicit strategy was developed and thoroughly tested. A detailed comparison of the influence of various finite element parameters on the simulations' results was performed, in particular regarding the choice between using the implicit and explicit strategies and the use of mass scaling to reduce the computation time.

In addition, many simulations were validated thanks to experimental results.

Moreover, the computation time required for simulations of the forming of parts with rotational symmetry was radically reduced by using a partial model with a new type of boundary conditions.

Finally, the material behavior occurring during incremental forming was analyzed.

Part I

State of the Art and Tools

Chapter 2

State of the Art

2.1 Incremental Forming Process

2.1.1 History

Although the basic ideas of incremental forming were patented almost forty years ago by Leszak [LES67], they were not put into practice until quite recently. One of the reasons for that fact is the development of the computer numerically controlled (CNC) machines. This topic started to appear in scientific literature beginning in the early nineties [POW92, ISE92, KIT93, MAT94]. This process has since been developed in response to the observed deficiency in rapid prototyping in the field of sheet metal forming.

Some classic sheet metal forming processes are stamping, drawing, and pressing. In these processes, commonly used in the industry, a blank of sheet metal is clamped around the edges, while the middle section is forced by a punch into a die to stretch the metal into the desired shape. A variant, called "deep drawing" because of the height of the formed part, is represented in Figure 2.1. The gap between the punch and the die has been exaggerated for illustration purposes. In this case, the sheet metal blank is clamped using a blank-holder subjected to a given pressure. Cups formed using this process are presented in Figure 2.2.

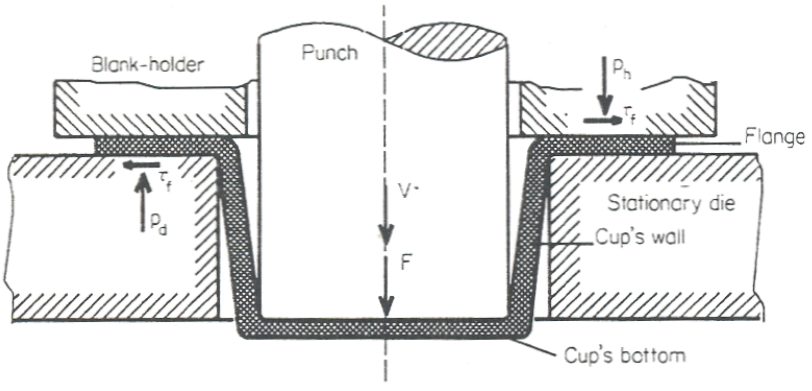


Figure 2.1: Deep drawing of a cup [TAL96, page 18]

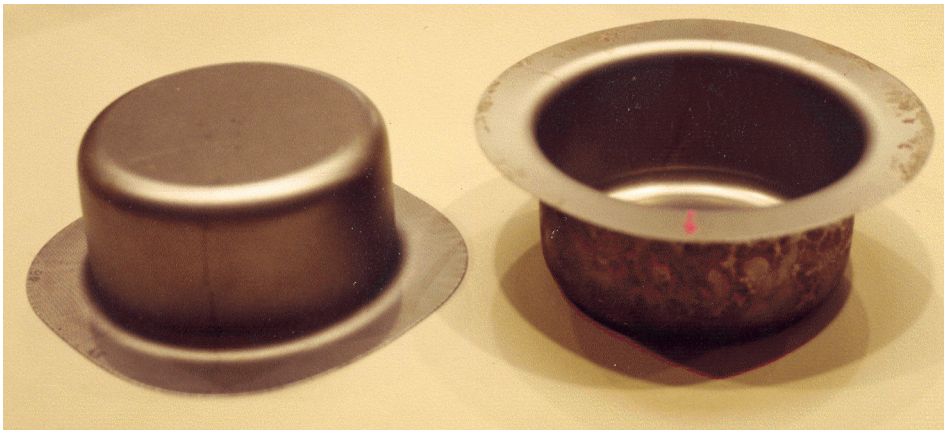


Figure 2.2: Cups produced by the deep drawing process; picture courtesy of Laurent DUCHÊNE (ULg)

For high volume production, this method is highly efficient since the production time of each part is considerably short and the high cost of the die and punch can be shared among a large number of parts. However, for prototyping or small batch production, the manufacturing time, as well as the cost, of the die and punch is a large burden.

2.1.2 Shear-Spinning Process

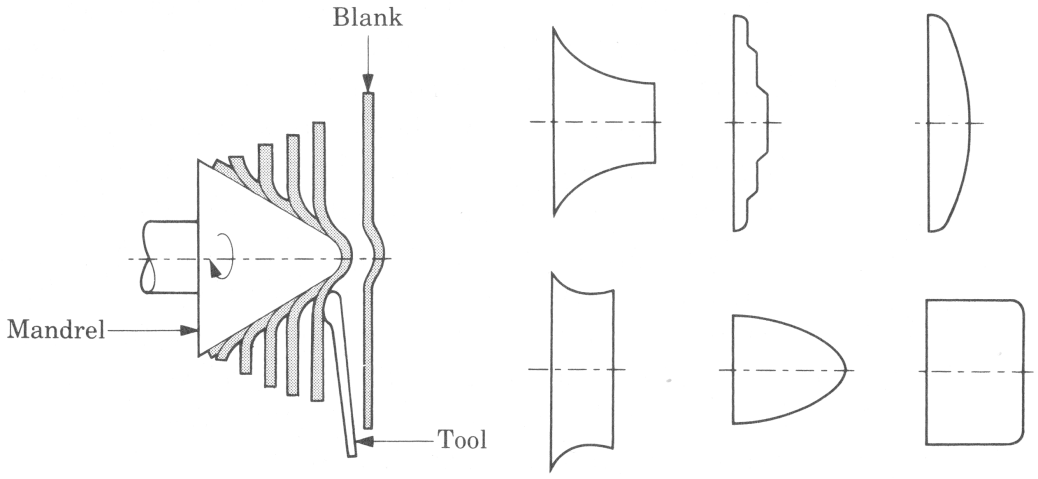
The shear-spinning process, a variant of the conventional spinning process, is often considered to be the ancestor of incremental forming [KIM00, HAG03]. Both processes have been in use for many years and "involve the forming of axisymmetric parts over a mandrel with tools or rollers. This process is somewhat similar to forming clay on a potter's wheel" [KAL95, page 480].

The conventional spinning process is illustrated in Figure 2.3*a*, along with some parts that can be made by using this approach. In his book, Kalpakjian described this process the following way:

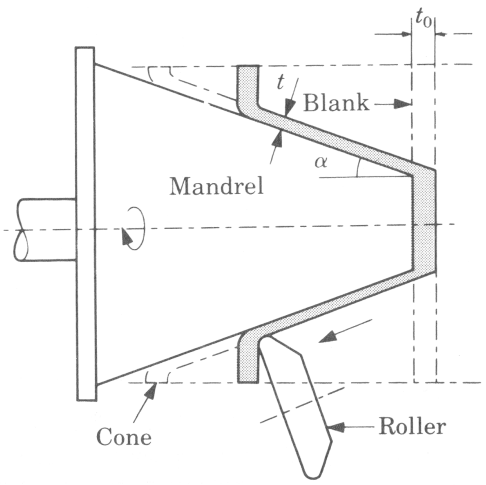
> A circular blank of flat or preformed sheet metal is held against a mandrel and rotated while a rigid tool deforms and shapes the material over the mandrel. The tools may be activated manually or by a computer-controlled hydraulic mechanism. The process involves a sequence of passes and requires considerable skill. Conventional spinning is particularly suitable for conical and curvilinear shapes, which otherwise would be difficult or uneconomical to produce. Part diameters may range up to six meters. Although most spinning is performed at room temperature, thick parts or metals with low ductility or high strength require spinning at elevated temperatures [KAL95, page 480].

The wall thickness of the formed part is, in theory, the same as the initial blank. More information about this process can be found in literature [SOR63].

More suitable for thicker materials, the shear-spinning process produces different results compared to those obtained by conventional spinning, as illustrated in Figure 2.3*b*. Kalpakjian described this other process as follows:



(*a*) Conventional spinning process



(*b*) Shear-spinning process

Figure 2.3: Description of the spinning process [KAL95, page 481]

> This process produces an axisymmetric conical or curvilinear shape while maintaining the part's maximum diameter and reducing the part's thickness. Although a single roller can be used, two rollers are preferable in order to balance the forces acting on the mandrel. Typical parts are rocket motor casings and missile nose cones. Parts up to about three meters in diameter can be formed. The operation wastes little material and can be completed in a relatively short time. [...] The spinnability of metal is defined as the maximum reduction in thickness to which a part can be subjected by spinning without fracture [KAL95, page 481].

Similarly to the conventional spinning process, more information about this variant can be found in literature [KOB61].

2.1.3 General Description

Unlike many other sheet metal forming processes, incremental forming does not require any dedicated dies or punches to form a complex shape and is therefore well adapted to rapid prototyping, as confirmed by several authors [DUF05a, HIR02, JES01b, LEA01, YOO03]. The process uses a standard smooth-end tool, the diameter of which is far smaller than the part being made, mounted on a three-axis CNC milling machine. A diagram of such a machine with a vertical axis is shown in Figure 2.4.

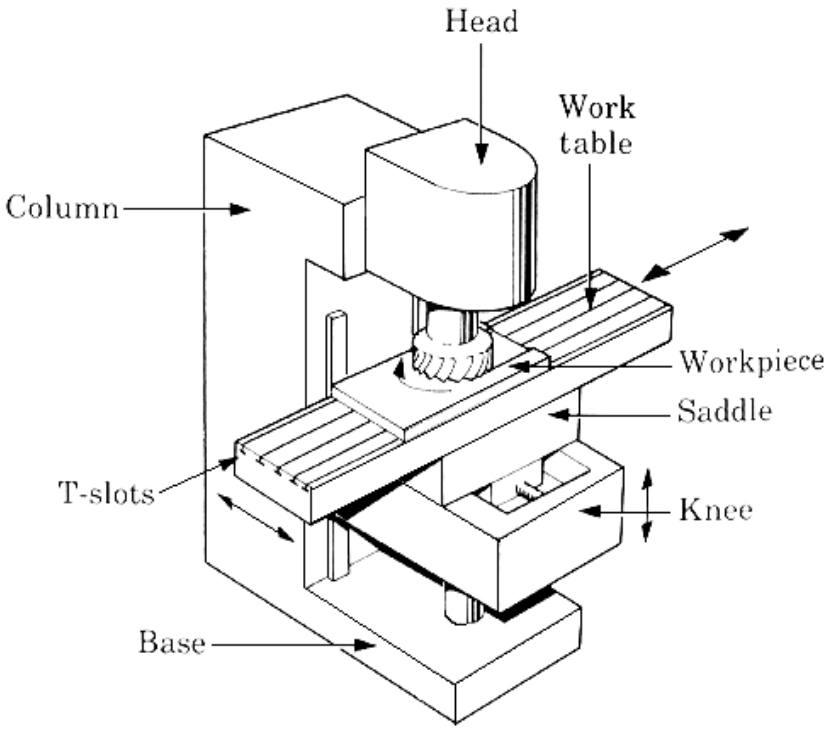


Figure 2.4: Vertical milling machine [KAL95, page 734]

The sheet metal blank is clamped around its edges using a blank-holder. During the forming process, the tool moves along a succession of contours, which follow the final geometry of the part, and deforms the sheet into its desired shape

incrementally. A schematic description of the process is shown in Figure 2.5 and an example of the tool path in Figure 2.6.

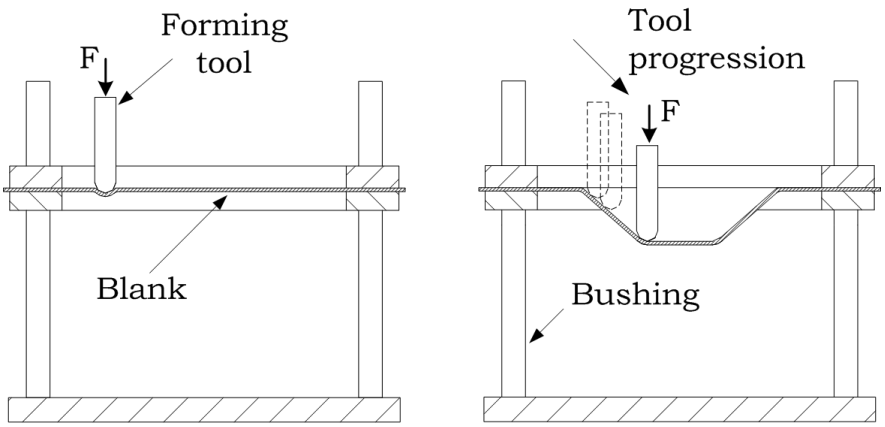


Figure 2.5: Single point incremental forming [DUF03]

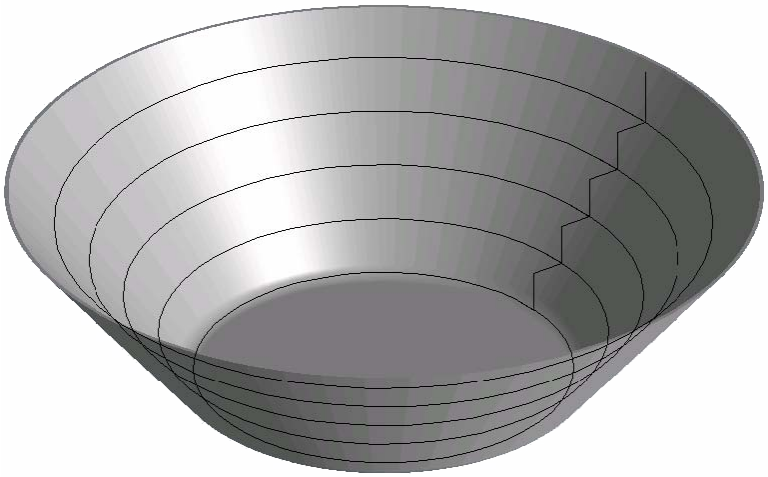


Figure 2.6: Tool path for a conical shape [HE05a]

The deformation is highly localized around the tool [HEN05c, HIR02]. As a result, the regions of the blank that have not yet been touched by the tool remain undeformed, even though they can undergo rigid displacement and rotation.

Another advantage of the process is that the deformation level that can be reached by the material is much higher than in conventional forming. More information about formability in incremental forming can be found in Section 2.2.

2.1.4 Experimental Setup

In addition to using a standard milling machine as shown in the previous section, other designs are possible, as described below.

In order to satisfy their own particular specifications, Allwood et al. designed their own incremental forming machine, which is presented in Figure 2.7 [ALL05].

Due to its particular design, this machine has the advantage of achieving a much larger axial load, up to 13 kN, which is not possible with a regular milling machine.

Another possible design is that used by Hirt et al., the purpose-built Amino machine, as shown in Figure 2.8 [HIR04b].

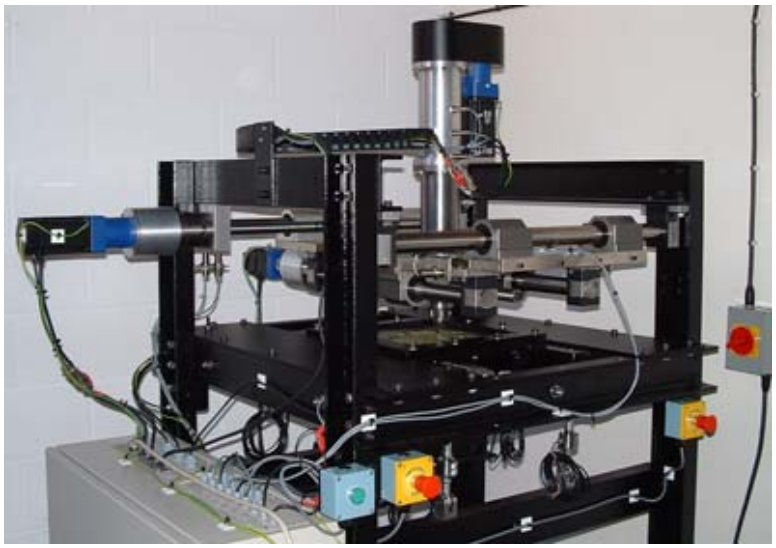


Figure 2.7: Allwood's SPIF machine [ALL05].

Figure 2.8: Purpose-built Amino machine [HIR04b]

Finally, some authors chose to use industrial robots, similarly to the one illustrated in Figure 2.9 [SCH04, LAM05]. The advantages of this approach over conventional forming are its flexibility and the possibility of combining several phases of the manufacturing chain, such as sheet positioning, forming, trimming, and surface treatments. Also, such robots usually have more than three degrees of freedom (DOFs). Therefore, they can change the orientation of the forming tool in addition to its position. These machines do not have to keep the forming tool perpendicular to the sheet metal at all times, as is the case with more classic

three-axis machines. This is especially useful for deep geometries with large wall angles because it can prevent a collision between the tool (or its holding fixture) and the sheet metal.

The major drawback of these robots is the fact that their arm is not as stiff as the frame of a milling machine, leading to a less accurate positioning, especially under high loading conditions.

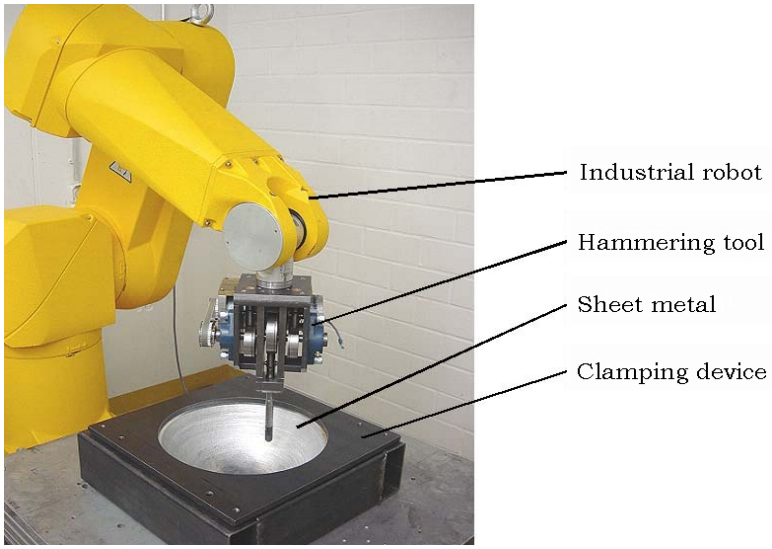


Figure 2.9: Industrial robot used for SPIF [SCH04]

2.1.5 Process Variants

a. SPIF

The process that was shown in Figure 2.5 is often called *single point incremental forming* (SPIF) in literature because there is theoretically only one point on the tool's surface in contact with the sheet metal. In practice, however, the contact zone extends to a small region of the tool, as shown by Eyckens, who performed a very detailed simulation of the contact [EYC08a].

Although this thesis focuses only on SPIF, other variants exist, as shown in the next sections.

b. TPIF

In addition to SPIF, another variant of the process exists called *two point incremental forming* (TPIF) for which a second tool or support is present on the opposite face of the sheet metal [JES01a, MAT01, SAW99, TAN99, JES05a].

In his PhD thesis, Shankar classified the TPIF processes into two main categories [SHA08, page 5]:

- TPIF with a static support;
- TPIF with a kinematic support.

For TPIF with a static support, there is a fixed partial positive die on the opposite face of the sheet metal with respect to the forming tool. This partial positive die

can either be generic, as in Figure 2.10, or specific to the shape to be produced, as in Figure 2.11. In the latter case, a new die must be manufactured for each type of part to be produced, which makes the process less flexible. For both cases, the support is fixed while the clamping system holding the sheet metal moves in the direction perpendicular to the tool.

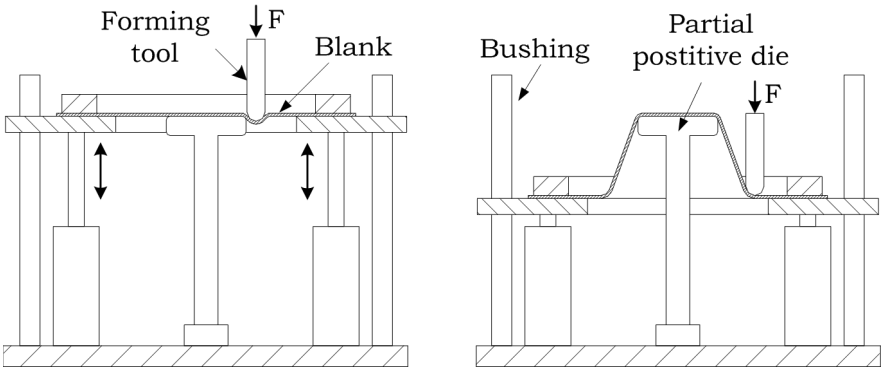


Figure 2.10: TPIF with a generic partial positive die [DUF03]

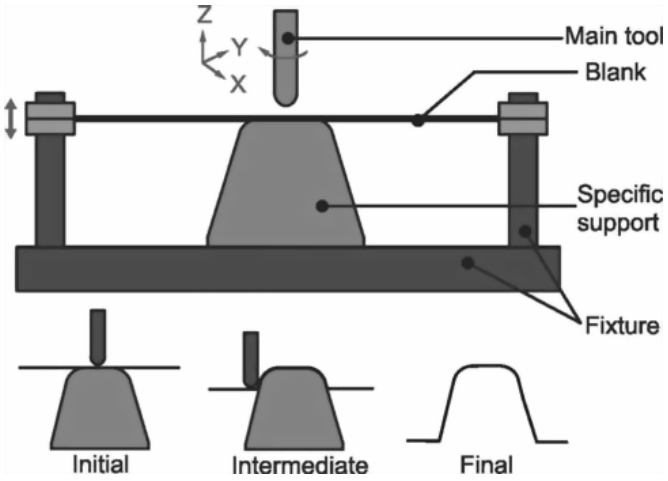


Figure 2.11: TPIF with a specific partial positive die [TEK07]

For TPIF with a kinematic support, the partial positive die moves at the same time as the forming tool. Several configurations for this variant exist in literature. Meier et al. designed a forming method in which two industrial robots move in a synchronized manner in order to form the sheet metal from both sides: in their first paper, one robot carried a forming tool and the other one, a partial support [MEI05]; in their second paper, both robots carried a forming tool [MEI07].

Maidagan et al. [MAI07] proposed a similar configuration having a tricept robot with a forming tool on one side and a three-axis moving table with a partial support on the other side.

Finally, Franzen et al. [FRA08] designed a unique system called “dyna-die”, which is shown in Figure 2.12. This system is composed of a forming tool and a moving partial support. This support is fixed on a rotating disk which moves at the same time as the forming tool on the other side of the sheet. This system has the advantage that it can be adapted to any single point incremental forming

platform and does not require two robots. However, it is only suitable for forming shapes with a rotational symmetry.

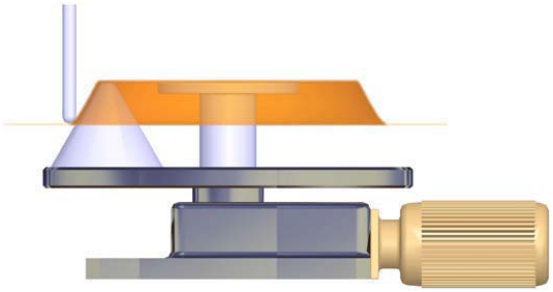


Figure 2.12: TPIF with kinematic support (dyna-die) [FRA08]

c. Laser-Supported SPIF

In addition to SPIF and TPIF, a third process variant, called laser-supported SPIF, was recently developed [DUF07a, DUF08a]. It was patented by Duflou et al. in 2006 [DUF06a].

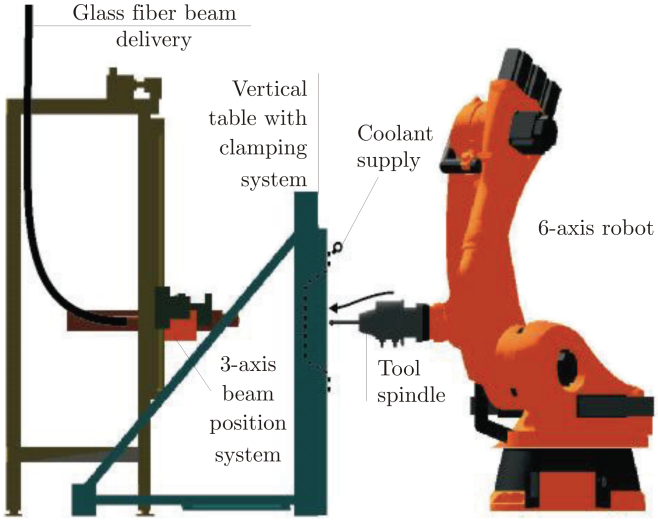


Figure 2.13: Laser-supported SPIF [DUF07a]

In this process, a six-axis robot forms the sheet metal on one side while a laser beam, mounted on a three-axis positioning system, heats the material dynamically under the tool from the opposite face of the sheet metal, as illustrated in Figure 2.13.

According to Duflou, "experimental results demonstrate that this process variant results in reduced process forces, improved dimensional accuracy and increased formability for a range of materials. Initial results also indicate that residual stresses can be significantly reduced by means of the dynamic heating system that was developed" [DUF07a].

2.1.6 Forming Tools

The tools used in incremental forming are rather simple, usually in the shape of a solid hard-steel hemispherical head mounted on a shaft. An example of this type of tool with a diameter of 25 mm is shown in Figure 2.14.

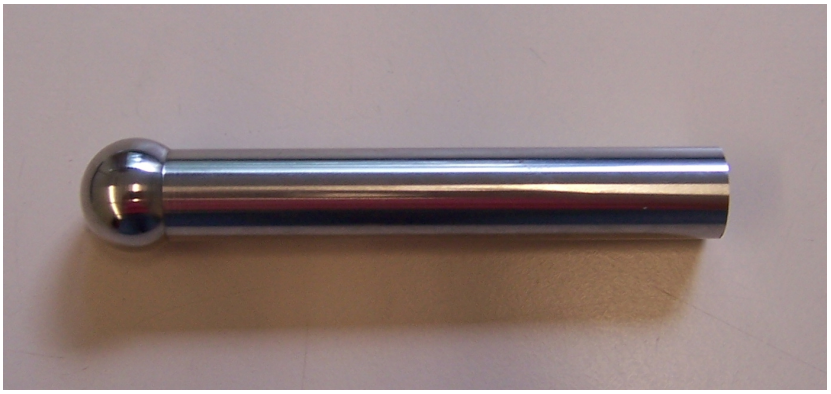


Figure 2.14: 25-mm tool; picture courtesy of Johan VERBERT (KUL)

In addition to this, other tool designs are also possible. Allwood et al. [ALL05] designed their own tool, as shown in Figure 2.15. Besides its primary advantage mentioned in Section 2.1.4, i.e., its large axial load capacity, it has the drawback that some parts are impossible to form due to the conical shape of the tool, such as parts with a steep wall angle.

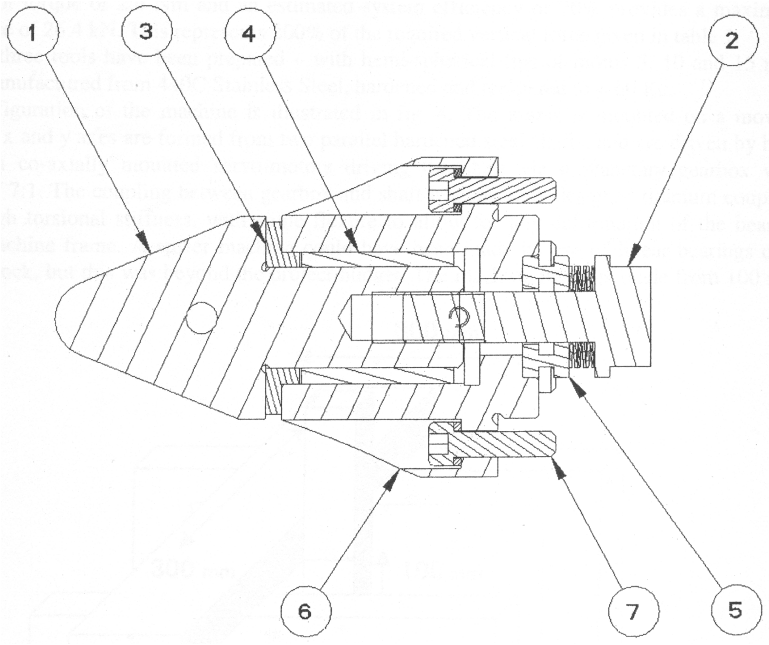


Figure 2.15: Allwood's tool; 1. tool, 2. preload screw, 3. thrust bearings, 4. radial bearings, 5. additional thrust bearing for the preload force, 6. tool-holder and 7. clamping bolts [ALL05].

Another interesting design was developed by Tanaka et al. who used, for TPIF, a second tool made of elastic materials [TAN99].

Finally, Hirt [HIR04b] used a universal tool head for which the ball is not attached to the shaft. Instead, the ball is supported by a high-pressure fluid and, as a result, can roll freely.

2.1.7 Force Measurement

During incremental forming, it is crucial to know the level of forces acting on the tool because these forces have an influence on the design of the tool, fixture, and machine. Indeed, experiments performed at the Catholic University of Leuven (KUL) and whose results were used in this thesis have been carried out on a milling machine which was not designed to work under a high level of axial force. Without particular care, the machine — or its bearings, at the very least — could be damaged. Moreover, the knowledge of the parameters governing the force level is an important step towards understanding the incremental forming process itself.

There are several ways of measuring the forces involved in incremental forming. Jeswiet et al. [JES05a] used three-strain-gauge Wheatstone bridges, one for each force component, mounted on the tool. For TPIF, an additional bridge was mounted on the support post.

Similarly, Attanasio et al. [ATT08] designed two devices to measure forces in TPIF: a strain gauge Wheatstone bridge mounted on the tool and a force platform. They performed a series of experiments in order to test and compare both systems and to study the influence of process parameters on the forces.

Allwood et al. [ALL05] used a force platform with six 10 kN load cells for the design of their incremental sheet forming machine.

Finally, Bologa et al. [BOL05] and Duflou et al. [DUF05b] chose a dynamometric table for their experimental research.

2.1.8 Limitations of Incremental Forming

The final thickness of a part formed by SPIF can be estimated using the *sine law*. This law was originally developed for the shear forming process, described in Section 2.1.2, and simply states that

$$t_f = t_0 \cdot \sin\left(\frac{\pi}{2} - \alpha\right), \tag{2.1}$$

where t_0 is the initial thickness, t_f the final thickness, and α the wall-angle measured with respect to the horizontal. This law is illustrated in Figure 2.16.

The sine law is based on the fact that the material is approximately in a plane strain state in the direction of the tool movement and is only able to move vertically toward its final position when starting from a horizontal flat sheet.

As shown in Figure 2.16, the comparison between the measured thickness and the one predicted by the sine law is very good. Matsubara [MAT01] also showed an accurate correlation in the thickness prediction of the wall of parts formed by TPIF.

Young and Jeswiet [YOU04] performed a more detailed analysis of the thickness and showed that this law is only correct on average but the thickness tends

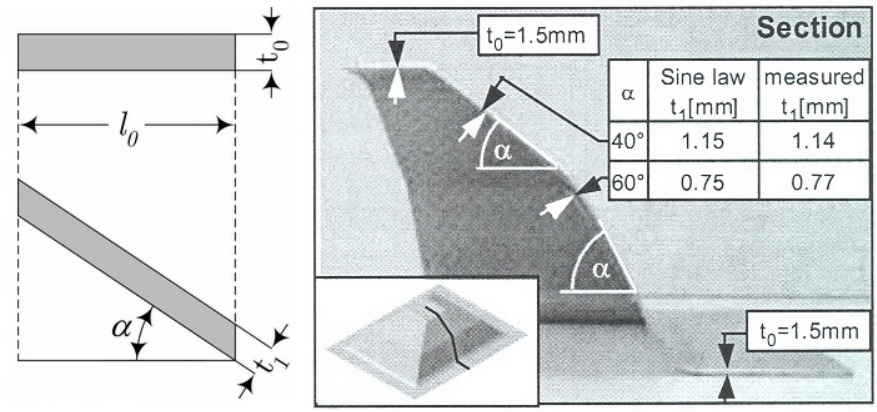


Figure 2.16: Prediction of the thickness in SPIF using the sine law [HIR02]

to be slightly thinner in some regions. This phenomenon, which is often called "overforming", was confirmed by Duflou et al. [DUF08b].

A direct consequence of the sine law is that the thickness decreases as the angle increases, as illustrated in Figure 2.17. In particular, the theoretical thickness of a part with vertical walls would be zero. The wall angle is a major limitation of SPIF. As an example, Duflou et al. [DUF08b] performed a large series of experiments in order to find the maximum wall angle of several materials. Their results are summarized in Table 2.1.

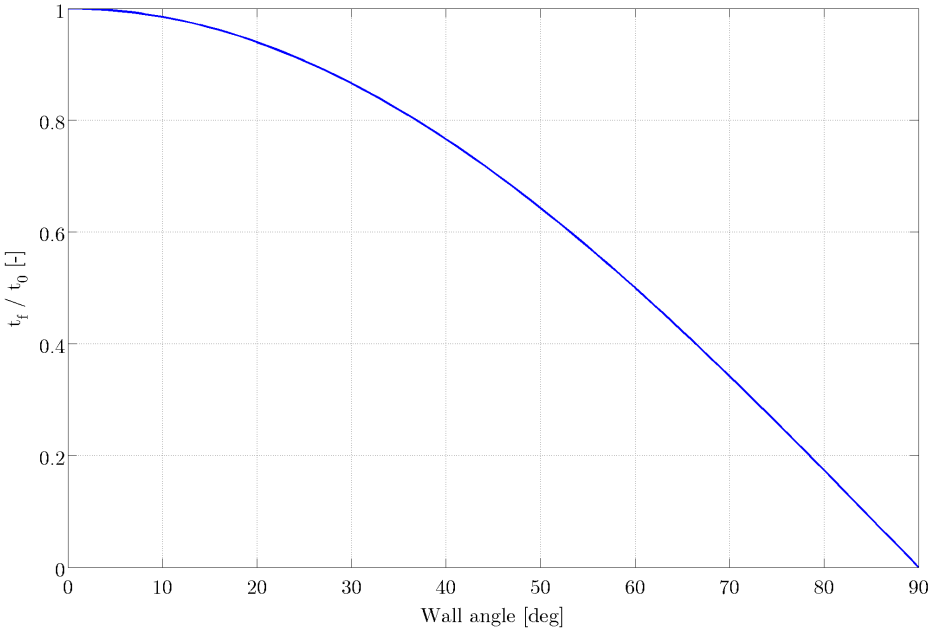


Figure 2.17: Thickness reduction in SPIF as predicted by the sine law

The limitation in the maximum wall angle is only valid for parts formed in one pass, i.e., when the tool follows the shape of the final geometry. Other strategies have been developed to overcome this limitation.

The first papers reporting the forming of a part with vertical walls with in-

Material	Thickness [mm]	Max. wall angle [°]
AA3003-O	1.2	71
AA3003-O	2.0	76
AA3103	0.85	71
AA3103	1.5	75
Ti Grade 2	0.5	47
DC01	1.0	67
AISI 304	0.4	63

Table 2.1: Maximum wall angles for cones made with SPIF with a tool diameter of 10 mm [DUF08b]

cremental forming were written by Kitazawa and Nakajima in 1997 and 1999 [KIT97, KIT99]. The authors were able to form a cylindrical cup using TPIF with a partial positive die. To do so, they developed a multistage strategy using conical intermediate shapes. Iseki also produced a very similar part with TPIF requiring two forming stages [ISE08].

Bambach et al. [BAM04, HIR04a] followed the same idea in 2004 and were able to form a square pyramid with a wall angle of 81 degrees with a similar multistage forming. This part is shown in Figure 2.18*m*. A year later, they were able to form a square pyramid with vertical walls [HIR05].

Iseki and Naganawa [ISE02] produced a shallow square box with a vertical wall using one-pass classic incremental forming followed by two additional forming stages: one to produce right angles and another one to flatten the bottom of the part. This method, however, required special tools, a significant disadvantage.

In 2008, two research teams published papers describing the strategy to produce cylindrical parts with SPIF without the need for special tools. Duflou et al. [VER08, DUF08b] developed an advanced strategy involving several intermediate shapes and were able to produce a cone with vertical walls using SPIF. Their idea was to form cones successively with a constant depth but increasing wall angles. This means that the region of the cone where the thickness is reduced is not limited to the vertical wall but extends to a larger region. It can be explained as if some material had been "moved" from regions that would otherwise have been untouched by the tool if it had formed the final cone in one pass.

Skjoedt et al. [SKJ08a] used a similar strategy but with slightly different intermediate shapes. In addition, they studied the influence of the direction of the tool movement (upwards or downwards) during the forming of the intermediate shapes on the thickness distribution.

2.1.9 Sample Parts

A wide variety of parts can be made with the incremental forming process. In order to illustrate these possibilities, a number of test parts are shown in Figures 2.18*a* to 2.18*r*. Certain characteristics make these parts difficult to manufacture using traditional methods:

- large wall-angles, e.g., (*b*), (*c*), (*e*), (*m*), and (*r*);
- changing slopes, e.g., (*f*), (*i*), (*j*), (*k*), and (*o*);
- inward curvature, e.g., (*e*), (*j*), and (*l*);
- asymmetric shapes, e.g., (*d*), (*e*), (*f*), (*g*), (*h*), (*i*);
- small details, e.g., (*o*), (*p*), and (*q*).

Incremental forming can also produce parts that can be used in the medical domain, as well as for aeronautic and automotive industries. In that case, a good surface finish is required. Examples are shown in Figures 2.19 and 2.20.

2.2 Formability in SPIF

2.2.1 Increased Formability

As stated before, the deformation level that can be reached by the material during SPIF is much higher than in conventional forming. In particular, several authors reported that the classic forming limit diagram (FLD) is not able to predict the failure of parts formed using incremental forming since a much larger deformation can be reached [YOU05, LAM05, MEY05].

The increased formability of incremental forming processes started to appear in literature at the beginning of the nineties and is still an important topic today. In order to measure the formability, Iseki developed a process called incremental sheet metal bulging, which he used to derive an empirical FLD for SPIF [ISE93, ISE94, ISE00, ISE01].

After Iseki, several other authors also published articles related to the development of similar types of FLDs based on an experimental investigation of the fracture occurring during the forming of various shapes [SHI01, FIL02, JES05d, BAM07]. Each shape would produce a set of fracture points from which an empirical FLD was derived.

Several possible explanations of this phenomenon of higher formability can be found in literature. They will be investigated in detail in the paragraphs below.

a. Strain Path

First, it was suggested that this formability was due to the *non-monotonic, serrated strain paths* to which the material is subjected during the SPIF process. An example of such a strain path is shown in Figure 2.21. This figure represents the evolution of the radial strain (approximately equal to the maximum principal strain) as a function of the tangential strain (minimum principal strain) of an element located in the region of the wall of a cone on the top layer of brick elements. This strain path evolution is due to the fact that the tool, which has a small radius and a limited influence around it, will approach and move away from the element several times, progressively increasing the level of plastic deformation.

Classic FLDs are derived from monotonic loading, which are very different from the loading occurring in SPIF, and are therefore not able to predict the onset of fracture [EYC07]. In order to prove this, van Bael et al. [VBA07] successfully

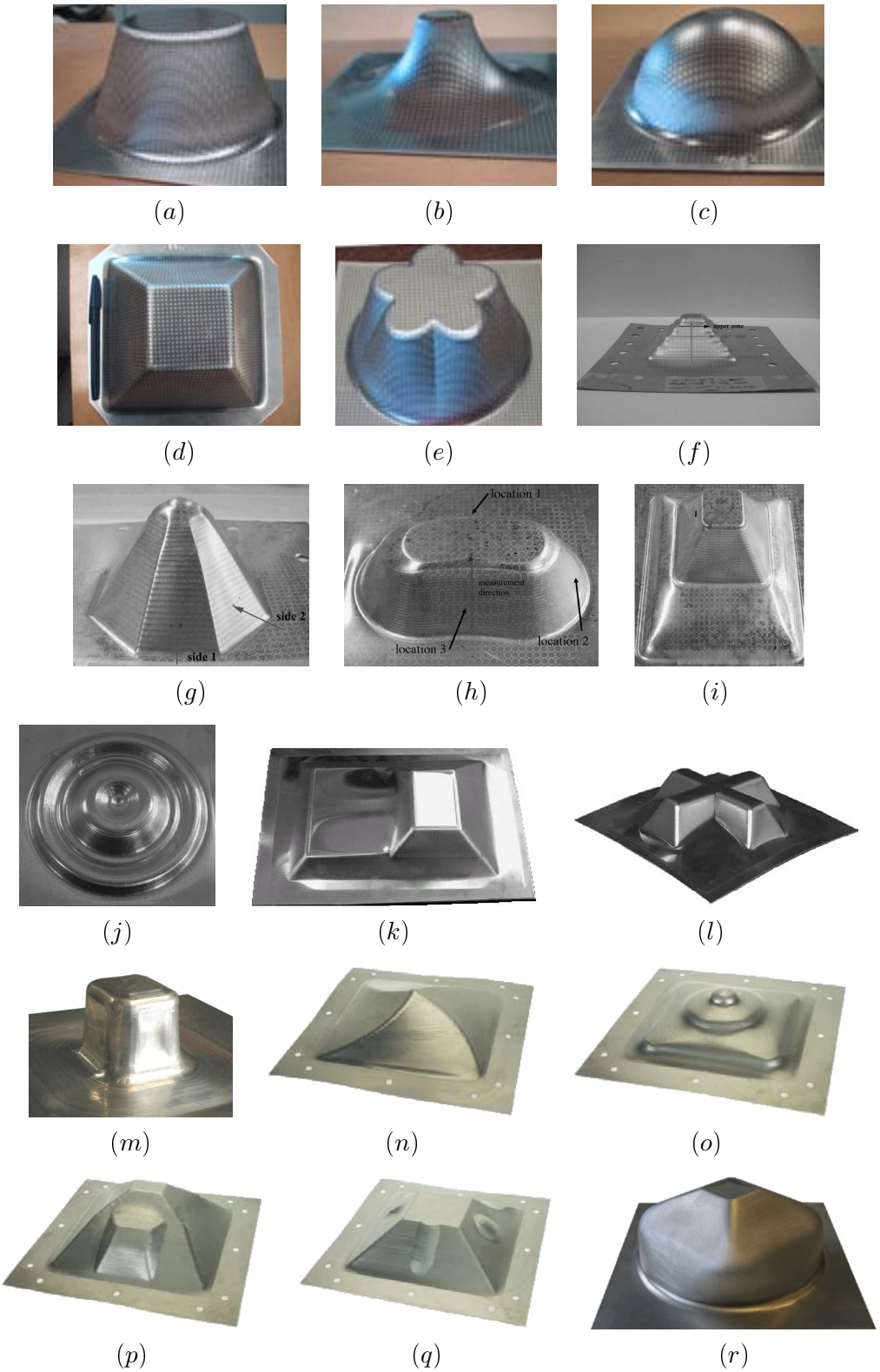
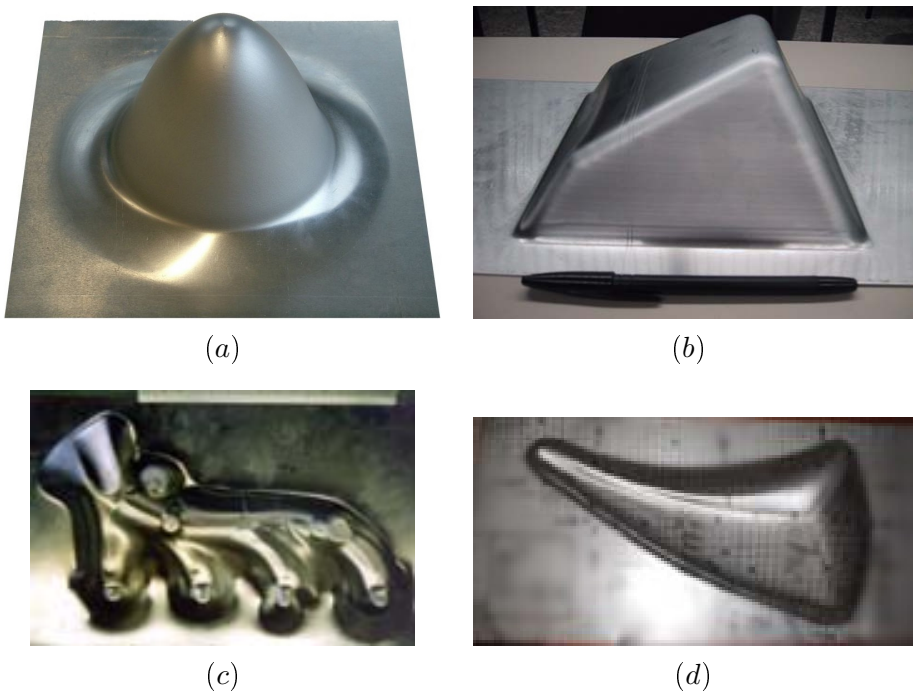


Figure 2.18: Parts used to test the capabilities of incremental forming; sources: (a)–(e) [JES05d]; (f)–(j) [PAR03]; (k)–(l) [AMB04b]; (m) [BAM04]; (n)–(q) [SCH04]; (r) [DUF08b]

(*a*) (*b*) (*c*)

Figure 2.19: Medical parts; (*a*) CAD model of an ankle support and (*b*) final product [AMB05a]; (*c*) cranial plate [DUF05a].



(*a*) (*b*)

(*c*) (*d*)

Figure 2.20: Industrial parts; (*a*) headlight reflector of a vehicule [JES01b]; (*b*) solar cooker [JES05b]; (*c*) automotive heat/noise shield and (*d*) motorbike gas tank [JES05c].

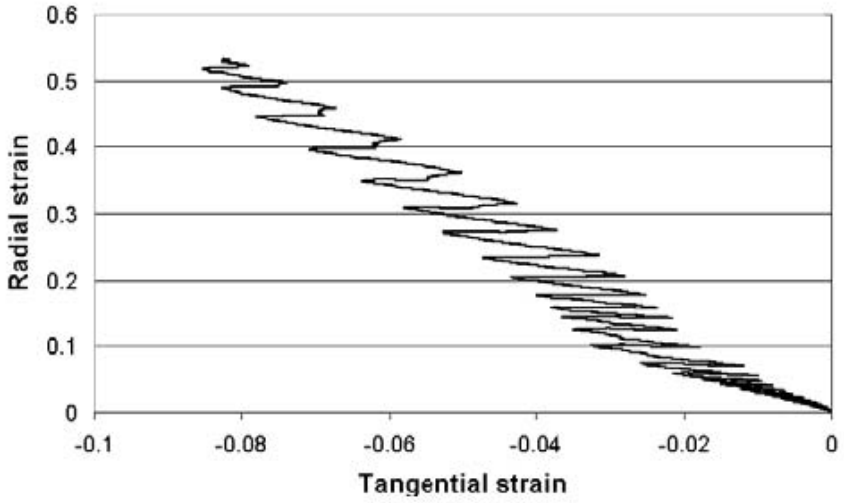


Figure 2.21: Strain path during SPIF [FLO05]

used a Marciniak-Kuczynski (MK) model to predict the onset of necking of a sheet subjected to the strain paths obtained by finite-element simulations.

b. Through-Thickness Shear

A second possible explanation which has often been mentioned as a likely phenomenon to postpone failure during SPIF is the *through-thickness shear*. Figure 2.22 shows the definition of a local reference frame centered at the tool contact point that will be used in the next sections to illustrate this shear phenomenon. In this figure, the direction of the tool movement during one contour is noted t, the direction of the normal to the sheet surface is noted n, and the direction g is perpendicular to t and parallel to the wall.[1]

Emmens and van den Boogaard [EMM07] published an interesting article about the *shearing mechanism* that is supposed to occur during SPIF and on which the sine law is based. This shearing mechanism occurs in a plane containing the directions n and g illustrated in Figure 2.22. In particular, their study concentrated on the implications of this mechanism on the determination of strains. Their conclusion was that a classic FLD is based on the fact that, during a tensile test, failure occurs because of necking. When a material undergoes shearing, "it may cause a macroscopic behavior in which necking does not occur or, at least, at levels other than in flat stretching" [EMM07].

Allwood et al. [ALL07] developed a simplified version of incremental forming called paddle forming. By modeling this process with a finite element (FE) method, the authors were able to demonstrate that the through-thickness shear is significant in the direction of the tool movement. This phenomenon is due to the fact that the material on the top surface of the sheet metal tends to move in the direction of the tool displacement because of friction. This shear mechanism occurs in a plane containing the directions n and t defined in Figure 2.22. Based on their measurement, Allwood et al. incorporated this shear component into an

[1] g stands for generatrix because, when the part is a cone, this direction is parallel to the generatrix of the cone.

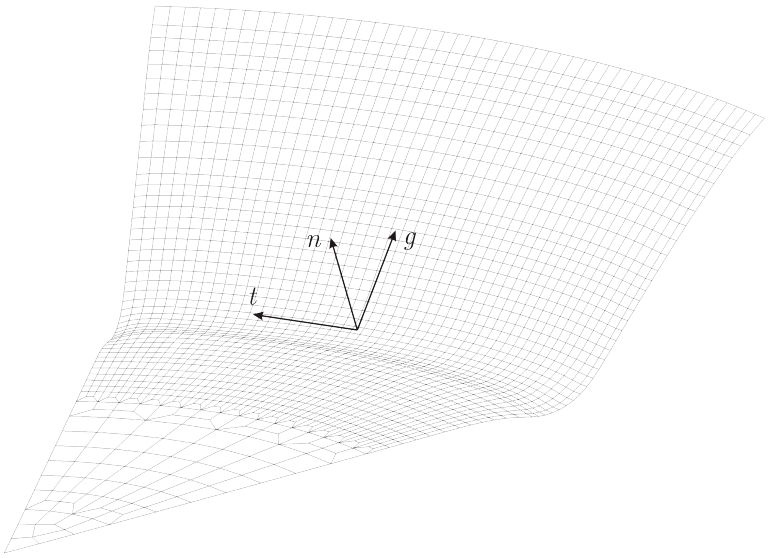


Figure 2.22: Definition of a local reference frame used to illustrate the through-thickness shear

MK model and showed that the forming limit curves increase with increasing thickness shear.

Similarly, Eyckens et al. [EYC08b] developed an MK model incorporating through-thickness shear in two planes, perpendicular to the in-plane major strain direction (plane containing the directions n and g) or the in-plane minor strain direction (n and t), respectively. The authors showed that only the shear component in the plane perpendicular to the minor strain has an effect on the FLD, which is again in the direction of the tool movement. In addition, they measured the through-thickness shear experimentally occurring during SPIF by drilling small holes into the sheet prior to forming and by analyzing their orientation at the end of the forming process. This study showed a significant amount of shear which, used with the MK model, predicted an increased formability.

c. Bending-Under-Tension

Another phenomenon, called *bending-under-tension*, has been proposed by Emmens and van den Boogaard [EMM08b, EMM08a] to account for higher formability. The conclusion of their study was that stretching with simultaneous bending does allow large uniform straining.

d. Triaxiality

Finally, Martins et al. [MAR08] performed an analytical computation of the strains which showed that the triaxiality ratio occurring during SPIF is smaller than that during a classic stamping process. This ratio is known to play a role in the formability of materials. They also showed that this ratio is larger in the corners of the parts than in the flat walls, which could explain the reason why the parts with corners always fracture in these regions.

e. Conclusions on the formability in SPIF

Seeing that these topics have only recently appeared in literature, there is no definite answer as to which one causes the very large formability occurring during SPIF. The final answer is probably a combination of two or more of the above-mentioned phenomena. One of the most likely is the through-thickness shear since it has been experimentally and numerically verified by Eyckens et al. The bending-under-tension mechanism proposed by Emmens and van den Boogaard is also very promising. Indeed, the material under compression due to bending probably delays the appearance of damage in the material, which increases the strain level that can be reached.

2.2.2 In-Process Failure Prediction

In addition to the development of FLDs, some authors developed alternative methods to predict part failure during the process. In particular, Ambrogio et al. [AMB06] developed an innovative approach based on force measurement. Their method is based on the observation that, during the forming of a cone, the force usually increases until it reaches a peak, then decreases before reaching a steady state.[2] This peak is due to two counteracting effects: the thinning of the material, which tends to reduce the forming force, and the material hardening, which increases it.

They also observed that, when a part is on the brink of failure, the force never reaches a steady state but instead decreases rapidly after the peak until failure. Therefore, they designed a failure criterion based on the force gradient after the peak, which is only a function of the sheet thickness.

2.3 Finite Element Modeling

This section will examine the state of the art in FE modeling of incremental forming.

2.3.1 Implicit or Explicit

The FE method is an analytical tool that can compute the solution of a continuum problem by reducing it to a system of algebraic equations [BATH76, chapter 3]. Very often, the governing equation is an equilibrium equation which can be, depending on the type of problem, either static or dynamic.

The integration of this equation over time is numerically computed and provides a solution at discrete times. Two time-integration schemes can be used: an *implicit* or an *explicit* one.

[2] Duflou et al. showed that the peak in the forming force only appears for cones with a large wall angle. When the wall angle is too small, i.e., smaller than 50 degrees for a cone made of aluminum AA3003-O, the force simply increases until it reaches its steady-state value [DUF05b].

Implicit For the implicit integration scheme, an assumption is made as to the values of the nodal velocities at the beginning of the time step. The nodal positions at the end of the time step can then be extrapolated. These positions are related to the strains, which provide the stress value using the constitutive law. The integration of the stresses over the volume of the elements, i.e., the internal forces, must satisfy the equilibrium equation. For this reason, the initial assumption on the nodal velocities must be corrected, which requires several iterations in order to converge to the equilibrium.

The advantage of this method is that it is unconditionally stable, meaning that it can provide the correct solution to the equation regardless of the size of the time step. The drawback is that it requires several iterations before converging to the exact solution. Moreover, at each iteration, a large linear system must be solved. Each time step requires a considerable amount of computation and is therefore time-consuming.

Explicit By contrast, the explicit integration scheme starts by computing the equilibrium equation at the beginning of the time step. This requires the computation of the internal forces using the strains and constitutive laws as was done for the implicit algorithm. The disequilibrium between internal and external forces provides the initial nodal accelerations. Finally, the nodal positions at the end of the time step are extrapolated using the initial nodal positions and accelerations.

With the explicit method, the equilibrium equation is never completely satisfied. This method is conditionally stable, which ensures that the remaining disequilibrium will remain insignificant provided that the amplitude of the step is kept smaller than a critical value. This value depends on the smallest size of the elements, on the elastic parameters (Young's modulus and Poisson's ratio), and on the specific mass of the material. This means that the maximum time step is always minute, leading to a large number of time steps before reaching the end of the simulation.[3] However, the computation of the initial nodal accelerations does not require any system solution. Each time step is therefore computed relatively quickly.

In general, the mass of the system can be artificially increased, leading to a larger critical time step, without a significant loss of precision, provided that the kinetic energy of the system remains much smaller than the internal energy, e.g., less than 5 to 10 % [AMB05b]. This "trick" is called *mass scaling*. Another trick, called *time scaling*, can be used to increase the speed of the tool artificially, which also decreases the total computation time. Unlike mass scaling, time scaling can only be used with materials that are not viscous, i.e., for which the stresses are

[3]The value of the critical time step depends on the units chosen for the lengths, the forces and the time. For example, if the lengths are in millimeters, the forces in Newtons, and the time in seconds, the order of magnitude of the critical time step is 10^{-7}s, for a mesh of an aluminum sheet metal with a characteristic mesh length of 1 mm. This was computed using the formula

$$\Delta t_c = L_{min} \cdot \sqrt{\frac{\rho}{E}},$$

where Δt_c is the critical time step, L_{min} the characteristic length of the smallest element of the mesh, ρ the specific mass, and E Young's modulus.

independent of the strain rate.

In literature, most of the authors using commercial FE codes claim that the explicit integration scheme is faster than the implicit one. The remaining paragraphs of this section will show various examples comparing explicit and implicit approaches.

Bambach et al. [BAM05a] used Abaqus/Standard, which uses an implicit time-integration scheme. In this article published in the SheMet2005 conference, it is mentioned that, on average, 44 % of the iterations performed during one time step resulted in a "severe discontinuity iteration" in order to achieve a consistent contact state. This indicates that the computation time of SPIF simulations using an implicit strategy is increased largely by the continuously alternating contact conditions, preventing the code from using large time increments.

In two different articles published in 2005, Bambach et al. [BAM05c, BAM05b] compared both the implicit and the explicit integration schemes for a cone simulation with a short tool path, shown in Figure 2.23, performed on Abaqus/Standard and Abaqus/Explicit with shell elements. This tool path is not classic for the forming of a cone since it does not follow the final geometry. Instead, the contours get larger and larger until the last one, which follows the bottom of the cone. Moreover, the tool path has large vertical stepdowns of 5 mm, instead of 0.5 to 1 mm, which are more commonly used. Experimentally, such a tool path produces a poor quality part as far as the surface finish is concerned. In addition, the large stepdowns induce large axial tool forces, which can not be sustained by standard CNC milling machine. The purpose of this tool path was to be as short as possible in order to reduce the computation time of the simulations.

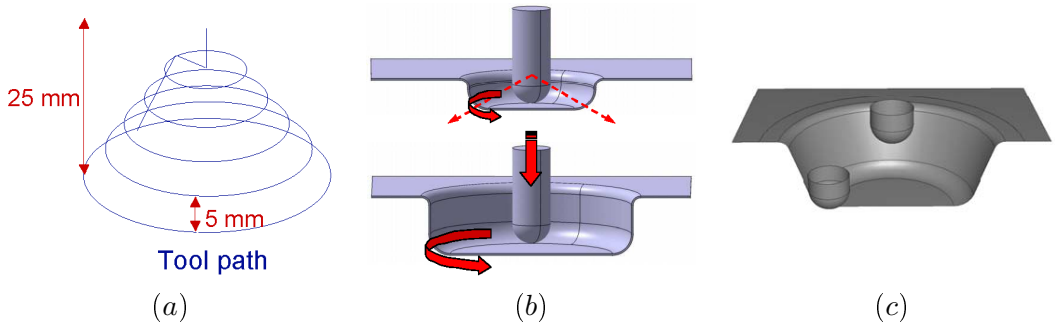


Figure 2.23: Bambach's cone with a short tool path; source: (*a*)–(*b*) [BAM05c], (*c*) [BAM05b]

The results of these simulations are summarized in Table 2.2. For both simulations, the deviation with respect to experimental measurements was computed in terms of sheet geometry and thickness. In this table, d_{max} is the maximum normal distance between the simulated and measured shapes, d_{av} is the average normal distance, and $d_{th,max}$ is the maximum deviation in thickness. The accuracy of the explicit simulation was of poorer quality than the implicit one. However, the CPU time was significantly reduced. This is because the time step in explicit Δt was artificially increased using mass scaling to a value close to the time step in implicit. The value of the mass-scaling factor was not given in the article.

Strategy	Δt [s]	d_{max} [mm]	d_{av} [mm]	$d_{th,max}$ [%]	CPU time [h]
Implicit	0.004	1.09	0.59	11.7	7.32
Explicit	0.0001	1.82	1.19	15.6	0.58

Table 2.2: Numerical comparison between implicit and explicit simulations [BAM05b]

Qin et al. [QIN05] used LS-Dyna with an explicit time-integration scheme to simulate the forming of a cone with SPIF and analyzed the influence of mass and time scaling. On the one hand, they used a maximum mass-scaling factor which multiplied the time step by a factor of two (compared to the case without mass scaling). With such a small factor, the computation time was only halved and had a negligible influence on the quality of the results, assuming that the simulation without mass scaling is the reference. On the other hand, they used a time-scaling factor up to 5000. Even with such a large factor, only a small deviation of approximately 0.2 mm was observed in the prediction of the geometry and the authors achieved a significant reduction in the computation time. However, for this simulation, they only simulated a small indent of the tool 4 mm inside the material instead of the whole SPIF simulation.

Skjoedt et al. [SKJ08b, SKJ08a] also used LS-Dyna with an explicit time-integration scheme. They used a time-scaling factor between 1000 and 1500 without additional mass scaling and were able to reduce the computation time significantly.

Ambrogio et al. [AMB04a] also used an explicit integration scheme with a commercial FE code. They used time scaling to speed up the simulations and increased the tool velocity to 5 m/s. In a separate article [AMB05b], the authors multiplied the feed rate by a factor of 40 and used a velocity of 40 m/s. They verified that the ratio between the kinetic energy and the total energy was always under 10 %.

Robert et al. [ROB08] used Abaqus/Explicit without mass scaling but used time scaling with an increased tool speed of up to 25 m/s. They also kept the energy ratio under 10 %.

Kim et al. [KIM08] also used Abaqus/Explicit to simulate incremental forming at an elevated temperature. The authors did not use time scaling because they used a material law taking the strain rate into account. On the other hand, they came to the conclusion that a mass-scaling factor of 100 is the best compromise between simulation speed and accuracy.

Dejardin et al. [DEJ08a, DEJ08b] also used LS-Dyna in an explicit strategy to form a cone. The main difference with the other authors is that after the forming stage, Dejardin et al. performed a springback analysis both numerically and experimentally. To do this, they sliced the cone into several rings. Then, they cut these rings open and observed how their shapes had changed. To simulate this elastic springback phenomenon, they used an implicit time-integration scheme with the same FE code and claimed that it was more accurate than an explicit one.

Lequesne et al. [LEQ08] used an implicit time-integration scheme but developed an adaptive remeshing technique to reduce the computation time. The

strategy adopted was to start the simulation with a coarse mesh and refine the elements that were close to the tool. These elements can be coarsened if several conditions are met: the tool must be distant from them, their curvature must be small and their stresses and state variables uniform. These conditions ensure that little information will be lost during coarsening.

Hadoush and van den Boogaard [HAD08] also used an implicit strategy with a refinement and coarsening technique.

The conclusion from this bibliographic study on the the choice between an implicit and an explicit time-integration scheme is the following. For the forming stage, all the authors using a commercial FE code claimed that it is possible to reduce the computation time without significantly deteriorating the quality of the results. For this, they used an explicit scheme with an appropriate mass- or time scaling instead of an implicit one. This will be the main argument used to justify the new developments in the field of the dynamic explicit integration scheme in the Lagamine FE code, as will be explained in Chapter 4.

Several authors had used simulations performed with an implicit strategy as a reference to validate the results obtained with the explicit strategy and to ensure that the mass- and time-scaling factors are not too large.

A different solution adopted by some authors to reduce the computation time is to use an implicit strategy with adaptive remeshing. This feature, however, is not available in every FE code.

Finally, it is advisable to use an implicit scheme to perform a springback analysis.

2.3.2 Finite Element Type

The element type itself can greatly influence the quality of the simulation's results. Bambach and Hirt [BAM05c] tested several element types available in the commercial FE code Abaqus using the same cone simulation as presented in the previous section in Figure 2.23. Their results are summarized in Table 2.3 with the same notations as in the previous section. All the element types starting with C3D8 are eight-node brick elements, whereas the element type S4R is a four-node shell element. The difference between all the brick element types comes from their choice of anti-hourglass and anti-shear-locking modes. More details can be found in Abaqus's user's manual [ABA03].

The best results were obtained with the shell element S4R, despite having assumed a plane-stress state. Moreover, this element corresponds to the fastest CPU time. One of the reasons for this conclusion is probably the fact that only two elements were used across the thickness for the simulations with brick elements, causing a loss of accuracy due to an inaccurate bending stiffness. On the other hand, using more elements across the thickness would increase the computation time further.

Most other authors also claim that shell elements are necessary to reduce the computation time. For instance, Skjoedt et al. [SKJ08b] used LS-Dyna fully integrated shell elements. Dejardin et al. [DEJ08a, DEJ08b] used LS-Dyna reduced integration shell elements with an adaptive mesh refinement.

Element	d_{max} [mm]	d_{av} [mm]	$d_{th,max}$ [%]	CPU time [h]
C3D8	2.59	1.54	14.52	17.6
C3D8R	6.35	3.56	15.09	15.4
C3D8R enh.	2.04	1.07	14.23	28.3
C3D8H	2.25	1.54	16.65	48.1
C3D8I	2.18	1.19	17.08	34.9
C3D8IH	2.17	1.07	17.07	103.3
C3D8RH	6.37	3.56	15.29	50.9
S4R	1.09	0.59	11.72	7.3

Table 2.3: Comparison between several element types of Abaqus [BAM05c]

On the other hand, brick elements should be used when an accurate description of the stress state occurring across the thickness is important for the accuracy of the results or when the user wants to use a complicated mechanical law. The reason for this is that classic shell elements assume a plane-stress state across the thickness and can not use 3D mechanical laws.

One way to overcome these limitations is to use an advanced type of shell element called solid-shell. These elements are able to use any 3D constitutive law. More details about shell and solid-shell elements will be given in Section 3.1.2.

The influence of the constitutive laws on simulations' predictions will be examined in the next section.

2.3.3 Constitutive Laws

The constitutive law describing the material behavior consists of two components: the shape of the yield surface (yield criterion) and its evolution with respect to time (hardening).

Regarding the yield criterion, most authors use a relatively simple law, e.g., Von Mises's [HE05a, SKJ08a, QIN05, ROB08] or Hill's [BOU08b]. The reasons for this choice are that more advanced yield criteria do not bring about a significant increase in the accuracy of the prediction, require more computation, and are hampered by difficult identification of their parameters.

As far as hardening is concerned, its choice is very important for the force prediction during SPIF. Indeed, as was already presented in Section 2.2, the tool follows a complex tool path, which generates in each element a serrated strain path. This strain path shows a cyclic evolution, meaning that the material will be loaded and unloaded several times. In this case, a mixed-type hardening law (isotropic and kinematic) should be used in order to predict the stress state and the forming forces accurately.

To prove this, Bambach [BAM05b] tested both a simple isotropic and a mixed-type isotropic-kinematic hardening law on the cone simulation already presented in Sections 2.3.1 and 2.3.2. As evidenced by the results shown in Figure 2.24, the deformation simulated using the mixed-type hardening law is more accurate than the one using a simple isotropic law. As mentioned before, the tool path

used for this simulation was not conventional: given the large vertical pitch, it only contained five contours. If a standard tool path were used, the effect of the cyclic loading on the mixed-type hardening law would be even greater.

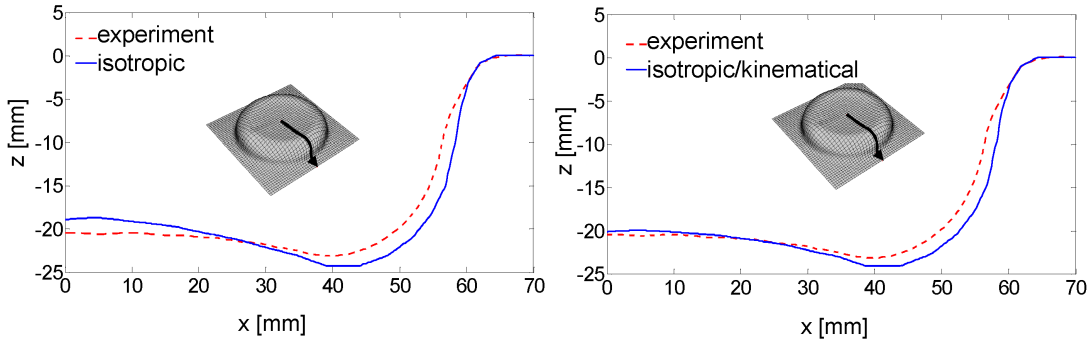


Figure 2.24: Comparison between isotropic and kinematic hardening [BAM05c]

He et al. [HE05a] also demonstrated that, when using an isotropic hardening law, the predicted force on the tool is consistently 20 to 30 % higher than that measured during experimental tests. Moreover, Flores et al. [FLO05] showed that using a mixed-type hardening law could reduce the predicted force by a similar factor.

To conclude this section about constitutive laws, it is important to observe that, even though the use of an advanced law is recommended for accurate force and springback prediction, these laws often require a large number of parameters. The identification of these parameters is quite difficult and can have a great influence on the results.

Bouffioux et al. [BOU08b, BOU08a] tested several procedures to identify the parameters of constitutive laws for incremental forming simulations. These procedures are based on an inverse method. The principle of this method is to choose a set of experimental tests whose results are sensitive to the material data to be adjusted. These tests are simulated using an initial set of material parameters, chosen arbitrarily, and the better this initial guess is, the faster the method will be. Then, the numerical results are compared with the experimental measurements and, using an optimization algorithm, the material data are iteratively adjusted until a sufficient level of accuracy is reached. This method offers the possibility of choosing a complex test to fit the parameters, inducing heterogeneous stress and strain states close to the ones reached during the process to be simulated with this material model.

In their articles, Bouffioux et al. compared the influence of the hardening law and the choice of the set of experimental tests on the quality of the material model obtained, especially, on the quality of the force prediction during a simple incremental forming test.[4]. To do so, the authors identified two hardening laws (isotropic and mixed-type hardening), each with two sets of material parameters using, on the one hand, classic tests only (i.e., a tensile test and a shear test),

[4]The simple incremental forming test used was a line test, which will be used later in this thesis, as in Section 4.1.1

and, on the other hand, a tensile test and an indent test.[5]. The conclusion was that the material model with a mixed-type hardening law identified with a tensile test and an indent test is more accurate than the other models.

2.4 Analytical Computations

Besides FE computations, several authors have tried to develop analytical computations to predict various quantities related to the incremental forming process. Some of these predictions will be analyzed in the next sections.

2.4.1 Strain Prediction

Predicting the strain of the final product is one of the most useful analytical computations. The main reason for this is that it gives an indication as to the feasibility of the part and helps localize the regions where failure is most likely to appear, modifying the tool path if necessary.

Kim and Yang [KIM00] developed a double-pass strategy to form a part with a more uniform thickness. To do so, the authors used the sine law, introduced in Section 2.1.8, to compute an approximation of the strain of the final product. From this approximate strain computation, they predicted to a certain degree the location of the zones having a high strain. They then used this computation to derive an intermediate shape having a large strain where a small one was observed in the final product, and vice-versa. Forming the part using this double-pass strategy led to a final product where the risk of failure was reduced.

Iseki [ISE01] developed an analytical model for strain prediction. This model is based on a plane-strain hypothesis (no deformation in the direction of the tool movement) and assumes that the sheet metal in contact with the forming tool stretches uniformly. The author also developed an analytical model for strain prediction during a multistage incremental forming of a cylindrical cup [ISE08]. Using this model, he was able to predict the thickness profile. However, no experimental verification was available at the time.

Martins et al. [MAR08, SIL08] developed a model of the fundamentals of SPIF based on a membrane analysis. From this model, the authors derived the stress and strain states occurring in flat surfaces, in rotationally symmetric surfaces, and in corners.

2.4.2 Force Prediction

The prediction of forming forces occurring during incremental forming is also a crucial piece of information. The main reason is that this governs the choice of the hardware setup. Indeed, most classic milling machines only tolerate a relatively limited axial force and exceeding this limit could potentially damage the machine or, at least, its bearings.

[5]An indent test is a test during which a spherical tool makes an indent in a clamped metal sheet. This is also the first step of the line test. This test is very useful for two reasons: it is performed with the same machine used for the SPIF process and generates a stress state in the material that is similar to the one occurring during SPIF

Several papers have been published about modeling forming forces as a function of experimental setup parameters. In a first paper, Duflou and Tunçkol [DUF06b] performed a series of experiments to study the influence of boundary conditions (distance between the part and the backing plate), the wall angle and the tool path. In particular, they studied pyramids with changing wall angles and proved that the local wall angle is not the only determining factor for the instantaneous forming force. As far as the tool path is concerned, they showed that having a spiraling or a stepdown tool path does not affect the forming forces, except for local peaks at the tool stepdown location.

Following this study, Duflou et al. [DUF07b] published a second paper in which they developed a model for force prediction during the forming of a cone as a function of the stepdown amplitude, the wall angle, the tool diameter and the sheet thickness. This model was a simple regression equation that could predict the peak, the steady-state, and the in-plane forces with a high degree of confidence. For more complex geometries, they could not reach such a simple equation but showed that the part geometry should be taken into account in the model.

This work was later continued by Aerens et al. [AER09]. They were able to develop a new model with a very efficient strategy to fit the parameters based on experimental measurements where several materials were tested. For each material, the authors fitted a law that was able to accurately predict the level of the steady-state tool force for various part geometries. They also fitted a general law valid for any material but in that case, the accuracy of the predictions was reduced. The ultimate tensile strength of a given material seemed to be the key material parameter governing the level of the forces.

2.5 Conclusions

The goal of this chapter was to give an overview of all the aspects of the SPIF process: its history, its principles, and its interests and limitations. In particular, one of the main contributions was the summary of the state of the art in the field of forming limit diagrams and concerning the increased formability observed during SPIF.

Since this thesis is mainly focused on modeling SPIF, a large part of this chapter was dedicated to its simulation techniques. With respect to FE simulations, the most effective technique adopted by many authors seems to be the use of shell elements for the mesh of the sheet metal, an explicit integration scheme, and a relatively simple constitutive law with a mixed-type isotropic-kinematic hardening law. These observations were the starting point of all the developments performed in the course of this thesis. In addition to these, a section was dedicated to analytical computations.

Chapter 3

Numerical Tools and Experimental Equipment

The purpose of this chapter is to describe the numerical tools and experimental equipment used to produce the results shown throughout this thesis.

The numerical tools include the FE code, the elements, the mechanical laws, the meshing tools and the post-processing tools. The latter tools are Matlab programs that were developed in order to be able to compare the results of several simulations and experimental results easily and to plot the curves in a consistent manner.

The experimental equipment consists of every piece of equipment needed for the incremental forming process: a machine, various forming tools, a clamping device to fix the metal sheet to the table of the machine, and measurement devices to extract useful data.

All the experimental tests of incremental forming and most of the measurements used in this thesis were performed by the team of Prof. Joost Duflou at the PMA Department of the Catholic University of Leuven. In addition, the team of Prof. Hugo Sol from the MEMC Department of the Free University of Brussels (VUB) performed the deformation measurements with a stereo camera system.

3.1 Numerical Tools

3.1.1 Finite Element Code: Lagamine

When starting a thesis in which a great portion of the work is dedicated to simulations, the choice of the FE code is important. For the present work, one of the goals was to find ways of reducing the computation time. Since this could mean reorganizing the code to best fit the needs of the process under investigation, it was crucial to have access to the source codes and to be able to modify everything. The self-made code Lagamine was chosen for these reasons.

Lagamine is a non-linear Lagrangian code that has been under development at the ArGEnCo Department of the University of Liège since 1982 and was started by Prof. Cescotto in order to simulate rolling [CES85].

In addition to rolling, the code has been applied to numerous other forming processes, such as forging [HAB90], continuous casting [CAST04], deep drawing [DUC05b], powder compaction [MOS99], cooling processes [CASO05], and incremental forming [HEN07a].

The code mainly focuses on modeling material behavior, especially metallic materials, and is suitable for large strains and large displacements. A large variety of constitutive laws are available, in particular in the fields of solid phase transformations [HAB92], recrystallization [HAB98b], damage models to predict crack appearance [ZHU92, ZHU95, CAST03], phenomenological laws with gradient plasticity [LI96], micro-macro and texture-based laws [DUC02, DUC03, HAB04b] or contact laws [CES93, HAB98a].

LAGAMINE has a large element library. Two particular elements are detailed in the next section. Other types can be found in Zhu and Cescotto [ZHU94]).

3.1.2 Elements

This section contains a detailed description of the two types of elements that are used in this thesis: a brick element and a shell element.

a. Brick Element: BWD3D

The following description of the BWD3D element is a simplified version of the description published by Laurent Duchêne in one of his papers [DUC07].

The BWD3D element, shown in Figure 3.1, is an 8-node 3D brick element with a mixed formulation adapted to large strains and large displacements. This element can be coupled with any 3D mechanical law and its stress and strain tensors are expressed in global axes.

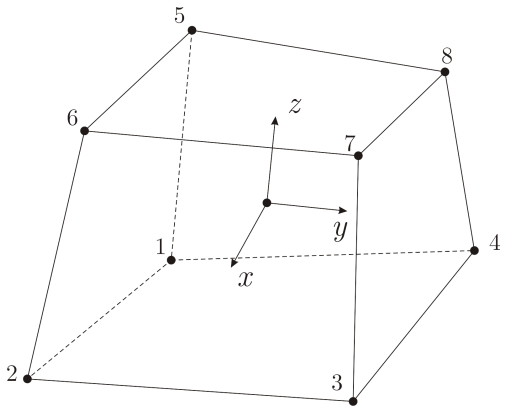


Figure 3.1: BWD3D brick element and its local axes

This element uses a reduced integration scheme (only one integration point in its center) and an hourglass control technique. It is based on the non-linear three-field[1] Hu-Washizu variational principle with the "assumed strain method" [SIM86, BEL91].

A first feature of the BWD3D element is a new shear locking treatment based on Wang and Wagoner's method [WAN04]. This method identifies the hourglass

[1]The three fields are the stress-, strain-, and displacement field.

modes responsible for the shear locking and removes them. The two bending hourglass modes and the warp (non-physical) hourglass mode are eliminated. The volumetric locking treatment is also based on the elimination of inconvenient hourglass modes.

A second feature of this element is the use of a corotational reference system. In order to be able to identify the hourglass modes, crucial to the method, the formulation of the element's kinematics must be expressed in a corotational reference system [BEL91], closely linked to element coordinates. This reference system must have its origin at the center of the element and its reference axes are aligned (as much as possible, depending on the element shape) with element edges. A fortunate consequence of this corotational reference system is a simple and accurate treatment of the hourglass stress objectivity, by using initial and final time step rotation matrices.

The shear locking and the volumetric locking methods proposed by Wang and Wagoner, together with the corotational reference system, have been successfully implemented in the BWD3D element. The Wang-Wagoner method, as opposed to other shear locking methods (see, for example, Li and Cescotto [LI97]), has deep physical roots, which makes it very efficient for various FE analyses. Further details about the hourglass and the locking treatments in the BWD3D element can be found in two articles by Duchêne et al. [DUC05a, DUC06]. Also, a more complete description of the rotation of the local axis can be found in a separate article by Duchêne et al. [DUC08].

b. Shell Element: COQJ4

LAGAMINE's shell element COQJ4 was originally developed for another FE code, FINELg, by Jetteur [JET85, JET86a, JET86b, JET86c]. It was then improved on by Jetteur et al. [JET87a, JET87b, JAA89] and was finally adapted to LAGAMINE by Li, whose PhD thesis is the most comprehensive work about this element [LI95]. Some information can also be found in the literature [BATO82].

This element is a 3D quadrilateral element with four nodes. It is based on Marguerre's shallow shell theory. The global behavior of the element can be divided into two modes. On the one hand, the bending behavior of the COQJ4 element is based on the classic Discrete Kirchhoff theory. It uses a bi-cubic interpolation and each node has 3 DOFs: w, φ_1, and φ_2, which are illustrated in Figure 3.2. On the other hand, the membrane behavior uses a bi-quadratic interpolation. Each node has therefore three additional DOFs: u_1, u_2, and the drilling DOF, φ_z. The drilling DOF, φ_z, was added to avoid zero stiffness in the case of co-planar elements meeting at one node. In total, then, each node has six DOFs. A detailed description of the interpolation functions of the element and the nodal DOFs can be found in Section 6.2.1.

If bending behavior is ignored, this element is reduced to a discrete Kirchhoff plate element. The basic assumptions of Kirchhoff's plate theory are the following:

1. Sections normal to the mid-surface remain normal, which means that there are no transversal shear strains;
2. The element is in a plane-stress state;

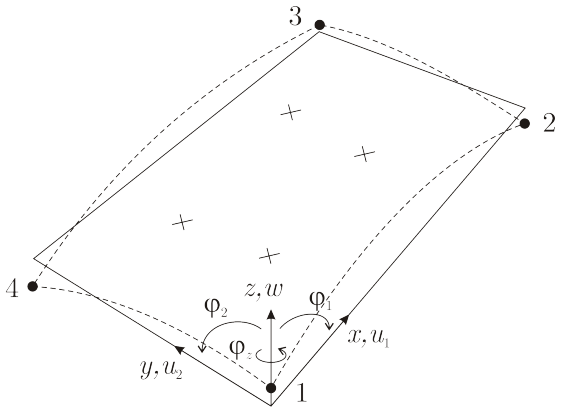


Figure 3.2: COQJ4 shell element and its nodal DOFs

3. There is no extension of the normal, i.e., the thickness is constant.

In the COQJ4 element, the first assumption is only imposed at the edges of the element. The third assumption is violated during FE simulations but is fulfilled during each time increment. This means that during one time increment, the thickness of the element is kept constant but updated incrementally at the end of each time step. The equation governing the thickness evolution can be chosen by the user. In his thesis, Li presents two approaches that can be used in the COQJ4 shell element to update the thickness [LI95, section 5.6.3]. The first approach, called the global approach, assumes that the volume of the element is constant. This is not completely accurate, because it neglects the elastic volumetric deformation, but is numerically very stable. The second approach is based on the plane stress hypothesis. Throughout this thesis, the constant volume hypothesis was chosen for all simulations. As will be shown in Section 9.2.2, this hypothesis gives, unlike the plane stress hypothesis, accurate predictions of the thickness during incremental forming, compared to experimental measurements.

The COQJ4 element is an extension of this discrete Kirchhoff plate formulation to shell elements. The shallow shell theory assumes that the slope (in local coordinates) is less than 5 degrees. Therefore, Cartesian local axes can be used instead of curvilinear coordinates (corotational coordinates). The computation of the local axes is presented in Appendix B.

Inside the element, both the stress and the strain tensors are expressed in local axes. The element uses two different stress and strain tensor conventions for the elastic and plastic parts.[2]. In addition, the stress components are integrated over the thickness. More details about the stress and strain components and the state variables of this element can be found in an internal report [HEN07c].

The element has four in-plane integration points, represented by four crosses in Figure 3.2, and from one to eleven integration points across the thickness, a number which is chosen by the user in the material law. It should be noted that this element can not be used with just any 3D constitutive law, but requires the use of dedicated plane stress laws. For this reason, a new class of shell element,

[2]The COQJ4 element assumes that the elastic part of the deformation is very small compared to the plastic part. In the elastic domain, it assumes that the strains are infinitesimal [LI95, section 5.2.1]. Therefore, it uses the engineering strains and Cauchy's stress tensor. In the plastic domain, however, it uses Green's strain tensor and Piola-Kirchhoff's stress (PK2) tensor

called solid-shell, is being investigated. This element will be based on Alves de Sousa and Fontes Valente's solid-shell element [ALV05, ALV06] and will be able to use classic 3D laws.

Much time was spent at the beginning of this thesis in order to grasp the use of this element. Indeed, the fact that the element uses two stress and strain tensors was not clearly stated in the element. Moreover, a few mistakes were discovered in the material law and in the contact element. In addition to this, no record was found as to the origin of the mass matrix in an explicit strategy, which caused a significant amount of difficulties, as will be shown later in the text.

c. Contact Element and Forming Tool

The contact elements available in the LAGAMINE FE code could only be used with an implicit strategy. To use an explicit strategy, a new approach had to be developed, as will be explained in Chapter 5.

The contact elements associated with both the BWD3D and the COQJ4 elements are based on the same principle. Details about this contact approach are available in literature [CES93, HAB98a]. This type of element uses a penalty approach and the contact pressure is computed at four in-plane integration points.

The principle of the penalty approach is the following. At each each integration point, the penetration distance of the tool inside the shell or brick element associated with the contact element is multiplied by a penalty coefficient, K_p, to generate a pressure field on the element. The nodal forces energetically equivalent to the pressure field are then computed. During the assembling step, i.e., when the nodal forces of all the brick or shell elements are regrouped in a global out-of-balance vector, the forces of the contact elements are added to the corresponding nodes and generate out-of-balance forces. During the equilibrium research, these out-of-balance forces will be gradually eliminated from iteration to iteration of the implicit algorithm by moving the nodes away from the tool, until they are considered negligible.

When contact occurs, the forces generated by the tool pressure are thus never completely removed, which means that the penetration distance never vanishes completely either. However, the higher the penalty coefficient is, the smaller the remaining penetration distance will be. The choice of the penalty coefficient's value results from a compromise between accuracy and convergence because a high penalty coefficient will produce a small remaining penetration distance but will introduce large terms in the stiffness matrix, which might then be ill-conditioned and lead to convergence problems. Therefore, this coefficient can not be too large.

Friction is treated by taking into account the relative velocity between the tool and the sheet, meshed with shell or brick elements. The relation between the pressure applied to the element's surface and the relative velocity is based on Coulomb's approach, which is similar to elastic-plastic behavior [CES93, CHA87].

As far as the spherical tool is concerned, its surface is always modeled using a rigid sphere. Indeed, it is assumed that the stiffness of the tool is much higher

than the stiffness of the sheet metal.[3] Therefore, the tool can be considered undeformable.

3.1.3 Constitutive Laws

This section, based on Flores's PhD thesis [FLO06, Section 2.4.1], introduces the concepts of mechanical laws and gives their general governing equations.

The description of an elastic-plastic material law usually involves the description of two concepts: the shape of the yield surface and its evolution with an increasingly plastic deformation. The yield surface divides the stress space into the elastic and plastic domains. In other words, the yield surface is the boundary between these two domains. A *yield criterion* F defines the shape and the size of the yield surface. The evolution of this surface during the plastic deformation can be described by a *hardening constitutive equation*.

In general, this yield criterion can be written as a function:

$$F(\underline{\underline{\sigma}}, \underline{\underline{X}}, \sigma_F) = \sigma_{eq}(\underline{\underline{\sigma}} - \underline{\underline{X}}) - \sigma_F(\epsilon^p_{eq}) = 0, \tag{3.1}$$

where $\underline{\underline{\sigma}}$ is the stress tensor, σ_{eq} is the equivalent stress and is a scalar, $\underline{\underline{X}}$ is the back-stress tensor, which defines the center of the yield surface, and σ_F represents the size of the yield surface and is a scalar function of ϵ^p_{eq}, the equivalent plastic strain. The elastic domain corresponds to $F < 0$, whereas the plastic domain corresponds to $F = 0$.

The concept of equivalent stress and strain is such that

$$\underline{\underline{\sigma}} : \underline{\underline{\epsilon}} = \sigma_{eq} \cdot \epsilon_{eq}. \tag{3.2}$$

For isotropic plasticity, Von Mises's definition, described in the next section, is applied. In all other cases, the equivalent norm must take into account the shape of the yield locus. In general, the equivalent stress is defined as

$$\sigma_{eq}(\underline{\underline{\sigma}} - \underline{\underline{X}}) = \sqrt{(\underline{\underline{s}} - \underline{\underline{X}}) : \underline{\underline{\mathbf{H}}} : (\underline{\underline{s}} - \underline{\underline{X}})}, \tag{3.3}$$

where $\underline{\underline{s}}$ is the deviatoric part of $\underline{\underline{\sigma}}$ and $\underline{\underline{\mathbf{H}}}$ is a fourth-order tensor that defines the anisotropy of the material. As far as the equivalent strain is concerned, it can be computed using Equation 3.2.

The following sections will give just a few examples of yield critera and hardening laws that are used in this thesis. A much more complete description can be found in Flores's PhD thesis [FLO06] or in several books [KHA95, DOG00].

All the simulations and experimental tests were performed with an AA3003-O aluminum alloy. The material parameters of these laws were identified using an inverse method to fit the simulation curves to the experimental measurements. The parameters of the different laws are given in Appendix C.

[3] More information about the tool material used within this thesis can be found in Section 3.2.2.

a. Yield Criteria

Cold-rolled metal sheets present anisotropic behavior of their mechanical properties due to the rolling procedure itself generating their crystallographic texture. The phenomenological anisotropic yield criteria described here express this anisotropic behavior in an orthogonal set of material axes. These material axes are taken in the rolling direction (RD), the transverse direction (TD) and the normal (through-thickness) direction of the sheet (ND).

Von Mises's Yield Criterion This is the simplest, although most widely used, yield criterion. The material's behavior is assumed to be isotropic and its definition is characterized by the following equation:

$$\begin{aligned}\sigma_{eq}^2(\underline{\underline{\sigma}}) &= \frac{1}{2}\Big[(\sigma_{xx}-\sigma_{yy})^2+(\sigma_{xx}-\sigma_{zz})^2+(\sigma_{yy}-\sigma_{zz})^2+\\ &\qquad 6\left(\sigma_{xy}^2+\sigma_{xz}^2+\sigma_{yz}^2\right)\Big].\end{aligned} \tag{3.4}$$

For this definition, the choice of the material axes has no effect.

Hill's 1948 Yield Criterion Hill's yield criterion is also very simple. It is a generalization of Von Mises's criterion to anisotropic material [HIL50]. It is a function of six material parameters, F, G, H, N, L, and M, although most of the time, it is assumed that $N = L = M$ and that $G + H = 2$. The equivalent stress is defined by the following equation:

$$\begin{aligned}\sigma_{eq}^2(\underline{\underline{\sigma}}) &= \frac{1}{2}\Big[H\left(\sigma_{xx}-\sigma_{yy}\right)^2+G\left(\sigma_{xx}-\sigma_{zz}\right)^2+F\left(\sigma_{yy}-\sigma_{zz}\right)^2+\\ &\qquad 2N\,\sigma_{xy}^2+2L\,\sigma_{xz}^2+2M\,\sigma_{yz}^2\Big].\end{aligned} \tag{3.5}$$

It can be seen that if $F = G = H = 1$ and $N = L = M = 3$, Von Mises's law is recovered.

For computational reasons, it is possible to rewrite this equation using the notation defined by Doghri [DOG00, Annexe C], which takes advantage of the tensor's symmetries. With this notation, the stress tensor is written as a vector and the fourth-order tensor $\underline{\underline{\mathbf{H}}}$ as a two-dimensional matrix $\underline{\underline{H}}$. This gives the following equation:

$$\sigma_{eq} = \sqrt{\frac{1}{2}\,\underline{\sigma}^T\cdot\underline{\underline{H}}\cdot\underline{\sigma}} \tag{3.6}$$

with

$$\underline{\underline{H}} = \begin{bmatrix} G+H & -H & -G & 0 & 0 & 0\\ -H & H+F & -F & 0 & 0 & 0\\ -G & -F & F+G & 0 & 0 & 0\\ 0 & 0 & 0 & 2N & 0 & 0\\ 0 & 0 & 0 & 0 & 2L & 0\\ 0 & 0 & 0 & 0 & 0 & 2M \end{bmatrix}$$

and

$$\underline{\sigma}^T = \left[\sigma_{xx}\ \sigma_{yy}\ \sigma_{zz}\ \sigma_{xy}\ \sigma_{xz}\ \sigma_{yz}\right]^T.$$

Others Numerous other yield criteria are available in literature and in the LAGAMINE FE code. More information can be found in Habraken's literature review on plastic anisotropy [HAB04a] or in Banabic's book [BAN00]. However, for the SPIF simulations, the computation time had to be as small as possible, which prevented the author of the present work from using more complex laws. The analysis of the simulations performed with Von Mises's and Hill's yield loci showed that these simple yield surfaces were accurate enough to predict the forming forces and the shape of parts formed with the material studied [HE05b]. Such a conclusion has been extended to the formability study during SPIF simulations performed by Eyckens [EYC08b]. Unlike the yield criterion, the hardening law, described in the next section, has a much bigger impact on the simulation results.

b. Hardening Laws

The hardening law describes the evolution of the size and the center position of the yield surface. There are three types of hardening:

- isotropic, where only the size of the yield surface can change;
- kinematic, where the size of the yield surface is fixed but its center position can vary;
- mixed isotropic-kinematic, where both the size and the center position are modified.

Three laws will be described here: one simple isotropic law, Swift's law, and two mixed hardening laws, Armstrong-Frederick's and Ziegler's laws.

Swift's Isotropic Hardening This hardening model describes the evolution of the size of the yield surface σ_F. It is defined as

$$\sigma_F(\epsilon_{eq}^p) = K\left(\epsilon_0 + \epsilon_{eq}^p\right)^n. \tag{3.7}$$

It is a simple power law without horizontal saturation. An example of a stress-strain curve is presented in Figure C.1 in Appendix C.

Armstrong-Frederick's Kinematic Hardening This model was introduced in 1966 by Armstrong and Frederick [ARM66]. This is LAGAMINE's first mixed-type hardening law. First, it used Swift's law to describe the evolution of the size of the yield locus. Then, the back-stress tensor, $\underline{\underline{X}}$, which represents the location of the center of the yield locus, evolves with the equivalent plastic strain:

$$\overset{\nabla}{\underline{\underline{X}}} = C_X\left(X_{sat}\,\underline{\underline{\dot{\epsilon}}}^p - \underline{\underline{X}}\,\dot{\epsilon}_{eq}^p\right), \tag{3.8}$$

where $\overset{\nabla}{\underline{\underline{X}}}$ is Jaumann's objective derivative of the back-stress tensor, C_X is the kinematic hardening saturation rate, X_{sat} is the saturation value of the kinematic hardening, and $\underline{\underline{\dot{\epsilon}}}^p$ is the plastic strain rate tensor. More information about this law can be find in an internal report by Pierre de Montleau [DEM03], in Flores's PhD thesis [FLO06] or in the literature [DET04].

Ziegler's Kinematic Hardening The basic concepts of this law were published by Ziegler in the late 1950s [ZIE59] and were an improvement on the first kinematic hardening law developed by Prager [PRA55]. This hardening law was added to LAGAMINE in order to have the same mixed hardening law as in Abaqus. Similarly to Armstrong-Frederick's law, Ziegler's uses Swift's law to describe the evolution of the size of the yield locus. As far as the back-stress is concerned, the governing equation of this law is the following:

$$\overset{\nabla}{\underline{\underline{X}}} = C_A \frac{1}{\sigma_F} \left(\underline{\underline{\sigma}} - \underline{\underline{X}}\right) \dot{\epsilon}^p_{eq} - G_A \underline{\underline{X}} \dot{\epsilon}^p_{eq}, \tag{3.9}$$

where C_A and G_A are material parameters.

3.1.4 Meshing Tools

A large number of simulations have been performed throughout this thesis. The LAGAMINE FE code is not a commercial code and the generation of the input files can be tedious. Therefore, several Matlab or Fortran programs have been developed in order to help the user with the creation of the meshes and the generation of all these input files.

Two shapes of metal sheets were commonly used for the simulations: square and circular. For the square meshes, a program was written in Matlab in order to help the user with the generation of all the simulation files.[4] For the circular meshes or the pie meshes (meshes for a portion of a circle), it is much more difficult to create a mesh with well-proportioned elements and adjust the element density. For this reason, another program was developed in Fortran. This program is generic and allows the user to define a variable mesh density easily. It can produce a mesh with shell elements or several layers of brick elements, with or without contact elements and with different types of boundary conditions.[5] Examples of such meshes can be seen in Figure 3.3.

3.1.5 Post-Processing Tools

a. Graphs of Part Shapes

Two Matlab programs `SelectNodesCut` and `GUI_IMP` were developed in order to produce plots of nodal coordinates. The purpose of these programs is to compare, in the same graph, the shape of several sheet metal simulations in a cross section. Many examples of such figures can be found throughout the thesis.

The creation of these plots is done in two steps. First, for each simulation, the user must select a list of nodes located, in the initial undeformed mesh, in a section plane and save the evolution of their coordinates in a text file. This

[4] This program, called `MESH_SQUARE_SHEET`, can generate not only the `lag` file with the mesh (with brick or shell elements) but also all the files necessary for the simulations: the execution file with the simulation parameters, the `dep` file with the displacement of the tool and the `pri` file with the data to be saved. This program is available in the meshing tool database of LAGAMINE.

[5] This program is called `MESH_CIRCULAR_SHEET`. In order to generate the mesh, the user can either input the necessary data interactively or write them in a text file in order to reuse them. This program is available in the meshing tool database of LAGAMINE.

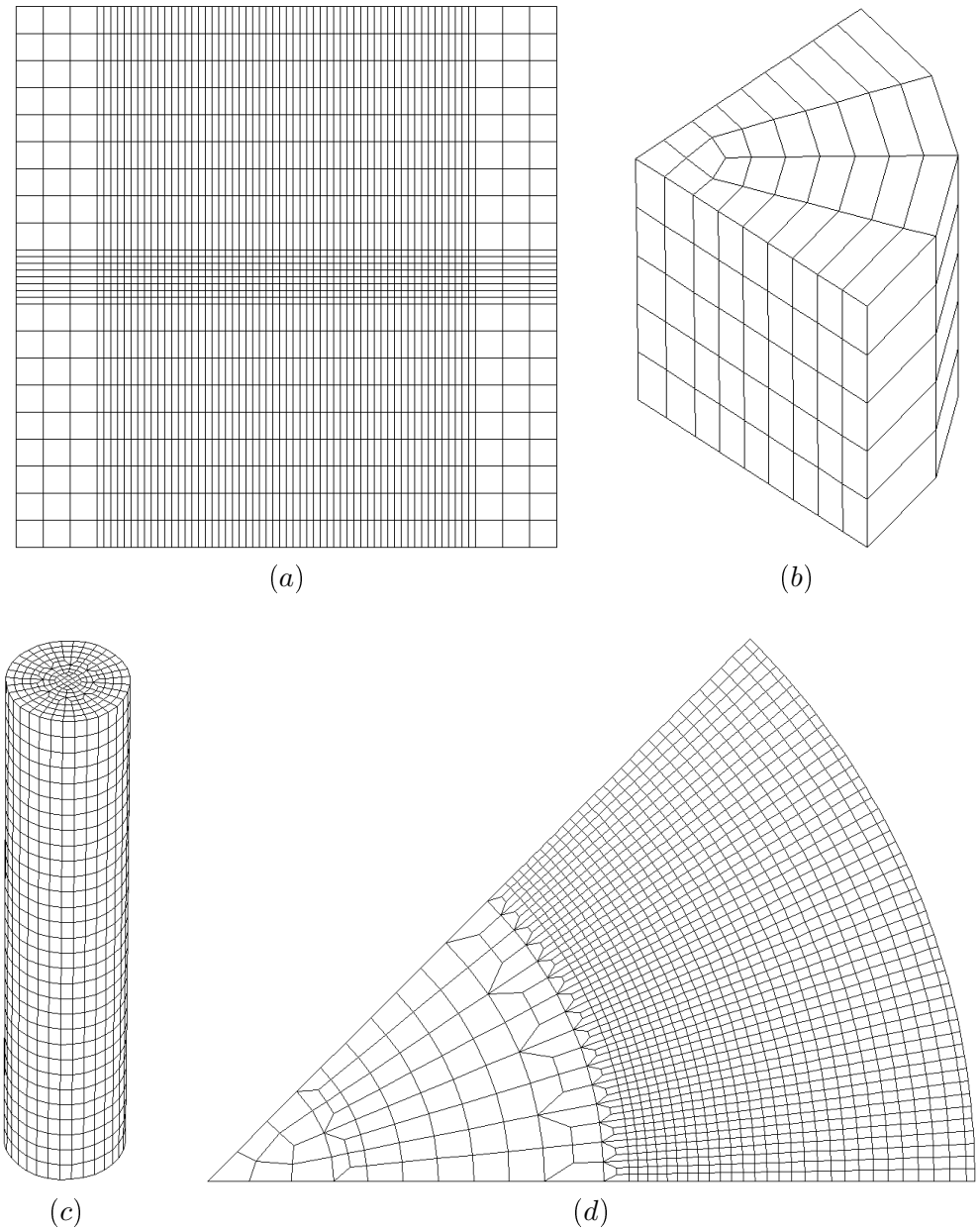


Figure 3.3: Square and circular mesh examples: (a) square mesh with shell elements, (b) 90-degree pie mesh with brick elements, (c) 360-degree circular mesh with brick elements, (d) 45-degree pie mesh with shell elements

is done using the program `SelectNodesCut`. This programs will ask the user to choose the desired section plane. Since the meshes of the sheet metal used in this thesis were always in the x-y plane with the thickness in the z-direction, it was decided to perform cross sections only in a plane parallel to the z-axis, which simplifies the equation of this plane. This equation can be one of the following:

- $x = \mathsf{a}$
- $y = \mathsf{a}$
- $x = \pm y$
- $y = \mathsf{m}\ x$
- $y = \mathsf{m}\ x + \mathsf{a}$

where a and m are user-defined constants. Some of these cross sections are illustrated in 2D in Figure 3.4.

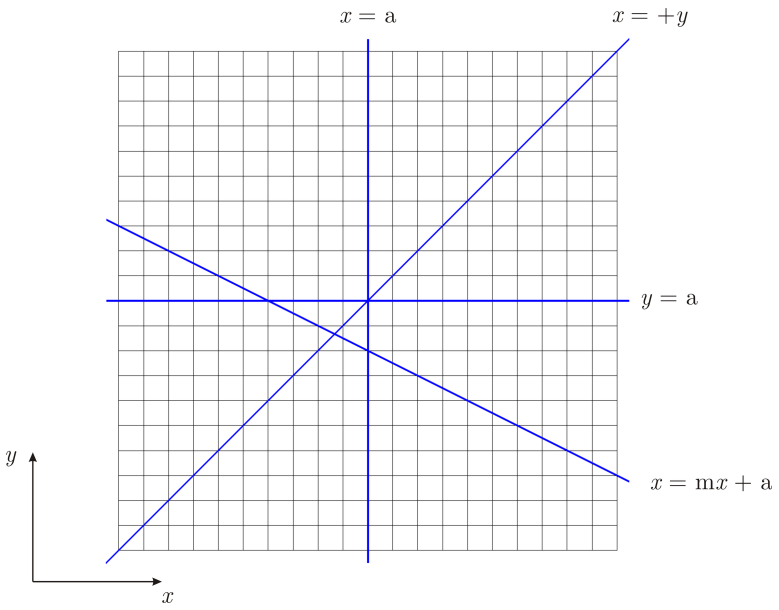


Figure 3.4: Example of cross sections used by `SelectNodesCut`

The program will then ask the user to locate both LAGAMINE's input and result files on the disk. It will read the input file, which contains the initial undeformed mesh, and generate the list of nodes. Then, it will automatically call the program `Select` in order to read the result file and save the coordinates of the nodes in a text file, called `imp` file. As an example, a few lines of such a file are shown in Figure 3.5.

After creating an `imp` file for each simulation, the second step is to read these files and plot the desired data. This is done using a graphical user interface (GUI) developed also in Matlab and called `GUI_IMP`. A screen capture of this program is shown in Figure 3.6.

This program is very intuitive and easy to use. The first step is to locate, for each simulation, the `imp` file by clicking on the "Browse" button. The user

```
Line test - impl simulation with shells
************************************************************************


================================================================================
TIME =    0.1000000        ALACUM =     0.000000        ALAMF =     0.000000
================================================================================

COORDONNEES ACTUALISEES -- TEMPS =  0.100
*****************************************

NODE   26    DOF 1 -91.0000        DOF 2  0.00000        DOF 3  0.00000        ...
NODE   77    DOF 1 -87.3553        DOF 2-0.367519E-05    DOF 3-0.627918E-01    ...
NODE  128    DOF 1 -83.7147        DOF 2-0.208977E-05    DOF 3-0.190372        ...
NODE  179    DOF 1 -80.0759        DOF 2 0.289303E-07    DOF 3-0.345642        ...
NODE  230    DOF 1 -76.4382        DOF 2 0.454224E-05    DOF 3-0.520029        ...
NODE  281    DOF 1 -72.8012        DOF 2 0.941654E-05    DOF 3-0.711676        ...
NODE  332    DOF 1 -69.1651        DOF 2 0.190559E-04    DOF 3-0.921626        ...
NODE  383    DOF 1 -65.5299        DOF 2 0.308034E-04    DOF 3 -1.15314        ...
NODE  434    DOF 1 -61.8961        DOF 2 0.598576E-04    DOF 3 -1.41102        ...
NODE  485    DOF 1 -58.2642        DOF 2 0.788061E-04    DOF 3 -1.70226        ...
...


================================================================================
TIME =    0.2000000        ALACUM =     0.000000        ALAMF =     0.000000
================================================================================

COORDONNEES ACTUALISEES -- TEMPS =  0.200
*****************************************

NODE   26    DOF 1 -91.0000        DOF 2  0.00000        DOF 3  0.00000        ...
NODE   77    DOF 1 -87.3445        DOF 2-0.305733E-04    DOF 3-0.126249        ...
NODE  128    DOF 1 -83.7059        DOF 2-0.420312E-04    DOF 3-0.383378        ...
NODE  179    DOF 1 -80.0723        DOF 2-0.513583E-04    DOF 3-0.686186        ...
NODE  230    DOF 1 -76.4413        DOF 2-0.554017E-04    DOF 3 -1.02086        ...
NODE  281    DOF 1 -72.8119        DOF 2-0.524140E-04    DOF 3 -1.38198        ...
NODE  332    DOF 1 -69.1838        DOF 2-0.381856E-04    DOF 3 -1.76884        ...
NODE  383    DOF 1 -65.5571        DOF 2 0.999124E-05    DOF 3 -2.18697        ...
NODE  434    DOF 1 -61.9327        DOF 2 0.104874E-03    DOF 3 -2.65124        ...
NODE  485    DOF 1 -58.3132        DOF 2 0.390541E-03    DOF 3 -3.19486        ...
...
```

Figure 3.5: Extract from an `imp` file

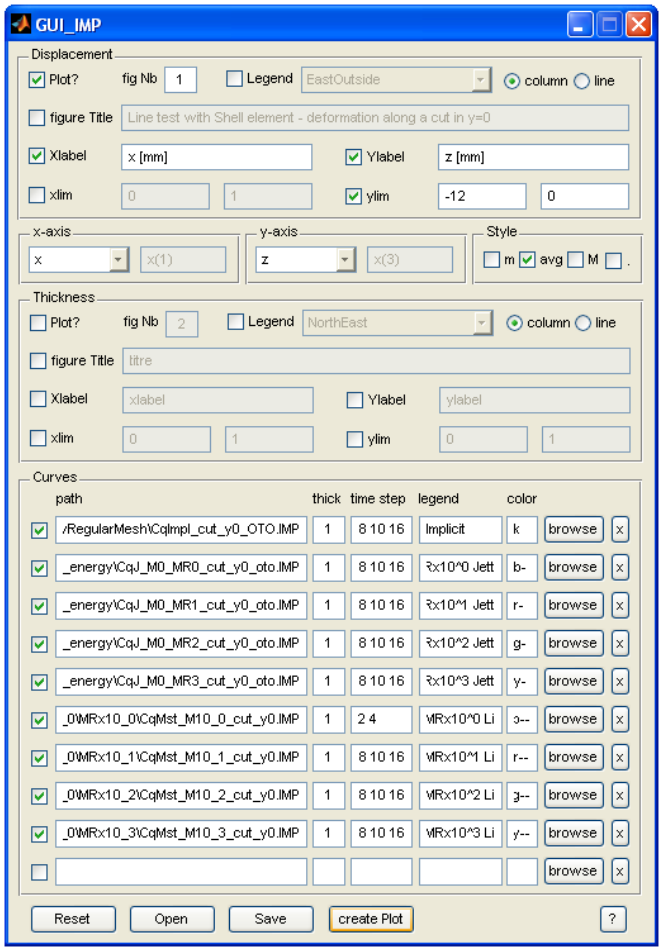


Figure 3.6: Graphical user interface used to read `imp` files

must also input the number of layers of nodes across the thickness[6], choose one or several simulation times, the string to be displayed in the legend for that curve, and the line style and color. The user can also choose the title of the graph, the position and orientation of the legend, the axes labels and data ranges, and which nodal coordinate (or function of a nodal coordinate) corresponds to which axis on the figure. As an example, if the mesh is a square metal sheet and the user is interested in the shape of the part in a cross section along the x-axis, he or she should plot the x-coordinate on the horizontal axis and the z-coordinate on the vertical axis. On the other hand, if the mesh is a circular metal sheet and the user wants to plot the shape of the part in a cross section at 45° from the x-axis, he or she should plot the radial distance, i.e., $\sqrt{x^2+y^2}$, on the horizontal axis and once again the z-coordinate on the vertical axis. More complex formulae could be used.

Finally, all the parameters of the GUI can be saved in a text file in order to reuse them later.

b. Graphs of Forces

The Matlab program `GUI_REA` was developed in order to read tool force data of one or several simulations and plot their evolution as a function of time. The files containing the force data must be `rea` files, which are a standard output of LAGAMINE.[7] The syntax of these files is very simple: each line contains the current time and the three tool force components. LAGAMINE always writes the forces in the global coordinate system. In the case of a simulation with rotational symmetry, it is usually preferable to analyze the force in terms of radial, circumferential and vertical components. In that case, an additional program, called `Convert_xyz_radius_theta_z`, can read the tool position for each time step, convert the forces into the new coordinate system, and create a new `rea` file.

Similarly to `GUI_IMP`, a GUI facilitates the use of the program `GUI_REA`. For each simulation, the user must locate the `rea` file by clicking on the "Browse" button. Then, he or she can choose the string to be displayed in the legend for that particular simulation and the line style and color. The user can also choose which force component must be plotted: this can be one or several individual components, or a function of several components like the norm of the force vector. Finally, an additional feature can be used in the case of forces with high oscillations in order to have a smoother curve. Large oscillations in the force often happen in FE simulations if the mesh is not fine enough. The user can then choose between two built-in smoothing methods (a moving average[8] or a

[6]For a simulation with shell elements, the number of layers of nodes is always one. On the other hand, for a simulation with brick elements, the number of layers of nodes is equal to the number of layer of elements across the thickness, plus one.

[7]Another file, `ipr`, is also available in LAGAMINE but can easily be converted into the required format using another program called `Convert_IPR_REA`. This program is located in the same directory as the program `GUI_REA`.

[8]For an n-point moving average, the value at each point is replaced by the average of the values of the $(2n+1)$ points around it:

$$y_{n-avg}(i) = \frac{1}{2\,n+1}\left[y(i-n) + y(i-n+1) + \ldots + y(i) + \ldots + y(i+n-1) + y(i+n)\right].$$

high-pass filter[9]) or input a user-defined one. Again, all the parameters of the GUI can be saved in a text file for later use.

A screen capture of this program is shown in Figure 3.7. An example of a plot generated with this program is shown in Figure 3.8.

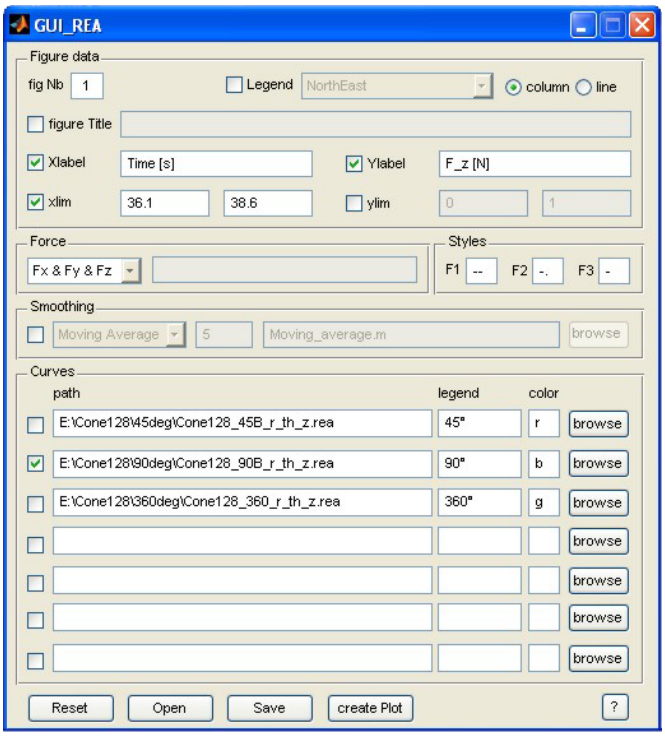


Figure 3.7: Graphical user interface used to read `rea` files

3.2 Experimental Equipment

3.2.1 Machine

Most users perform the experimental forming tests on a standard three-axis CNC milling machine. This is the choice that was made within the SeMPeR project. The machine was a 3-axis CNC ACIERA milling machine with a horizontal spindle. Two DOFs were obtained by the movement of the table along the horizontal and vertical axes, a third resulting from the displacement of the spindle along its axis. The main characteristics of the machine are summarized in Table 3.1.

For incremental forming, the tool speed was reduced to 2000 mm/min and the spindle rotation speed was kept between 80 and 250 RPM, depending on the wall angle of the part being made and on the tool size.

[9]The high-pass filter uses a Butterworth filter of order eight [BUT30]. The input argument of this function is the normalized cutoff frequency ω_n. This is a real number between 0 and 1, where 1 corresponds to Nyquist's frequency, i.e., half the sample rate. More information can be found in Matlab help or in literature [SMI02].

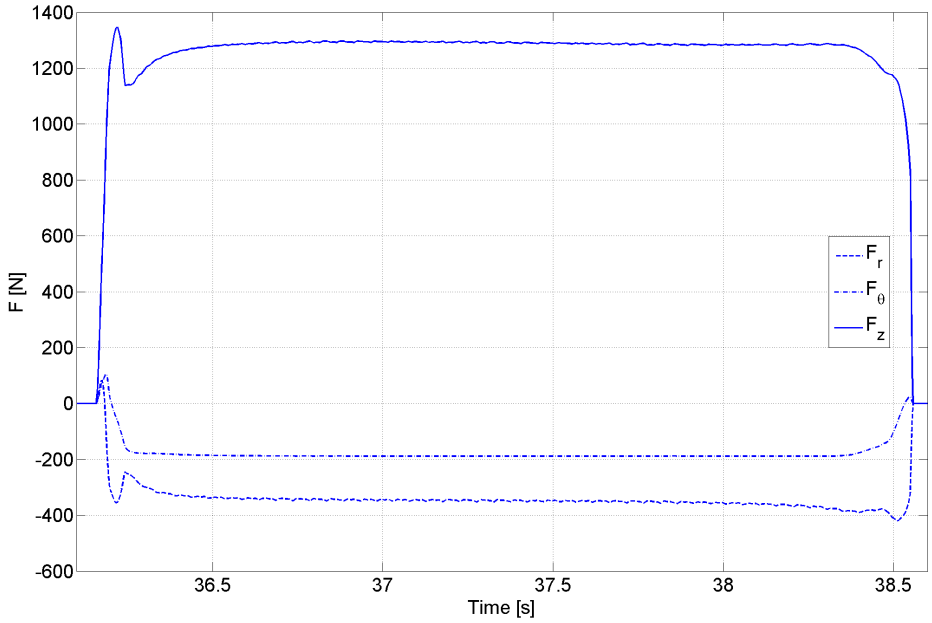


Figure 3.8: Example of a graph showing the evolution over time of the radial, circumferential and vertical components of the tool force during one contour of the forming of a 50-degree cone

Characteristics	Value	[Unit]
Total power	27	[kW]
Spindle power	9	[kW]
Spindle rotation speed	50–4000	[RPM]
Displacement of table	500 × 400	[mm]
Displacement of spindle	300	[mm]
Maximum tool speed	6000	[mm/min]

Table 3.1: Characteristics of the ACIERA machine

3.2.2 Forming Tools

The tools used in the SeMPeR project were a solid hard-steel hemispherical head mounted on a shaft, as shown in Figure 2.14.

The tools were manufactured from a hard steel, called Vanadis® 23, manufactured by Uddeholm [UDDEL]. This steel was then heat-treated. Vanadis® 23 is a high-performance powder metallurgical steel used for cutting tools. More specifically, it is a chromium-molybdenum-tungsten-vanadium alloyed high-speed steel which is characterized by

- high wear resistance (abrasive profile),
- high compressive strength,
- superior through-hardening properties,
- good toughness,
- excellent dimensional stability on heat treatment,
- and excellent temper resistance.

This alloyed steel is suitable for cutting tools such as reamers, taps, milling cutters, and broaches. The measured hardness of the tools is 859.30±7.22 HV (Vickers hardness). Several diameters of the tool tip were used: 10, 12.7, 15, 20, and 25 mm. Two pictures of the tools are shown in Figures 2.14 and 3.9.

The tools were inserted into a holding fixture designed for the milling machine, as shown in Figure 3.9.

Figure 3.9: 5-mm tool inserted in the tool holding fixture; picture courtesy of Johan Verbert (KUL)

3.2.3 Clamping Device

A clamping device was used to fix the blank on the table. Since a milling machine with a vertical spindle was used at the beginning of the SeMPeR project, a first version had been designed for this machine. An exploded assembly view and a cross section view are presented respectively in Figures 3.10 and 3.11. This device is composed of several parts:

- a support, leaving some space underneath the part so that it can deform freely, and so that the on-line optical measuring system (see Section 3.2.5) can "see" a good portion of its bottom surface;

- the top plate, which is the main support of the part, used to center the backing plate and provide the system with a sufficient amount of rigidity;
- the backing plate, which is needed to prevent global deformation of the part and to obtain sharp edges;
- the clamping plate, which is bolted upon the top plate in order to compress the edges of the part between the top plate and the backing plate, thereby preventing it from sliding.

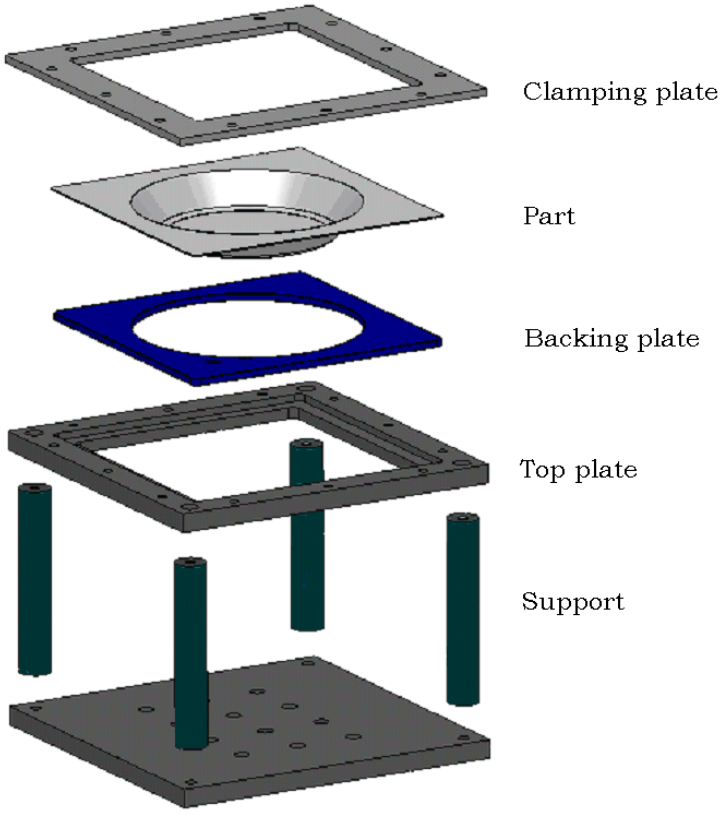


Figure 3.10: Exploded assembly view of the clamping device; picture courtesy of Johan VERBERT (KUL)

In the example of the milling machine with a horizontal spindle, the support differs slightly from the above design, but the other parts remain the same. Some pictures of this clamping device are shown in Figure 3.12. In this case, the device is placed vertically, which solves the problem of having enough space under the part for the optical system and provides easy access to its bottom surface. Unfortunately, this type of orientation excludes the possibility of using an oil bath on the top surface of the sheet metal, making lubrication of the part more difficult during forming. The solution chosen was to use a cooling fluid[10] which also lubricates this surface.

[10]A cooling fluid is a fluid which is continuously sprayed on the tool and on the metal sheet during the whole process to prevent them from overheating.

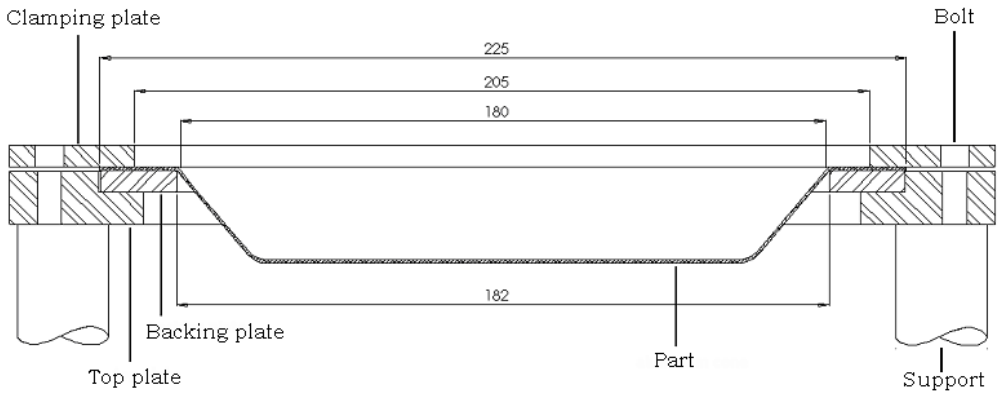
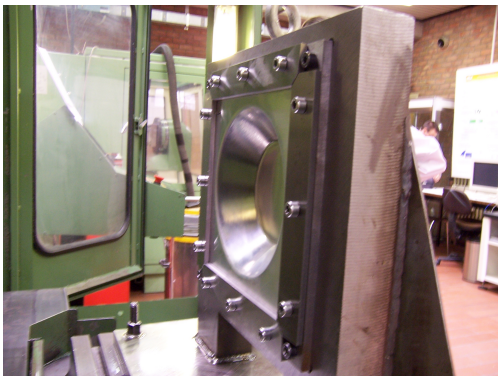
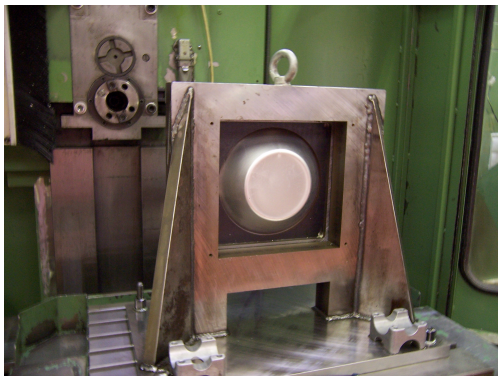


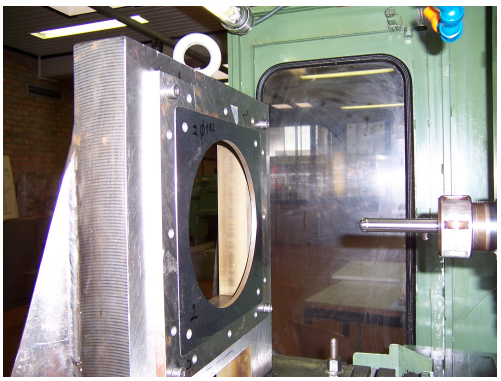
Figure 3.11: Cross section of the clamping device; picture courtesy of Johan Verbert (KUL)



(*a*) Side view of the complete system



(*b*) View from behind



(*c*) Close-up view on the backing plate and the tool

Figure 3.12: Clamping device for the horizontal spindle milling machine; pictures courtesy of Johan Verbert (KUL)

3.2.4 Force Measurement

In the SeMPeR project, the forces acting on the tool were measured using a force platform [DUF05b]. This platform was a Kistler 9265B six-component force dynamometer, as shown in Figure 3.13, capable of measuring a vertical force between -15 and 30 kN and two horizontal forces of ±15 kN.

Figure 3.13: Kistler force platform [KISTL]

This device was connected to a complementary Kistler 5017B 8-channel charge amplifier, then was placed between the clamping device and the work table of the milling machine. The forces applied on the tool were then recorded on a computer equipped with a data acquisition system.

3.2.5 Deformation Measurement

Two techniques were used for measuring the deformation of the parts: Digital Image Correlation Technique (DICT)n to measure parts during forming, and laser scanning for the off-line measurements.

a. DICT

According to the Limess website [LIMESS],

> Digital Image Correlation is a data analysis method which uses a proprietary mathematical correlation method to analyze digital image data taken while samples are subjected to mechanical stresses. Consecutive image captures taken during the testing phase will 'show' a change in surface characteristics as the specimen is affected by the mechanical stresses imposed upon it.

The system is composed of two cameras mounted on a fixing device, a computer with a data acquisition device and sample plates with a given pattern used during the system calibration, as shown in Figure 3.14. A light source must also be available.

As illustrated by the images in Figure 3.15, the work principle is the following [LIMESS]:

> The sample is prepared for testing by the application of a random dot pattern to its surface. An image is acquired of the newly 'speckled' sample before loading. The sample is subject to surface deformation as a load is applied. A series of images are taken during this loading.

Figure 3.14: DICT components [GOM]

> These other images show a deformed random dot pattern relative to the initial, undeformed random dot pattern. From the difference in the deformed images and the undeformed image, a deformation plot can be calculated for each image.

The principal advantage of DICT is that it can take on-line measurements, without contact with the part. Its drawbacks are that:

- the lighting needs to be uniform for the camera to be able to recognize all the points of the patterns on the images;
- both the density of the dot pattern which is applied on the surface of the sample and the size of the dots have to be carefully chosen;
- the calibration process of the system can be time-consuming, especially for 3D measurements[11];
- the cameras need to "see" the part from a certain angle, which prevents them from seeing the whole part;
- only the bottom surface of the sheet metal can be scanned, since the tool blocks the view of the top surface for the camera;
- the cameras must be placed at a certain distance to be able to focus on the part, which is sometimes difficult to achieve because of the limited space available behind the part.

For the SeMPeR project, two different DICT systems were available: Aramis® (at the KUL) and Limess® (at the VUB). The validation of the Limess® system with FE simulations was discussed in an article published during Esaform'05 conference [WAT05].

b. Laser Scanning

For the SeMPeR project, Metris® LC50, a laser scanning device, was also available for measuring formed parts. According to this device's developers [METRIS],

> laser scanning enables complete modeling and inspection of parts with complex free-form surfaces or features such as holes, slots, etc.

[11] Indeed, for 2D measurements, only one camera is needed, which makes the process easier.

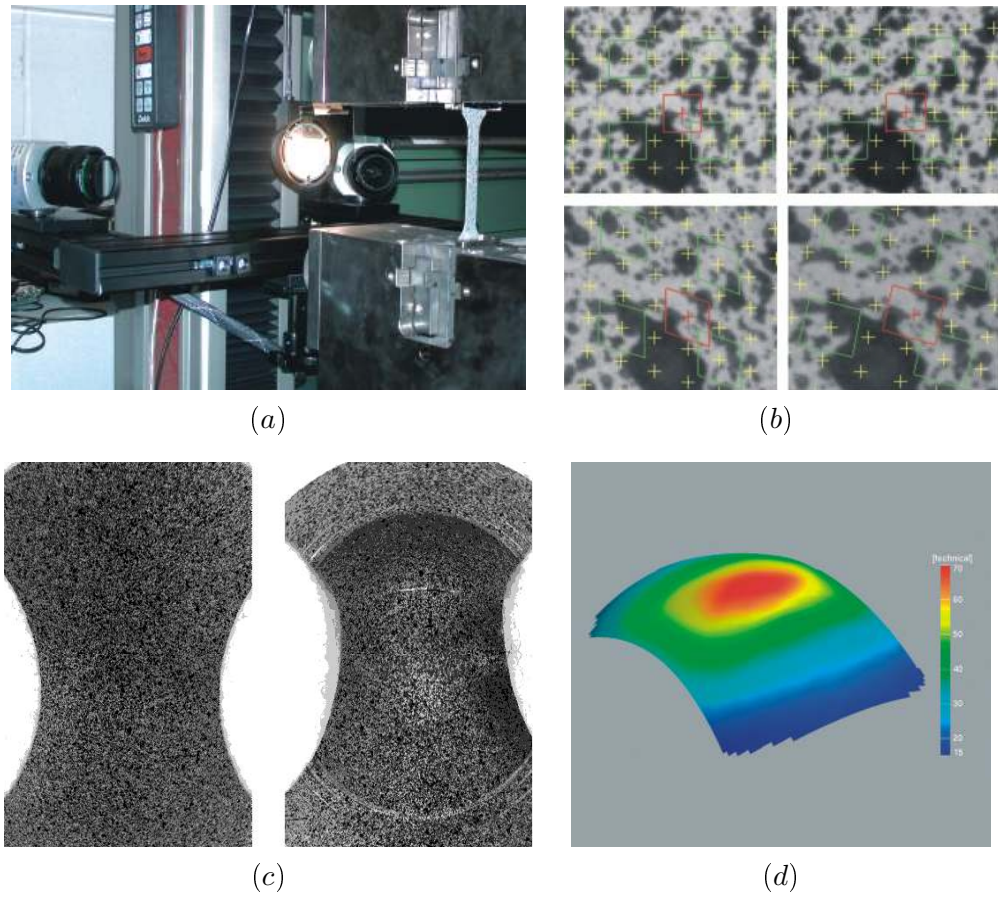


Figure 3.15: Work principle of DICT [GOM]; (*a*) close-up view of the specimen, the camera and the lighting; (*b*) evolution of the dot pattern of the specimen during deformation; (*c*) specimen before and after deformation; (*d*) plot of the results.

> Whereas with traditional touch probes, it can take hours or even days to measure full parts, laser scanning is a non-contact scanning technology which gives the results within minutes and is suited for measuring flexible or fragile materials, which often present severe challenges for touch probes due to the risk of indentations or surface scratches.

The main advantage is that this device can scan both sides of the surface of the part when the forming process is finished and gives, by taking the difference, the thickness of the part.

Figure 3.16: Laser scanning device [METRIS]

Part II

Dynamic Explicit Simulations of SPIF

Chapter 4

Motivations

The purpose of this chapter is to explain the motivations for the developments in the dynamic explicit field in the Lagamine code. This will be done by showing the performance of the simulation tool using an implicit strategy, before making any modifications to the code.

4.1 Simulation Examples

In order to evaluate the performance of the available simulation tool, two simulations were performed: a line test and a cone. For both simulations, the focus will be on the computation time, rather than on the accuracy of the results. For this reason, this chapter will not contain a description of all the material parameters, finite element type, mesh, etc. More detailed analyses will be done later in Chapters 8 and 9.

4.1.1 Line Test

A line test can be considered a simplified version of the SPIF process. The tool path consists of two successive linear displacements of 100 mm in the middle of the metal sheet at a depth of 5 and 10 mm, respectively, as illustrated in Figure 4.1. The lengths in the figure are expressed in millimeters, as will always be the case throughout this thesis (unless otherwise specified).

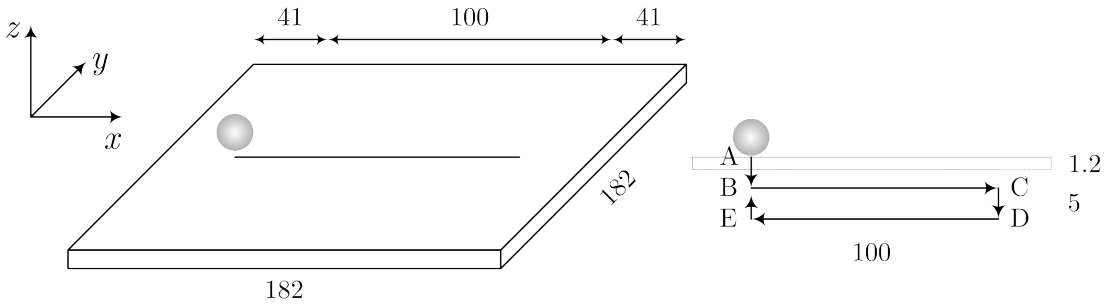


Figure 4.1: The line test and its tool path

Using the LAGAMINE FE code with an implicit strategy and a mesh made with either shell or brick elements, the computation times are given in Table 4.1. These simulations were all performed on the same computer, with one CPU. Two methods for storing a linear system of equations were tested:

- KNSYM=3 uses a sparse storage method;
- KNSYM=4 uses a skyline storage method.

The skyline system is usually faster but requires a significant amount of storage space. The sparse system takes up less memory space and is independent on the numbering of the DOFs, but is usually slower. Both meshes — brick and shell — used the same in-plane element density shown in Figure 4.2. The mesh with shell elements contains 1612 elements, whereas the one with bricks contains 6448 elements, i.e., 3 layers of bricks and 1 layer of contact elements. [1] The smallest elements have an in-plane area of 2.275×2.275 mm^2. The symmetry of the problem was not taken into account here because the purpose was to have an order of magnitude of the computation time for a simple SPIF test, not to reduce that time as much as possible, as is the case for later simulations.

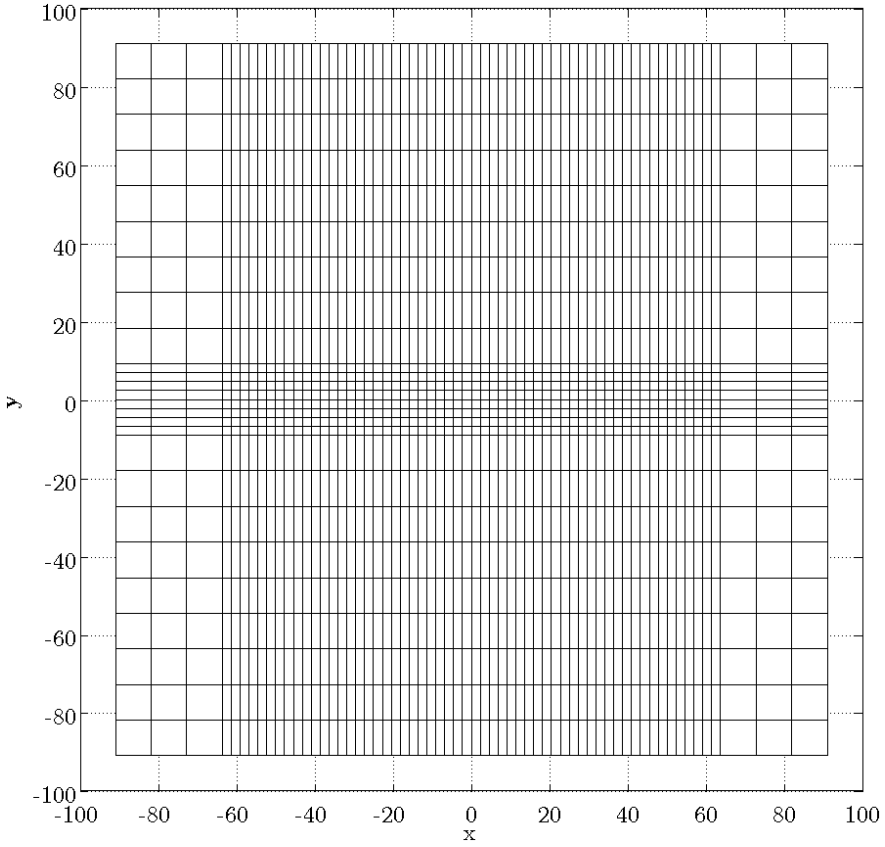


Figure 4.2: Mesh used for the line test

The conclusion from this test is that even for such a small SPIF simulation, the computation times are relatively large and can vary greatly – from about

[1] For the shell element, the contact element is included in the element itself, whereas it needs to be explicitly defined for the brick element.

Element type	Computation time [s]	
	KNSYM=3	KNSYM=4
Shell element	4 986	1 314
Brick element	65 762	31 728

Table 4.1: Computation times in seconds for the line test simulation

20 minutes to 18 hours – depending on the element type and system solution method. With an optimized mesh of shell elements simulating only half of the sheet metal with appropriate boundary conditions[2], the simulation time could be reduced to less than 3 minutes.

4.1.2 Cone Simulation

A cone is the most common part made with the SPIF process. It is widely tested in literature because the wall angle of the cone corresponds to the difficulty in producing the part. In this section, a 50-degree cone with a depth of 30 mm was simulated. This angle is far from the limiting angle for the material used, which is around 73 degrees.

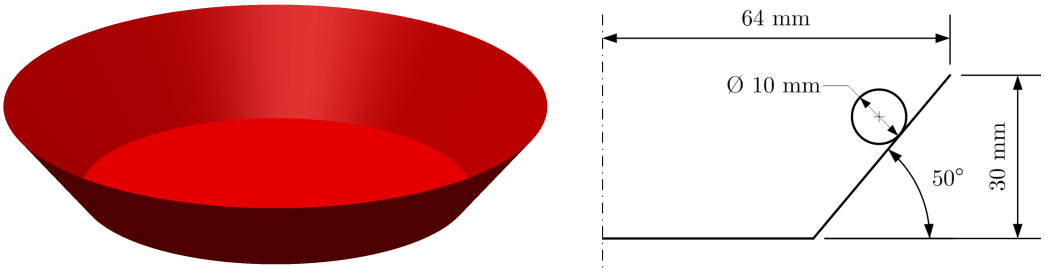


Figure 4.3: 3D view and dimensions of the 50-degree cone

The tool path consists of a succession of 30 circular contours with a vertical stepdown of 1 mm between two successive ones, as shown in Figure 4.4.

In order to speed up the simulation, only one quarter of the cone was simulated with appropriate boundary conditions. More details about the boundary conditions will be given in Chapter 8, and in particular in Section 8.1.2. The mesh, shown in Figure 4.5, was designed with shell elements. It contained 4 448 nodes (26 392 DOFs) and 4 506 elements.

The norm of the total displacement is shown in Figure 4.6. The computation time was approximately 95 hours or almost four days.

The same simulation was also run without any simplification: the full 360° mesh contained 17 569 nodes (104 262 DOFs) and 17 376 elements. In that case, the computation time was around 1640 hours, i.e., more than 3 months, far from the four days needed for a quarter of the cone. The thickness profile at the end of this simulation is shown in Figure 4.7.

[2] 418 elements and 468 nodes with symmetry boundary conditions

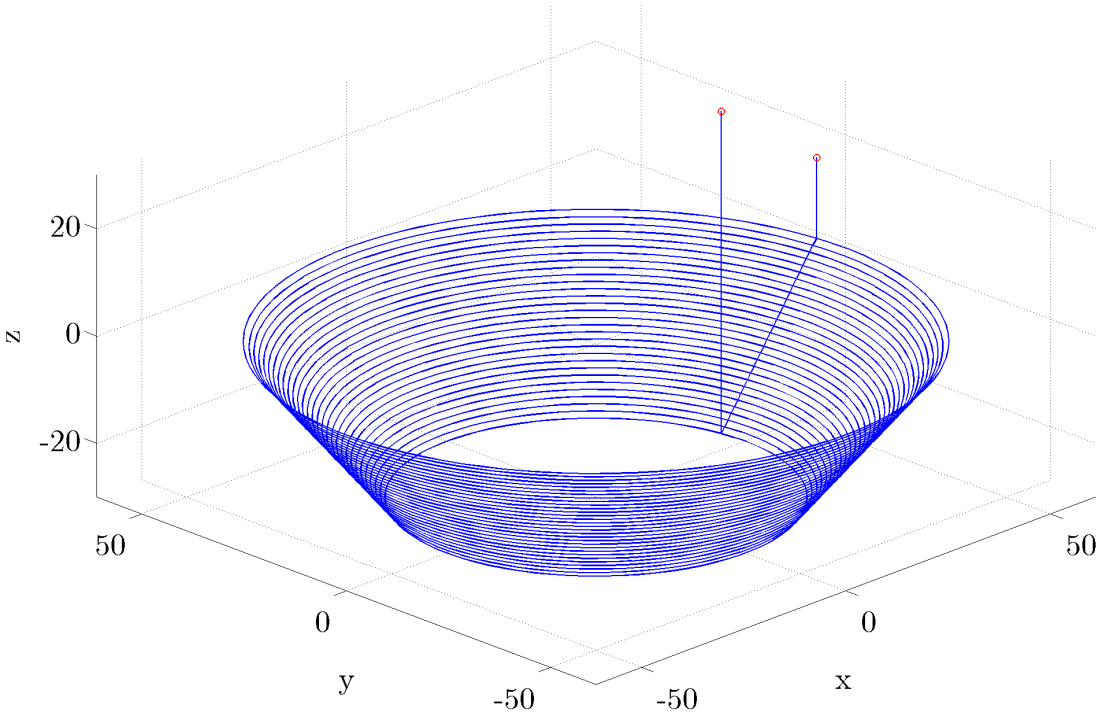


Figure 4.4: Tool path of a 50-degree cone

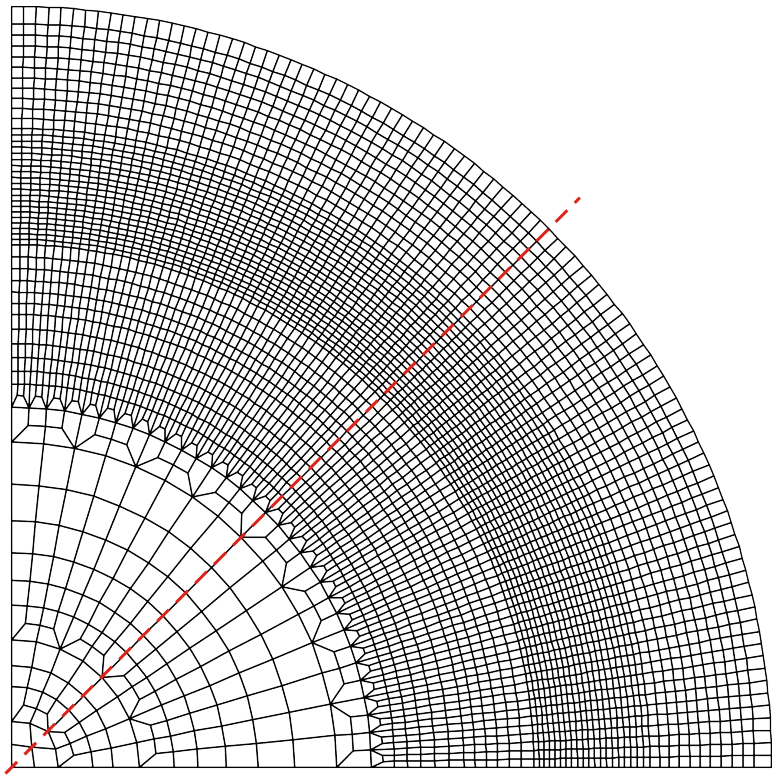


Figure 4.5: 90-degree pie mesh of the cone

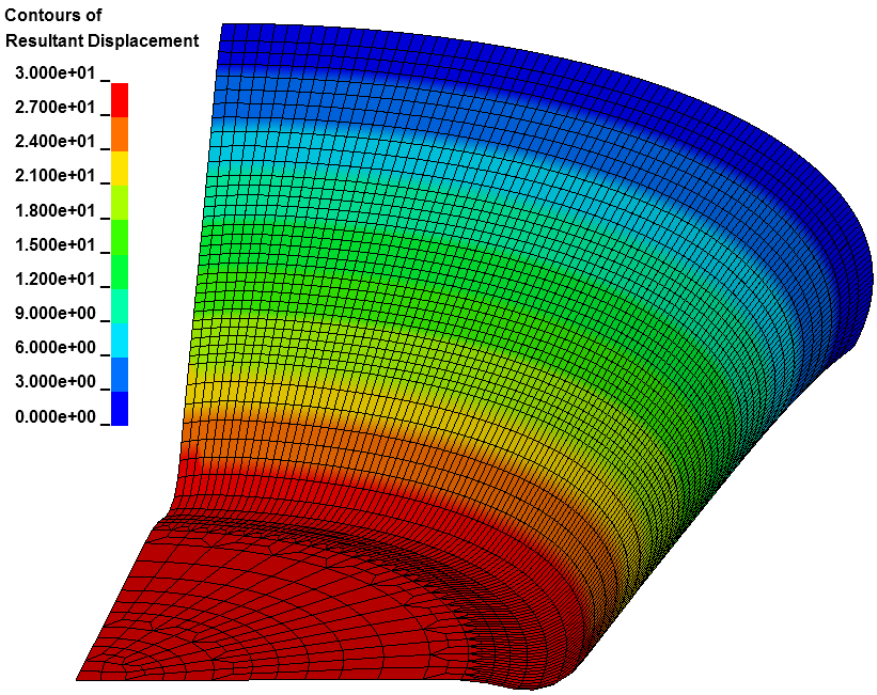


Figure 4.6: Norm of the displacement of a 50-degree cone – partial model

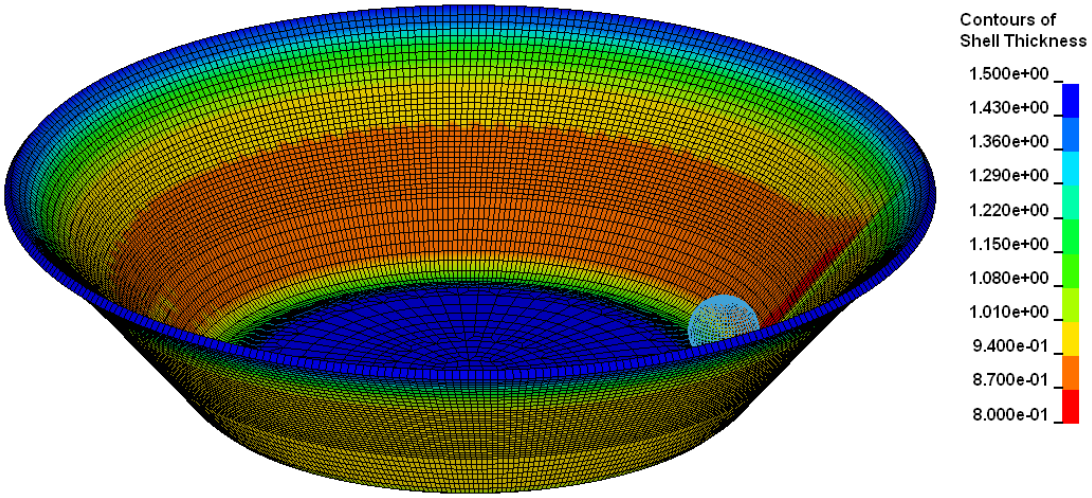


Figure 4.7: Thickness of a 50-degree cone – global model

4.2 Conclusions

As illustrated in the previous section, the computation time of SPIF simulations can be enormous, especially if no simplification is applied.

Several authors claim that an explicit strategy can decrease the computation time, as explained in the literature review in Section 2.3.1. At the beginning of this thesis, the LAGAMINE FE code was already able to perform dynamic explicit simulations but a major problem had to be solved in order to simulate the SPIF process: the existing contact element could not work with the explicit strategy and a new one had to be developed. At that stage, it was decided to develop a special contact element that would only work with the SPIF process but would be highly efficient in the search for the contact points and the accuracy of their location, since the position of the tool and its geometry is always known in advance.

The details of these developments will be explained in the next chapter.

Chapter 5

Moving Spherical Tool Method

The basic concepts of this new method were developed with the help of André Godinas.[1] This method was also the subject of the present author's Master's thesis, finished in September 2005 [HEN05a]. Although a few sections of this chapter are directly taken from that thesis, the presentation adopted here is slightly different and was chosen because it corresponds more accurately to the equations programmed into the code.

5.1 Structure of the Lagamine Explicit Code

Before going into detail on the new developments added to the finite element code Lagamine, this section will briefly summarize the structure of the code when working with the explicit strategy.

5.1.1 Equilibrium Equation

Within the Lagamine code, the general dynamic equilibrium equation is written at the *end* of the time step:

$$\underline{\underline{M}}\,\underline{\ddot{x}}_1 + \underline{\underline{C}}\,\underline{\dot{x}}_1 + \underline{F}_{int} = \underline{F}_{fixed} + \underline{F}_{contact} + \underline{F}_{ext} \tag{5.1}$$

where $\underline{\underline{M}}\,\underline{\ddot{x}}_1$ = inertia forces,
$\underline{\underline{C}}\,\underline{\dot{x}}_1$ = damping forces,
$\underline{F}_{int}$ = internal forces (integration of stresses),
$\underline{F}_{fixed}$ = supporting forces (reactions on fixed DOFs),
$\underline{F}_{contact}$ = contact forces,
$\underline{F}_{ext}$ = externally applied forces.

This equation is discretized at the nodes of the system. The left-hand side represents the total internal forces, which were split into the inertia forces (due to the acceleration of the nodal masses), the damping forces (due to velocity), and the internal forces (due to displacement). The latter term is the most difficult to compute. Starting from the nodal displacements, it involves computing the

[1] Godinas's report, written in French, is included in the appendix of an internal report [HEN05b, Appendix B].

strain at the integration points. Then, the stresses are computed (also at the integration points) using the constitutive law of the material. [2] Finally, those stresses are integrated over the element's volume (or surface for 2D elements) in order to compute the internal forces.

The right-hand side of the equation contains the load vector, which again was split into three components: the supporting forces (due to the fact that fixing a DOF introduces a force in the same direction), the contact forces (due to the contact between the tool and the sheet), and the externally applied forces.

Since Equation 5.1 is solved using an explicit integration scheme, the matrices $\underline{\underline{M}}$ and $\underline{\underline{C}}$ must be diagonal.

On the one hand, for the mass matrix $\underline{\underline{M}}$, it is quite intuitive to find a diagonal matrix, at least for the translational DOFs: each of them contains the mass of its corresponding node. For the rotational DOFs, however, it is more difficult. Each of them contains some kind of rotational inertia, which depends on the geometry of the element. For a more detailed explanation, the next chapter will show how to compute the complete mass matrix, starting from the interpolation functions of the element, and how to diagonalize that matrix.

On the other hand, the damping matrix $\underline{\underline{C}}$ is not always based on physical grounds and can be chosen in many different ways. In the LAGAMINE code, the simple approach chosen is either simply to ignore this term, which can sometimes cause instabilities, or to use proportional damping. In that case, $\underline{\underline{C}}$ is chosen to be equal to a linear combination of the mass matrix and the elastic stiffness matrix:

$$\underline{\underline{C}} = \mathrm{c}_1 \, \underline{\underline{M}} + \mathrm{c}_2 \, \underline{\underline{K}}^e,$$

both matrices being diagonal and computed once at the beginning of the simulation.[3].

5.1.2 Time Step Algorithm

The computation of a time step is schematically represented in Figure 5.1. At the beginning of each time step, the complete state of the system is known, i.e., the position $\underline{x}_0$, the velocity $\dot{\underline{x}}_0$ and the acceleration $\ddot{\underline{x}}_0$ of every DOF. The position at the end of the time step is computed using a simple extrapolation rule:

$$\underline{x}_1 = \underline{x}_0 + \dot{\underline{x}}_0 \, \Delta t + \ddot{\underline{x}}_0 \, \frac{\Delta t^2}{2}, \tag{5.2}$$

[2] In non-linear FE codes, every quantity is computed by increments. The stresses do not depend on the position at a given time, but rather on the history of the displacements, i.e., the sum of all the small displacement increments starting with the initial state of the material. The input of the constitutive law is then the strain rate — or the velocity gradient — and the time step Δt instead of the position. At each time step, the variations in stresses and internal variables over the time step are computed and added to the total stresses and internal variables.

[3] For the shell element COQJ4, these two matrices are functions of the orientation of the element. Therefore, they are computed in the local axes of the elements in the preprocessor and need to be rotated in the global axes at every time step within the LAGAMINE code. The choice of the initial elastic stiffness matrix results from the fact that the complete stiffness matrix is not computed when using an explicit strategy

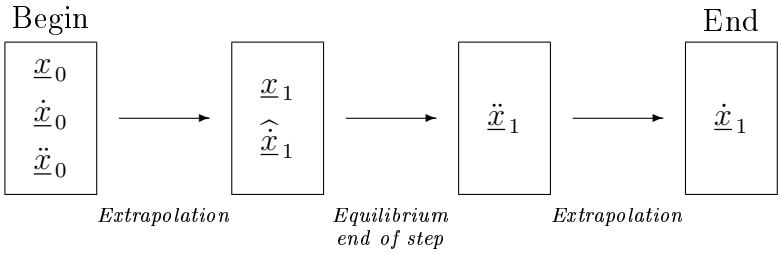


Figure 5.1: Time step in a dynamic explicit strategy

where Δt is the time step. An estimate of the final velocity is also computed using the following equation:

$$\hat{\dot{\underline{x}}}_1 = \dot{\underline{x}}_0 + \ddot{\underline{x}}_0 \, \Delta t. \tag{5.3}$$

The hat above the variable $\dot{\underline{x}}_1$ is there as a reminder that it is only an estimate and that this variable will be recomputed later. The reason for that is that this quantity is needed to compute the damping term when solving the equilibrium equation. Moreover, the end-of-step velocity is also needed to compute the contact forces, as will be explained in the next section.

Using the end-of-step position $\underline{x}_1$, the strains can be computed, which, in turn, are used to obtain the stresses using the constitutive laws of the material. The integration of these stresses provides the internal forces $\underline{F}_{int}$. If the three force components of the right-hand side of the equilibrium equation are also given, the only unknown in the equilibrium equation (5.1) is the final acceleration $\ddot{\underline{x}}_1$, which can be computed using all the other terms and the inverse of the diagonal mass matrix $\underline{\underline{M}}$:

$$\ddot{\underline{x}}_1 = \underline{\underline{M}}^{-1} \cdot \left(-\underline{\underline{C}} \, \hat{\dot{\underline{x}}}_1 - \underline{F}_{int} + \underline{F}_{fixed} + \underline{F}_{contact} + \underline{F}_{ext} \right). \tag{5.4}$$

This system of equations is diagonal and does not need any complicated matrix computation. Each DOF can be computed independently.

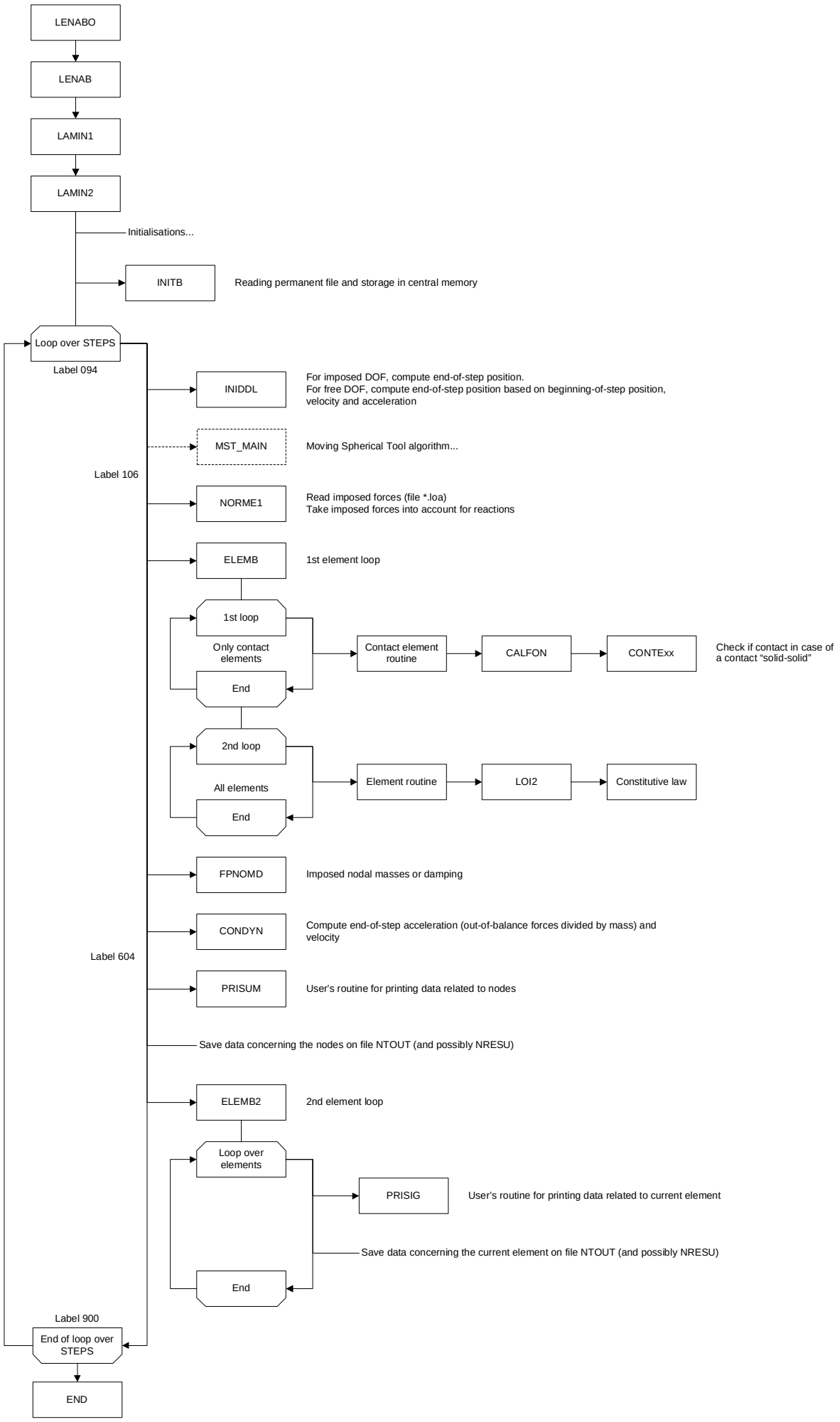
Finally, the end-of-step velocity $\dot{\underline{x}}_1$ can be computed using the linear acceleration method (simplified version of the Wilson θ method where $\theta = 1$), for which the acceleration is assumed to vary linearly over the time step [BATH76, p. 319]. This method is a simple extrapolation algorithm using the average acceleration over the time step instead of the initial acceleration:

$$\dot{\underline{x}}_1 = \dot{\underline{x}}_0 + \frac{\ddot{\underline{x}}_0 + \ddot{\underline{x}}_1}{2} \cdot \Delta t. \tag{5.5}$$

5.1.3 Lagamine Routines

The flow chart presenting the general architecture of the code in a dynamic explicit strategy is shown on the next page. It presents the main routines used, along with a short description of each of them. The place where the new routines are added is called `MST_MAIN`. This flow chart is a simplified version of the global flow chart of the Lagamine code, which is valid for both the implicit and explicit strategies and is found in the user manuals of the code.[4]

[4] See file "`Manuels\Lagamine\Organigramme\Lagamine_routines_en.pdf`"


LENABO
LENAB
LAMIN1
LAMIN2
Initialisations...
INITB
Reading permanent file and storage in central memory
Loop over STEPS
Label 094
INIDDL
For imposed DOF, compute end-of-step position.
For free DOF, compute end-of-step position based on beginning-of-step position, velocity and acceleration
MST_MAIN
Moving Spherical Tool algorithm...
Label 106
NORME1
Read imposed forces (file *.loa)
Take imposed forces into account for reactions
ELEMB
1st element loop
1st loop
Only contact elements
End
Contact element routine
CALFON
CONTExx
Check if contact in case of a contact "solid-solid"
2nd loop
All elements
End
Element routine
LOI2
Constitutive law
FPNOMD
Imposed nodal masses or damping
CONDYN
Compute end-of-step acceleration (out-of-balance forces divided by mass) and velocity
Label 604
PRISUM
User's routine for printing data related to nodes
Save data concerning the nodes on file NTOUT (and possibly NRESU)
ELEMB2
2nd element loop
Loop over elements
PRISIG
User's routine for printing data related to current element
Save data concerning the current element on file NTOUT (and possibly NRESU)
End
Label 900
End of loop over STEPS
END

5.2 General Principal of the New Method

The new method added to the Lagamine code is called the *Moving Spherical Tool* (MST), because it computes the contact forces between a spherical tool and a metal sheet.

As presented in the previous section, the equilibrium equation is written at the end of the time step. When solving this equation, it was assumed that all the external forces were known, and that the only unknown was the acceleration. This is the case for all types of loading, except in the case of contact forces with the spherical tool. With a classic contact element, these are computed by multiplying a penalty coefficient by the distance of penetration of the tool inside the metal sheet. The approach used here is different and consists in computing the contact forces *a posteriori* in order to meet suitable contact conditions at the next time step.

The main idea of the Moving Spherical Tool method is to split the computation of the acceleration $\underline{\ddot{x}}_1$ into two parts:

$$\underline{\ddot{x}}_1 = \underbrace{\underline{\underline{M}}^{-1} \cdot \left(-\underline{\underline{C}}\, \underline{\widehat{\dot{x}}}_1 - \underline{F}_{int} + \underline{F}_{fixed} + \underline{F}_{ext} \right)}_{\underline{\ddot{x}}_1^*} + \underbrace{\underline{\underline{M}}^{-1} \cdot \underline{F}_{contact}}_{\underline{\Delta\ddot{x}}_1}, \tag{5.6}$$

which are computed at different stages of the time step.

The first part of the acceleration $\underline{\ddot{x}}_1^*$ contains all the terms of the equilibrium equation except the contact forces. This is symbolized by an asterisk in superscript. The size of this vector is the number of DOFs in the whole structure. The second part $\underline{\Delta\ddot{x}}_1$ deals solely with the contact forces, and therefore contains the DOFs of the elements that are in contact with the tool (or the elements directly adjacent to them) only.

The goal of the method is to *choose the contact forces necessary to achieve a physically acceptable relative position and velocity between the tool and the sheet at the next step, i.e., a position for which the interpenetration is exactly zero and Coulomb's friction law is satisfied.* Ignoring the contact forces, the first part can be computed by the Lagamine code without any change to its classic algorithm, as illustrated in Figure 5.2. Since the computation of the second part involves the position and velocity of the tool and the sheet one step ahead, this can be done efficiently at the beginning of the next step as will be explained below.

At the beginning of the next step, the end-of-step position and an estimate of the end-of-step velocity will be computed, using the equations

$$\underline{x}_2^* = \underline{x}_1 + \underline{\dot{x}}_1^*\, \Delta t + \underline{\ddot{x}}_1^*\, \frac{\Delta t^2}{2} \tag{5.7}$$

and

$$\underline{\widehat{\dot{x}}}_2^* = \underline{\dot{x}}_1^* + \underline{\ddot{x}}_1^*\, \Delta t. \tag{5.8}$$

Once again, the asterisk means that the contact forces are not taken into account; the hat represents an estimate of the variable.

For a given modification of the initial acceleration $\underline{\Delta\ddot{x}}_1$ due to contact forces

$$\underline{\ddot{x}}_1 = \underline{\ddot{x}}_1^* + \underline{\Delta\ddot{x}}_1,$$

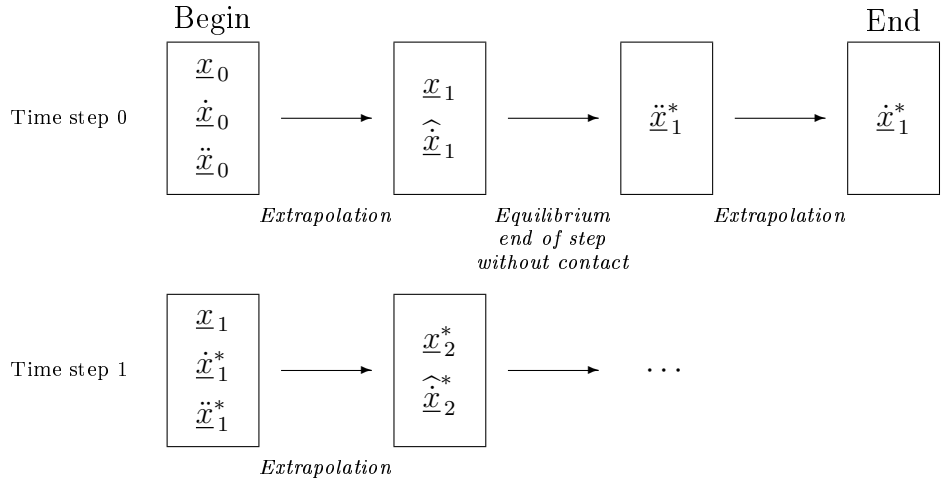


Figure 5.2: Two consecutive time steps in a dynamic explicit strategy without contact forces

the initial velocity is modified by $\underline{\Delta\dot{x}}_1$

$$\dot{\underline{x}}_1 = \dot{\underline{x}}^*_1 + \underline{\Delta\dot{x}}_1$$

with

$$\underline{\Delta\dot{x}}_1 = \underline{\Delta\ddot{x}}_1 \frac{\Delta t}{2},$$

as shown by equation (5.5). Consequently, the final position and velocity are modified as follows:

$$\underline{\Delta x}_2 = \underline{\Delta\dot{x}}_1 \, \Delta t + \underline{\Delta\ddot{x}}_1 \frac{\Delta t^2}{2} = \underline{\Delta\ddot{x}}_1 \, \Delta t^2 \tag{5.9}$$

and

$$\underline{\Delta\hat{\dot{x}}}_2 = \underline{\Delta\dot{x}}_1 + \underline{\Delta\ddot{x}}_1 \, \Delta t = \underline{\Delta\ddot{x}}_1 \frac{3}{2}\Delta t. \tag{5.10}$$

To summarize, the sensibility of the final position to a modification of the initial acceleration is equal to

$$\frac{\partial \underline{x}_2}{\partial \underline{\Delta\ddot{x}}_1} = \frac{\underline{\Delta x}_2}{\underline{\Delta\ddot{x}}_1} = \Delta t^2 \tag{5.11}$$

and, for the final velocity, the sensibility is equal to

$$\frac{\partial \hat{\dot{\underline{x}}}_2}{\partial \underline{\Delta\ddot{x}}_1} = \frac{\underline{\Delta\hat{\dot{x}}}_2}{\underline{\Delta\ddot{x}}_1} = \frac{3}{2}\Delta t. \tag{5.12}$$

Once the desired relative position and velocity between the tool and the sheet are known, finding which contact forces are needed is simple using these sensibilities.

5.3 Detailed Algorithm of the New Method

This section will focus on the details of the algorithm implemented, in particular the search for the contact points using an oil-stain method and the adjustment of the contact forces in order to fulfill the geometric conditions. A description of each routine added in the code is also given in Appendix A.

5.3.1 Contact Search

During the incremental forming process, the tool moves around the metal sheet. The location of the contact point(s) changes constantly but follows the same trend as the tool path. Therefore, the contact search should be focused around the tool center. The strategy adopted for this search is given in the following paragraphs.

a. Neighborhood

The first step is to select a neighborhood around the position of the tool center at the end of the time step (with the contact forces still unknown), i.e., a set of elements satisfying a proximity condition. This condition compares the distance from the tool center to a function of the characteristic size of the elements and the tool radius:

$$D^2 \leq \mathrm{s}\left(L^2 + R^2\right) \tag{5.13}$$

where D is the minimum distance between the tool center and the four nodes of the contact element;
L is the length of the longest diagonal of the element;
R is the radius of the tool;
s is a coefficient adjusting the size of the neighborhood and is usually chosen to be 1.5.

The size of the neighborhood will increase if the tool radius or the element size increases. An example of neighborhood (with $\alpha = 1.5$) is shown for three different tool positions in Figure 5.3.

Instead of inspecting each element of the metal sheet one by one, an oil-stain strategy was adopted.[5] This strategy has three simple rules. First, it starts by inspecting the elements that were in the neighborhood at the previous time step. Then, when an element satisfies the proximity criteria, it is added to the current neighborhood and its neighbors are also inspected. Finally, the algorithm stops when no neighbor satisfies the criteria any longer. The algorithm is very efficient since only a small number of elements are inspected.

b. Potential contact points

The next step is to search for potential contact points among the elements in the neighborhood. These points can be anywhere inside, on an edge, or on a corner of the elements. Their distance with respect to the tool center does not matter here.

As illustrated on a 2D example in Figure 5.4, a potential contact point on an element is defined as the point where the perpendicular to the element's surface goes through the tool center. If the elements are considered infinite, each element has one and only contact point.[6] Since they have a finite size, two cases are

[5] It is called an oil-stain method because the elements are added from neighbor to neighbor, hence the neighborhood grows like a stain that spreads over the metal sheet.

[6] For shell elements, since their surface is quadratic, this statement might not be true. In order to avoid problems, it will be assumed that the curvature of the elements will always remain smaller than the tool radius. This is a reasonable assumption since the out-of-plane displacement of the shell elements (and then the curvature) is supposed to be a small correction to the approximate linear geometry.

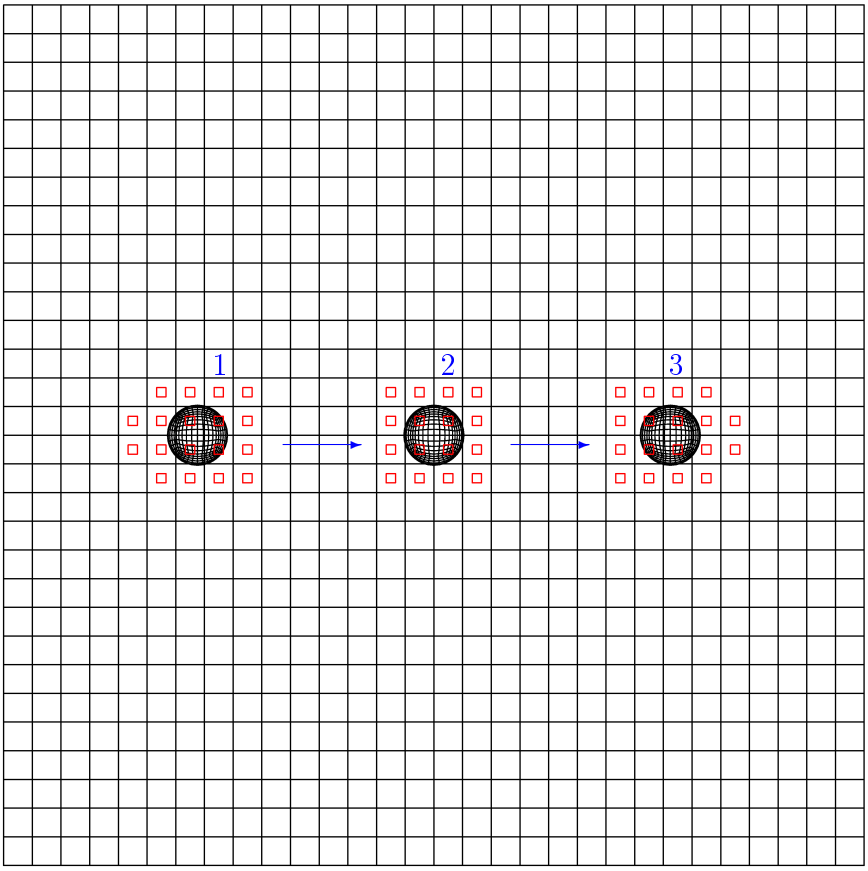


Figure 5.3: Elements in the neighborhood for three different tool positions

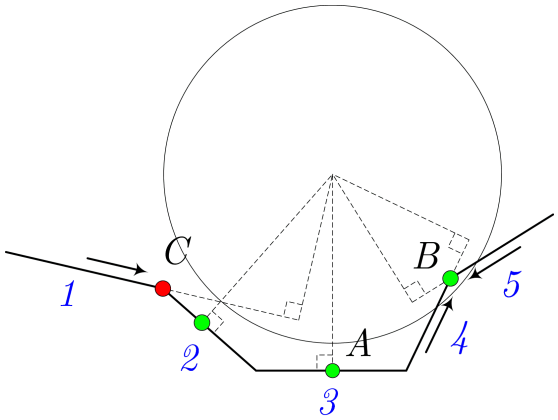


Figure 5.4: Definition of the potential contact points and the normal direction

possible. On the one hand, if the potential contact point lies inside the element's boundaries — like point A for element 3 in Figure 5.4 — it is considered a valid point. On the other hand, if it is outside the element's boundaries, this point is projected to the closest point along the boundary. In order to decide if the point is valid or not, all the elements adjacent to the boundary (edge or corner) must be considered. If all the adjacent elements indicate that the contact is on the boundary — like elements 4 and 5, which indicate a contact in B — the point is valid. The normal in that case is the direction from the contact point to the tool center, even if this direction may not be perpendicular to any of the adjacent elements. If the adjacent elements do not indicate the same point — for example, element 1 indicates a contact in C while element 2 does not — the point is not valid.

Those points are called *potential contact points* because it is not guaranteed that they will be in contact with the tool. Their validity in that case will depend on their distance from the tool center. The only certainty is that *if* there is contact, it will happen in one (or several) of those points.

5.3.2 Shape of the System of Equations

As stated before, the algorithm has to adjust the intensity of the contact forces at the beginning of the time step (old contact points) in order to satisfy geometric conditions at the contact points at the end of the time step (new contact points). To achieve this, the algorithm has to solve a system of equations. It is therefore essential to have the same number of equations as unknowns, hence the same number of new contact points as old contact points.

In general, among the potential contact points, only a few will actually be in contact at the end of the time step. In most cases, these points are approximately located at the old contact points' locations, shifted in the same direction and distance as the tool displacement during the time step. This remark provides the algorithm with a strategy for selecting the new contact points among the potential contact points:

> *For each old contact point, one and only new contact point is selected. This is done by choosing the potential contact point nearest the old contact point location shifted in the same direction and distance as the tool displacement during the time step.*

The remaining potential contact points are temporarily ignored, as their contact situation will be analyzed at the end of the algorithm. If the tool penetrates the material at those points, they will be considered contact points for the next time step; if not, they are ignored. This mechanism is used to increase the number of contact points, which is essential especially at the beginning of the simulation. When a new contact point appears, there is always a delay of one time step between the detection of the contact and the instant when the algorithm considers it to be a contact point.

In the (rare) cases where there are fewer potential contact points than old contact points, a special treatment is performed. The algorithm simply ignores certain old contact points when establishing the one-to-one relation between the old and new contact points. The former contact forces will be set to zero. This

might happen, for instance, if two old contact points were close to each other on both sides of an edge, but become only one new contact point located on the edge because the angle between both elements had changed.

The geometric conditions at the new contact point only concern three quantities, which are dependent on the three components of both the position and the velocity vectors: the distance d with respect to the tool measured along the normal, which was defined in Section 5.3.1, and the relative velocity between the contact point on the metal sheet and on the tool surface in two orthogonal directions in the tangential plane (perpendicular to the normal direction[7]) v_s and v_t. These three quantities are expressed in a reference frame local to each element at the end of the time step.

For consistency reasons, the contact forces at the old contact points are also expressed in a reference frame local to the element at the beginning of the time step. These are the normal force, P, and the friction force in two orthogonal directions, F_s and F_t.

If n is the number of old contact points (which is also the number of new contact points given the one-to-one relationship established before), $3n$ equations can be written, giving the $3n$ geometric conditions of the new contact points at the end of the time step — $d^{(i)}$, $v_s^{(i)}$ and $v_t^{(i)}$ for $i = 1 \dots n$ — as a function of those same quantities before applying the contact forces — $d^{*(i)}$, $v_s^{*(i)}$ and $v_t^{*(i)}$ — and the values of the $3n$ contact forces applied on the old contact points — $P^{(j)}$, $F_s^{(j)}$ and $F_t^{(j)}$ for $j = 1 \dots n$.

In addition to this set of $3n$ equations, an additional unknown, the tool angular speed (or rotation speed) at the end of the time step ω_2, can be added along with an equation giving these quantities as a function of the angular speed at the beginning of the time step ω_1 and the $2n$ friction forces. This is only useful when this rotation is free. In the case where it is imposed by the machine's rotation speed ω, this equation is trivial, and equal to

$$\omega_2 = \omega_1 = \omega \tag{5.14}$$

at any time.

The set of $(3n + 1)$ equations is presented in Figure 5.5. The computation of the individual terms of the matrix is explained in the next section.

5.3.3 Computation of the Matrix

This section will focus on the computation of all the terms of the matrix shown in Figure 5.5, i.e., the matrix linking the contact forces on the old contact points to the positions and velocities of the new contact points. This requires a detailed description of the local reference frame $(\underline{r}, \underline{s}, \underline{t})$, in which the forces, positions and velocities of the contact points are expressed.

[7]This plane might be different from the mid-plane of the element, defined as the plane containing the two diagonals of the quadrilateral, for two reasons. First, in the case of shell elements, the element's surface is quadratic and has therefore an out-of-plane component, thereby changing the curvature and the orientation of the normal direction. Secondly, if the contact point is located on an edge or a corner, the normal direction is not normal to any of the elements as explained in Section 5.3.1.

$$
\left(\begin{array}{c|c}
\omega_2 & \\ \hline
d^{(1)} & \\
v_s^{(1)} & \\
v_t^{(1)} & \\ \hline
\vdots & \\ \hline
d^{(n)} & \\
v_s^{(n)} & \\
v_t^{(n)} &
\end{array}\right)
=
\left(\begin{array}{c|c}
0 & \\ \hline
d^{*(1)} & \\
v_s^{*(1)} & \\
v_t^{*(1)} & \\ \hline
\vdots & \\ \hline
d^{*(n)} & \\
v_s^{*(n)} & \\
v_t^{*(n)} &
\end{array}\right)
+
\left(\begin{array}{c|ccc|ccc|c|ccc}
1 & 0 & \frac{\partial\omega}{\partial F_s^{(1)}} & \frac{\partial\omega}{\partial F_t^{(1)}} & \cdots & 0 & \frac{\partial\omega}{\partial F_s^{(n)}} & \frac{\partial\omega}{\partial F_t^{(n)}} & \omega_1 \\ \hline
0 & \frac{\partial d^{(1)}}{\partial P^{(1)}} & \frac{\partial d^{(1)}}{\partial F_s^{(1)}} & \frac{\partial d^{(1)}}{\partial F_t^{(1)}} & \cdots & \frac{\partial d^{(1)}}{\partial P^{(n)}} & \frac{\partial d^{(1)}}{\partial F_s^{(n)}} & \frac{\partial d^{(1)}}{\partial F_t^{(n)}} & P^{(1)} \\
\frac{\partial v_s^{(1)}}{\partial\omega_1} & \frac{\partial v_s^{(1)}}{\partial P^{(1)}} & \frac{\partial v_s^{(1)}}{\partial F_s^{(1)}} & \frac{\partial v_s^{(1)}}{\partial F_t^{(1)}} & \cdots & \frac{\partial v_s^{(1)}}{\partial P^{(n)}} & \frac{\partial v_s^{(1)}}{\partial F_s^{(n)}} & \frac{\partial v_s^{(1)}}{\partial F_t^{(n)}} & F_s^{(1)} \\
\frac{\partial v_t^{(1)}}{\partial\omega_1} & \frac{\partial v_t^{(1)}}{\partial P^{(1)}} & \frac{\partial v_t^{(1)}}{\partial F_s^{(1)}} & \frac{\partial v_t^{(1)}}{\partial F_t^{(1)}} & \cdots & \frac{\partial v_t^{(1)}}{\partial P^{(n)}} & \frac{\partial v_t^{(1)}}{\partial F_s^{(n)}} & \frac{\partial v_t^{(1)}}{\partial F_t^{(n)}} & F_t^{(1)} \\ \hline
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \hline
0 & \frac{\partial d^{(n)}}{\partial P^{(1)}} & \frac{\partial d^{(n)}}{\partial F_s^{(1)}} & \frac{\partial d^{(n)}}{\partial F_t^{(1)}} & \cdots & \frac{\partial d^{(n)}}{\partial P^{(n)}} & \frac{\partial d^{(n)}}{\partial F_s^{(n)}} & \frac{\partial d^{(n)}}{\partial F_t^{(n)}} & P^{(n)} \\
\frac{\partial v_s^{(n)}}{\partial\omega_1} & \frac{\partial v_s^{(n)}}{\partial P^{(1)}} & \frac{\partial v_s^{(n)}}{\partial F_s^{(1)}} & \frac{\partial v_s^{(n)}}{\partial F_t^{(1)}} & \cdots & \frac{\partial v_s^{(n)}}{\partial P^{(n)}} & \frac{\partial v_s^{(n)}}{\partial F_s^{(n)}} & \frac{\partial v_s^{(n)}}{\partial F_t^{(n)}} & F_s^{(n)} \\
\frac{\partial v_t^{(n)}}{\partial\omega_1} & \frac{\partial v_t^{(n)}}{\partial P^{(1)}} & \frac{\partial v_t^{(n)}}{\partial F_s^{(1)}} & \frac{\partial v_t^{(n)}}{\partial F_t^{(1)}} & \cdots & \frac{\partial v_t^{(n)}}{\partial P^{(n)}} & \frac{\partial v_t^{(n)}}{\partial F_s^{(n)}} & \frac{\partial v_t^{(n)}}{\partial F_t^{(n)}} & F_t^{(n)}
\end{array}\right)
\cdot
\left(\begin{array}{c|c}
\omega_1 & \\ \hline
P^{(1)} & \\
F_s^{(1)} & \\
F_t^{(1)} & \\ \hline
\vdots & \\ \hline
P^{(n)} & \\
F_s^{(n)} & \\
F_t^{(n)} &
\end{array}\right)
$$

Figure 5.5: System of equations

Let $C^{(i)}$ be a contact point located at latitude $\varphi^{(i)}$ on the surface on the tool, as shown in Figure 5.6. The first base vector, $\underline{r}$, goes from the contact point toward the tool center. The second base vector, $\underline{s}$, is perpendicular to $\underline{r}$ and lies in the plane containing $\underline{r}$ and the tool rotation axis, which is assumed to be $\underline{z}$ here. Vector $\underline{s}$ is thus tangent to a meridian. The third base vector, $\underline{t}$, is perpendicular to both other vectors and is defined by

$$\underline{t} = \underline{r} \wedge \underline{s}.$$

This last vector is thus tangent to a parallel.

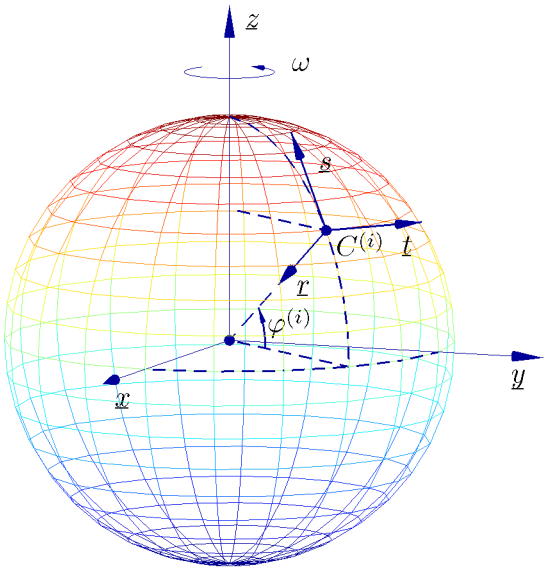


Figure 5.6: Local axis at the contact point

a. First Line of the Matrix

The *first line* of the matrix contains the equilibrium of momentum about the tool axis, which is assumed to be $\underline{z}$:

$$\sum_{i=1}^{n} \left(\underline{F}^{(i)} \wedge \underline{R}^{(i)} \right) \cdot \underline{z} = I \, \frac{d\omega}{dt} = I \, \dot{\omega} \tag{5.15}$$

where n is the number of (old) contact points, I is the inertia of the tool (mounted on the machine) about its rotation axis, $\dot{\omega}$ is the angular acceleration of the tool, $\underline{F}^{(i)}$ is the i^{th} friction force, $\underline{R}^{(i)}$ is the vector going from the tool center to the point of application of this force on the tool surface and $\underline{z}$ is a unit vector pointing in the direction of the tool axis, going from the tool center toward the machine.

If the vectors are expressed in the local reference frame $(\underline{r}, \underline{s}, \underline{t})$, they become

$$\underline{F}^{(i)} = \left(F_r^{(i)}, F_s^{(i)}, F_t^{(i)} \right) \tag{5.16}$$

and

$$\underline{R}^{(i)} = \left(R_r^{(i)}, R_s^{(i)}, R_t^{(i)} \right) = -R \cdot \underline{r}, \tag{5.17}$$

where R is the tool radius.

For a given i, the term in the sum of the left-hand side of Equation 5.15 becomes

$$\left(\underline{F}^{(i)} \wedge \underline{R}^{(i)}\right) \cdot \underline{z} = R\left(F_s^{(i)}\underline{t} - F_t^{(i)}\underline{s}\right) \cdot \underline{z}. \tag{5.18}$$

Since $\underline{t}$ is orthogonal to $\underline{z}$ and $\underline{s} \cdot \underline{z} = \cos\varphi^{(i)}$, Equation 5.18 becomes

$$\left(\underline{F}^{(i)} \wedge \underline{R}^{(i)}\right) \cdot \underline{z} = R\cos\varphi^{(i)}\ F_t^{(i)}. \tag{5.19}$$

Finally, the integration of Equation 5.15 over one time step gives

$$\omega_2 = \omega_1 + \frac{R}{I}\sum_{i=1}^{n}\cos\varphi^{(i)}\ F_t^{(i)}\,\Delta t. \tag{5.20}$$

The terms of the first line of the matrix are then equal to:

$$\frac{\partial\omega}{\partial F_s^{(i)}} = 0 \tag{5.21a}$$

$$\frac{\partial\omega}{\partial F_t^{(i)}} = \frac{R\cos\varphi^{(i)}\Delta t}{I}. \tag{5.21b}$$

b. First Column of the Matrix

The *first column* of the matrix, which deals with the influence of rotation speed of the tool on the relative velocities $v_s^{(i)}$ and $v_t^{(i)}$, is easy to compute. The equation linking the rotational velocity ω and the tangential velocity $v_t^{(i)}$ is

$$v_t^{(i)} = R\cos\varphi^{(i)}\ \omega. \tag{5.22}$$

The partial derivatives are then given by

$$\frac{\partial v_s^{(i)}}{\partial\omega_1} = 0 \tag{5.23a}$$

$$\frac{\partial v_t^{(i)}}{\partial\omega_1} = R\cos\varphi^{(i)}. \tag{5.23b}$$

c. Other Terms of the Matrix

The other terms of the matrix involve one new contact point i and one old contact point j. For a given i and j, the terms of the matrix form a sub-matrix:

$$\begin{pmatrix} \dfrac{\partial d^{(i)}}{\partial P^{(j)}} & \dfrac{\partial d^{(i)}}{\partial F_s^{(j)}} & \dfrac{\partial d^{(i)}}{\partial F_t^{(j)}} \\ \dfrac{\partial v_s^{(i)}}{\partial P^{(j)}} & \dfrac{\partial v_s^{(i)}}{\partial F_s^{(j)}} & \dfrac{\partial v_s^{(i)}}{\partial F_t^{(j)}} \\ \dfrac{\partial v_t^{(i)}}{\partial P^{(j)}} & \dfrac{\partial v_t^{(i)}}{\partial F_s^{(j)}} & \dfrac{\partial v_t^{(i)}}{\partial F_t^{(j)}} \end{pmatrix}. \tag{5.24}$$

For all these nine terms, the computation is similar. Therefore, it will only be illustrated here for the first one $\partial d^{(i)}/\partial P^{(j)}$. This term represents the influence of a normal force located on the j^{th} old contact point on the distance between the i^{th} new contact point and the tool center. The computation is performed in several stages. First, an arbitrary normal force $\Delta P^{(j)}$ is placed at the old contact point. Using the interpolation functions of the element, four energetically-equivalent nodal forces $F_n^{(j)}$ can be computed:

$$F_n^{(j)} = \Phi^n(\xi^{(j)}, \eta^{(j)})\Delta P^{(j)},\ n = 1 \ldots 4, \tag{5.25}$$

where $\Phi^n(\xi^{(j)}, \eta^{(j)})$ is the n^{th} interpolation function of element j evaluated at the point where the force $P^{(j)}$ is applied. These forces induce a modification of the initial acceleration of those nodes $\underline{\Delta\ddot{x}}_{n,1}$, as shown by Equation (5.6), which in turn causes a modification of the final position $\underline{\Delta x}_{n,2}$, as shown by Equation (5.9). If elements i and j have at least one node in common, these modifications will have an influence on the position of the i^{th} new contact point $\underline{\Delta x}_c^{(i)}$, which can be computed using the interpolation functions of the i^{th} element:

$$\underline{\Delta x}_c^{(i)} = \sum_{n=1}^{4} \Phi^n(\xi_c^{(i)}, \eta_c^{(i)})\ \underline{\Delta x}_{n,2} \tag{5.26}$$

where the interpolation functions are evaluated at the local coordinate of the new contact point i. The modification of the normal distance due to the arbitrary normal force is then given by this vector $\underline{\Delta x}_c^{(i)}$ projected along the normal unit vector $\hat{n}^{(i)}$:

$$\Delta d^{(i)} = \underline{\Delta x}_c^{(i)} \cdot \hat{n}^{(i)}. \tag{5.27}$$

The term is finally equal to

$$\frac{\partial d^{(i)}}{\partial P^{(j)}} = \frac{\Delta d^{(i)}}{\Delta P^{(j)}}. \tag{5.28}$$

5.3.4 Types of Contact

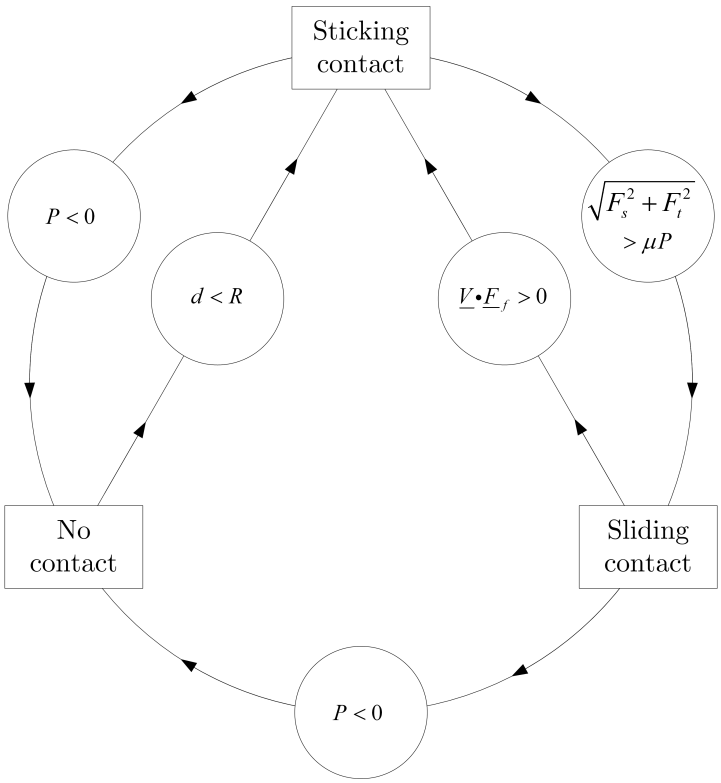
In general, three different types of contact may occur: a sticking contact, a sliding contact or no contact. According to the type of contact, the constraints that must be set by the equations are different.

1. For a sticking contact, the distance d to the tool center must be equal to the tool radius and the relative velocities v_s and v_t of the contact point on the metal sheet and on the tool must be zero. The normal contact force P must be positive and the norm of the friction force must be smaller than the critical value given by Coulomb's law, i.e., $F_f = \sqrt{F_s^2 + F_t^2} \leq \mu P$ where μ is the friction coefficient.
2. For a sliding contact, the distance d must also be equal to the tool radius. The normal contact force P must be positive and the friction forces must be equal to the critical value given by Coulomb's law, i.e., $F_f = \mu P$. The direction of the relative velocity vector must be approximately opposite to the direction of the friction force vector; in other words, their scalar product must be negative.

3. If there is no contact, the contact forces are set to zero. In that case, the distance d must be larger than the tool radius.

The problem with this algorithm is that it does not know in advance which contact condition must be applied to each of the contact points. As a result, an iterative strategy was adopted.

For the first iteration, a sticking contact is assumed for every contact point. Then, from one iteration to the other, the hypothesis on each contact point may vary independently. At the end of each iteration, a test is performed for each contact point depending on its current type of contact. If the test is not satisfied, the hypothesis of this contact point is modified according to the rules defined in Figure 5.7. If at least one hypothesis is modified, a new iteration is performed. If not, the algorithm is terminated.



Contact	Hypothesis	Output	Tests
Sticking	$d = R$	P, F_s & F_t	If $P < 0 \Rightarrow$ No contact If $F_f > \mu P \Rightarrow$ Sliding Else OK
Sliding	$d = R$ $F_f = \mu P$	P, v_s & v_t	If $P < 0 \Rightarrow$ No contact If $\underline{V} \cdot \underline{F}_f > 0 \Rightarrow$ Sticking Else OK
No contact	$P = F_s = F_t = 0$	d	If $d < R \Rightarrow$ Sticking Else OK

Figure 5.7: Contact conditions

A sticking contact is the default case. The equations presented in the previous section correspond to this situation. The other two types of contact are slightly

different.

If no contact is assumed, the contact forces must be equal to zero and the three equations concerning this new contact point are replaced by the following ones:

$$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} P^{(1)} \\ F_s^{(1)} \\ F_t^{(1)} \end{pmatrix}. \tag{5.29}$$

For a sliding contact, modifying the equation is more difficult. Indeed, the constraints are that the distance must be equal to the radius of the tool and the norm of the friction forces $\sqrt{F_s^2 + F_t^2}$ must be equal to the critical value μP. Since this condition can not be applied with a linear system of equations, the solution that was finally chosen was to impose that:

$$F_s = \mu P \frac{F_s^*}{\sqrt{(F_s^*)^2 + (F_t^*)^2}} \tag{5.30a}$$

$$F_t = \mu P \frac{F_t^*}{\sqrt{(F_s^*)^2 + (F_t^*)^2}} \tag{5.30b}$$

where F_s^* and F_t^* are the values of the friction forces obtained in the previous iteration. As shown in Figure 5.7, the contact state at the previous iteration can only be either a sticking or a sliding contact. This condition is then equivalent to imposing a norm of the friction force equal to the critical value and oriented in the direction obtained in the previous iteration. In other words, the magnitude of the friction force is scaled down to the correct value without changing its direction. Equations 5.30 can be imposed in the linear system of equations by changing two of the three equations for this new contact point:

$$\begin{pmatrix} d^{(i)} \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} d^{*(i)} \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} \frac{\partial d^{(1)}}{\partial P^{(1)}} & \frac{\partial d^{(1)}}{\partial F_s^{(1)}} & \frac{\partial d^{(1)}}{\partial F_t^{(1)}} \\ \frac{-\mu F_s^*}{\sqrt{(F_s^*)^2+(F_t^*)^2}} & 1 & 0 \\ \frac{-\mu F_t^*}{\sqrt{(F_s^*)^2+(F_t^*)^2}} & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} P^{(1)} \\ F_s^{(1)} \\ F_t^{(1)} \end{pmatrix} \tag{5.31}$$

where $d^{(i)}$ must be equal to the tool radius R. After the resolution of the system, the algorithm must check that the velocity of the contact point is indeed in the direction opposite to the friction force. This is achieved by evaluating the scalar product between the velocity $\underline{V} = [v_s, v_t]^T$ and the friction force $\underline{F}_f = [F_s, F_t]^T$, which should be negative:

$$\underline{V} \cdot \underline{F}_f \leq 0. \tag{5.32}$$

If this is not the case, the friction force is too high and the hypothesis on the type of contact should be changed to a sticking contact.

This iterative process should converge after a few iterations. If this is not the case, a small interpenetration is allowed for the non-contact case.

Chapter 6

Mass Matrix of the COQJ4 Shell Element

The goal of this chapter is to check the computation of the mass matrix programmed inside Lagamine's shell element. Indeed, as will be shown in the next chapter, some simulations experienced convergence problems and it was suspected that the problem stemmed from the rotational DOFs of the mass matrix, whose computation was uncertain.

The dynamic explicit code only uses a diagonal mass matrix. In order to verify this matrix, it was decided first to recompute the complete matrix starting from the interpolation functions of the element. Then, using diagonalization techniques, the diagonal matrix was derived from the previous result.

Before diving into the details of the interpolation functions of the COQJ4 shell element, the first section summarizes the global ideas behind the computation of a mass matrix and illustrates the computation on a hypothetical four-node element having only three DOFs per node.

The content of this chapter can also be found in an internal report from the University of Liège [HEN07b].

6.1 Simplified Mass Matrix

Let $\underline{x}$ be the position vector of a point inside an element E, $\underline{\dot{x}}$ its velocity, and $\underline{\ddot{x}}$ its acceleration. Finally, let $\underline{\delta x}$ be a virtual displacement, which produces a virtual strain increment of $\underline{\underline{\delta\epsilon}}$. The virtual work principle applied to the volume of element E states that

$$\int_V \underline{\underline{\sigma}} : \underline{\underline{\delta\epsilon}}\, dv + \int_V (\rho\underline{\ddot{x}}) \cdot \underline{\delta x}\, dv = \int_V \underline{F} \cdot \underline{\delta x}\, dv. \tag{6.1}$$

Using the index notation, Equation 6.1 becomes[1]

$$\underbrace{\int_V \sigma_{ij}\,\delta\epsilon_{ij}\,dv}_{\textit{Internal forces}} + \underbrace{\int_V (\rho\ddot{x}_i)\,\delta x_i\,dv}_{\textit{Inertia term}} = \underbrace{\int_V F_i\,\delta x_i\,dv}_{\textit{External forces}}\,. \tag{6.2}$$

The basic principles of the finite element method require that the position vector x_i be expressed as a function of the nodal positions x_i^J of the four J nodes defining the element using the interpolation functions Φ^J:

$$x_i = \sum_{J=1}^{4} \Phi^J x_i^J. \tag{6.3}$$

This equation is valid for a simple four-node element with three DOFs per node. For the COQJ4 element, which has six DOFs per node and more complex interpolation functions, the detailed computation will be performed in the next section.

After taking its second derivative, Equation 6.3 becomes:

$$\ddot{x}_i = \sum_{J=1}^{4} \Phi^J \ddot{x}_i^J. \tag{6.4}$$

Similarly, the virtual displacement is equal to

$$\delta x_i = \sum_{J=1}^{4} \Phi^J \delta x_i^J. \tag{6.5}$$

Using Equations 6.4 and 6.5, the inertia term of Equation 6.2 becomes

$$I = \int_V (\rho\ddot{x}_i)\,\delta x_i\,dv = \int_V \rho \sum_{I=1}^{4}\sum_{J=1}^{4} \Phi^I\Phi^J \ddot{x}_i^I \delta x_i^J dv. \tag{6.6}$$

The notation of this expression can be vectorized if, for each combination of I and J, a new matrix $\underline{\underline{M}}^{IJ}$ is introduced:

$$\underline{\underline{M}}^{IJ} = \rho \begin{bmatrix} \Phi^I\Phi^J & 0 & 0 \\ 0 & \Phi^I\Phi^J & 0 \\ 0 & 0 & \Phi^I\Phi^J \end{bmatrix} = \rho\,\Phi^I\Phi^J \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{6.7}$$

In that case, Equation 6.6 can be written as

$$I = \int_V \sum_{I=1}^{4}\sum_{J=1}^{4} \left(\underline{\ddot{x}}^I\right)^T \underline{\underline{M}}^{IJ}\,\underline{\delta x}^J dv. \tag{6.8}$$

[1] As a reminder, unless explicitly specified, every index appearing twice in the same factor implies a summation over the three DOFs:

$$a_i b_i \triangleq \sum_{i=1}^{3} a_i b_i = \underline{a}\cdot\underline{b}.$$

Each of these $\underline{\underline{M}}^{IJ}$ matrices represents the contribution of the interaction between two terms — the acceleration of node I of the element $\underline{\ddot{x}}^I$, and the virtual displacement of node J, $\underline{\delta x}^J$ — to the computation of the virtual energy.

Finally, the nodal vectors $\underline{\ddot{x}}^I$ and $\underline{\delta x}^J$ can be stored in two (12×1) vectors,

$$\left[\underline{\ddot{x}}^N\right] = \left\langle\, \ddot{x}_1^1 \;\; \ddot{x}_2^1 \;\; \ddot{x}_3^1 \;\vdots\; \ddot{x}_1^2 \;\; \ddot{x}_2^2 \;\; \ddot{x}_3^2 \;\vdots\; \ddot{x}_1^3 \;\; \ddot{x}_2^3 \;\; \ddot{x}_3^3 \;\vdots\; \ddot{x}_1^4 \;\; \ddot{x}_2^4 \;\; \ddot{x}_3^4 \,\right\rangle^T \tag{6.9}$$

and

$$\left[\underline{\delta x}^N\right] = \left\langle\, \delta x_1^1 \;\; \delta x_2^1 \;\; \delta x_3^1 \;\vdots\; \delta x_1^2 \;\; \delta x_2^2 \;\; \delta x_3^2 \;\vdots\; \delta x_1^3 \;\; \delta x_2^3 \;\; \delta x_3^3 \;\vdots\; \delta x_1^4 \;\; \delta x_2^4 \;\; \delta x_3^4 \,\right\rangle^T, \tag{6.10}$$

where the super-script N stands for nodal unknowns, and the sixteen $\underline{\underline{M}}^{IJ}$ matrices can be stored in a (12×12) matrix $\underline{\underline{M}}$:

$$\underline{\underline{M}} = \left[\begin{array}{c|c|c|c} \underline{\underline{M}}^{11} & \underline{\underline{M}}^{12} & \underline{\underline{M}}^{13} & \underline{\underline{M}}^{14} \\ \hline \underline{\underline{M}}^{21} & \underline{\underline{M}}^{22} & \underline{\underline{M}}^{23} & \underline{\underline{M}}^{24} \\ \hline \underline{\underline{M}}^{31} & \underline{\underline{M}}^{32} & \underline{\underline{M}}^{33} & \underline{\underline{M}}^{34} \\ \hline \underline{\underline{M}}^{41} & \underline{\underline{M}}^{42} & \underline{\underline{M}}^{43} & \underline{\underline{M}}^{44} \end{array}\right]. \tag{6.11}$$

Using these new vectors and this matrix, the inertia term of the virtual work principle becomes

$$I = \int_V \left[\underline{\ddot{x}}^N\right]^T \underline{\underline{M}} \;\left[\underline{\delta x}^N\right] dv. \tag{6.12}$$

The matrix $\underline{\underline{M}}$ is called the mass matrix. Its complete form is shown in Figure 6.1.

$$M = \rho \left[\begin{array}{ccc|ccc|ccc|ccc} \Phi^1\Phi^1 & 0 & 0 & \Phi^1\Phi^2 & 0 & 0 & \Phi^1\Phi^3 & 0 & 0 & \Phi^1\Phi^4 & 0 & 0 \\ 0 & \Phi^1\Phi^1 & 0 & 0 & \Phi^1\Phi^2 & 0 & 0 & \Phi^1\Phi^3 & 0 & 0 & \Phi^1\Phi^4 & 0 \\ 0 & 0 & \Phi^1\Phi^1 & 0 & 0 & \Phi^1\Phi^2 & 0 & 0 & \Phi^1\Phi^3 & 0 & 0 & \Phi^1\Phi^4 \\ \hline \Phi^2\Phi^1 & 0 & 0 & \Phi^2\Phi^2 & 0 & 0 & \Phi^2\Phi^3 & 0 & 0 & \Phi^2\Phi^4 & 0 & 0 \\ 0 & \Phi^2\Phi^1 & 0 & 0 & \Phi^2\Phi^2 & 0 & 0 & \Phi^2\Phi^3 & 0 & 0 & \Phi^2\Phi^4 & 0 \\ 0 & 0 & \Phi^2\Phi^1 & 0 & 0 & \Phi^2\Phi^2 & 0 & 0 & \Phi^2\Phi^3 & 0 & 0 & \Phi^2\Phi^4 \\ \hline \Phi^3\Phi^1 & 0 & 0 & \Phi^3\Phi^2 & 0 & 0 & \Phi^3\Phi^3 & 0 & 0 & \Phi^3\Phi^4 & 0 & 0 \\ 0 & \Phi^3\Phi^1 & 0 & 0 & \Phi^3\Phi^2 & 0 & 0 & \Phi^3\Phi^3 & 0 & 0 & \Phi^3\Phi^4 & 0 \\ 0 & 0 & \Phi^3\Phi^1 & 0 & 0 & \Phi^3\Phi^2 & 0 & 0 & \Phi^3\Phi^3 & 0 & 0 & \Phi^3\Phi^4 \\ \hline \Phi^4\Phi^1 & 0 & 0 & \Phi^4\Phi^2 & 0 & 0 & \Phi^4\Phi^3 & 0 & 0 & \Phi^4\Phi^4 & 0 & 0 \\ 0 & \Phi^4\Phi^1 & 0 & 0 & \Phi^4\Phi^2 & 0 & 0 & \Phi^4\Phi^3 & 0 & 0 & \Phi^4\Phi^4 & 0 \\ 0 & 0 & \Phi^4\Phi^1 & 0 & 0 & \Phi^4\Phi^2 & 0 & 0 & \Phi^4\Phi^3 & 0 & 0 & \Phi^4\Phi^4 \end{array}\right]$$

Figure 6.1: Mass matrix of a four-node element with three DOFs per node

6.2 General Mass Matrix

In the previous section, the mass matrix was developed for a simplified case: a four-node element with three DOFs per node using only first-order interpolation functions. However, the shell element used in the LAGAMINE code is more complex than this simplified element.

In order to derive the general mass matrix for this shell element, this section will first introduce the interpolation functions of the shell element, then develop the consistent mass matrix, and finally simplify it to obtain the lumped diagonal mass matrix.

6.2.1 Interpolation Functions

The shell element available in LAGAMINE is called COQJ4. It consists of four nodes and uses the Discrete Kirchhoff Theory (DKT). Every node has six DOFs, i.e., three degrees of translation and three degrees of rotation. More details can be found in Section 3.1.2.

The geometry of this element is described using eight interpolation functions: four first-order interpolation functions that describe the global geometry of the shell (just as for any four-node element) and four second-order interpolation functions that describe the curvature of the element, namely, by adding a slight correction in the direction of the normal to the mid-plane of the element. The shell elements are then oriented, which means that their geometry depends on the direction of the normal and, therefore, requires the definition of a local reference frame for each element. These two kinds of interpolation functions are described further in the next sections.

a. First-Order Interpolation Functions

The reference element is presented in Figure 6.2. Its local coordinates are $-1 \leq \xi \leq +1$ and $-1 \leq \eta \leq +1$.

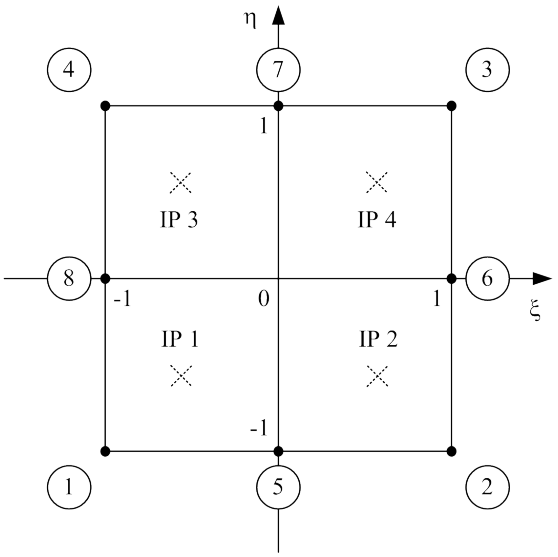


Figure 6.2: Reference element

The first-order interpolation functions $\Phi^I(\xi,\eta)$ are defined on the reference element. The mapping between the local coordinates of the reference element (ξ,η) and the global position vector $\underline{x}$ of any point inside an element is given by

$$\underline{x} = \sum_{I=1}^{4} \Phi^I(\xi,\eta)\, \underline{x}^I \tag{6.13}$$

where $\underline{x}^I$ is the position vector of node I of the element. The interpolation functions are bilinear functions which satisfy

$$\Phi^I(\xi,\eta) = \begin{cases} 1 & \text{at node } I \\ 0 & \text{at other nodes.} \end{cases} \tag{6.14}$$

The functions are plotted in Figure 6.3 and are equal to

$$\Phi^1 = \frac{(1-\xi)(1-\eta)}{4}; \tag{6.15a}$$

$$\Phi^2 = \frac{(1+\xi)(1-\eta)}{4}; \tag{6.15b}$$

$$\Phi^3 = \frac{(1+\xi)(1+\eta)}{4}; \tag{6.15c}$$

$$\Phi^4 = \frac{(1-\xi)(1+\eta)}{4}. \tag{6.15d}$$

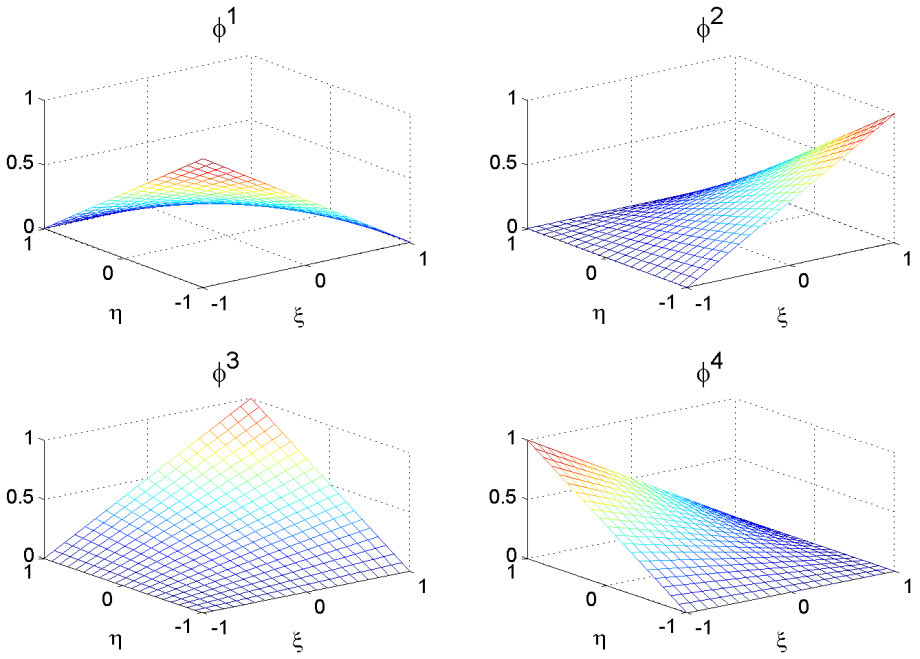


Figure 6.3: Interpolation functions of the corner nodes

b. Second-Order Interpolation Functions

The second-order interpolation functions are used to compute a correction to the geometry added in the direction of the normal to the mid-plane of the element.

The definition of this normal direction is not as straightforward as it may seem, since the element is not necessarily planar; in that case, it is not possible to find a plane that contains all four nodes.

The definition of the local axes of the element starts by choosing the reference plane of the element, defined as the plane containing the medians of the element, which are also equal to the derivative of the global coordinates with respect to the intrinsic coordinates $\partial\underline{x}/\partial\xi$ and $\partial\underline{x}/\partial\eta$. Since these two directions are not necessarily perpendicular to each other, they must be transformed into two new co-planar directions $\underline{R}$ and $\underline{S}$. Finally, the third direction $\underline{T}$ is chosen perpendicular to the first two axes:

$$\underline{T} = \underline{R} \wedge \underline{S}. \tag{6.16}$$

A detailed computation of the local axes is given in Appendix B.[2]

The formulation of the correction to the approximate geometry can be easily introduced if the rotational DOFs of the nodes are replaced by fictitious nodes placed at the midpoint of each edge. In that case, the global displacement field can be written as

$$\underline{x} = \sum_{I=1}^{4} \Phi^I(\xi,\eta)\,\underline{x}^I + \underline{T} \sum_{I=1}^{4} \Phi^{I+4}(\xi,\eta)\,w^{I+4}, \tag{6.17}$$

where w^{I+4} is the relative displacement, in the direction normal to the mid-plane of the element, of node $I+4$ compared to its initial position, which is located at midpoint of its two adjacent nodes I and $I+1$.[3] $\Phi^{I+4}(\xi,\eta)$ denotes the interpolation functions of the mid-side nodes $I+4$. These functions are given by

$$\Phi^5 = \frac{(1-\xi^2)(1-\eta)}{2}, \tag{6.18a}$$

$$\Phi^6 = \frac{(1-\eta^2)(1+\xi)}{2}, \tag{6.18b}$$

$$\Phi^7 = \frac{(1-\xi^2)(1+\eta)}{2}, \tag{6.18c}$$

$$\Phi^8 = \frac{(1-\eta^2)(1-\xi)}{2}, \tag{6.18d}$$

and are illustrated in Figure 6.4.

It can be shown [HEN05b, Appendix B] that the relation between the relative displacement of the mid-side node w^{I+4} of an edge and the three rotational DOFs of its adjacent nodes $\underline{\theta}^I$ and $\underline{\theta}^{I+1}$ is given by

$$w^{I+4} = \underbrace{\left[\underline{\theta}^{I+1} - \underline{\theta}^I\right]^T}_{(1\times3)} \cdot \underbrace{\left[\underline{I} - \underline{T}\underline{T}^T\right]}_{(3\times3)} \cdot \underbrace{\left[\underline{x}^{I+1} - \underline{x}^I\right]}_{(3\times1)} /8, \tag{6.19}$$

[2] This is also explained in an internal report [HEN07c] and in Kaiping Li's thesis [LI95, p. 3.11].

[3] For node I=1, 2, 3, and 4, the notation $(I+1)$ produces respectively 2, 3, 4, and 1. Therefore, a more accurate notation of node $(I+1)$ would be $((I \bmod 4)+1)$, since it can only take a value between 1 and 4.

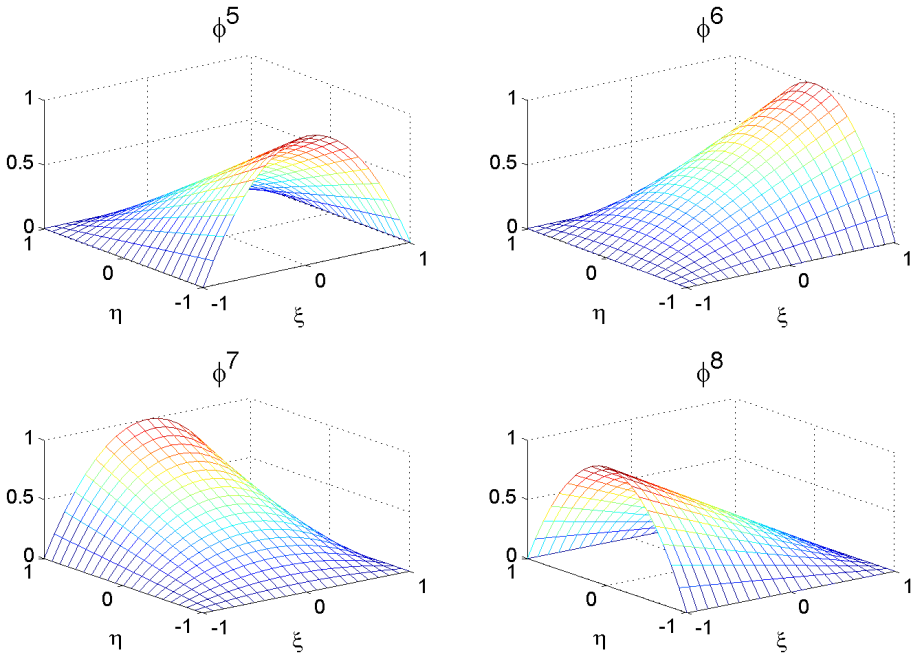


Figure 6.4: Interpolation functions of the mid-side nodes for the shell element

where $\underline{\underline{I}}$ is a unit tensor represented by ones on the diagonal and zeros everywhere else. It should be noted that w^{I+4} is independent of the choice of the directions $\underline{R}$ and $\underline{S}$, but depends only on the direction $\underline{T}$. In this formula, the vector $\underline{\theta}^I$ is given by

$$\underline{\theta}^I = \begin{bmatrix} \theta_1^I \\ \theta_2^I \\ \theta_3^I \end{bmatrix} = \begin{bmatrix} \varphi_2^I \\ -\varphi_1^I \\ \varphi_3^I \end{bmatrix}, \tag{6.20}$$

where
- φ_i^I is the rotation around the i-axis in the clockwise direction (this convention is called the right-hand-side rule);
- θ_1^I and θ_2^I are the rotations in planes (1,3) and (2,3) respectively, the positive rotations being the rotations for which the 3-axis comes closer to the 1- and 2-axes, respectively;
- θ_3^I is equal to φ_3^I.

The convention used for $\underline{\theta}^I$ is different from the classic right-hand-side rule as illustrated in Figure 6.5. This convention is often used in literature on shell elements [BATO92, LI95]. When writing the mass matrix in the next section, this will have to be taken into account since Lagamine always uses the right-hand-side convention.

c. Particular Case: Local Reference Frame

The development of the consistent mass matrix will be performed in the local reference frame. In that case, the expression of the interpolation functions can

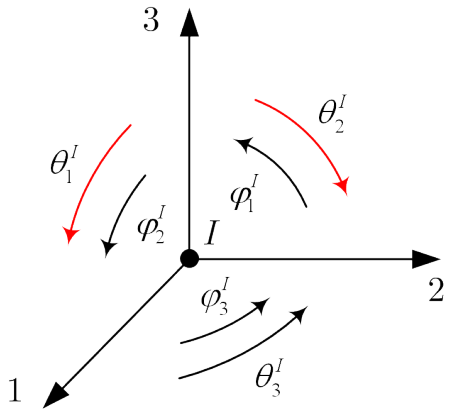


Figure 6.5: Rotation convention for the interpolation functions

be simplified. Indeed, the normal direction is always equal to

$$\underline{T} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{6.21}$$

Therefore, the three components of the vector Equation 6.17 become

$$\begin{cases} x_1 &= \displaystyle\sum_{I=1}^{4} \Phi^I(\xi,\eta)\, x_1^I \\ x_2 &= \displaystyle\sum_{I=1}^{4} \Phi^I(\xi,\eta)\, x_2^I \\ x_3 &= \displaystyle\sum_{I=1}^{4} \Phi^I(\xi,\eta)\, x_3^I + \sum_{I=1}^{4} \Phi^{I+4}(\xi,\eta)\, w^{I+4}. \end{cases} \tag{6.22}$$

The second derivatives of these equations are

$$\begin{cases} \ddot{x}_1 &= \displaystyle\sum_{I=1}^{4} \Phi^I(\xi,\eta)\, \ddot{x}_1^I \\ \ddot{x}_2 &= \displaystyle\sum_{I=1}^{4} \Phi^I(\xi,\eta)\, \ddot{x}_2^I \\ \ddot{x}_3 &= \displaystyle\sum_{I=1}^{4} \Phi^I(\xi,\eta)\, \ddot{x}_3^I + \sum_{I=1}^{4} \Phi^{I+4}(\xi,\eta)\, \ddot{w}^{I+4} \end{cases} \tag{6.23}$$

and the three components of the virtual displacement are given by

$$\begin{cases} \delta x_1 &= \sum_{I=1}^{4} \Phi^I(\xi,\eta)\, \delta x_1^I \\ \delta x_2 &= \sum_{I=1}^{4} \Phi^I(\xi,\eta)\, \delta x_2^I \\ \delta x_3 &= \sum_{I=1}^{4} \Phi^I(\xi,\eta)\, \delta x_3^I + \sum_{I=1}^{4} \Phi^{I+4}(\xi,\eta)\, \delta w^{I+4}. \end{cases} \tag{6.24}$$

Equation 6.19, which expresses the relative displacement of the middle-side node as a function of the DOFs of the adjacent nodes, becomes

$$w^{I+4} = \left[\underline{\theta}^{I+1} - \underline{\theta}^I\right]^T \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot \left[\underline{x}^{I+1} - \underline{x}^I\right]/8. \tag{6.25}$$

The important factor in this expression is the difference between the rotations of the two adjacent nodes $\underline{\theta}^I$ and $\underline{\theta}^{I+1}$. If the membrane deformation remains small, the factor involving the position vectors of the two adjacent nodes can be considered constant:

$$\left[\underline{x}^{I+1} - \underline{x}^I\right] \approx \left[\underline{X}^{I+1} - \underline{X}^I\right], \tag{6.26}$$

where a vector with a capital letter represents the position in the initial configuration and a vector with a small letter, the actualized coordinate. The final expression for the displacements of the four mid-side nodes are then given by

$$w^{I+4} = \frac{1}{8}\left[\left(\theta_1^{I+1} - \theta_1^I\right)\left(X_1^{I+1} - X_1^I\right) + \left(\theta_2^{I+1} - \theta_2^I\right)\left(X_2^{I+1} - X_2^I\right)\right]. \tag{6.27}$$

Since the factors involving the position vectors are assumed to be constant, the second derivative of this expression is

$$\ddot{w}^{I+4} = \frac{1}{8}\left[\left(\ddot{\theta}_1^{I+1} - \ddot{\theta}_1^I\right)\left(X_1^{I+1} - X_1^I\right) + \left(\ddot{\theta}_2^{I+1} - \ddot{\theta}_2^I\right)\left(X_2^{I+1} - X_2^I\right)\right]. \tag{6.28}$$

Similarly, the virtual displacements of the mid-side nodes are given by

$$\delta w^{I+4} = \frac{1}{8}\left[\left(\delta\theta_1^{I+1} - \delta\theta_1^I\right)\left(X_1^{I+1} - X_1^I\right) + \left(\delta\theta_2^{I+1} - \delta\theta_2^I\right)\left(X_2^{I+1} - X_2^I\right)\right]. \tag{6.29}$$

6.2.2 Consistent Mass Matrix

The computation of the consistent mass matrix follows the same methodology as in Section 6.1. Using the interpolation functions of the shell element developed in the previous section, i.e., Equations 6.23 and 6.24, the inertia term of Equation

6.6 (the virtual work principle) becomes

$$
\begin{aligned}
I &= \int_V \rho \left[\ddot{x}_1 \delta x_1 + \ddot{x}_2 \delta x_2 + \ddot{x}_3 \delta x_3\right] dv && \text{(6.30a)}\\
&= \int_V \rho \sum_{I=1}^{4}\sum_{J=1}^{4} \Big[\Phi^I \Phi^J \left(\ddot{x}_1^I \delta x_1^J + \ddot{x}_2^I \delta x_2^J + \ddot{x}_3^I \delta x_3^J\right) \\
&\qquad + \Phi^{I+4}\Phi^{J+4}\ddot{w}^{I+4}\delta w^{J+4} \\
&\qquad + \Phi^I \Phi^{J+4}\ddot{x}_3^I \delta w^{J+4} + \Phi^{I+4}\Phi^J \ddot{w}^{I+4}\delta x_3^J\Big] dv && \text{(6.30b)}\\
&\triangleq \int_V \sum_{I=1}^{4}\sum_{J=1}^{4} \left(\underline{\ddot{x}}^I\right)^T \underline{\underline{M}}^{IJ}\, \underline{\delta x}^J dv && \text{(6.30c)}\\
&= \int_V \left[\underline{\ddot{x}}^N\right]^T \underline{\underline{M}}\ \left[\underline{\delta x}^N\right] dv, && \text{(6.30d)}
\end{aligned}
$$

where
$$
\begin{aligned}
\underline{\ddot{x}}^I &= \left\langle\ \ddot{x}_1^I\ \ddot{x}_2^I\ \ddot{x}_3^I\ \ddot{\varphi}_1^I\ \ddot{\varphi}_2^I\ \ddot{\varphi}_3^I\ \right\rangle^T,\\
\underline{\delta x}^J &= \left\langle\ \delta x_1^I\ \delta x_2^I\ \delta x_3^I\ \delta\varphi_1^I\ \delta\varphi_2^I\ \delta\varphi_3^I\ \right\rangle^T,\\
\left[\underline{\ddot{x}}^N\right] &= \left\langle\ \left(\underline{\ddot{x}}^1\right)^T \,\vdots\, \left(\underline{\ddot{x}}^2\right)^T \,\vdots\, \left(\underline{\ddot{x}}^3\right)^T \,\vdots\, \left(\underline{\ddot{x}}^4\right)^T\ \right\rangle^T,\\
\left[\underline{\delta x}^N\right] &= \left\langle\ \left(\underline{\delta x}^1\right)^T \,\vdots\, \left(\underline{\delta x}^2\right)^T \,\vdots\, \left(\underline{\delta x}^3\right)^T \,\vdots\, \left(\underline{\delta x}^4\right)^T\ \right\rangle^T,\\
\underline{\underline{M}}^{IJ} &= (6\times 6) \text{ sub-matrix of the mass matrix } \underline{\underline{M}}\ (24\times 24).
\end{aligned}
$$

Determining each of the sub-matrices of the mass matrix will be performed by identifying the corresponding terms in Equations 6.30b and 6.30c. Equation 6.30b contains three kinds of terms:

- Translational terms $\ddot{x}_i^I \delta x_i^J$
- Rotational terms $\ddot{w}^{I+4}\delta w^{J+4}$
- Mixed terms $\ddot{x}_3^I \delta w^{J+4}$ and $\delta x_3^J \ddot{w}^{I+4}$

Each of these will be explained in the following three sections. In order to simplify the notation, two functions can be introduced:[4]

$$
P^{I+4}(\xi,\eta) = \frac{1}{8}\left(X_2^{I+1} - X_2^I\right)\Phi^{I+4}(\xi,\eta) \qquad (6.31)
$$

and

$$
Q^{I+4}(\xi,\eta) = \frac{1}{8}\left(X_1^I - X_1^{I+1}\right)\Phi^{I+4}(\xi,\eta). \qquad (6.32)
$$

Additionally, the dependence on the two variables ξ and η will generally not be written explicitly.

[4] These functions are computed by the preprocessor of the FE code LAGAMINE.

a. Translational Terms

The translational term in Equation 6.30b is equal to

$$\rho\,\Phi^I\Phi^J\left(\ddot{x}_1^I\delta x_1^J+\ddot{x}_2^I\delta x_2^J+\ddot{x}_3^I\delta x_3^J\right), \tag{6.33}$$

for a given value of I and J (no summation). It can be seen that this is equivalent to

$$\left(\underline{\ddot{x}}^I\right)^T\underline{\underline{M}}^{IJ}\,\underline{\delta x}^J=\left[\begin{array}{c}\ddot{x}_1^I\\ \ddot{x}_2^I\\ \ddot{x}_3^I\\ \hdashline \ddot{\varphi}_1^I\\ \ddot{\varphi}_2^I\\ \ddot{\varphi}_3^I\end{array}\right]^T\left[\begin{array}{ccc:ccc}M_{11}^{IJ}&0&0&\times&\times&\times\\ 0&M_{22}^{IJ}&0&\times&\times&\times\\ 0&0&M_{33}^{IJ}&\times&\times&\times\\ \hdashline \times&\times&\times&\times&\times&\times\\ \times&\times&\times&\times&\times&\times\\ \times&\times&\times&\times&\times&\times\end{array}\right]\left[\begin{array}{c}\delta x_1^J\\ \delta x_2^J\\ \delta x_3^J\\ \hdashline \delta\varphi_1^J\\ \delta\varphi_2^J\\ \delta\varphi_3^J\end{array}\right], \tag{6.34}$$

where $M_{11}^{IJ}=M_{22}^{IJ}=M_{33}^{IJ}=\rho\,\Phi^I\Phi^J$.

b. Rotational Terms

This section focuses on the rotational terms in Equation 6.30b, for a given value of I and J (no summation):

$$\Phi^{I+4}\Phi^{J+4}\ddot{w}^{I+4}\delta w^{J+4}\rho. \tag{6.35}$$

Substituting Equations 6.28 and 6.29 into this equation gives

$$\begin{aligned}&\frac{1}{8}\Phi^{I+4}\left[\left(\ddot{\theta}_1^{I+1}-\ddot{\theta}_1^I\right)\left(X_1^{I+1}-X_1^I\right)+\left(\ddot{\theta}_2^{I+1}-\ddot{\theta}_2^I\right)\left(X_2^{I+1}-X_2^I\right)\right]*\ldots\\ &\frac{1}{8}\Phi^{J+4}\left[\left(\delta\theta_1^{J+1}-\delta\theta_1^J\right)\left(X_1^{J+1}-X_1^J\right)+\left(\delta\theta_2^{J+1}-\delta\theta_2^J\right)\left(X_2^{J+1}-X_2^J\right)\right]\rho.\end{aligned} \tag{6.36}$$

Next, introducing the two functions P^{I+4} and Q^{I+4}, this equation becomes

$$\begin{aligned}&\left[\left(\ddot{\theta}_1^{I+1}-\ddot{\theta}_1^I\right)\left(-Q^{I+4}\right)+\left(\ddot{\theta}_2^{I+1}-\ddot{\theta}_2^I\right)P^{I+4}\right]*\ldots\\ &\left[\left(\delta\theta_1^{J+1}-\delta\theta_1^J\right)\left(-Q^{J+4}\right)+\left(\delta\theta_2^{J+1}-\delta\theta_2^J\right)P^{J+4}\right]\rho.\end{aligned} \tag{6.37}$$

Finally, taking into account the fact that Lagamine uses a different sign convention, as shown in Equation 6.20 (p. 89), Equation 6.37 becomes

$$\begin{aligned}&\left[\left(\ddot{\varphi}_2^{I+1}-\ddot{\varphi}_2^I\right)\left(-Q^{I+4}\right)+\left(\ddot{\varphi}_1^{I+1}-\ddot{\varphi}_1^I\right)\left(-P^{I+4}\right)\right]*\ldots\\ &\left[\left(\delta\varphi_2^{J+1}-\delta\varphi_2^J\right)\left(-Q^{J+4}\right)+\left(\delta\varphi_1^{J+1}-\delta\varphi_1^J\right)\left(-P^{J+4}\right)\right]\rho.\end{aligned} \tag{6.38}$$

In this case, the placement of the different terms in the sub-matrix $\underline{\underline{M}}^{IJ}$ is more complicated since Equation 6.38 involves rotational DOFs for nodes I, $I+1$, J, and $J+1$, whereas Equation 6.30c only involves DOFs for nodes I and J. This

seeming problem can be solved because Equation 6.30c contains a double sum over I and J. Indeed, it can be seen that

$$\sum_{I=1}^{4} x^{I+1}y^{I+1} = \left(x^2y^2 + x^3y^3 + x^4y^4 + x^1y^1\right) = \sum_{I=1}^{4} x^I y^I. \tag{6.39}$$

Therefore, adding or subtracting a given number to I or J can be performed if the same operation is performed for all the terms of a monomial, i.e., the terms separated by multiplication or division. For example,

$$\begin{aligned}\left(\ddot{\varphi}_2^{I+1} - \ddot{\varphi}_2^{I}\right) Q^{I+4} &= \ddot{\varphi}_2^{I+1}Q^{I+4} - \ddot{\varphi}_2^{I}Q^{I+4} \\ &= \ddot{\varphi}_2^{I}Q^{I+3} - \ddot{\varphi}_2^{I}Q^{I+4} \\ &= \ddot{\varphi}_2^{I}\left(Q^{I+3} - Q^{I+4}\right).\end{aligned} \tag{6.40}$$

Using similar transformations, Equation 6.38 can then be transformed into the following:[5]

$$\begin{aligned}&\left[\left(-\ddot{\varphi}_2^{I}Q^{I+3} + \ddot{\varphi}_2^{I}Q^{I+4}\right) + \left(-\ddot{\varphi}_1^{I}P^{I+3} + \ddot{\varphi}_1^{I}P^{I+4}\right)\right] * \ldots \\ &\left[\left(-\delta\varphi_2^{J}Q^{J+3} + \delta\varphi_2^{J}Q^{J+4}\right) + \left(-\delta\varphi_1^{J}P^{J+3} + \delta\varphi_1^{J}P^{J+4}\right)\right]\rho \\ &= \left[\ddot{\varphi}_2^{I}\left(Q^{I+4} - Q^{I+3}\right) + \ddot{\varphi}_1^{I}\left(P^{I+4} - P^{I+3}\right)\right] * \ldots \\ &\quad \left[\delta\varphi_2^{J}\left(Q^{J+4} - Q^{J+3}\right) + \delta\varphi_1^{J}\left(P^{J+4} - P^{J+3}\right)\right]\rho,\end{aligned} \tag{6.41}$$

where the rotational and virtual DOFs always appear with an index I and J, respectively.

The identification with Equation 6.30c can now be performed. The result is the following:

$$\left(\underline{\ddot{x}}^I\right)^T \underline{\underline{M}}^{IJ}\,\underline{\delta x}^J = \begin{bmatrix} \ddot{x}_1^I \\ \ddot{x}_2^I \\ \ddot{x}_3^I \\ \hdashline \ddot{\varphi}_1^I \\ \ddot{\varphi}_2^I \\ \ddot{\varphi}_3^I \end{bmatrix}^T \left[\begin{array}{ccc:ccc} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \hdashline \times & \times & \times & M_{44}^{IJ} & M_{45}^{IJ} & 0 \\ \times & \times & \times & M_{54}^{IJ} & M_{55}^{IJ} & 0 \\ \times & \times & \times & 0 & 0 & 0 \end{array}\right] \begin{bmatrix} \delta x_1^J \\ \delta x_2^J \\ \delta x_3^J \\ \hdashline \delta\varphi_1^J \\ \delta\varphi_2^J \\ \delta\varphi_3^J \end{bmatrix} \tag{6.42}$$

where $M_{44}^{IJ} = \rho\left(P^{I+4} - P^{I+3}\right)\left(P^{J+4} - P^{J+3}\right)$,
$M_{45}^{IJ} = \rho\left(P^{I+4} - P^{I+3}\right)\left(Q^{J+4} - Q^{J+3}\right)$,
$M_{54}^{IJ} = \rho\left(Q^{I+4} - Q^{I+3}\right)\left(P^{J+4} - P^{J+3}\right)$,
$M_{55}^{IJ} = \rho\left(Q^{I+4} - Q^{I+3}\right)\left(Q^{J+4} - Q^{J+3}\right)$.

[5] Similarly to footnote 3 page 88, it should be noted that the superscripts of P and Q can only take a value between 5 and 8. Therefore, P^{I+3} is equal to 8, 5, 6, and 7 for I equal to 1, 2, 3, and 4, respectively. As a result, an alternative notation for the superscript $(I+3)$ should be $((I+2 \bmod 4)+5)$.

It should be noted that the sum of each of the rotational terms of any line or column vanishes. For example, for a given line I, the sum of M_{44}^{IJ} is equal to

$$\begin{aligned}\sum_{J=1}^{4} M_{44}^{IJ} &= \rho\left(P^{I+4}-P^{I+3}\right)\sum_{J=1}^{4}\left(P^{J+4}-P^{J+3}\right)\\ &= \rho\left(P^{I+4}-P^{I+3}\right)\left[(P^5+P^6+P^7+P^8)-(P^8+P^5+P^6+P^7)\right]\\ &= 0.\end{aligned} \tag{6.43}$$

c. Mixed Terms

This section focuses on the mixed terms in Equation 6.30b. For a given value of I and J (no summation):

$$\rho\,\Phi^I\Phi^{J+4}\ddot{x}_3^I\delta w^{J+4}+\rho\,\Phi^{I+4}\Phi^J\ddot{w}^{I+4}\delta x_3^J. \tag{6.44}$$

Substituting Equations 6.28 and 6.29 into this equation and introducing the functions P^{I+4} and Q^{I+4} gives

$$\begin{aligned}&\rho\,\Phi^I\ddot{x}_3^I\left[\left(\delta\theta_1^{J+1}-\delta\theta_1^J\right)\left(-Q^{J+4}\right)+\left(\delta\theta_2^{J+1}-\delta\theta_2^J\right)P^{J+4}\right]+\ldots\\ &\rho\,\Phi^J\delta x_3^J\left[\left(\ddot{\theta}_1^{I+1}-\ddot{\theta}_1^I\right)\left(-Q^{I+4}\right)+\left(\ddot{\theta}_2^{I+1}-\ddot{\theta}_2^I\right)P^{I+4}\right].\end{aligned} \tag{6.45}$$

Then, using Lagamine's sign convention for the rotational DOFs gives the following equation

$$\begin{aligned}&\rho\,\Phi^I\ddot{x}_3^I\left[\left(\delta\varphi_2^{J+1}-\delta\varphi_2^J\right)\left(-Q^{J+4}\right)+\left(\delta\varphi_1^{J+1}-\delta\varphi_1^J\right)\left(-P^{J+4}\right)\right]+\ldots\\ &\rho\,\Phi^J\delta x_3^J\left[\left(\ddot{\varphi}_2^{I+1}-\ddot{\varphi}_2^I\right)\left(-Q^{I+4}\right)+\left(\ddot{\varphi}_1^{I+1}-\ddot{\varphi}_1^I\right)\left(-P^{I+4}\right)\right].\end{aligned} \tag{6.46}$$

The next step in the transformation of this formula consists in renaming the indices and substituting $I+1$ by I and $J+1$ by J:

$$\begin{aligned}&\rho\,\Phi^I\ddot{x}_3^I\left[\delta\varphi_2^J\left(Q^{J+4}-Q^{J+3}\right)+\delta\varphi_1^J\left(P^{J+4}-P^{J+3}\right)\right]+\ldots\\ &\rho\,\Phi^J\delta x_3^J\left[\ddot{\varphi}_2^I\left(Q^{I+4}-Q^{I+3}\right)+\ddot{\varphi}_1^I\left(P^{I+4}-P^{I+3}\right)\right].\end{aligned} \tag{6.47}$$

The identification with Equation 6.30c gives the final terms of the sub-matrix $\underline{\underline{M}}^{IJ}$:

$$\left(\underline{\ddot{x}}^I\right)^T\underline{\underline{M}}^{IJ}\,\underline{\delta x}^J=\begin{bmatrix}\ddot{x}_1^I\\ \ddot{x}_2^I\\ \ddot{x}_3^I\\ \hdashline \ddot{\varphi}_1^I\\ \ddot{\varphi}_2^I\\ \ddot{\varphi}_3^I\end{bmatrix}^T\left[\begin{array}{ccc:ccc}\times&\times&\times&0&0&0\\ \times&\times&\times&0&0&0\\ \times&\times&\times&M_{34}^{IJ}&M_{35}^{IJ}&0\\ \hdashline 0&0&M_{43}^{IJ}&\times&\times&\times\\ 0&0&M_{53}^{IJ}&\times&\times&\times\\ 0&0&0&\times&\times&\times\end{array}\right]\begin{bmatrix}\delta x_1^J\\ \delta x_2^J\\ \delta x_3^J\\ \hdashline \delta\varphi_1^J\\ \delta\varphi_2^J\\ \delta\varphi_3^J\end{bmatrix} \tag{6.48}$$

where $M_{34}^{IJ} = \rho\,\Phi^I\left(P^{J+4} - P^{J+3}\right)$,
$M_{35}^{IJ} = \rho\,\Phi^I\left(Q^{J+4} - Q^{J+3}\right)$,
$M_{43}^{IJ} = \rho\,\Phi^J\left(P^{I+4} - P^{I+3}\right)$,
$M_{53}^{IJ} = \rho\,\Phi^J\left(Q^{I+4} - Q^{I+3}\right)$.

It should be noted that the sum of the mixed terms for a line or a column does not always vanish. Instead, it gives the following result for a line:

$$\begin{cases} \displaystyle\sum_{J=1}^{4} M_{34}^{IJ} \;=\; \sum_{J=1}^{4} M_{35}^{IJ} \;=\; 0, \\ \displaystyle\sum_{J=1}^{4} M_{43}^{IJ} \;=\; \rho\left(P^{I+4} - P^{I+3}\right), \\ \displaystyle\sum_{J=1}^{4} M_{53}^{IJ} \;=\; \rho\left(Q^{I+4} - Q^{I+3}\right), \end{cases} \tag{6.49}$$

and for a column:

$$\begin{cases} \displaystyle\sum_{I=1}^{4} M_{34}^{IJ} \;=\; \rho\left(P^{J+4} - P^{J+3}\right), \\ \displaystyle\sum_{I=1}^{4} M_{35}^{IJ} \;=\; \rho\left(Q^{J+4} - Q^{J+3}\right), \\ \displaystyle\sum_{I=1}^{4} M_{43}^{IJ} \;=\; \sum_{I=1}^{4} M_{53}^{IJ} \;=\; 0, \end{cases} \tag{6.50}$$

using the basic property of the interpolation functions:

$$\sum_{I=1}^{4} \Phi^I = 1. \tag{6.51}$$

d. Complete Matrix

In the previous three sections, the different parts of the sub-matrices $\underline{\underline{M}}^{IJ}$ were computed separately. They can now be assembled. The final result is the following:

$$\underline{\underline{M}}^{IJ} = \left[\begin{array}{ccc:ccc} M_{11}^{IJ} & 0 & 0 & 0 & 0 & 0 \\ 0 & M_{22}^{IJ} & 0 & 0 & 0 & 0 \\ 0 & 0 & M_{33}^{IJ} & M_{34}^{IJ} & M_{35}^{IJ} & 0 \\ \hdashline 0 & 0 & M_{43}^{IJ} & M_{44}^{IJ} & M_{45}^{IJ} & 0 \\ 0 & 0 & M_{53}^{IJ} & M_{54}^{IJ} & M_{55}^{IJ} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \tag{6.52}$$

where $M_{11}^{IJ} = M_{22}^{IJ} = M_{33}^{IJ} = \rho\,\Phi^I\Phi^J$,

$$
\begin{aligned}
M_{34}^{IJ} &= \rho\,\Phi^I\left(P^{J+4} - P^{J+3}\right),\\
M_{35}^{IJ} &= \rho\,\Phi^I\left(Q^{J+4} - Q^{J+3}\right),\\
M_{43}^{IJ} &= \rho\,\Phi^J\left(P^{I+4} - P^{I+3}\right),\\
M_{53}^{IJ} &= \rho\,\Phi^J\left(Q^{I+4} - Q^{I+3}\right),\\
M_{44}^{IJ} &= \rho\left(P^{I+4} - P^{I+3}\right)\left(P^{J+4} - P^{J+3}\right),\\
M_{45}^{IJ} &= \rho\left(P^{I+4} - P^{I+3}\right)\left(Q^{J+4} - Q^{J+3}\right),\\
M_{54}^{IJ} &= \rho\left(Q^{I+4} - Q^{I+3}\right)\left(P^{J+4} - P^{J+3}\right),\\
M_{55}^{IJ} &= \rho\left(Q^{I+4} - Q^{I+3}\right)\left(Q^{J+4} - Q^{J+3}\right).
\end{aligned}
$$

The complete mass matrix is also shown in Figure 6.6. The terms located on the diagonal are printed using a bold blue font.

It can be seen that the sixth DOF does not contribute to the mass. However, it is necessary to keep this DOF because, otherwise, this can lead to numerical problems in case of a discontinuity between the normal of two adjacent elements [BATO92, page 371]. Within the Lagamine code, a correction is added to each of the sub-matrices $\underline{\underline{M}}^{IJ}$:

$$
\underline{\underline{M}}^{IJ} = \left[\begin{array}{ccc:cc:c}
M_{11}^{IJ} & 0 & 0 & 0 & 0 & -M_{34}^{IJ}\\
0 & M_{22}^{IJ} & 0 & 0 & 0 & -M_{35}^{IJ}\\
0 & 0 & M_{33}^{IJ} & M_{34}^{IJ} & M_{35}^{IJ} & 0\\
\hdashline
0 & 0 & M_{43}^{IJ} & M_{44}^{IJ} & M_{45}^{IJ} & 0\\
0 & 0 & M_{53}^{IJ} & M_{54}^{IJ} & M_{55}^{IJ} & 0\\
\hdashline
-M_{43}^{IJ} & -M_{53}^{IJ} & 0 & 0 & 0 & M_{44}^{IJ} + M_{55}^{IJ}
\end{array}\right]. \tag{6.53}
$$

6.2.3 Diagonal Mass Matrix

Several techniques can be used to diagonalize the mass matrix. The easiest one consists in extracting the diagonal terms of the matrix, which gives the following diagonal vector

$$
\begin{aligned}
\underline{M}_d = \rho \Big\langle\ & (\Phi^1)^2\ ,\ (\Phi^1)^2\ ,\ (\Phi^1)^2\ ,\ (P^5 - P^8)^2\ ,\ (Q^5 - Q^8)^2\ ,\ 0\ ,\ \ldots\\
& (\Phi^2)^2\ ,\ (\Phi^2)^2\ ,\ (\Phi^2)^2\ ,\ (P^6 - P^5)^2\ ,\ (Q^6 - Q^5)^2\ ,\ 0\ ,\ \ldots\\
& (\Phi^3)^2\ ,\ (\Phi^3)^2\ ,\ (\Phi^3)^2\ ,\ (P^7 - P^6)^2\ ,\ (Q^7 - Q^6)^2\ ,\ 0\ ,\ \ldots\\
& (\Phi^4)^2\ ,\ (\Phi^4)^2\ ,\ (\Phi^4)^2\ ,\ (P^8 - P^7)^2\ ,\ (Q^8 - Q^7)^2\ ,\ 0\ \Big\rangle^T.
\end{aligned} \tag{6.54}
$$

Using the sub-matrix notation, this equation is equivalent to:

$$
\underline{M}_d = \rho \left\langle\ \left(\underline{M}_d^1\right)^T \,\middle|\, \left(\underline{M}_d^2\right)^T \,\middle|\, \left(\underline{M}_d^3\right)^T \,\middle|\, \left(\underline{M}_d^4\right)^T\ \right\rangle^T \tag{6.55}
$$

where

$$
\underline{M}_d^I = \left\langle\ D_1^I\ ,\ D_2^I\ ,\ D_3^I\ ,\ D_4^I\ ,\ D_5^I\ ,\ D_6^I\ \right\rangle^T,\ I = 1\ldots4 \tag{6.56}
$$

$$
\underline{\underline{M}} = \rho
\begin{bmatrix}
\boldsymbol{(\Phi^1)^2} & 0 & 0 & 0 & 0 & 0 & \Phi^1\Phi^2 & 0 & 0 & 0 & 0 & \cdots \\
0 & \boldsymbol{(\Phi^1)^2} & 0 & 0 & 0 & 0 & 0 & \Phi^1\Phi^2 & 0 & 0 & 0 & \cdots \\
0 & 0 & \boldsymbol{(\Phi^1)^2} & \Phi^1(P^5-P^8) & \Phi^1(Q^5-Q^8) & 0 & 0 & 0 & \Phi^1\Phi^2 & \Phi^1(P^6-P^5) & \Phi^1(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & \boldsymbol{(P^5-P^8)^2} & (P^5-P^8)(Q^5-Q^8) & 0 & 0 & 0 & \Phi^2(P^5-P^8) & (P^5-P^8)(P^6-P^5) & (P^5-P^8)(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & (Q^5-Q^8)(P^5-P^8) & \boldsymbol{(Q^5-Q^8)^2} & 0 & 0 & 0 & \Phi^2(Q^5-Q^8) & (Q^5-Q^8)(P^6-P^5) & (Q^5-Q^8)(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & 0 & 0 & \mathbf{0} & 0 & 0 & 0 & 0 & 0 & \cdots \\
\Phi^1\Phi^2 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{(\Phi^2)^2} & 0 & 0 & 0 & 0 & \cdots \\
0 & \Phi^1\Phi^2 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{(\Phi^2)^2} & 0 & 0 & 0 & \cdots \\
0 & 0 & \Phi^1\Phi^2 & \Phi^2(P^5-P^8) & \Phi^2(Q^5-Q^8) & 0 & 0 & 0 & \boldsymbol{(\Phi^2)^2} & \Phi^2(P^6-P^5) & \Phi^2(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & (P^6-P^5)(P^5-P^8) & (P^6-P^5)(Q^5-Q^8) & 0 & 0 & 0 & \Phi^2(P^6-P^5) & \boldsymbol{(P^6-P^5)^2} & (P^6-P^5)(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & (Q^6-Q^5)(P^5-P^8) & (Q^6-Q^5)(Q^5-Q^8) & 0 & 0 & 0 & \Phi^2(Q^6-Q^5) & (Q^6-Q^5)(P^6-P^5) & \boldsymbol{(Q^6-Q^5)^2} & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} & \cdots \\
\Phi^1\Phi^3 & 0 & 0 & 0 & 0 & 0 & \Phi^2\Phi^3 & 0 & 0 & 0 & 0 & \cdots \\
0 & \Phi^1\Phi^3 & 0 & 0 & 0 & 0 & 0 & \Phi^2\Phi^3 & 0 & 0 & 0 & \cdots \\
0 & 0 & \Phi^1\Phi^3 & \Phi^3(P^5-P^8) & \Phi^3(Q^5-Q^8) & 0 & 0 & 0 & \Phi^2\Phi^3 & \Phi^3(P^6-P^5) & \Phi^3(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & (P^7-P^6)(P^5-P^8) & (P^7-P^6)(Q^5-Q^8) & 0 & 0 & 0 & \Phi^2(P^7-P^6) & (P^7-P^6)(P^6-P^5) & (P^7-P^6)(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & (Q^7-Q^6)(P^5-P^8) & (Q^7-Q^6)(Q^5-Q^8) & 0 & 0 & 0 & \Phi^2(Q^7-Q^6) & (Q^7-Q^6)(P^6-P^5) & (Q^7-Q^6)(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\
\Phi^1\Phi^4 & 0 & 0 & 0 & 0 & 0 & \Phi^2\Phi^4 & 0 & 0 & 0 & 0 & \cdots \\
0 & \Phi^1\Phi^4 & 0 & 0 & 0 & 0 & 0 & \Phi^2\Phi^4 & 0 & 0 & 0 & \cdots \\
0 & 0 & \Phi^1\Phi^4 & \Phi^4(P^5-P^8) & \Phi^4(Q^5-Q^8) & 0 & 0 & 0 & \Phi^2\Phi^4 & \Phi^4(P^6-P^5) & \Phi^4(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & (P^8-P^7)(P^5-P^8) & (P^8-P^7)(Q^5-Q^8) & 0 & 0 & 0 & \Phi^2(P^8-P^7) & (P^8-P^7)(P^6-P^5) & (P^8-P^7)(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & (Q^8-Q^7)(P^5-P^8) & (Q^8-Q^7)(Q^5-Q^8) & 0 & 0 & 0 & \Phi^2(Q^8-Q^7) & (Q^8-Q^7)(P^6-P^5) & (Q^8-Q^7)(Q^6-Q^5) & \cdots \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots
\end{bmatrix}
$$

$$
\begin{array}{c|cccccc|cccccc|cccccc|cccccc}
\cdots & \Phi^1\Phi^3 & 0 & 0 & 0 & 0 & 0 & \Phi^1\Phi^4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & \Phi^1\Phi^3 & 0 & 0 & 0 & 0 & 0 & \Phi^1\Phi^4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & 0 & \Phi^1\Phi^3 & \Phi^1(P^7-P^6) & \Phi^1(Q^7-Q^6) & 0 & 0 & 0 & \Phi^1\Phi^4 & 0 & 0 & 0 & \Phi^1(P^8-P^7) & 0 & 0 & \Phi^1(Q^8-Q^7) & 0 \\
\hline
\cdots & 0 & 0 & \Phi^3(P^5-P^8) & (P^5-P^8)(P^7-P^6) & (P^5-P^8)(Q^7-Q^6) & 0 & 0 & 0 & \Phi^4(P^5-P^8) & 0 & 0 & 0 & (P^5-P^8)(P^8-P^7) & 0 & 0 & (P^5-P^8)(Q^8-Q^7) & 0 \\
\cdots & 0 & 0 & \Phi^3(Q^5-Q^8) & (Q^5-Q^8)(P^7-P^6) & (Q^5-Q^8)(Q^7-Q^6) & 0 & 0 & 0 & \Phi^4(Q^5-Q^8) & 0 & 0 & 0 & (Q^5-Q^8)(P^8-P^7) & 0 & 0 & (Q^5-Q^8)(Q^8-Q^7) & 0 \\
\cdots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
\cdots & \Phi^2\Phi^3 & 0 & 0 & 0 & 0 & 0 & \Phi^2\Phi^4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & \Phi^2\Phi^3 & 0 & 0 & 0 & 0 & 0 & \Phi^2\Phi^4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & 0 & \Phi^2\Phi^3 & \Phi^2(P^7-P^6) & \Phi^2(Q^7-Q^6) & 0 & 0 & 0 & \Phi^2\Phi^4 & 0 & 0 & 0 & \Phi^2(P^8-P^7) & 0 & 0 & \Phi^2(Q^8-Q^7) & 0 \\
\hline
\cdots & 0 & 0 & \Phi^3(P^6-P^5) & (P^6-P^5)(P^7-P^6) & (P^6-P^5)(Q^7-Q^6) & 0 & 0 & 0 & \Phi^4(P^6-P^5) & 0 & 0 & 0 & (P^6-P^5)(P^8-P^7) & 0 & 0 & (P^6-P^5)(Q^8-Q^7) & 0 \\
\cdots & 0 & 0 & \Phi^3(Q^6-Q^5) & (Q^6-Q^5)(P^7-P^6) & (Q^6-Q^5)(Q^7-Q^6) & 0 & 0 & 0 & \Phi^4(Q^6-Q^5) & 0 & 0 & 0 & (Q^6-Q^5)(P^8-P^7) & 0 & 0 & (Q^6-Q^5)(Q^8-Q^7) & 0 \\
\cdots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
\cdots & \mathbf{(\Phi^3)^2} & 0 & 0 & 0 & 0 & 0 & \Phi^3\Phi^4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & \mathbf{(\Phi^3)^2} & 0 & 0 & 0 & 0 & 0 & \Phi^3\Phi^4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & 0 & \mathbf{(\Phi^3)^2} & \Phi^3(P^7-P^6) & \Phi^3(Q^7-Q^6) & 0 & 0 & 0 & \Phi^3\Phi^4 & 0 & 0 & 0 & \Phi^3(P^8-P^7) & 0 & 0 & \Phi^3(Q^8-Q^7) & 0 \\
\hline
\cdots & 0 & 0 & \Phi^3(P^7-P^6) & \mathbf{(P^7-P^6)^2} & (P^7-P^6)(Q^7-Q^6) & 0 & 0 & 0 & \Phi^4(P^7-P^6) & 0 & 0 & 0 & (P^7-P^6)(P^8-P^7) & 0 & 0 & (P^7-P^6)(Q^8-Q^7) & 0 \\
\cdots & 0 & 0 & \Phi^3(Q^7-Q^6) & (Q^7-Q^6)(P^7-P^6) & \mathbf{(Q^7-Q^6)^2} & 0 & 0 & 0 & \Phi^4(Q^7-Q^6) & 0 & 0 & 0 & (Q^7-Q^6)(P^8-P^7) & 0 & 0 & (Q^7-Q^6)(Q^8-Q^7) & 0 \\
\cdots & 0 & 0 & 0 & 0 & 0 & \mathbf{0} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
\cdots & \Phi^3\Phi^4 & 0 & 0 & 0 & 0 & 0 & \mathbf{(\Phi^4)^2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & \Phi^3\Phi^4 & 0 & 0 & 0 & 0 & 0 & \mathbf{(\Phi^4)^2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & 0 & \Phi^3\Phi^4 & \Phi^4(P^7-P^6) & \Phi^4(Q^7-Q^6) & 0 & 0 & 0 & \mathbf{(\Phi^4)^2} & 0 & 0 & 0 & \Phi^4(P^8-P^7) & 0 & 0 & \Phi^4(Q^8-Q^7) & 0 \\
\hline
\cdots & 0 & 0 & \Phi^3(P^8-P^7) & (P^8-P^7)(P^7-P^6) & (P^8-P^7)(Q^7-Q^6) & 0 & 0 & 0 & \Phi^4(P^8-P^7) & \mathbf{(P^8-P^7)^2} & (P^8-P^7)(Q^8-Q^7) & 0 & 0 & 0 & 0 & 0 & 0 \\
\cdots & 0 & 0 & \Phi^3(Q^8-Q^7) & (Q^8-Q^7)(P^7-P^6) & (Q^8-Q^7)(Q^7-Q^6) & 0 & 0 & 0 & \Phi^4(Q^8-Q^7) & (Q^8-Q^7)(P^8-P^7) & \mathbf{(Q^8-Q^7)^2} & 0 & 0 & 0 & 0 & 0 & \mathbf{0} \\
\cdots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}
$$

Figure 6.6: Consistent mass matrix of the COQJ4 element

and $D_1^I = D_2^I = D_3^I = \rho \left(\Phi^I\right)^2$,
$D_4^I = \rho \left(P^{I+4} - P^{I+3}\right)^2$,
$D_5^I = \rho \left(Q^{I+4} - Q^{I+3}\right)^2$,
$D_6^I = 0.$

Like the transformation of Equation 6.53, the term corresponding to the sixth DOF of each sub-matrix could also be corrected by summing the terms of the fourth and fifth DOFs:

$$D_6^I = \rho \left[\left(P^{I+4} - P^{I+3}\right)^2 + \left(Q^{I+4} - Q^{I+3}\right)^2\right].$$

Another diagonalization technique consists in summing up the corresponding terms of the different blocks of the matrix:

$$D_i^I = \sum_{J=1}^{4} M_{ii}^{IJ}, \tag{6.57}$$

where no summation is implied on the repeated index i of the right-hand side of this equation. This gives the following diagonal matrix:

$$D_1^I = D_2^I = D_3^I = \rho\,\Phi^I \left(\Phi^1 + \Phi^2 + \Phi^3 + \Phi^4\right) = \rho\,\Phi^I \tag{6.58}$$

and

$$D_4^I = D_5^I = D_6^I = 0. \tag{6.59}$$

However, this matrix would not be acceptable, since no mass is assigned to the rotational DOFs. Therefore, a correction must be added to the rotational DOFs.

a. Li's Matrix

Within LAGAMINE, two different diagonal matrices are programmed. The first one, Li's matrix, is defined by

$$\begin{cases} D_1^I = D_2^I = D_3^I = \rho\,\Phi^I, \\ D_4^I = \rho \left(P^{I+4} - P^{I+3}\right)\left(P^{I+4} - P^{I+3} - P^{I+6} + P^{I+5}\right) \triangleq PP^I, \\ D_5^I = \rho \left(Q^{I+4} - Q^{I+3}\right)\left(Q^{I+4} - Q^{I+3} - Q^{I+6} + Q^{I+5}\right) \triangleq QQ^I, \\ D_6^I = PP^I + QQ^I. \end{cases} \tag{6.60}$$

This is the default diagonal matrix. Unfortunately, no record was found regarding the origin of this matrix.

b. Jetteur's Matrix

The second matrix, Jetteur's matrix, was added following the apparition of some problems with Li's matrix. With this matrix, the rotational masses are simply proportional to the translational ones. It is defined by

$$\begin{cases} D_1^I = D_2^I = D_3^I = \rho\,\Phi^I, \\ D_4^I = D_5^I = \mathrm{c}\rho\,\Phi^I, \\ D_6^I = D_4^I + D_5^I, \end{cases} \tag{6.61}$$

and

$$\mathrm{c} = \max\left(t^2/12; S/18\right), \tag{6.62}$$

where t is the thickness of the element and S, its surface. This matrix was suggested by Jetteur, who programmed the shell element within another code developed at the University of Liège, FINELg. It was added to Lagamine in order to try to solve problems regarding mass in rotation. The definition of Equations 6.61 and 6.62 represents the fact that the masses of the translational DOFs are the only essential terms of the mass matrix. The masses of the rotational DOFs, on the other hand, do not play an important role and can, therefore, be chosen more or less arbitrarily.

Using the explicit strategy, the time step is conditionally stable and is proportional to

$$\Delta t \approx L\sqrt{\frac{\rho}{E}}.$$

Therefore, the rotational terms of the mass matrix must be chosen in such a way that they do not reduce the maximum stable time step excessively. This choice is made by computing the natural frequencies of a rectangular sheet. The coefficient $t^2/12$ comes from the bending behavior, whereas the coefficient $S/18$ comes from the membrane behavior.

Chapter 7

Evaluation of the Moving Spherical Tool Method

The goal of this chapter is to analyze the overall performance of the Moving Spherical Tool method.

7.1 Indent Test

The simplest dynamic explicit simulation that can be performed is an indent test, shown in Figure 7.1.[1] A clamped square metal sheet of 10×10 mm^2 with a thickness of 0.3 mm is punched in its center by a spherical tool whose radius is 5 mm. At the end of the simulation, after 1 s, the vertical displacement of the tool is 2 mm. The mesh is a regular grid of $10 \times 10 = 100$ shell elements.

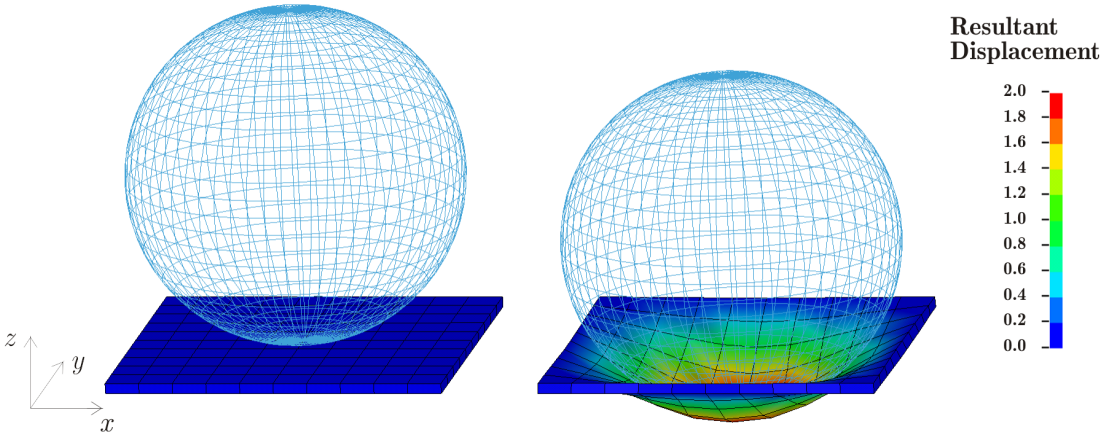


Figure 7.1: Indent Test

A mass-scaling factor of 100 was used. In addition to that, the rotational masses were multiplied by 1000. The reason for this multiplication will be explained in the next section. Figure 7.2 shows the deformation in a cross section along the vertical plane x = 0. This simulation required 714 000 time steps. The computation time was 47 minutes on a regular computer (laptop with Pentium M

[1] The displacements were not amplified in the drawing.

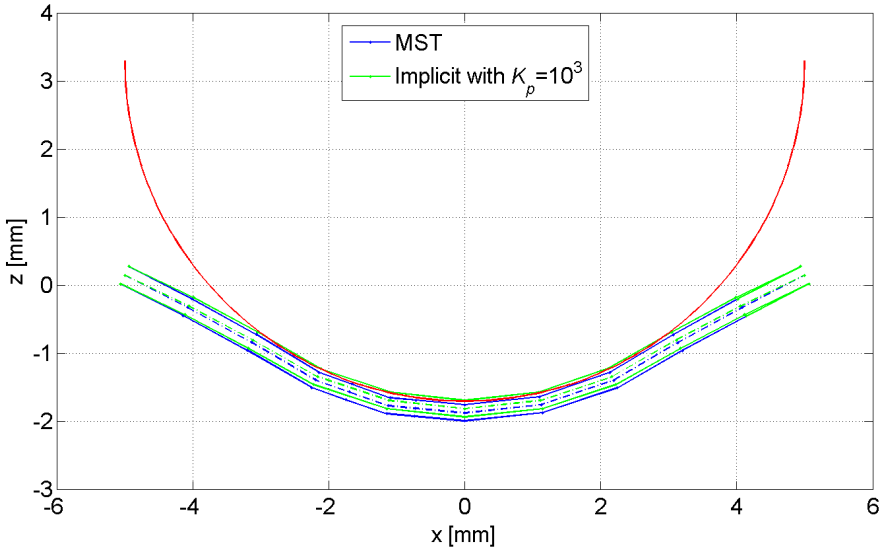


Figure 7.2: Comparison between the implicit and MST dynamic explicit simulations for the prediction of the shape (top, middle and bottom surfaces) of the indent simulation. Cross section along x=0.

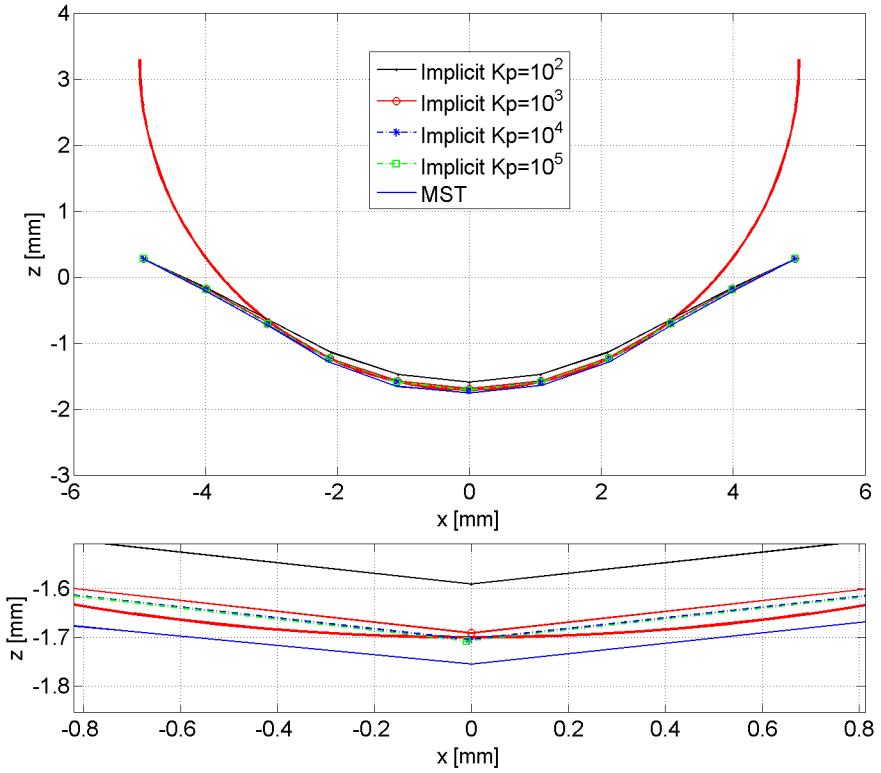


Figure 7.3: Influence of the penalty coefficient on the prediction of the shape (top surface) of the indent simulation. Cross section along x=0. Global view and zoom.

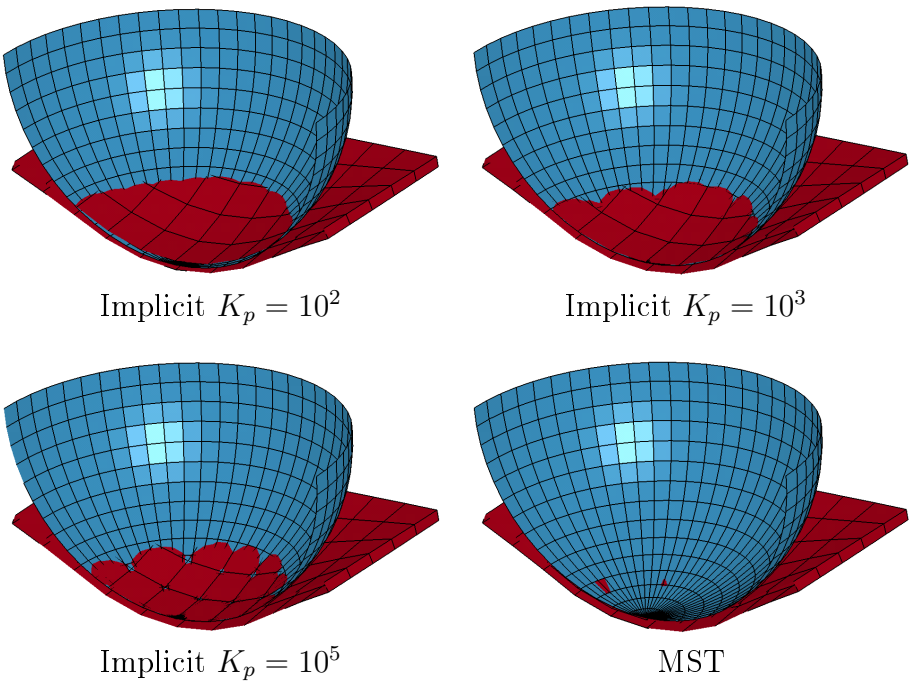


Figure 7.4: Comparison of the tool's penetration for different values of the penalty coefficient for the implicit simulations and for the MST dynamic explicit simulation.

1.86 GHz). In comparison, the same simulation with an implicit strategy required 110 time steps and took only 12 seconds.

In an implicit strategy, the accuracy of the contact is a function of the penalty coefficient. Figures 7.3 and 7.4 show the penetration of the tool in an implicit strategy for different values of the penalty coefficient, as well as in a dynamic explicit strategy with the MST method. $K_p = 10^3$ is the best compromise between accuracy and good convergence[2], which is why that value was chosen for most of the implicit simulations performed throughout this thesis. The dynamic explicit simulation with MST does not have any interpenetration, as it does not use a penalty method.

On this first very simple example, the new simulation method seems to provide accurate results but the computation times are much higher. This might be due to the fact that this simulation has a very small number of elements, which is in favor of the implicit integration scheme. This issue will be discussed more in detail in the next example.

7.2 Mesh Sensitivity with the Explicit Strategy

After this first basic example, another more complicated simulation was tested: the line test. This simulation was already introduced in Section 4.1.1 page 61.

[2] $K_p = 10^5$ was the highest value for which convergence was achieved. On the other hand, $K_p = 10^2$ led to a maximum interpenetration of about 0.1 mm in the center of the sheet, which is more than 30 % of the initial sheet thickness and is close to 50 % of the deformed sheet thickness. There is no significant difference in accuracy between $K_p = 10^3$ and $K_p = 10^5$.

After a few runs, this simulation appeared to be highly unstable. Figure 7.5 shows the result of a simulation at two different stages: after the first indent (point B in Figure 4.1 page 61) and during the first line (between points B and C). Whereas for the first stage, the simulation seemed to run normally, it stopped at the second stage after an obvious convergence problem. It had already been mentioned in the present author's Master's thesis [HEN05a] that there was a convergence problem with LAGAMINE's shell element in an explicit strategy but without understanding what was causing it.

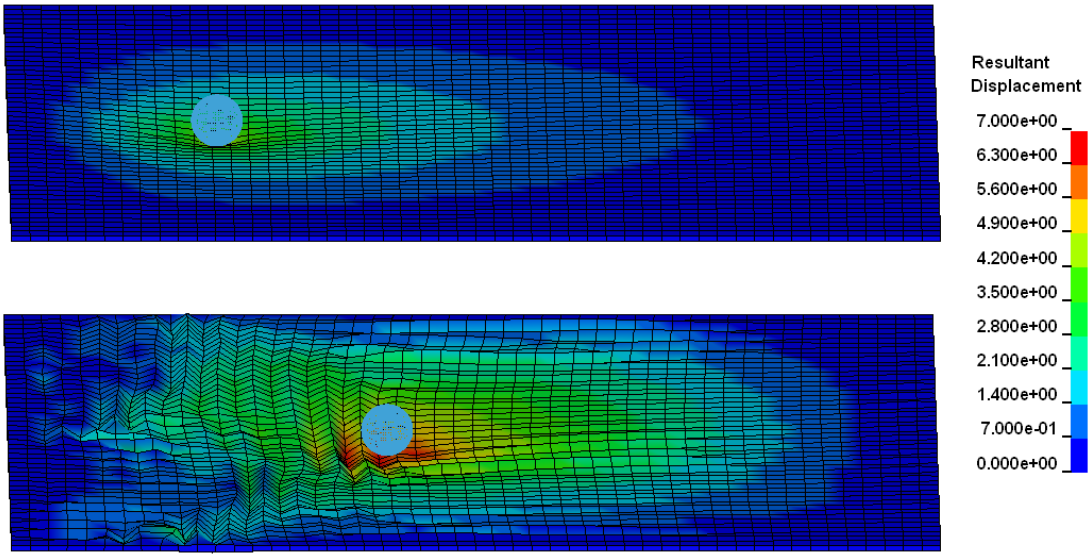


Figure 7.5: Total displacement at two stages of the simulation without mass scaling – illustration of the convergence problems in a dynamic explicit strategy

Four possible sources for these instabilities in the explicit strategy were identified: the mesh (in particular the aspect ratio of the elements), the rotational DOFs of the mass matrix, the absence of numerical damping, and the new MST method added in the LAGAMINE code. This section will focus on the influence of the mesh.

In order to eliminate the possible source of error coming from the MST algorithm, a simpler simulation was chosen: a square metal sheet of 182×182 mm^2 was deformed by applying, in 0.15 s, a vertical displacement of 5 mm to its center node. Since the problem is symmetric, only one fourth of the mesh was modeled with symmetry boundary conditions. This simulation uses an imposed nodal displacement and not the new contact method developed with the new MST dynamic explicit method.

When creating a mesh with thin shell elements, it is often stated in literature that each element should follow two basic design rules in order to behave properly:

1. Its in-plane shape should be as square as possible:

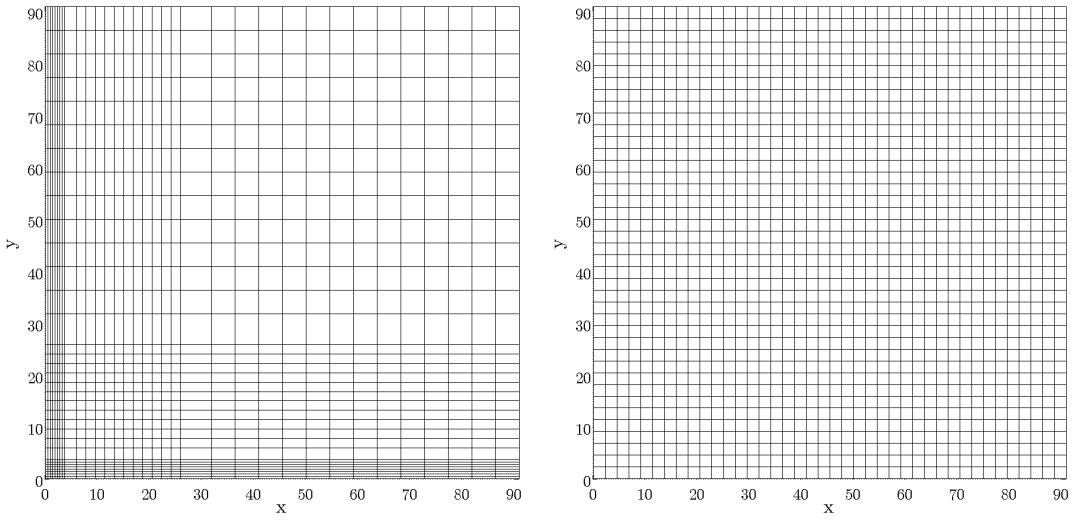
$$\frac{l_{max}}{l_{min}} \cong 1 \tag{7.1}$$

where l_{min} and l_{max} are the two in-plane lengths of the element.

2. Its thickness t should be much smaller than its characteristic in-plane length l:

$$\frac{t}{l} << 1. \tag{7.2}$$

To test if these two properties are important for Lagamine's element, two in-plane mesh densities were used, as shown in Figures 7.6*a* and 7.6*b*. On the one hand, the variable mesh density (*a*) contains elements of several sizes, including small elements (0.455×0.455 mm^2), large elements (4.55×4.55 mm^2) and elongated elements (0.455×4.55 mm^2). On the other hand, all the elements of the constant mesh density (*b*) have the same size, i.e., 2.275×2.275 mm^2. Then, for each of these two mesh densities, two thicknesses were tested. This resulted in four different meshes, whose characteristics are summarized in Table 7.1.



(*a*) Variable mesh density (*b*) Constant mesh density

Figure 7.6: In-plane mesh densities used for the study of the influence of the elements' aspect ratio in an explicit strategy

Mesh	Density	Thickness	t/l	l_{max}/l_{min}
Mesh 1	(*a*)	1.20 mm	**3**	**10**
Mesh 2	(*a*)	0.04 mm	0.1	**10**
Mesh 3	(*b*)	7.00 mm	**3**	1
Mesh 4	(*b*)	0.20 mm	0.1	1

Table 7.1: Characteristics of the four meshes used for the study of the influence of the mesh in an explicit strategy

As can be seen, simulations with meshes 1 and 3 diverge rapidly. As an example, Figure 7.7 shows the deformed shape of mesh 1 after an imposed displacement of only 16 μm.

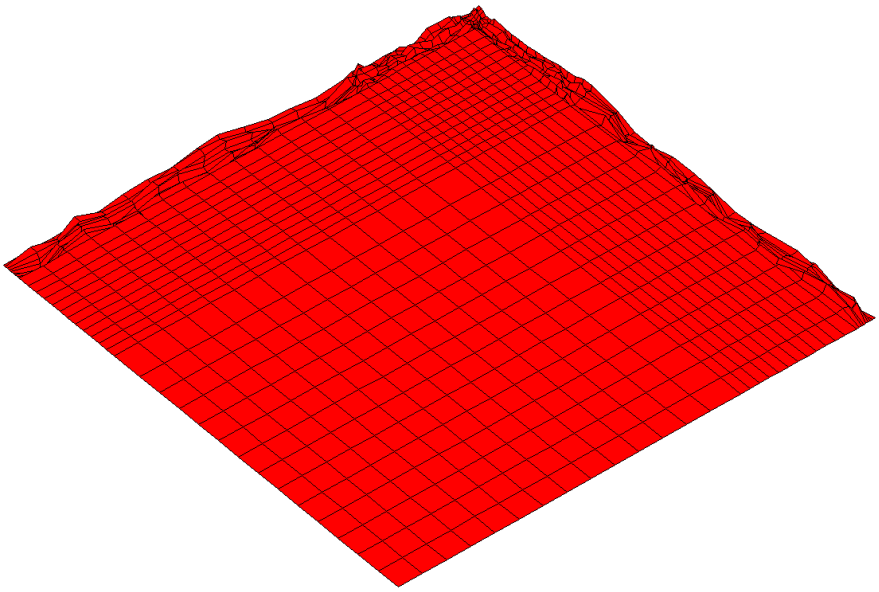


Figure 7.7: Deformed shape with Mesh 1

Thickness [mm]	t/l	Results?
7.0	3.0769	Diverge
6.0	2.6374	Diverge
5.0	2.1978	Diverge
4.0	1.7582	Diverge
3.0	1.3187	Diverge
2.2	0.9670	Diverge
2.0	0.8791	Diverge
1.8	0.7912	Diverge
1.6	0.7033	Diverge
1.4	0.6154	Diverge
1.3	0.5714	Diverge
1.2	0.5275	Starts to diverge
1.1	0.4835	Starts to diverge
1.0	0.4396	Starts to diverge
0.9	0.3956	Good results
0.8	0.3516	Good results
0.6	0.2637	Good results
0.4	0.1758	Good results
0.2	0.0879	Good results

Table 7.2: Influence of the relative thickness on the quality of the mesh in an explicit strategy

On the other hand, simulations with meshes 2 and 4 produce results close to the ones obtained in the implicit strategy. It seems that the thickness is more important in determining the quality of a mesh, compared to the in-plane aspect ratio of the elements. In order to be more specific on which thickness is acceptable and which is not, the same simulation was run using a constant mesh density (b) with a thickness ranging from 0.2 mm to 7 mm. As a reminder, the elements have an in-plane size of 2.275×2.275 mm^2. The results of the simulations are shown in Table 7.2. It can be seen that the thickness must be smaller than or equal to 40 % of the characteristic in-plane length of the elements. To be on the safe side, a value of one third is recommended.

7.3 Line Test

The previous section focused on the analysis of the mesh characteristics on the stability of the results in a dynamic explicit strategy. This resulted in a criterion that the shell elements should fulfill.

The current section focuses on the second possible source for the instabilities in an explicit strategy, i.e., the rotational DOFs of the mass matrix. The simulation used was again the line test. The mesh that was originally used for this simulation, presented in Figure 4.2 page 62, had to be redesigned in order to fulfill the stability criteria. The new mesh was composed of 2500 regularly distributed square elements, as shown in Figure 7.8. All the elements had an in-plane size of 3.62×3.62 mm, to be compared with a thickness of 1.2 mm.

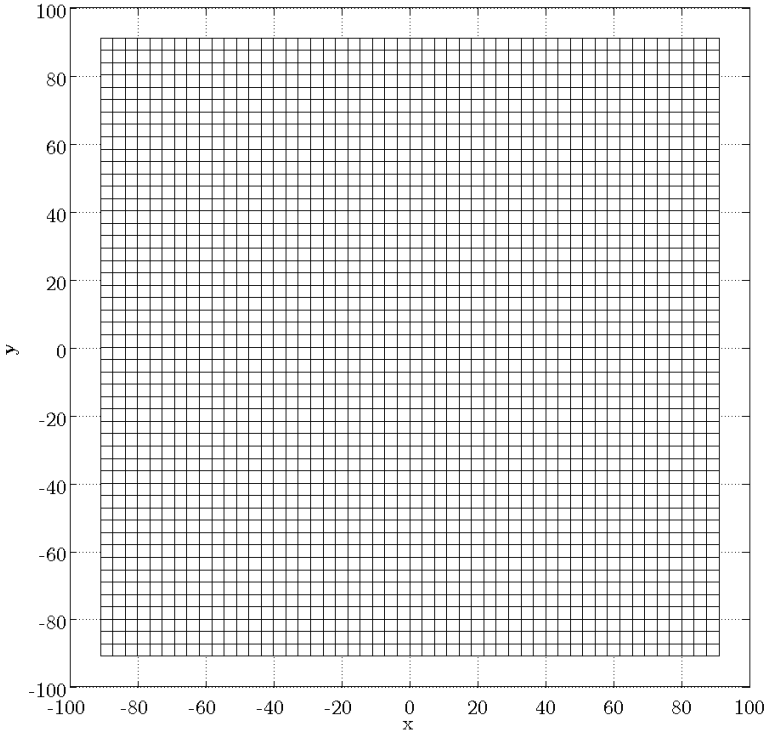


Figure 7.8: New regular mesh for the line test simulation (units in mm)

7.3.1 Results without Mass Scaling

First of all, the simulation was run without any mass scaling and using Li's diagonal mass matrix, introduced in Section 6.2.3. This simulation was unstable. The deformed shape is shown in Figure 7.5 and was already commented on in Section 7.2.

When the simulation stopped, the result showed an accordion-like deformation mode, which indicated that the elements did not behave properly in rotation. In order to solve that problem, the rotational DOFs of the mass matrix were multiplied by an arbitrary factor, ranging from 10 to 1000. The deformation of the sheet after the first linear tool displacement, i.e., for the tool located at point C in Figure 4.1, is presented in a cross section along $y = 0$ in Figure 7.9 for different factors of multiplication.

In the legends of the following figures, M stands for mass and the number following it is the mass-scaling factor — one, in this case, meaning no mass scaling. Similarly, MR stands for rotational mass and the number following it gives the additional multiplication of the rotational DOFs of the mass matrix. Finally, the simulations that did not reach the desired time step because of convergence problems and instabilities are marked using an asterisk.

It can be seen in Figure 7.9 that multiplying the rotational DOFs produces the targeted stabilization. For this simulation, there is almost no difference between a factor of 10 and 1000.

The same simulation was also run with the other diagonal mass matrix, Jetter's matrix, introduced in Section 6.2.3. The result is once again shown after the first linear tool displacement (point C) in Figure 7.10.

Jetteur's mass matrix seems to produce more stable results, even without scaling the mass in rotation. This might be due to the fact that both figures are shown at the end of the first line. The level of deformation reached at that stage of the simulation is quite limited. However, Figure 7.11 confirms these results. This figure shows the results using both matrices at the end of the second line, for the tool located at point E in Figure 4.1. The only simulation that did not reach this point is the one using Li's matrix without mass scaling in rotation. Even though there are some oscillations with Jetteur's matrix when using a rotational mass-scaling factor of 1 or 10, the results are more stable.

However, as far as accuracy is concerned, the results are not satisfactory. Figure 7.12 shows the same curves as in the previous figure but the scale of the vertical axis was changed in order to magnify the differences between the curves. It can be seen that a scaling factor of 100 or 1000 produces the same results with both Li's and Jetteur's matrices but these results are different from the ones obtained in the implicit strategy. The maximum discrepancies between the curves is around 0.8 mm. As a reminder, the initial sheet thickness is 1.2 mm.

7.3.2 Influence of Mass Scaling

In order to speed up the simulation, it was necessary to use mass scaling, since it increases the largest stable time step.

Figures 7.13 and 7.14 compare the influence of mass scaling on the shape in a cross section along $y = 0$ at the end of the first line (point C) when using,

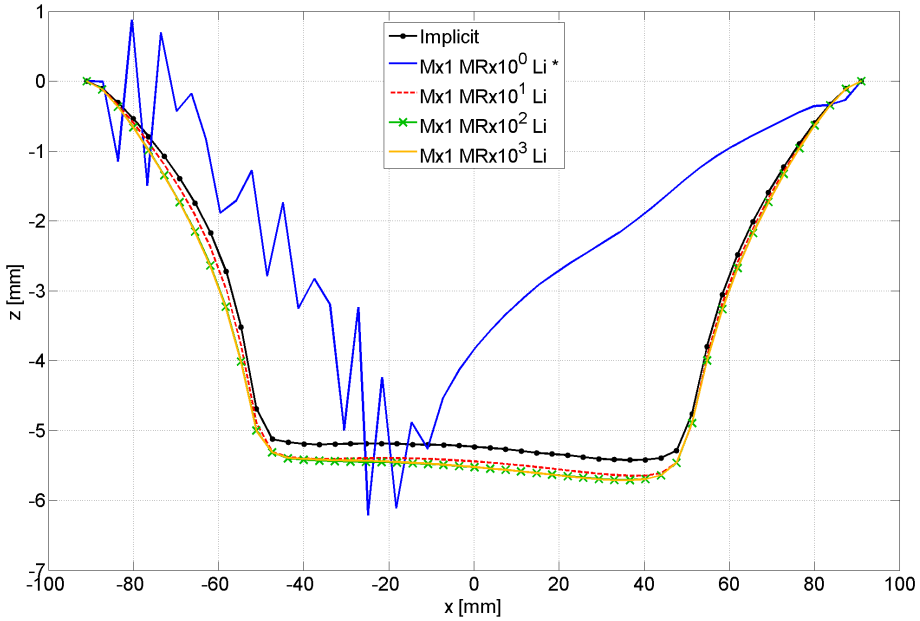


Figure 7.9: Comparison of the influence of mass scaling in rotation when using Li's matrix at the end of the first line (point C)

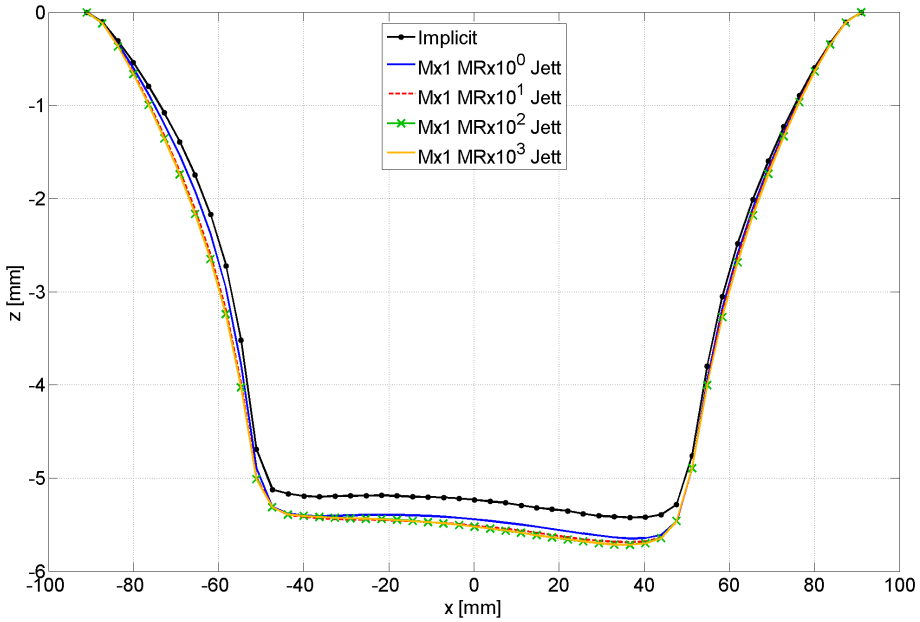


Figure 7.10: Comparison of the influence of mass scaling in rotation when using Jetter's matrix at the end of the first line (point C)

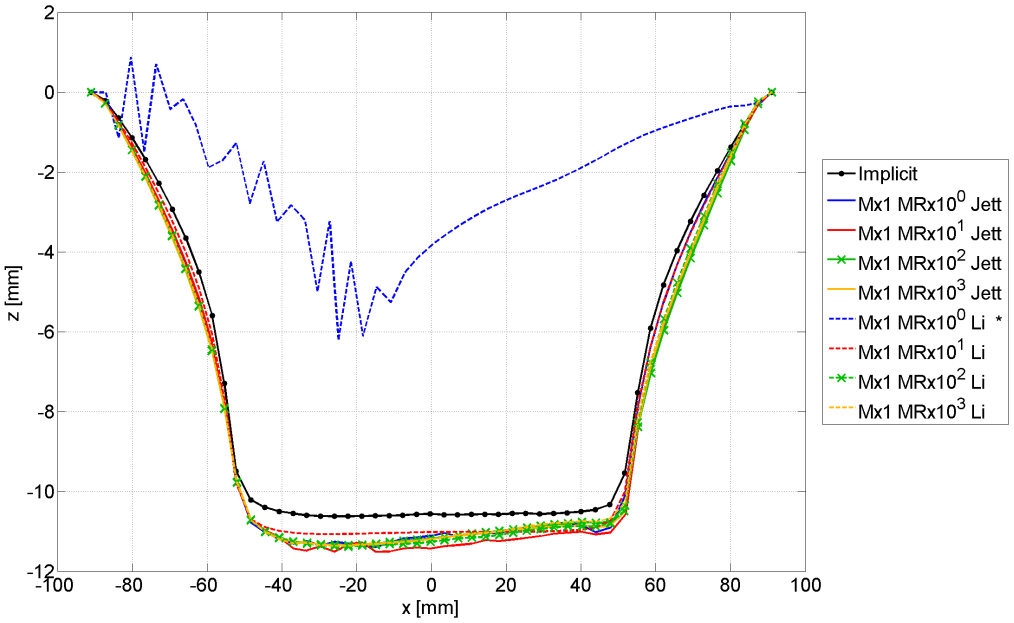


Figure 7.11: Comparison of the influence of mass scaling in rotation when using Li's and Jetteur's matrices at the end of the second line (point E)

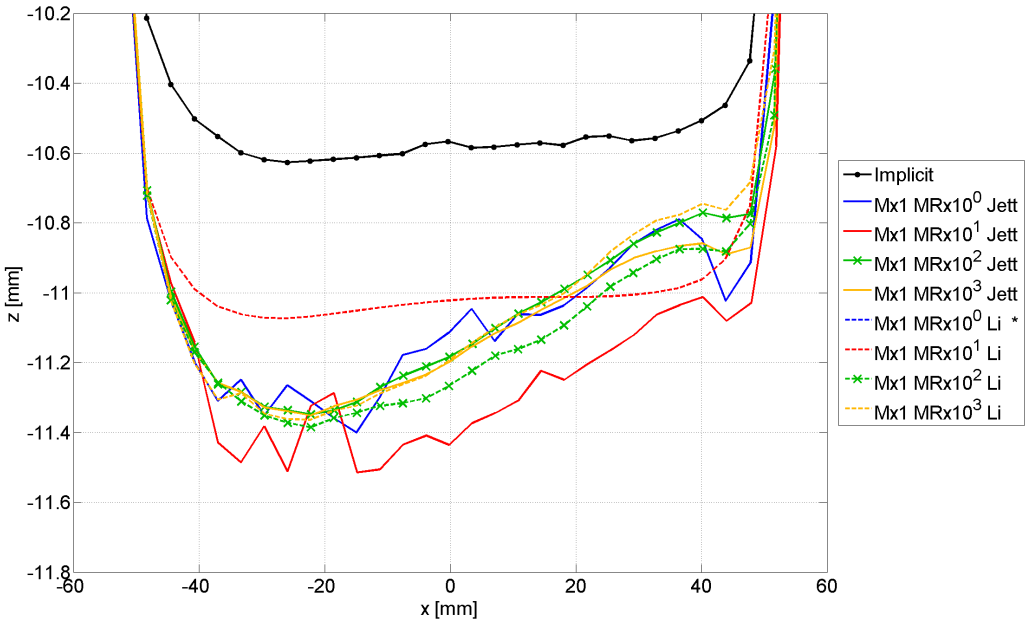


Figure 7.12: Comparison of the influence of mass scaling in rotation when using Li's and Jetteur's matrices at the end of the second line (point E) – zoom on the vertical axis

respectively, Li's and Jetteur's diagonal matrices without any additional mass scaling in rotation. In these figures, it is obvious that Jetteur's matrix produces more stable results if no additional mass scaling in rotation is used.

The same results are shown at the end of the second line (point E) in Figures 7.15 and 7.16, respectively. These figures lead to the same conclusions as at the end of the first line (point C), as far as stability and accuracy are concerned.

If an additional mass-scaling factor in rotation of 1000 is applied, the results are different, especially when using Li's matrix. Figures 7.17 and 7.18 compare the influence of mass scaling on the shape in a cross section along $y = 0$ at the end of the first line (point C) when using Li's and Jetteur's diagonal matrices, respectively, with an additional mass-scaling factor in rotation of 1000. It can be seen that in this case, there is no significant difference between Li's and Jetteur's matrices when the mass-scaling factor is smaller than 10^3 to 10^4. The simulations with Jetteur's matrix seem to be slightly less stable. Indeed, the results with a mass-scaling factor of 10^4 already had some small parasitic oscillations, which was not the case with Li's matrix.

The same results are shown at the end of the second line (point E) in Figures 7.19 and 7.20, respectively. Similar conclusions can be drawn as far as the maximum mass-scaling factor and the stability of Jetteur's and Li's matrices.

In conclusion, in all the results presented in this section (Figures 7.13 to 7.20), the use of an additional scaling factor for the rotation terms of the mass matrix greatly improved the stability of the results. Moreover, the results obtained with Li's matrix are slightly more stable than the one obtained with Jetteur's matrix, especially for large mass-scaling factors.

A mass-scaling factor of up to 10^4 can be used without any significant deterioration of the results. This factor leads to a multiplication of the maximum stable time step by the square root of this number, hence a reduction in the computation time by a factor of 100. This can be verified in Table 7.3, which presents the computation times of all the line test simulations. The computation time of the simulation with a mass-scaling factor of 10^4 is around 50 % of the computation time of the implicit simulation. However, a mass-scaling factor of 10^3, which is safer concerning stability, does not decrease the CPU time.

In terms of accuracy, the shape of all the dynamic explicit simulations is consistently too deep (up to 1 mm).

7.4 Conclusions on the MST Method

The purpose of the present chapter was to evaluate the performance — both in terms of accuracy and computation time — of the dynamic explicit strategy with shell elements, and the MST method in particular, to simulate the SPIF process.

The main conclusion of this study is that, on the one hand, it is possible to reduce the computation time of the simulation by using a dynamic explicit approach instead of a classic implicit strategy. On the other hand, this approach has several drawbacks with the Lagamine code.

First, the use of a dynamic explicit strategy introduces inertia terms into the equilibrium equation and parasitic oscillations. With a more advanced commer-

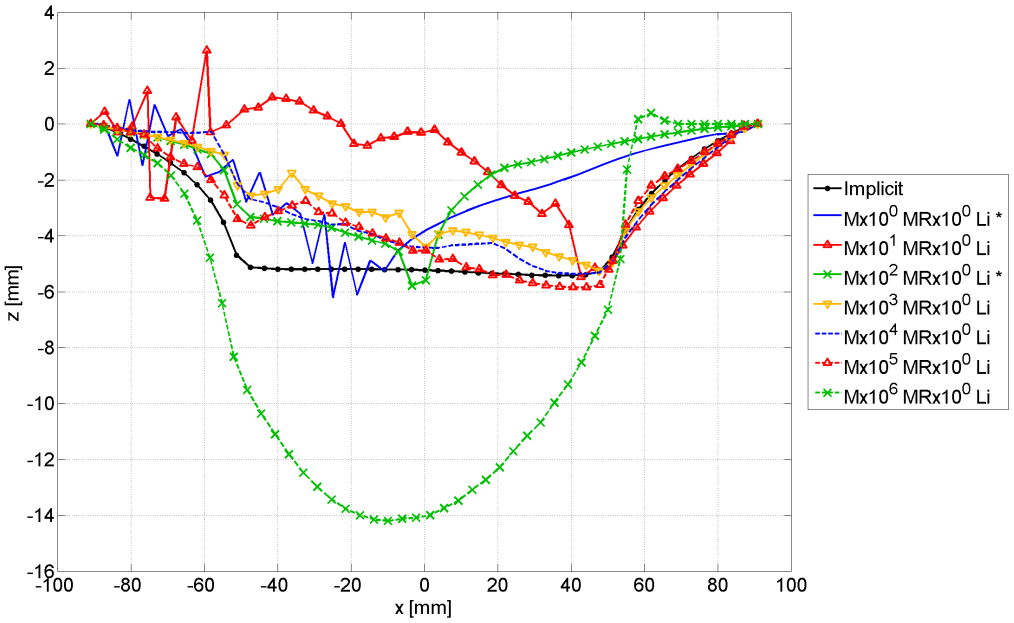


Figure 7.13: Comparison of the influence of mass scaling (without any additional scaling in rotation) when using Li's matrix at the end of the first line (point C)

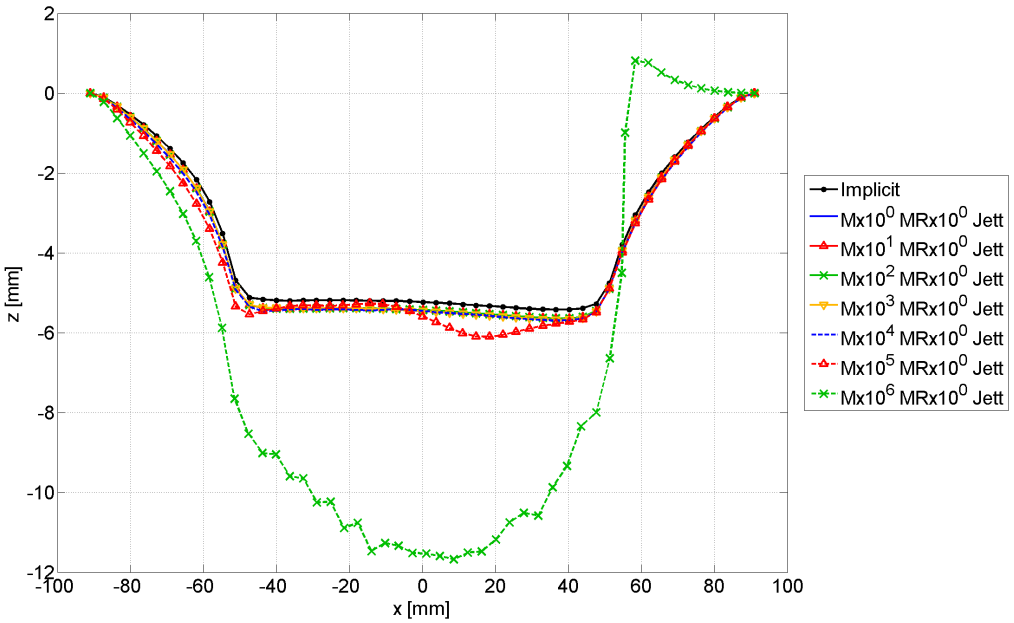


Figure 7.14: Comparison of the influence of mass scaling (without any additional scaling in rotation) when using Jetteur's matrix at the end of the first line (point C)

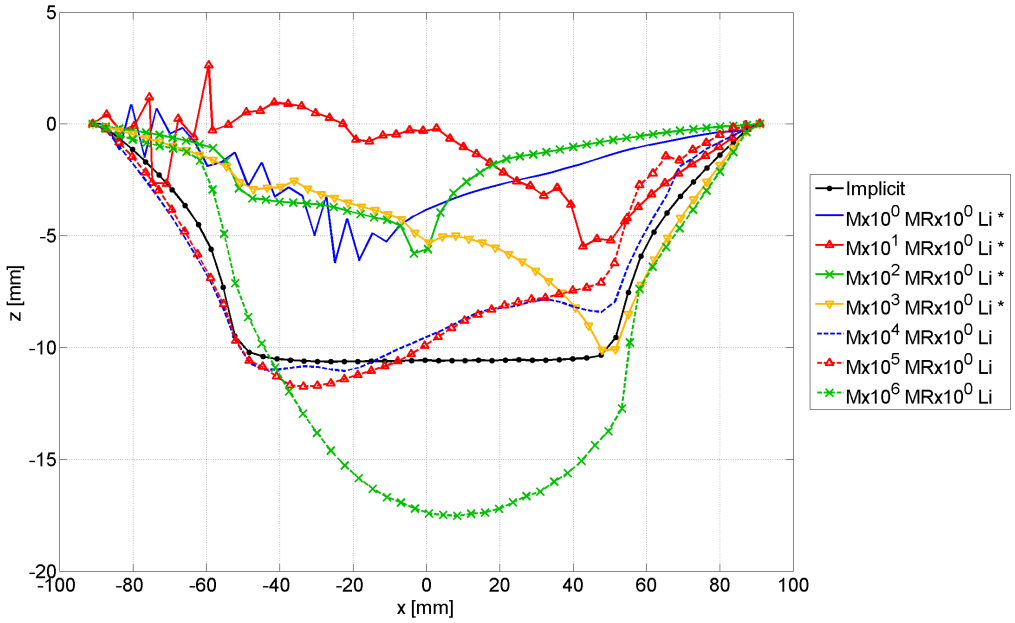


Figure 7.15: Comparison of the influence of mass scaling (without any additional scaling in rotation) when using Li's matrix at the end of the second line (point E)

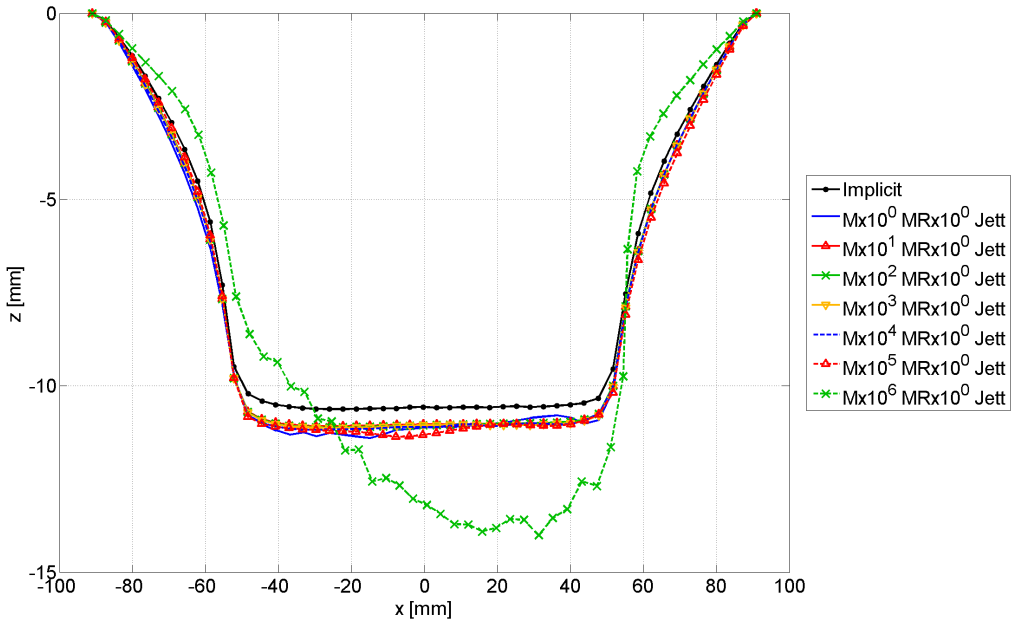


Figure 7.16: Comparison of the influence of mass scaling (without any additional scaling in rotation) when using Jetteur's matrix at the end of the second line (point E)

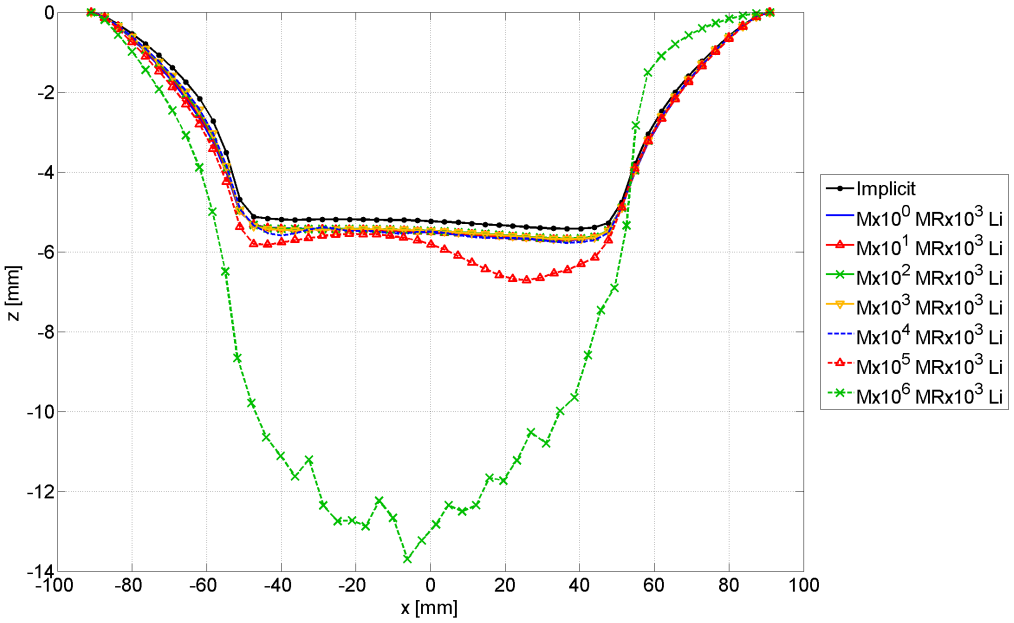


Figure 7.17: Comparison of the influence of mass scaling (with an additional scaling in rotation of 1000) when using Li's matrix at the end of the first line (point C)

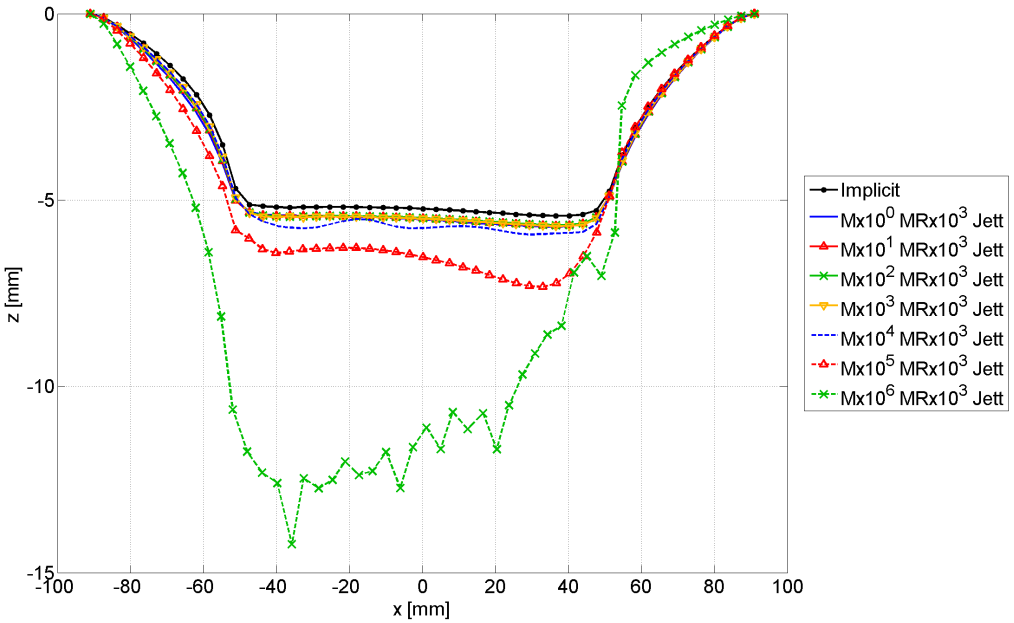


Figure 7.18: Comparison of the influence of mass scaling (with an additional scaling in rotation of 1000) when using Jetteur's matrix at the end of the first line (point C)

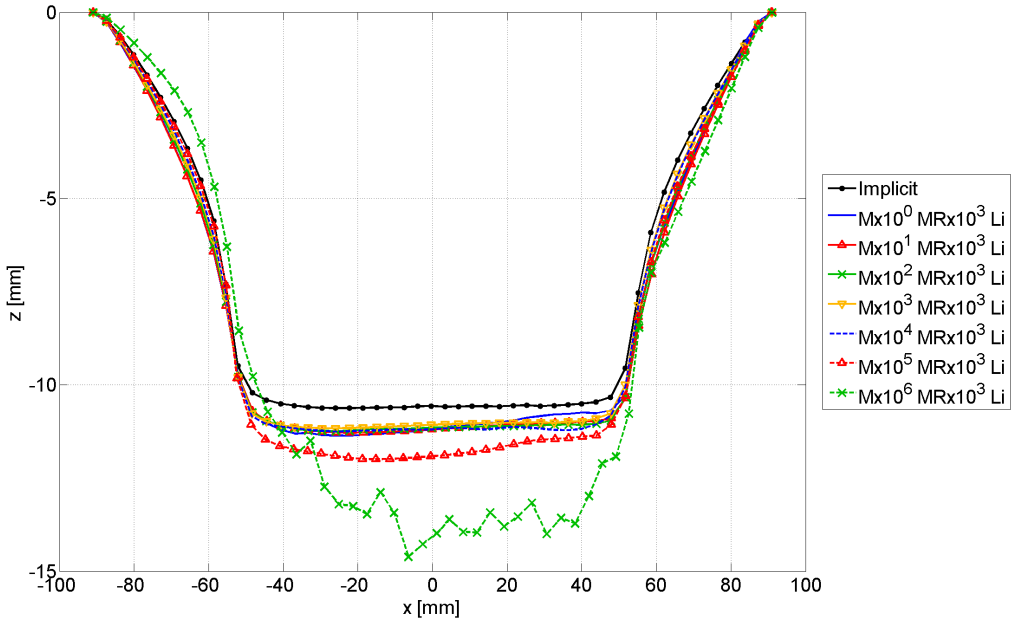


Figure 7.19: Comparison of the influence of mass scaling (with an additional scaling in rotation of 1000) when using Li's matrix at the end of the second line (point E)

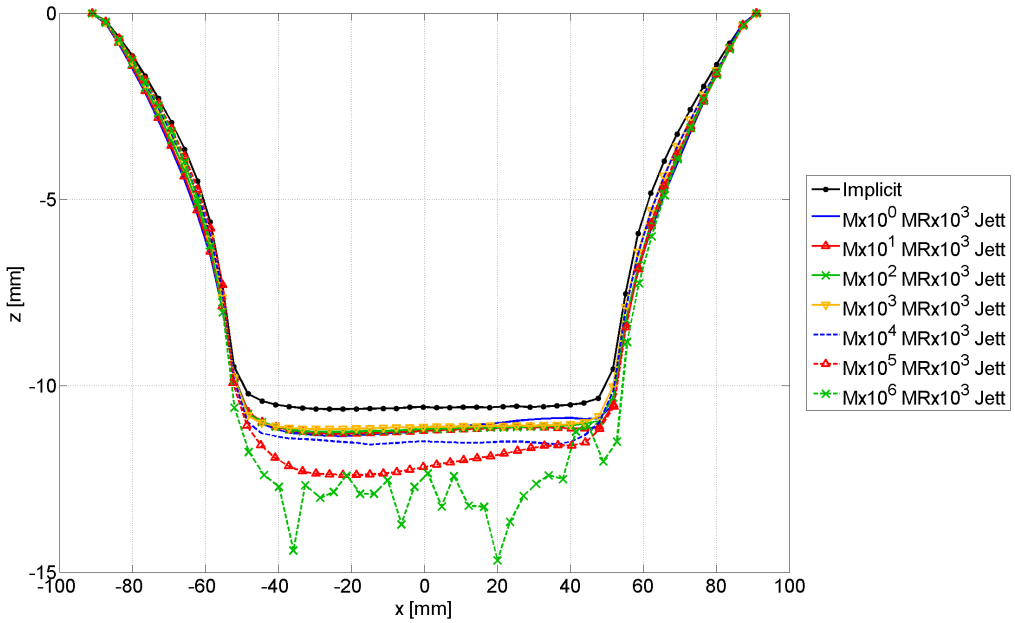


Figure 7.20: Comparison of the influence of mass scaling (with an additional scaling in rotation of 1000) when using Jetteur's matrix at the end of the second line (point E)

Simulation	M	CPU time [s]	Relative CPU time
Implicit	—	7 273	100 %
MST Li	10^0	315 926	4 344 %
MST Li	10^1	89 519	1 231 %
MST Li	10^2	30 461	419 %
MST Li	10^3	9 786	135 %
MST Li	*10^4*	*3 446*	*47 %*
MST Li	10^5	1 034	14 %
MST Li	10^6	381	5 %
MST Jett	10^0	302 291	4 156 %
MST Jett	10^1	104 080	1 431 %
MST Jett	10^2	31 003	426 %
MST Jett	10^3	11 308	155 %
MST Jett	*10^4*	*3 292*	*45 %*
MST Jett	10^5	1 164	16 %
MST Jett	10^6	376	5 %

Table 7.3: Comparison of the computation times of line test simulations for different mass-scaling factors

cial code like Abaqus/Explicit or LS-Dyna, numerical damping limits the effect of these terms and make the use of very large mass-scaling factors possible. For example, Abaqus could produce stable results for the line test with a mass-scaling factor of up to 10^5, which significantly reduces the computation time. With the LAGAMINE code, only a very simplified damping model exists but no satisfactory set of damping parameters could be found to stabilize the results. Therefore, the shape can not be predicted as accurately with an explicit strategy as with the implicit strategy.

Secondly, the choice of the mass-scaling factor results from a compromise between accuracy and computation time. It is always recommended to compare the results obtained with different values. With LAGAMINE, the mass-scaling factor is limited to approximately 10^3. Unfortunately, this factor did not produce any reduction in the computation time, compared to the implicit simulation. This conclusion could be modified if the mesh had many more DOFs.[3] However, with shell elements, the in-plane element size can not be reduced too much because it needs to be larger than the thickness. Therefore, the number of shell elements is limited by the size of the sheet metal.

Thirdly, the explicit strategy is much more unstable than the implicit one. Even when the time step is smaller than the maximum stable time step, it is necessary with LAGAMINE's shell element to increase the value of the masses of the rotational DOFs in order to stabilize the results.

Finally, the simulations using a dynamic explicit strategy are much more sensitive to the quality of the mesh than the simulations with an implicit one.

[3]Indeed, the computation time in an implicit strategy is proportional to the square of the number of DOFs, but increases approximately linearly with this number when using a dynamic explicit strategy.

Therefore, the user should be very careful when designing a mesh. In particular, the thickness of the elements should always remain much smaller than the in-plane length. When using an implicit strategy, this condition is less strict. For example, Mesh 1 of Table 7.1 could produce accurate results with an implicit strategy, even though none of the two design rules was respected.

With all these drawbacks, Lagamine does not seem to be very appropriate for simulating SPIF with a dynamic explicit strategy. To make up for this lack, it would be necessary to add an efficient damping algorithm. Otherwise, the implicit strategy is recommended.

The next chapters of this thesis will investigate the deformation mechanism occurring during SPIF. This will be done on more complex geometries like cones using an implicit strategy.

Part III

Implicit Simulations of SPIF

Chapter 8

Analysis of the Boundary Conditions

In order to reduce the computation time of part having a rotational symmetry, a possible solution was to simulate only part of the mesh (a pie) and apply boundary conditions to both edges of the pie in order to replace the missing material. The purpose of this chapter is to analyze the influence of this simplification. This will be done on a complex part: a cone with vertical walls performed in five different steps.

8.1 Description of the Cone with Vertical Walls

8.1.1 Geometry and Material

Figure 8.1 shows a 3D view and a schematic view with the dimensions of the cone. It is a cone with a vertical wall (a cylinder) with a depth of 30 mm. The tool is a sphere with a diameter of 10 mm.

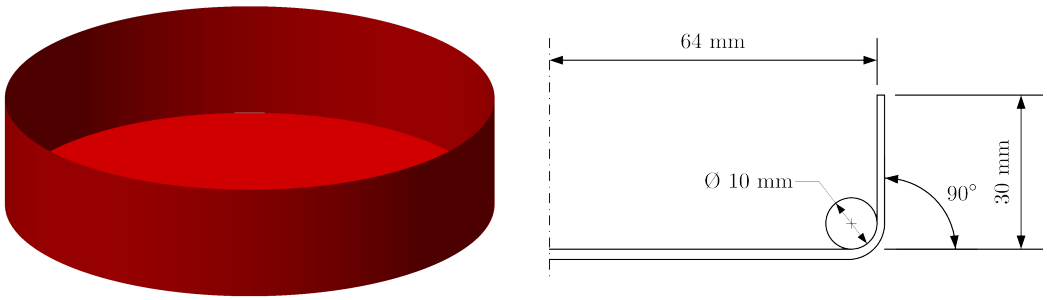


Figure 8.1: 3D view and dimensions of the cone with vertical walls

The material used is an AA3003-O aluminum alloy, whose characteristics are presented in Appendix C. The initial thickness of the metal sheet is 1.5 mm. When forming a cone with this material, the maximum wall angle that can be achieved before failure is between 71 and 76 degrees [DUF08b]. Forming a cone with vertical walls is then impossible since the theoretical thickness of such a part is zero and the part would break, at least with a classic tool path where the tool

follows the final geometry. As explained in the literature review in Section 2.1.8, it is possible to overcome this limitation by using an advanced tool path.

Verbert et al. developed a multi-stage forming strategy [DUF08b]. Instead of forming a 90-degree wall-angle cone directly, the authors started by forming a 50-degree cone, followed by a 60, 70, 80, and 90-degree cone. The global tool path is then composed of a succession of five stages during which the tool forms cones with increasing wall angles, as presented in Figures 8.2*a* to 8.2*e*. So, the global tool path is composed of five times 30 contours, with a stepdown of 1 mm between two successive contours.

The location of the stepdown in the circumferential direction is different in each of the five stages, as can be seen in the five figures.[1] The reason for this is that fracture would most likely occur during the stepdown because of excessive thinning if it were always performed at the same location. A zoom on the transition between successive contours is shown in Figure 8.2*f*.

8.1.2 Finite Element Model

The simulation of the cone's forming was performed using the LAGAMINE FE code.

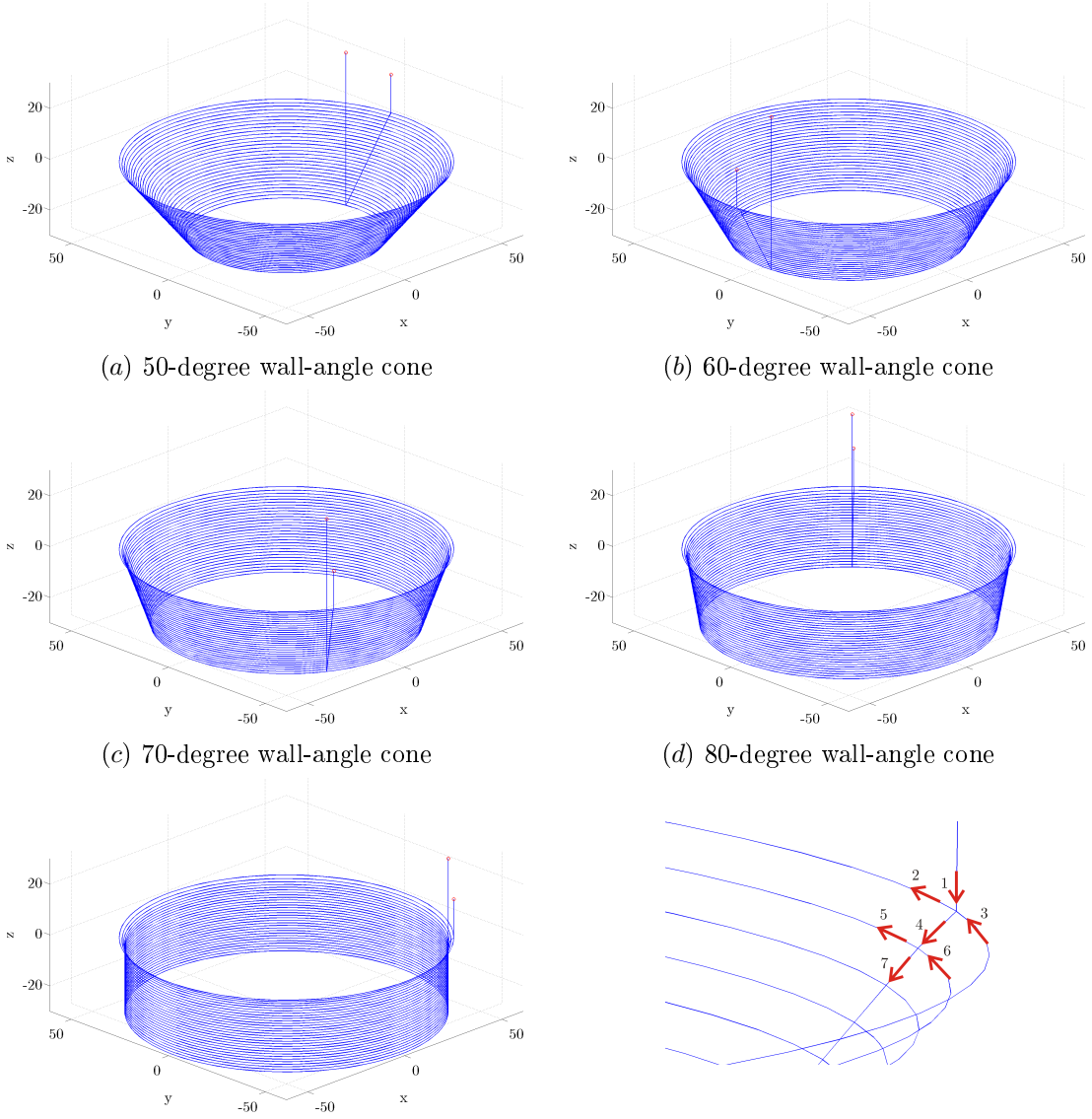
As stated in the introduction of this chapter, the goal is to analyze the influence of using a partial mesh to simulate the forming of a part with a rotational symmetry. For this reason, three different meshes were tested: a 45-degree pie mesh, a 90-degree pie mesh, and a full 360-degree mesh. These meshes are shown in Figure 8.3 and the orientation of the global axes for the three meshes is shown in Figure 8.3*a*. The characteristics of the meshes are given in Table 8.1. In this table, the last column corresponds to the number of elements in the circumferential direction in the part of the mesh where it is regular, i.e., the elements for which their radial position is larger than $r = 33$ mm. The three meshes then have more than one element per degree in the circumferential direction.

Regarding the density of the meshes in the radial direction, it can be seen that the elements are regularly distributed in the region between $r = 33$ and $r = 64$ mm.[2] Its density is higher in the region located between $r = 45$ and $r = 55$ mm, the region that undergoes the highest deformation, as will be seen in the analysis of the results.

The element type used was the COQJ4 element, presented in Section 3.1.2. The mechanical law had four in-plane integration points and, for each of these, five integration points across the thickness. Of the latter, the first one is located close to the lower surface, the third is in the mid-plane of the element, and the fifth is close to the upper surface. This gives a total of 20 integration points per element. In comparison, a mesh designed with three layers of LAGAMINE's brick

[1] The location of the stepdown is, for each of the five stages, at 6, -177, -118, 43, and -45 degrees, respectively, from the x-axis.

[2] The cone is formed with a backing plate. This plate is a stiff metal sheet in which a hole slightly larger than the cone is drilled. This plate is placed under the metal sheet in which the cone is formed. It is used to ensure that the edge of the cone is sharp without too much elastic deformation. This cone had a diameter of 128 mm and the backing plate, a hole diameter of 131 mm. For this reason, an additional element was added between $r = 64$ and $r = 65.5$ mm in order to take this backing plate into account.



(*a*) 50-degree wall-angle cone

(*b*) 60-degree wall-angle cone

(*c*) 70-degree wall-angle cone

(*d*) 80-degree wall-angle cone

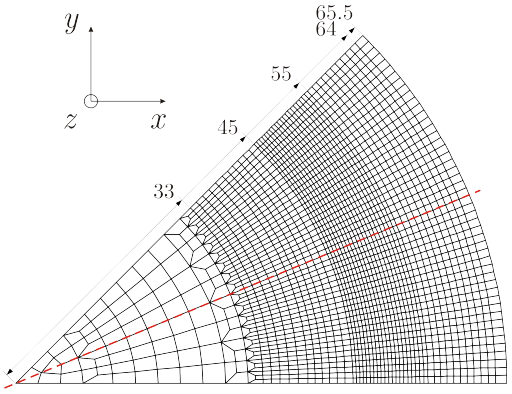
(*e*) 90-degree wall-angle cone

(*f*) Zoom on the transition between contours

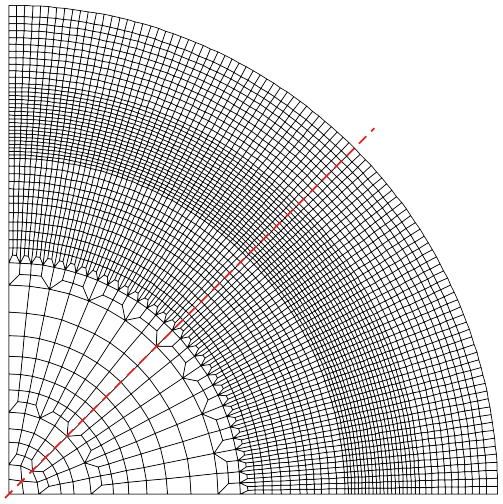
Figure 8.2: Five steps of the tool path of the cone with vertical walls and zoom on the transitions between contours

Mesh	Number of elements	Number of nodes	Number of DOFs	Number of elts in circum. dir.
45-degree pie	2 320	2 244	13 300	48
90-degree pie	4 506	4 450	26 392	96
360-degree pie	17 376	17 571	104 262	384

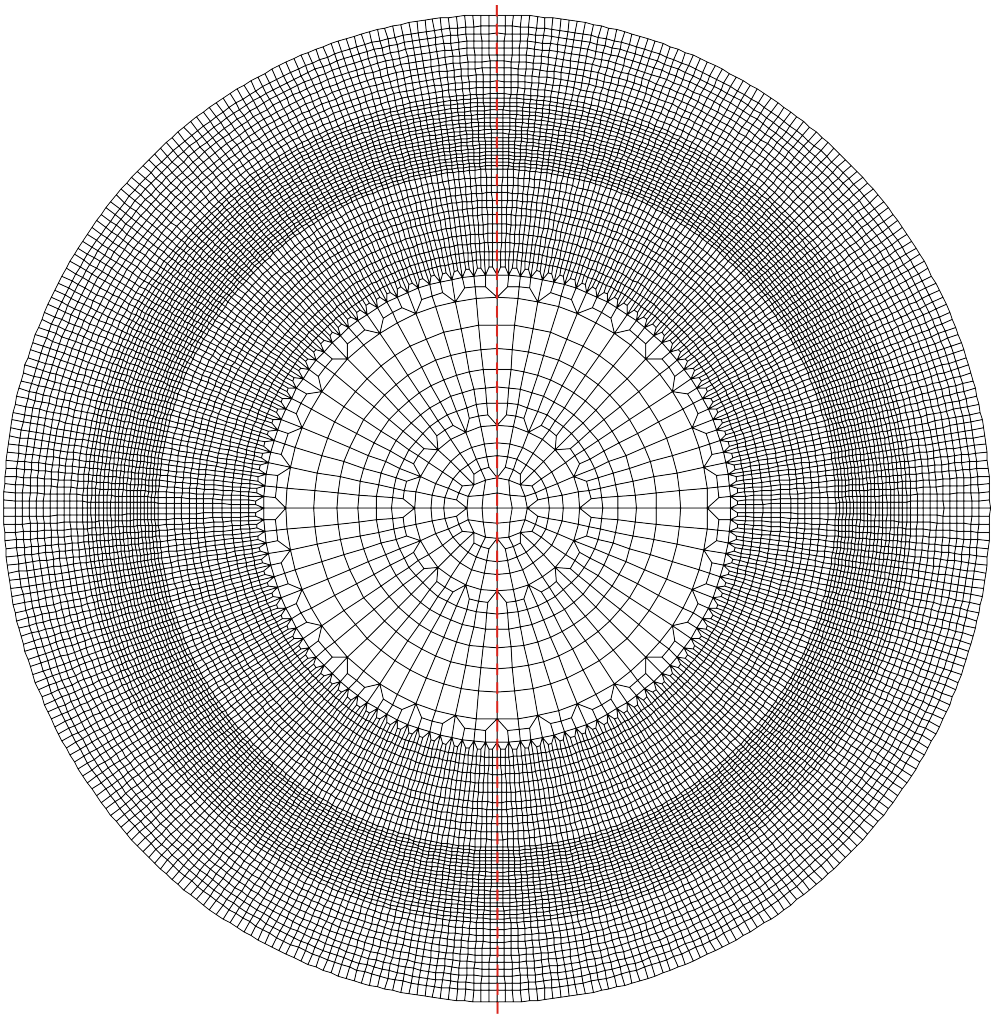
Table 8.1: Characteristics of the meshes used for the cone with vertical walls



(*a*) 45-degree pie mesh



(*b*) 90-degree pie mesh



(*c*) 360-degree full mesh

Figure 8.3: Meshes used for the cone with vertical walls

reduced-integration element BWD3D would have only three integration points for the same in-plane area as one shell element.

The contact between the tool and the metal sheet is taken into account using contact elements. The penalty coefficient of these elements was chosen to be 1000 N/mm^3. Given the absence of experimental data, the friction coefficient was assumed to be zero.

In order to minimize the impact of the material missing at both edges of the pie, *rotational* boundary conditions were imposed, which consist of a link between the displacements of both edges, as presented schematically in Figure 8.4. Even though these conditions do not exactly reproduce the missing material, they induce a smaller deviation than the one induced by classic *symmetry* boundary conditions [HEN05a, Section 7.2.1].[3] The reason for this is that these boundary condition do not prevent the material at the edges of the pie mesh to be carried along in the circumferential direction of the tool movement. Indeed, even without friction between the tool and the sheet metal, the tool induces a force component in the circumferential direction because of the material that surrounds the tool during its movement. Since the tool always moves in the same circumferential direction, the whole cone has a tendency to twist around its axis of rotational symmetry, a movement that can be simulated with these boundary conditions even at the edges of the pie, but can not be with the symmetry boundary conditions. This twist effect will be analyzed further in the text.

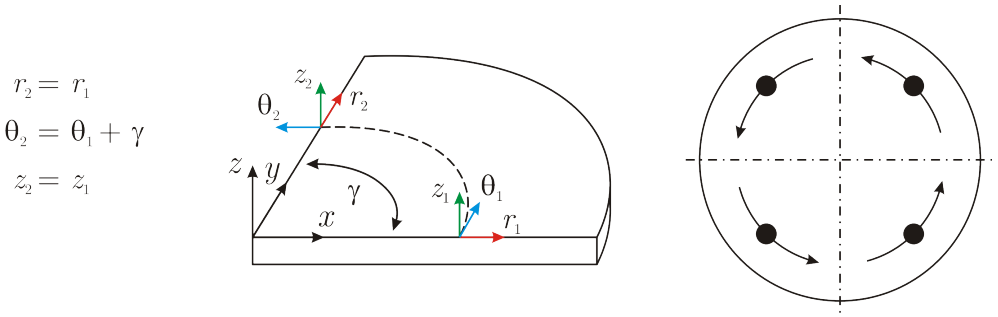


Figure 8.4: Rotational boundary conditions and their associated virtual tools (schematic view of a 90-degree pie)

When using a pie mesh, the tool path, shown in Figure 8.2, has to be modified; otherwise, the tool would not touch the sheet metal most of the time, which would increase the computation time unnecessarily. Figure 8.5 shows the first step of the tool path transformed in the case of a 45-degree pie mesh. This must be compared to Figure 8.2*a*. As can be seen, at the end of each contour, the tool goes up, then moves in a horizontal plane up to the a position located on top of the beginning of the next contour and then goes back down.

[3] With symmetry boundary conditions, the circumferential coordinate θ_i of the nodes of both edges is fixed. This means that the nodes of one edge can move either in the radial or in the vertical direction but not out of the vertical plane containing their initial positions.

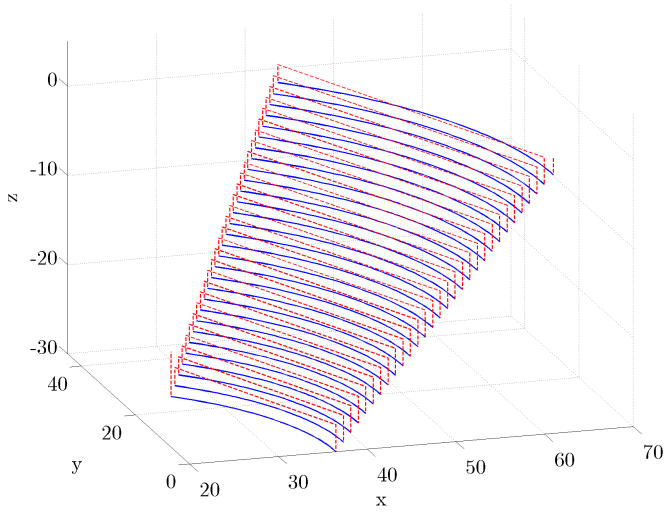


Figure 8.5: Modified tool path used for the 50-degree wall-angle cone with a 45-degree pie mesh

8.2 Prediction of the Shape and Thickness

8.2.1 Shape

This section discusses the results concerning the shape of the cone. The thickness will be analyzed in the next section, although the graphs showing the thickness at various stages of the simulation will be shown on the same page as the corresponding graphs of the shape.

The shape of the simulations is presented in cross sections in the middle of the pie mesh for the 45- and 90-degree meshes. The reason for this choice is that the boundary conditions always introduce some error. The quality of the results improves at greater distances from the boundary conditions, which is the case in the middle of the mesh. Regarding the full cone, the cross section was performed in the plane $x = 0$ because the stepdown is never performed close to this plane. The three sections are illustrated by a red dotted line in Figure 8.3. Since the meshes are composed of shell elements, only the average surface is shown.

For the experimental tests performed at KUL, the global shape of the outer surface (opposite face with respect to the tool) of the cone was measured with a laser scanner after each of the five stages and a point cloud was extracted. From this point cloud, an average cross section was extracted along with the 90 % confidence interval of each point of the section.[4] This section in the outer surface was shifted by half the initial thickness (0.75 mm), so that it would match the results of the mid-surface of the simulations. This introduces an error in the regions where the thickness was reduced, mainly in the wall region, but this error

[4]The computation of this average cross section is performed using the following strategy. First, the point cloud is centered around the origin of the axes with the z-axis being positive towards the tool and negative towards the bottom of the cone. The coordinate $z = 0$ corresponds to the lower surface of the initial metal sheet. Then, for each point $\langle P \rangle$ of the pointcloud, the angle between the z-axis and the line going from the origin to $\langle P \rangle$ is computed. All the points having approximately the same angle are regrouped and averaged. The 90 % confidence interval of these points is also computed.

remains very small.[5].

Since the tool path is composed of five stages, the results will be presented at the end of each of the five stages. Due to excessive computation time, the full simulation, with the 360-degree mesh, is not available for the last stage.

The first stage of the tool path produces a cone with a wall-angle of 50 degrees. The predicted shape and the measured shape are compared to the CAD surface (theoretical model of the part used to design the tool path) in Figure 8.6. The measured surface is plotted with an error bar representing the confidence interval or variability of the measurement. At this stage, the variability is slight and barely visible on the plot.

As can be seen in the figure, all the simulations have, compared to the measured profile, an error of approximately 0.5 mm close to the backing plate, i.e., around the largest radius $r = 65.5$ mm. This is due to the fact that the clamping of the sheet metal was modeled by fixing the outer nodes in translation only. Better results would have been obtained by fixing also the rotations of the nodes or, even better, by modeling the real clamping system. However, that would have increased the computation time.

Besides this shortcoming, the prediction made with the full (360-degree) mesh is almost perfect everywhere compared to the measured profile: there is an error of only 0.01 mm at the cone center.

The shape predictions obtained with the 90- and 45-degree pie meshes are also quite accurate: the error at the cone center is 0.39 and 0.62 mm, respectively. As will be explained below, the shape of the bottom part of the cone is always the most troublesome to predict. So, an error of approximately 0.5 mm is acceptable.

The first conclusion that can be drawn is that for a simple geometry such as this 50-degree wall-angle cone, a pie mesh can predict the shape of the final part with a sufficient level of accuracy.

The second stage of the tool path produces a cone with a wall-angle of 60 degrees. The shape at the end of this stage is shown in Figure 8.8. It can be seen that the bottom of the cone in the experimental section was lowered during this new stage. However, the bottom of the cone predicted by all the simulations did not follow the same trend, but rather remained at the same depth or moved slightly upwards.

A similar conclusion can be made about stages three, four, and five, shown in Figures 8.10, 8.12, and 8.14, respectively: the prediction of the bottom of the cone becomes less and less accurate as the simulation progresses. This effect can be seen more clearly in Figure 8.16, which shows the evolution of the depth of the center of the cone measured experimentally and predicted by all the simulations at the end of each stage. The position of the bottom of the cone is accurately predicted only after the first stage but not for the following ones.

The reason for this poor prediction of the bottom of the cone is the following.

[5]The theoretical thickness predicted by the sine law assumes that the material is in a plane-strain state and that the deformation mechanism is dominated by shear. The material is supposed to be projected vertically onto the final shape. If this were exact, the vertical distance between the mid-surface and the outer surface would everywhere be equal to half the initial thickness, even in the wall region

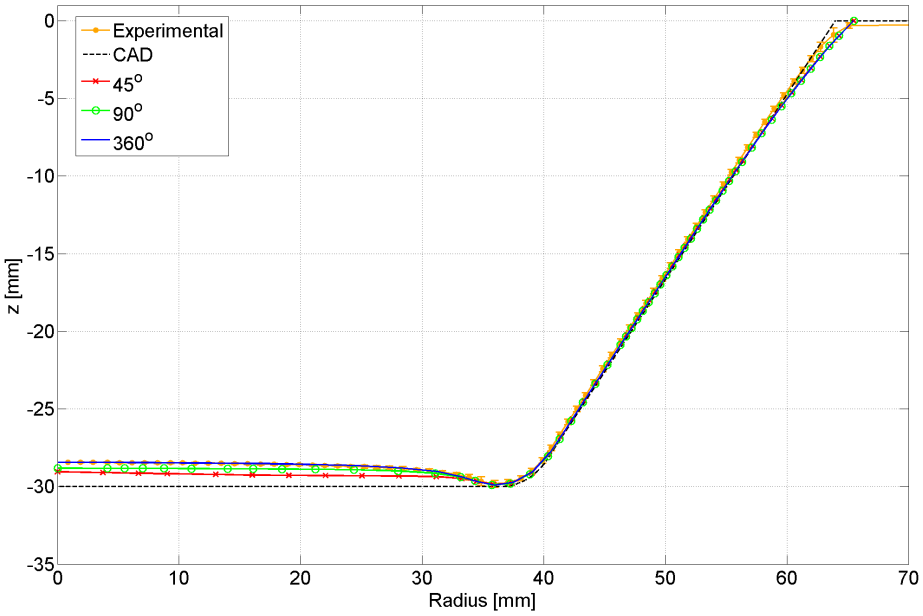


Figure 8.6: Shape of the cone with vertical walls after the first stage (50-degree cone)

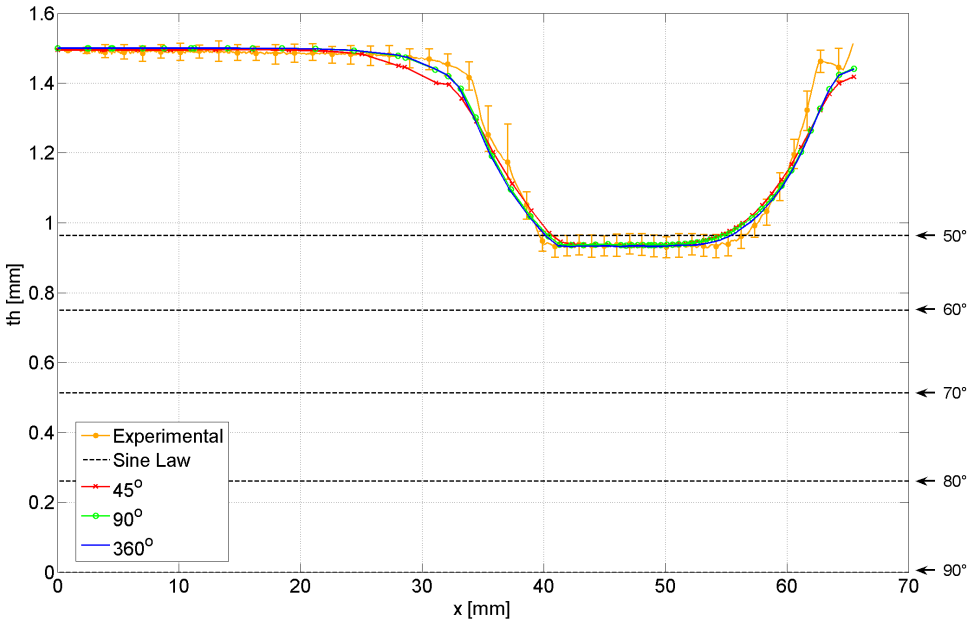


Figure 8.7: Thickness profile of the cone with vertical walls after the first stage (50-degree cone)

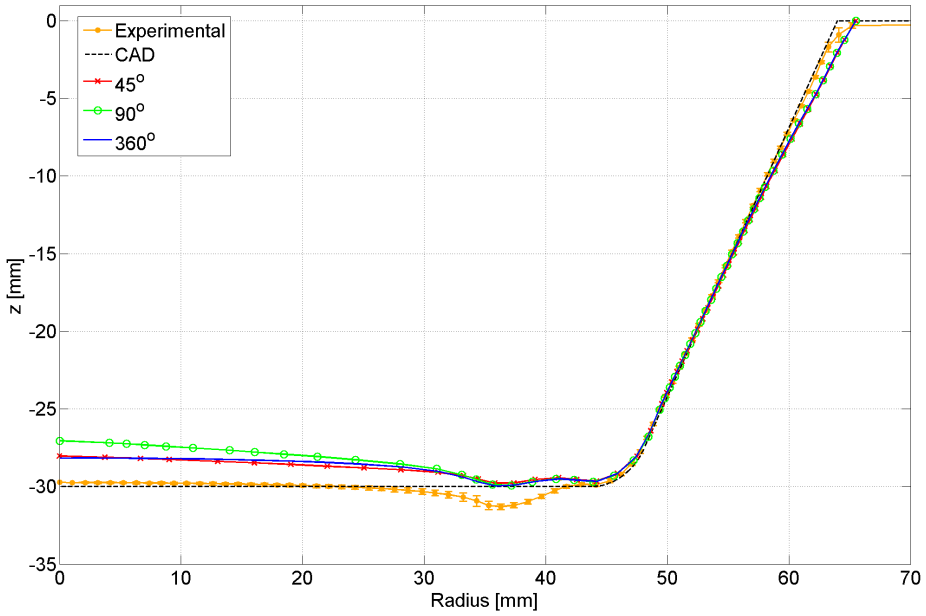


Figure 8.8: Shape of the cone with vertical walls after the second stage (60-degree cone)

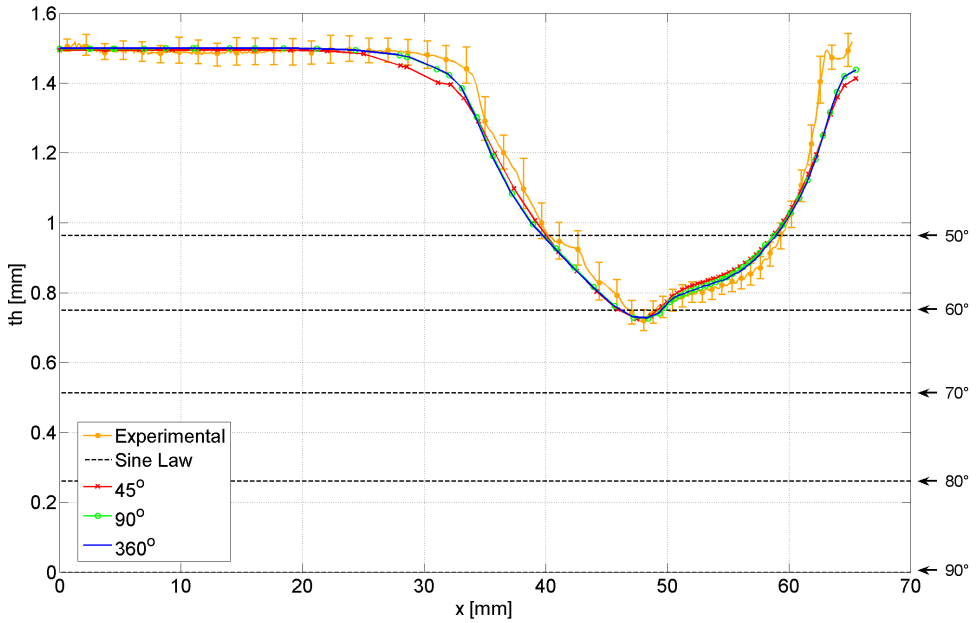


Figure 8.9: Thickness profile of the cone with vertical walls after the second stage (60-degree cone)

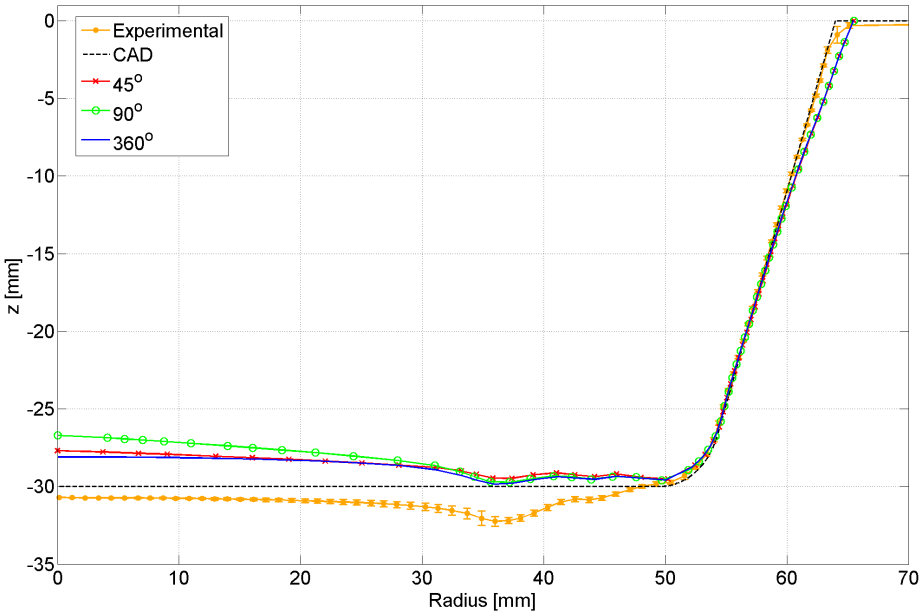


Figure 8.10: Shape of the cone with vertical walls after the third stage (70-degree cone)

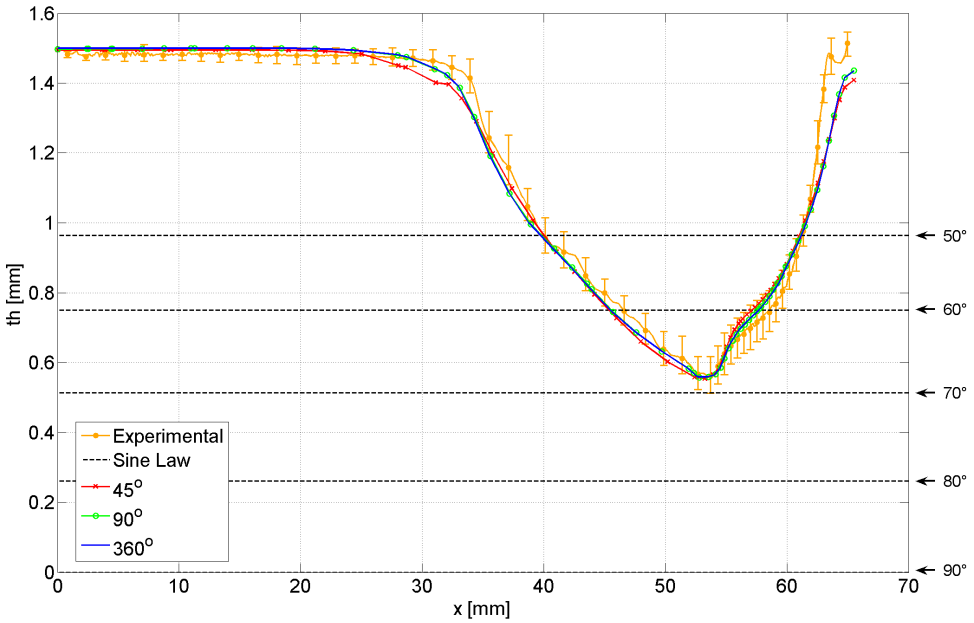


Figure 8.11: Thickness profile of the cone with vertical walls after the third stage (70-degree cone)

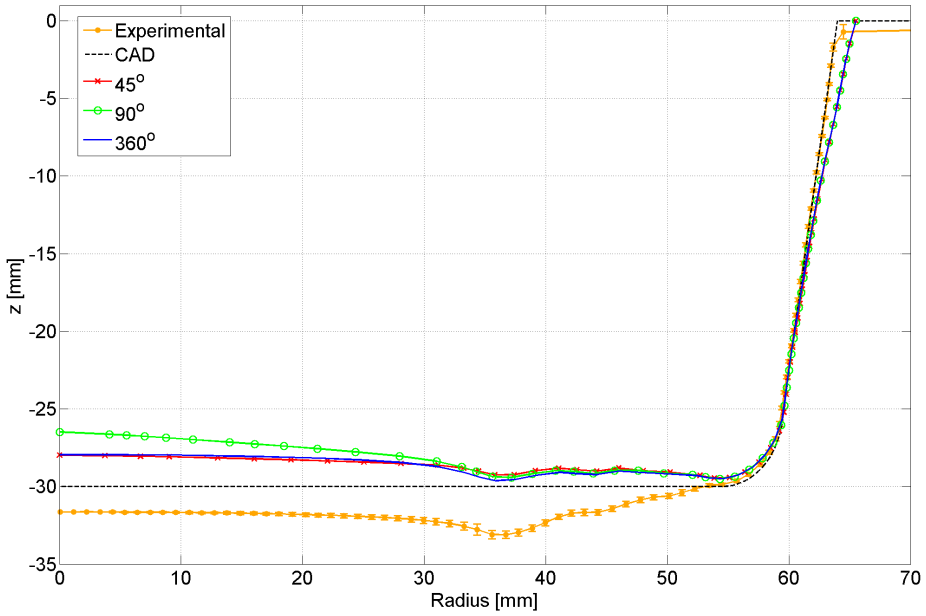


Figure 8.12: Shape of the cone with vertical walls after the fourth stage (80-degree cone)

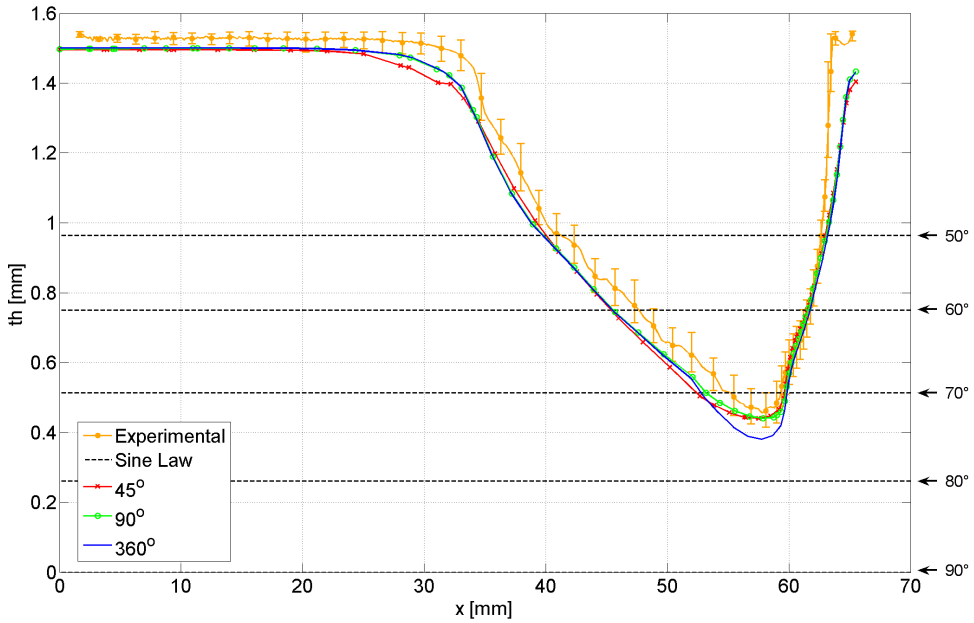


Figure 8.13: Thickness profile of the cone with vertical walls after the fourth stage (80-degree cone)

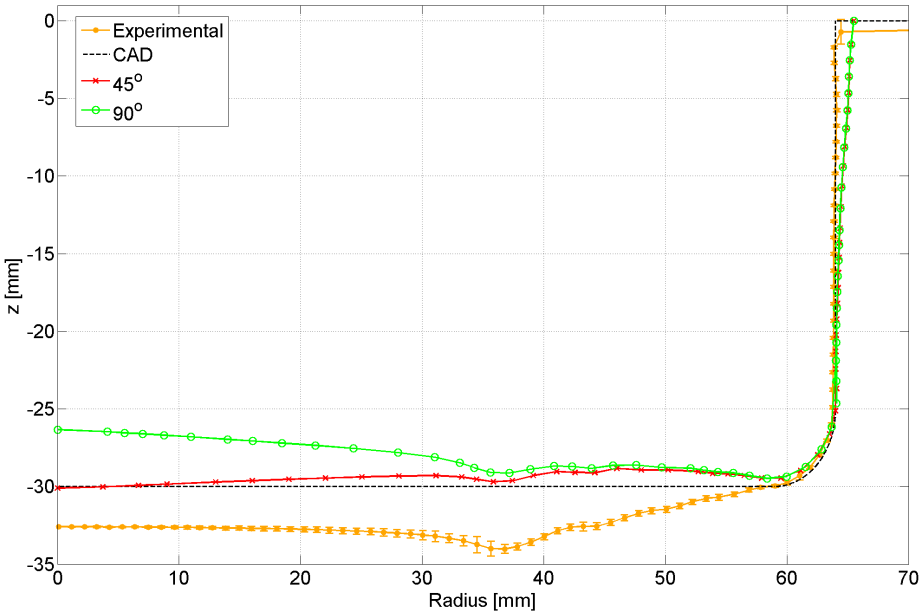


Figure 8.14: Shape of the cone with vertical walls after the last stage (90-degree cone); the simulation with the full 360° mesh was not available at this stage

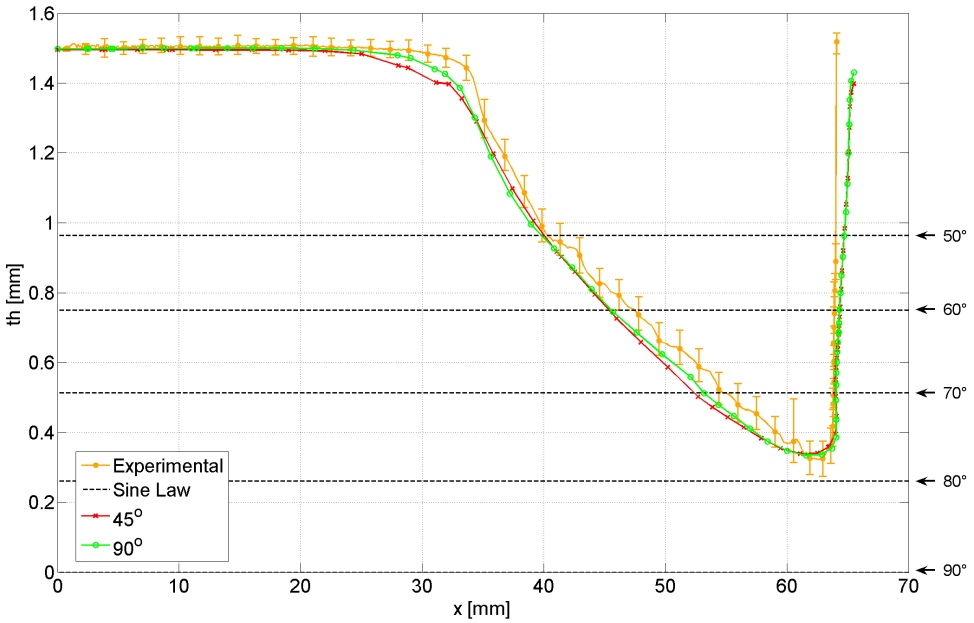


Figure 8.15: Thickness profile of the cone with vertical walls after the last stage (90-degree cone); the simulation with the full 360° mesh was not available at this stage

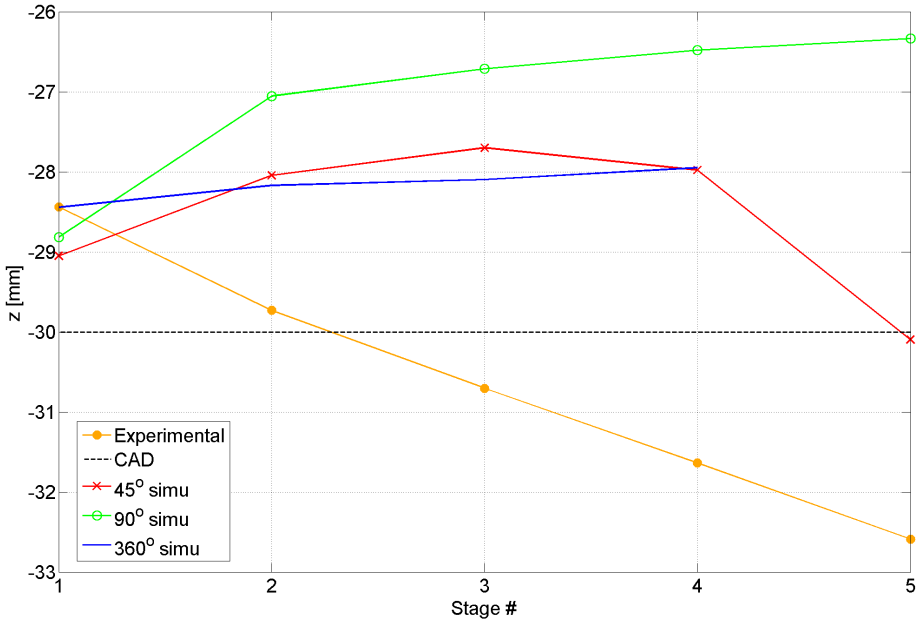


Figure 8.16: Depth of the center of the cone with vertical walls at the end of the five stages

First, the tool never touched this region. Its final position is then not constrained by the tool displacement as is the case for the wall region, but rather is dependent on the stiffness of the elements that are bent under the tool in the corner of the cone, i.e., at the transition between the wall and the bottom. Since the COQJ4 element uses a plane stress law, the stress distribution across the thickness is not accurate, especially under the tool, which causes some error in the stiffness of these elements.

Another important factor is that this simulation applies an extreme level of deformation, compared to more classic parts like a cone with a smaller wall angle processed in a single stage. This is illustrated in Figure 8.17, whose lower part shows the displacement of all the nodes in a cross section for the 90-degree pie mesh. The shape of the cone after each of the five stages is plotted in red.

The upper part of the figure shows the elongation of all the elements of the cross section in the radial direction at the end of the simulation. It can be seen that the length of some elements have been multiplied by a factor of 3.5, which is considerably more than the maximum elongation of 1.56 obtained after the first stage.[6] With such a large elongation, it is very difficult to maintain an acceptable aspect ratio for these elements. For such a simulation, it might be useful to perform remeshing when the elongation becomes too large, e.g., larger than two.

Figure 8.17 can also help verify that the mesh density was chosen in an appropriate way: the mesh density in the radial direction is higher in the region where the elements undergo the largest elongation, i.e., the region of the corner of the cone.

[6] The maximum elongation of the elements at the end of each of the five stages is the following: stage 1, 1.56; stage 2, 1.89; stage 3, 2.31; stage 4, 2.79; stage 5, 3.51.

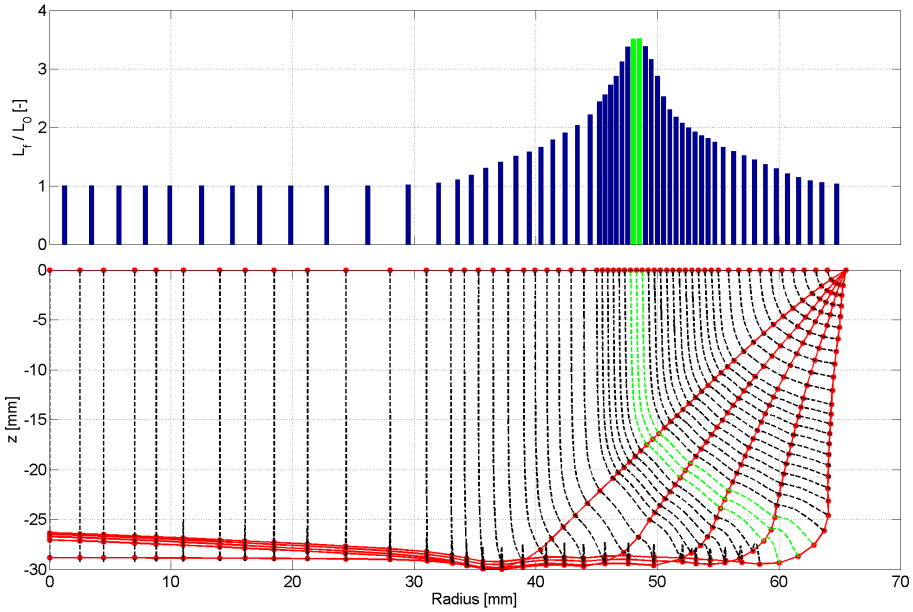


Figure 8.17: Nodal displacement during the simulation (bottom figure) and element elongation at the end of the simulation (top figure) with the 90-degree pie mesh

8.2.2 Thickness

This section focuses on the analysis of the prediction of the cone's thickness. This result is shown in a cross section at the end of each of the five stages in Figures 8.7, 8.9, 8.11, 8.13, and 8.15.

In each figure, the five horizontal black dashed lines represent the theoretical thickness of the wall region as predicted by the sine law, for a wall angle respectively equal to 50, 60, 70, 80, and 90 degrees. This theoretical law, which was introduced in the literature review in Section 2.1.8, assumes that the deformation mechanism is purely in plane strain. As was already mentioned in that section, the actual thickness is usually slightly thinner. This phenomenon is called "overforming" [DUF08b].

This overforming phenomenon can be seen clearly in Figure 8.7, which shows the thickness at the end of the first stage, a stage during which the tool path produces a cone with a wall-angle of 50 degrees. Both the predicted thickness and the measured one are slightly thinner in the wall region than the one predicted by the sine law for a 50-degree cone.

Concerning the accuracy of the thickness prediction, it can be seen that it is very precise everywhere for all the simulations except close to the backing plate ($r = 65.5$ mm) and around the corner of the cone. The error close to the backing plate was also detected in the prediction of the shape and is due to the clamping modeling of the metal sheet, which has been simplified for computational considerations. As to the error in the corner of the cone, it is probably due to the fact that the elements in that region were the last ones to be touched by the tool. Since the shell element uses a plane stress hypothesis, the stress state of these elements was probably not accurate and, as a result, neither were the strain and thickness predictions. When the tool is not in contact with an element, however,

the predictions are highly accurate.

Like the prediction of the shape, the prediction of the thickness at the end of the next stages becomes less and less accurate as the simulation progresses, even though the error remains within an acceptable range: there is a maximum deviation of around 0.06 mm (when the initial thickness is 1.5 mm). The experimental measurements show a minimum in the thickness and both the location and the value of this thickness are accurately predicted by all the simulations, with a maximum deviation between 0.01 and 0.015 mm for some stages, the last one in particular.

8.3 Prediction of the Force

This section focuses on the analysis of the prediction of the tool force. The experimental force has, unfortunately, never been measured for this cone, so the aim is to verify whether or not the partial meshes are able to reproduce the force predicted by the global mesh. A comparison with experimental measurements will be performed on a different cone geometry in Chapter 9, and the evolution of the force will be analyzed in much more detail as well.

Figures 8.18, 8.19, and 8.20 show the three components of the tool force in a radial coordinate system for the 45-, 90-, and 360-degree mesh simulations, respectively. These results were not represented in the same figure because the lengths of the tool paths of the three simulations are different, hence the time scale as well, since the tool moves at a constant velocity of 2 m/min.

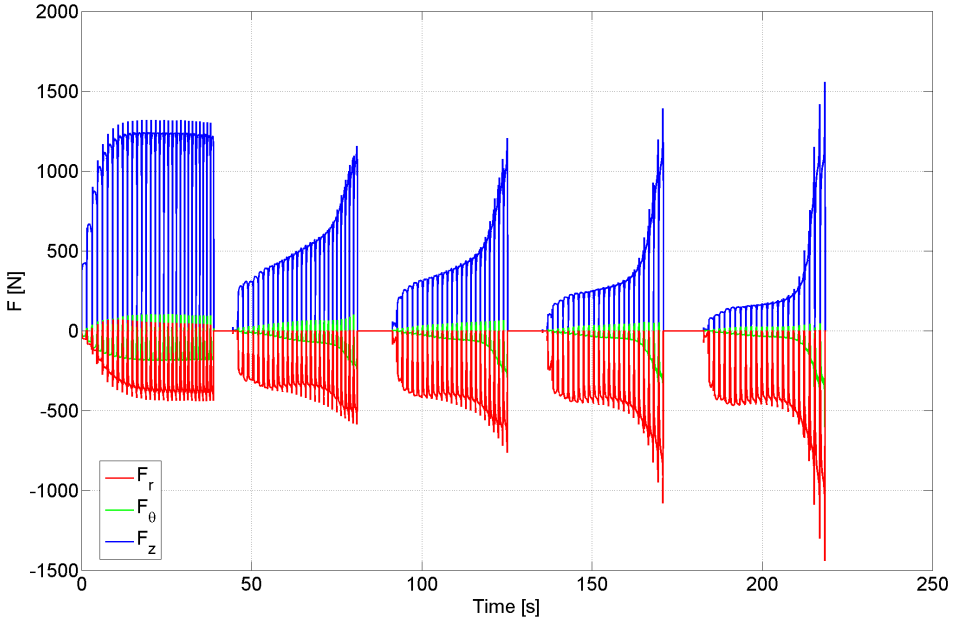


Figure 8.18: Three components of the tool force for the 45-degree pie-mesh simulation

Even though their time scales are different, it can be seen that the general evolution of the three figures is identical. The five stages of the simulation can be

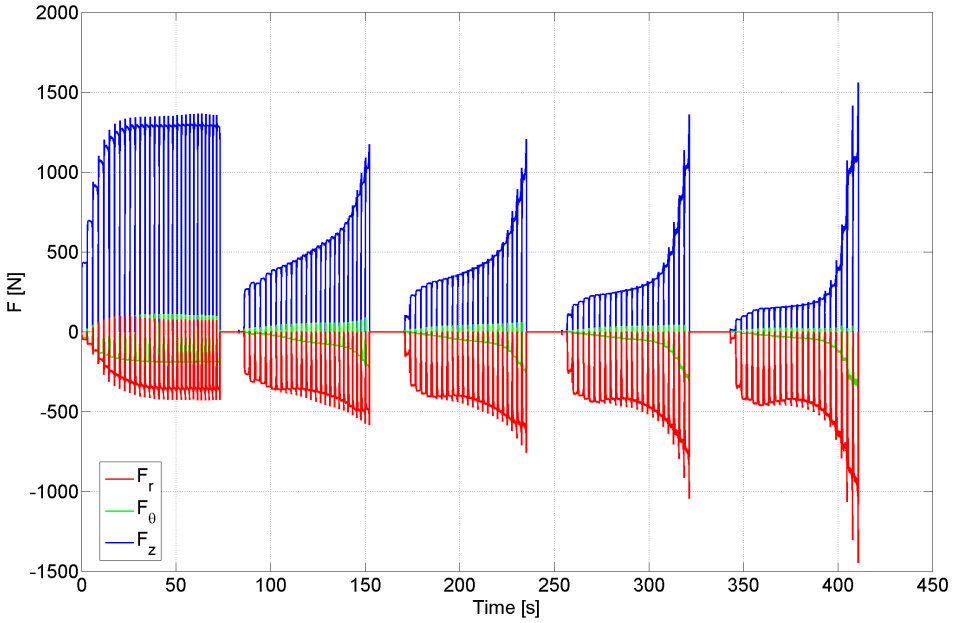


Figure 8.19: Three components of the tool force for the 90-degree pie-mesh simulation

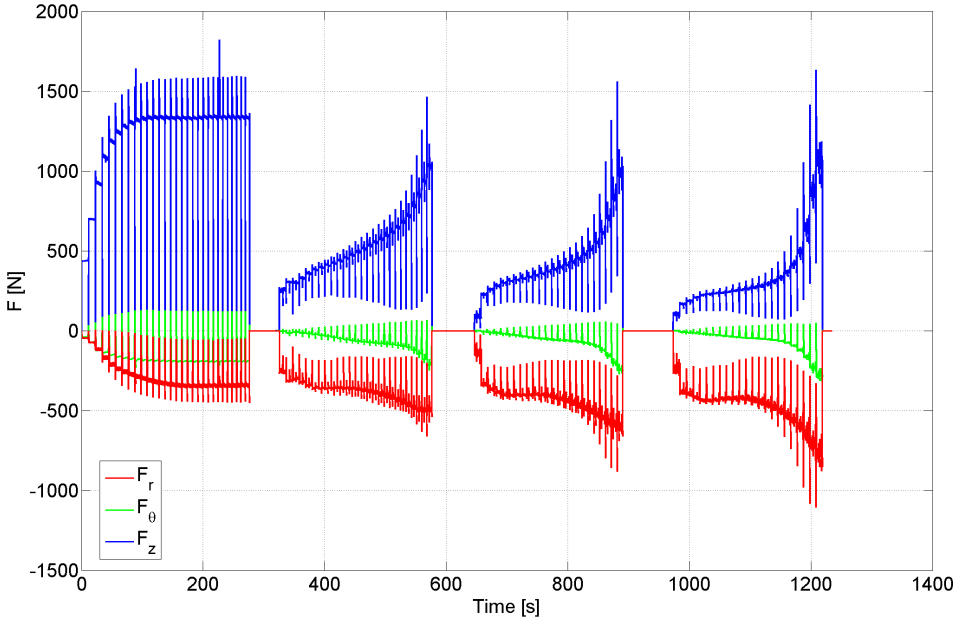


Figure 8.20: Three components of the tool force for the 360-degree mesh simulation

clearly identified. Once again, it should be noted that the results of the 360-degree mesh are only available up to the end of the fourth stage.

Large oscillations can be seen in the three figures. These oscillations actually only appear at the transition between contours and close to the boundary conditions.[7] However, a zoom on an individual contour would show that the force components are fairly stable once the tool is away from the boundary conditions. For example, Figure 8.21 shows the evolution of the three components of the tool force for the 90-degree pie mesh simulation during contours 15 to 17.

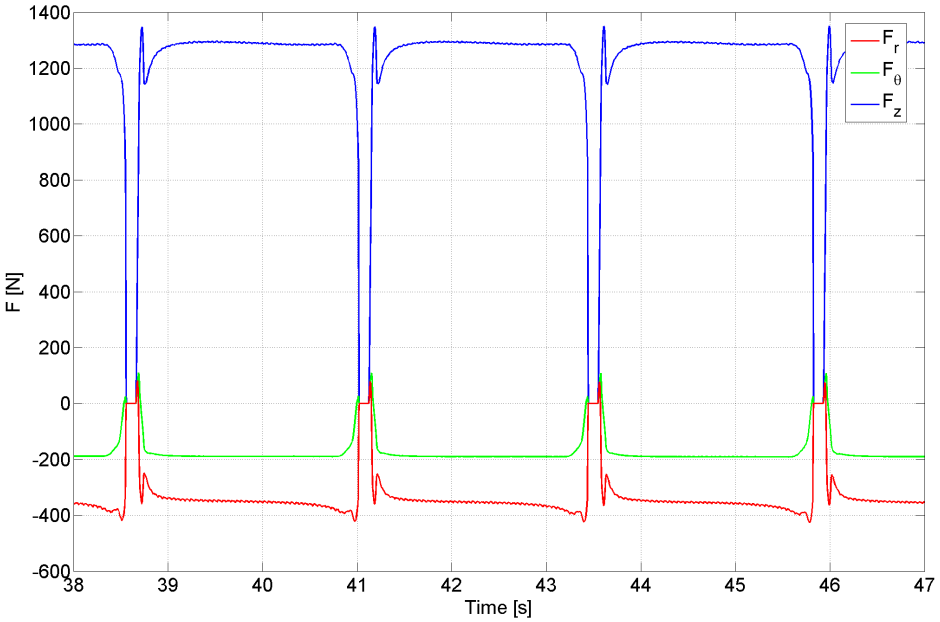


Figure 8.21: Evolution of the three components of the tool force for the 90-degree pie-mesh simulation during contours 15 to 17

In order to be able to compare the level of forces for the three simulations quantitatively, Figure 8.22 shows the average value of the three components of the force during each of the 150 contours (as mentioned before, the global tool path is composed of five stages of 30 contours each).[8] As can be seen in the figure, the three meshes predict almost exactly the same average force components. The maximum deviation occurs at contour 30 for the axial force F_z: for the 360-degree mesh, F_z=1333 N, for the 90-pie mesh, F_z=1285 N, and for the 45-degree mesh, F_z=1211 N. The maximum relative error in the axial force is therefore less than 10 %.

To conclude on the tool force prediction, using a pie mesh to simulate the forming of a part with rotational symmetry does not introduce an unacceptable

[7] For the full mesh, the tool is never completely removed (except at the end of each of the five stages). At the transition between successive contours, the force decreases, but not always to zero. For the pie mesh simulations, however, the tool path was modified and the tool draws back from the metal sheet at the transition between contours. Therefore, the tool force always drops to zero.

[8] Instead of taking the average of the force during the whole contour, only the central third of the contour was taken into account, in order to ensure that the force had reached a steady-state value.

level of error.

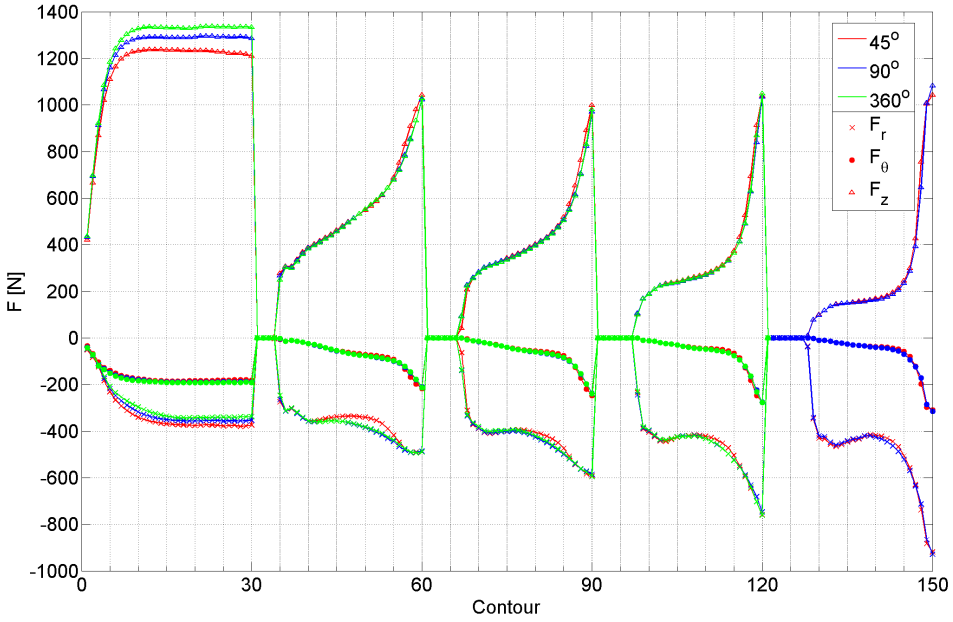


Figure 8.22: Comparison of the average value of the three components of the tool force for the three simulations

8.4 Computation Times

This section aims to analyze the computation times of the three simulations.

The simulations were all performed on the same machine using only one CPU.[9] The computation times are given in Table 8.2. Since the simulation with the full mesh did not reach the end, the time given in the table is an extrapolation based on the finished part (four stages out of five).

Mesh	Number of DOFs	Computation time (in hours)	Computation time (in days)
45-degree pie	13 300	101	4
90-degree pie	26 392	499	21
360-degree pie	104 262	7 354	306

Table 8.2: Computation times of the three different simulations

The computation time is usually proportional to the square of the number of DOFs because most of the time is spent on the resolution of the linear system of equations. Indeed, it can be seen that the computation time of the simulation with the full mesh is approximately 16 times larger (the mesh having four times as many DOFs) than the 90-degree pie simulation. Similarly, the computation time of the 90-degree pie simulation is approximately 5 times larger (the mesh having twice as many DOFs) than the 45-degree pie simulation.

[9]The machine used was the parallel computing server (SUN Cluster) available at the University of Liège, called NIC2.

The computation time is reduced by a factor 76 if the 45-degree pie mesh is used instead of a full mesh.

8.5 Conclusions

In conclusion, it can be said that the pie meshes are almost as accurate as the full 360-degree mesh in terms of shape, thickness, and force prediction when simulating the forming of a part with rotational symmetry boundary conditions.

This results in a significant benefit because it implies that the number of elements can be divided by a factor of 4 (for a 90-degree pie) or 8 (for a 45-degree pie) compared to a full mesh of the metal sheet, which can lead to a drastic reduction in the computation time.

Chapter 9

Analysis of the Deformation Mechanism during SPIF

The purpose of this chapter is to analyze the deformation mechanism during the SPIF process. This analysis will be carried out on the 70-degree wall-angle cone simulation.

9.1 Description of the 70-Degree Cone

9.1.1 Geometry and Material

Figure 9.1 shows a 3D view and a schematic view with the dimensions of the cone. It is a 70-degree wall-angle cone with a depth of 30 mm. In addition to that geometry, a cone with a depth of 50 mm was also used for certain results, but the experimental measurements were not available for this depth. The tool, shown on the right side of the figure, is a sphere with a diameter of 10 mm.

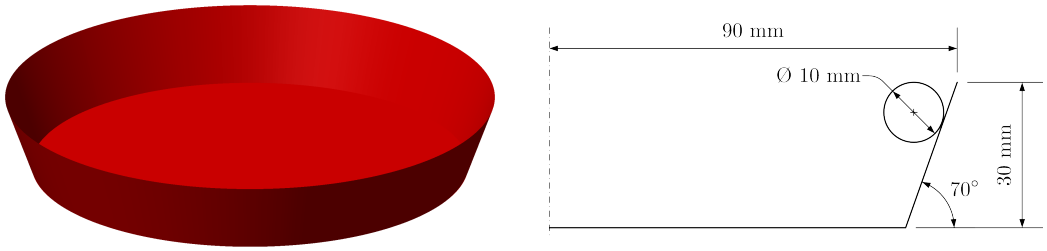


Figure 9.1: 3D view and dimensions of the 70-degree cone

The tool path of the cones with a depth of 30 and 50 mm is composed of 60 and 100 contours, respectively, with a stepdown of 0.5 mm between two successive contours, as shown in Figure 9.2.

The material used is an AA3003-O aluminum alloy, whose characteristics are presented in Appendix C. The initial thickness of the metal sheet is 1.2 mm.

The wall-angle of the cone was chosen to be 70 degrees because this value is close to the failure angle, which is about 71 degrees for this particular alloy and this thickness [DUF08b].

9.1.2 Finite Element Model

The forming simulation of a 70-degree cone part was performed with an implicit strategy using the LAGAMINE FE code. In order to reduce the computation time, only a 45-degree pie of the metal sheet was meshed with COQJ4 shell elements. As demonstrated in Chapter 8, this simplified mesh has a relatively limited influence on the accuracy of the results, provided that the results be analyzed far from the boundaries. The tool path, shown in Figure 9.2, was transformed accordingly: at the end of each contour, the tool is lifted, moved horizontally to the beginning of the next contour and then lowered.

Similarly to the study in Chapter 8, the element type used was the COQJ4 element and the mechanical law had four in-plane integration points and, for each of them, five integration points across the thickness. Of the latter, the first one is located close to the lower surface, the third is in the mid-plane of the element, and the fifth is close to the upper surface (closer to the tool).

The contact between the tool and the metal sheet is taken into account using contact elements. The penalty coefficient of these elements was chosen to be 1000 N/mm^3. Given the absence of experimental data, the friction coefficient was assumed to be zero.

Two different meshes were tested:

- a coarse mesh with 1556 nodes and 1492 elements (see Figure 9.3*a*);
- a fine mesh with 2242 nodes and 2164 elements (see Figure 9.3*b*).

Both meshes have 48 elements in the circumferential direction (except around the corner of the pie) but different numbers of elements in the radial direction.

Again, as in Chapter 8, *rotational* boundary conditions were imposed at both edges of the pie in order to take into account the missing material.

9.2 Results Concerning the Geometry

9.2.1 Shape of the Cone

This section focuses on the prediction of the shape of the cone. The thickness will be analyzed in the next section.

The simulation was run with both the coarse and fine meshes shown in Figure 9.3 using the tool path with 60 contours, which corresponds to a cone with a depth of 30 mm. The shape of both simulations is presented in Figure 9.4 in a cross section at 22.5 degrees from the x-direction, i.e., in the middle of the 45-degree pie. In the figure, the horizontal axis represents the radial distance from the center of the cone ($r = \sqrt{x^2 + y^2}$) and the vertical axis is the z-coordinate. Since the mesh is composed of shell elements, only the average surface (in the middle of the thickness) was available.

As to the experimental tests, both the top and bottom surfaces were measured with a laser scanning system. The output was a point cloud, from which a cross section of the full cone was extracted. Both surfaces of the cone are also plotted in Figure 9.4.

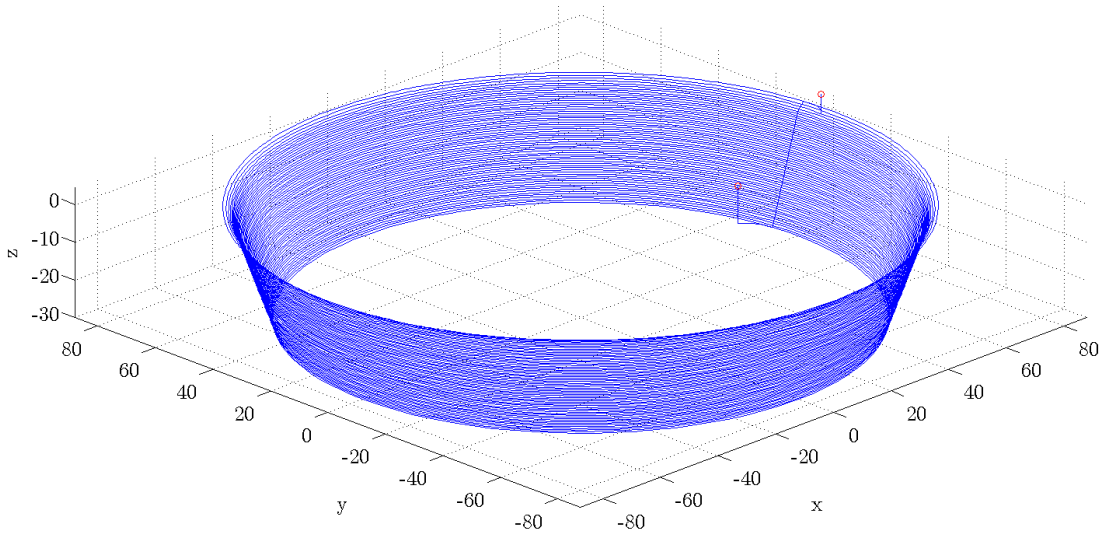
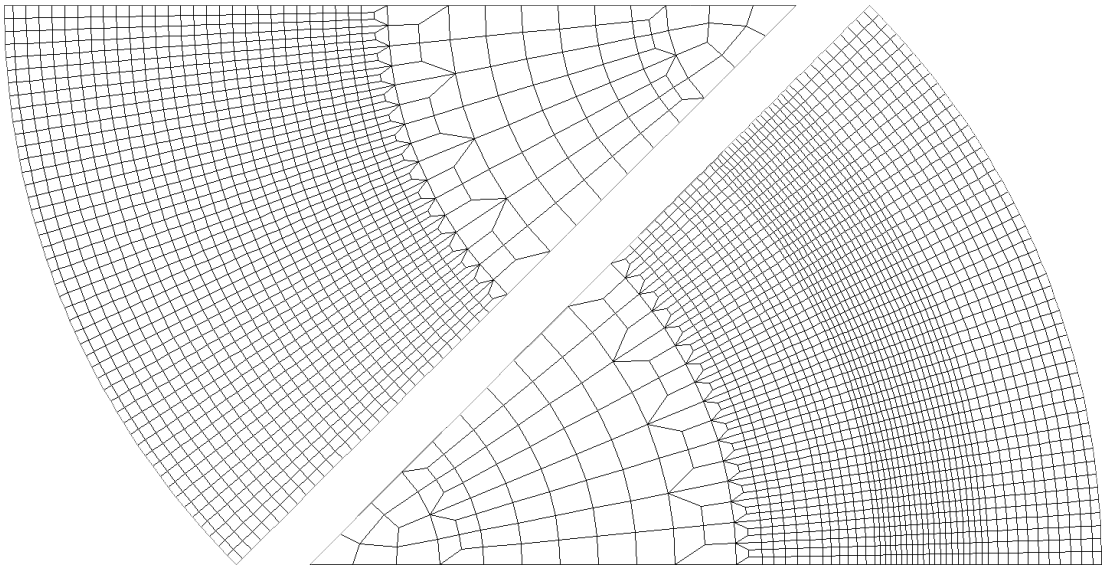


Figure 9.2: Tool path of the 70-degree cone with 60 contours



(*a*) Coarse mesh (*b*) Fine mesh

Figure 9.3: Meshes used for the simulation of the 70-degree cone

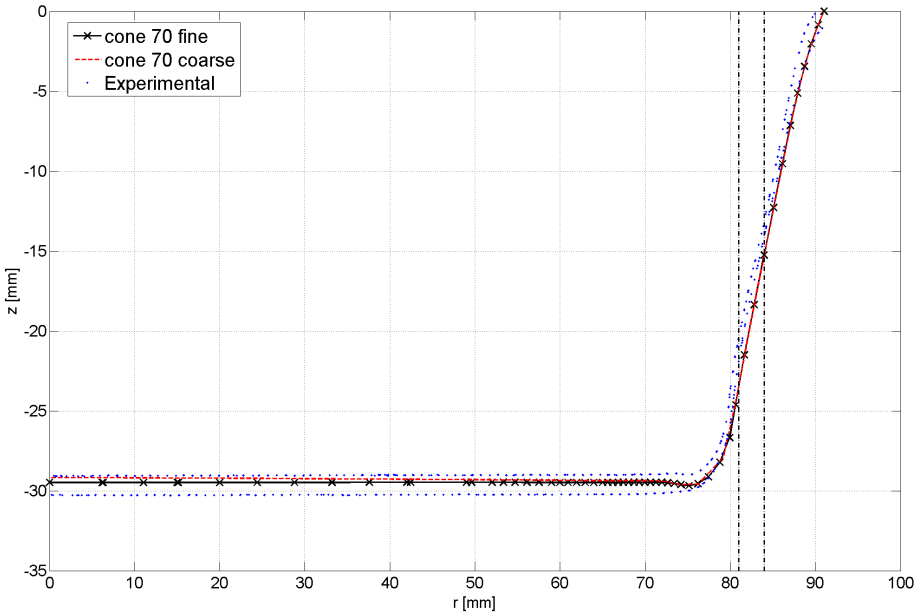


Figure 9.4: Shape of the 70-degree cone after 60 contours

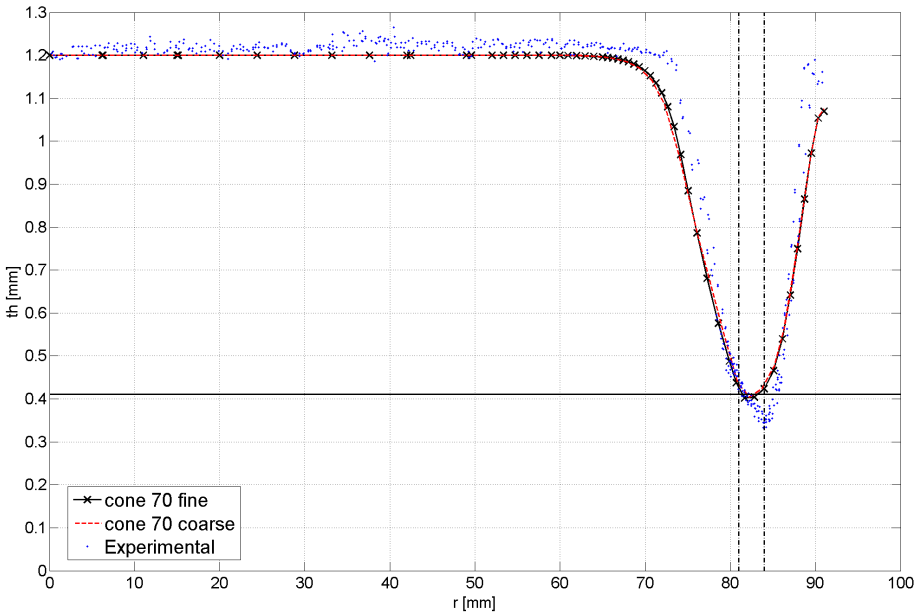


Figure 9.5: Thickness of the 70-degree cone after 60 contours

A relatively good correlation was found between the simulation results and the experimental measurements in the 70-degree wall. The error is limited to a fraction of millimeter, although it might be due to a centering error of the experimental point cloud.

At the bottom of the cone, the accuracy of the simulation is also quite precise. As explained in Section 8.2, this part is always the hardest to predict. The maximum deviation between the simulated and the measured shapes is located at the center of the cone and is smaller than 0.4 mm for the coarser mesh and about 0.1 mm for the finer mesh (less than 10 % of the initial thickness). It should be noted that with a coarser mesh made with three layers of brick elements, the bottom of the cone was more than two millimeters too deep compared to the experimental results [HE05b, Figure 1].

Both mesh densities predict the same geometry. A maximum deviation of 0.29 mm is observed at the center between the two meshes. The same conclusion can be drawn after 100 contours (i.e., a depth of 50 mm), as shown in Figure 9.6. In that case, the maximum deviation, still at the center of the cone, becomes 0.49 mm. As mentioned before, the experimental measurements were not available for this depth for comparison.

9.2.2 Thickness of the Cone

The thickness of the cone after 60 contours is presented in Figure 9.5 in the same cross section as the two figures of the previous section. The shell element COQJ4 used a constant volume hypothesis in order to compute the thickness. It can be observed that this hypothesis is not too far from the experimental measurements. A second computation was performed using a plane stress hypothesis but the results were comparatively much worse.

In Figure 9.5, the horizontal solid black line represents the theoretical thickness t_f given by the sine law (theoretical law introduced in Section 2.1.8)

$$t_f = t_0 \sin(90 - \alpha), \tag{9.1}$$

where t_0 is the initial thickness of the sheet and α is the wall angle. This formula predicts a thickness of 0.41 mm for a 70-degree wall angle. It can be seen that the simulated and measured thicknesses barely reach the theoretical value predicted by the sine law in the wall region. The region that follows the sine law approximately has been marked by two vertical dotted black lines in both Figures 9.4 and 9.5. This cone is probably not deep enough so that a steady state in the thickness was not reached yet. In order to check this, Figure 9.7 shows the thickness of the cone after 100 contours, i.e., at a depth of 50 mm. In this case, a plateau in the thickness is clearly seen. Again, the region following the sine law is marked by two vertical dotted black lines. This region is much wider here.

A zoom on Figures 9.5 and 9.7 would show that the finer mesh predicts a slight minimum in the thickness, whereas the coarser mesh does not. This effect is easier to see in Figures 9.8 and 9.9, where the thickness is plotted as a function of the depth of the cone after 60 and 100 contours, respectively. These two figures show that:

- the thickness of the metal sheet is close to its initial value of 1.2 mm at a depth of 0 mm, i.e., at the outer radius of the cone;

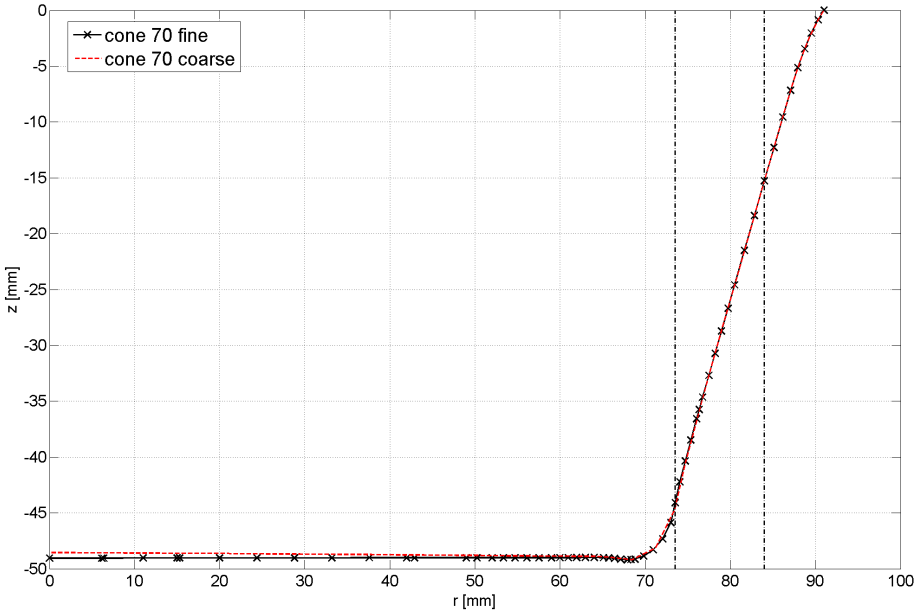


Figure 9.6: Shape of the 70-degree cone after 100 contours (no experimental results were available at that depth)

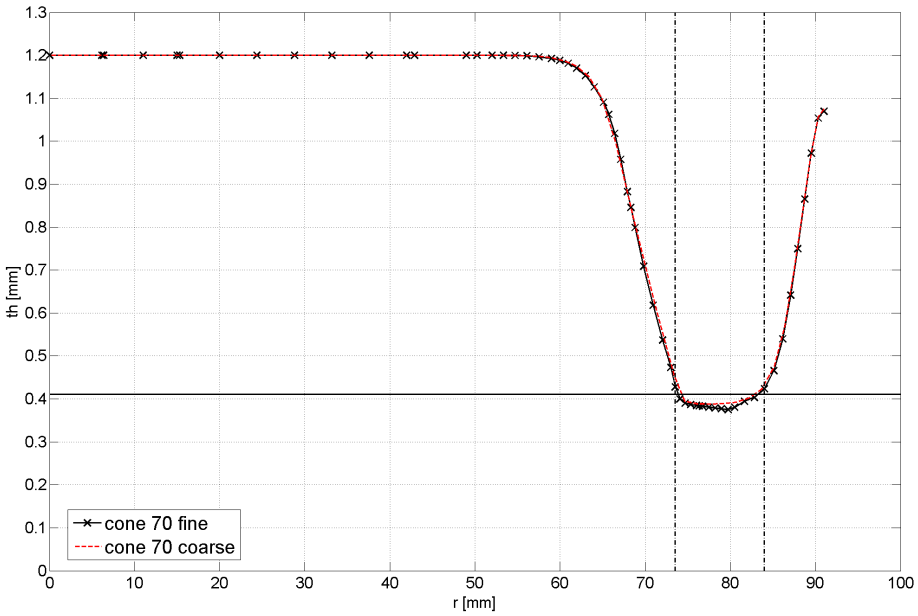


Figure 9.7: Thickness of the 70-degree cone after 100 contours (no experimental results were available at that depth)

- the thickness is also close to its initial value at a depth of 30 mm, i.e., at the bottom of the cone;
- when the depth increases from 0 to 30 mm, i.e., in the wall region, the thickness decreases rapidly until it reaches a steady-state value close to the theoretical sine-law value.

Moreover, Figure 9.8 shows that in the wall region, a slight minimum in the thickness is predicted by the simulations at a depth of approximately 21 mm whereas it appears at a depth of about 13 mm in the experimental measurements.

Since the prediction of the location of the minimum thickness was not accurate, it was suspected that friction might be one of the reasons. To verify this, the simulation with the finer mesh was run two more times with a non-zero friction coefficient Φ equal to 0.064 and 0.15. Figures 9.10 and 9.11 show the thickness as function of the depth of the cone at the end of contours 60 and 100.

First, in looking at the figures, it should be noted that the friction coefficient has an effect on the thickness prediction: the thickness is smaller at almost every depth when friction is introduced. After 60 contours, Figure 9.10 shows that the location of the minimum thickness seems to be slightly deeper than in the prediction of the simulations without friction. This effect is confirmed after 100 contours in Figure 9.11: both friction coefficients predict a minimum thickness at a depth of about 25 mm, whereas it was 21 mm without friction. As a results, this prediction is worse when friction is introduced. It should be noted, however, that the friction coefficient did not have a significant impact on the shape profile.

The shell element used in Lagamine, as for most shell elements, assumes that the through-thickness stress is equal to zero. This hypothesis, which is not true under the tool, is acceptable enough in most cases. The result is that, even though this element provides a good indication of the general trends of the thickness profile, it is not able to predict the location of the minimum thickness accurately. This would require the use of a more advanced shell element, such as a solid-shell element, which can use a full 3D material law instead of a 2D plane-stress law. Another solution would be to mesh the sheet metal with several layers of brick elements with a very fine mesh, but this would increase the computation time greatly, which is impractical.

9.3 Results Concerning the Tool Force

The norm of the tool force is plotted in Figure 9.12 for the simulations with both meshes and measured experimentally. This figure is relatively difficult to analyze for different reasons:

- the simulations show small parasitic oscillations due to the FE mesh and in particular to the discrete location of the integration points of the contact elements;
- the force drops to (almost) zero at the transition between successive contours;

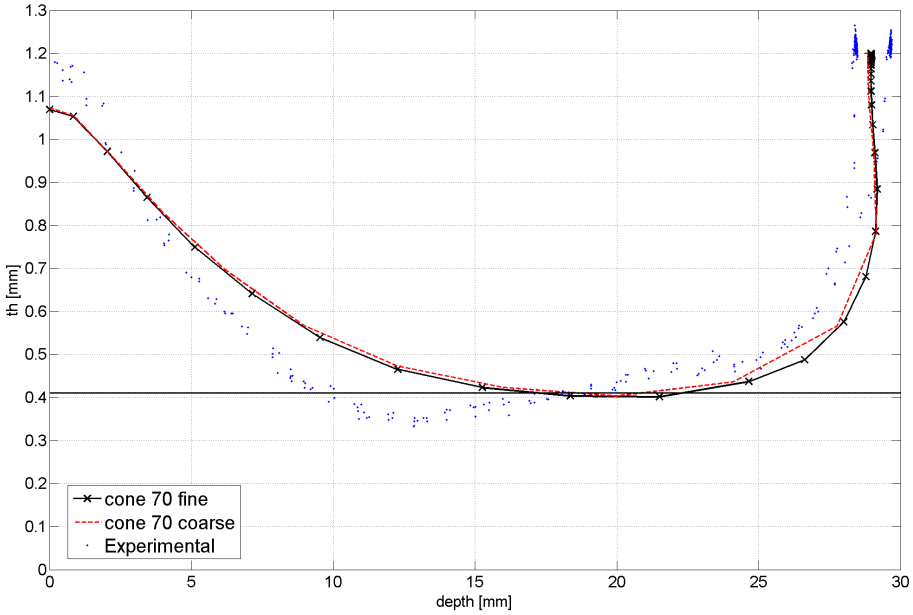


Figure 9.8: Thickness of the 70-degree cone after 60 contours as a function of depth

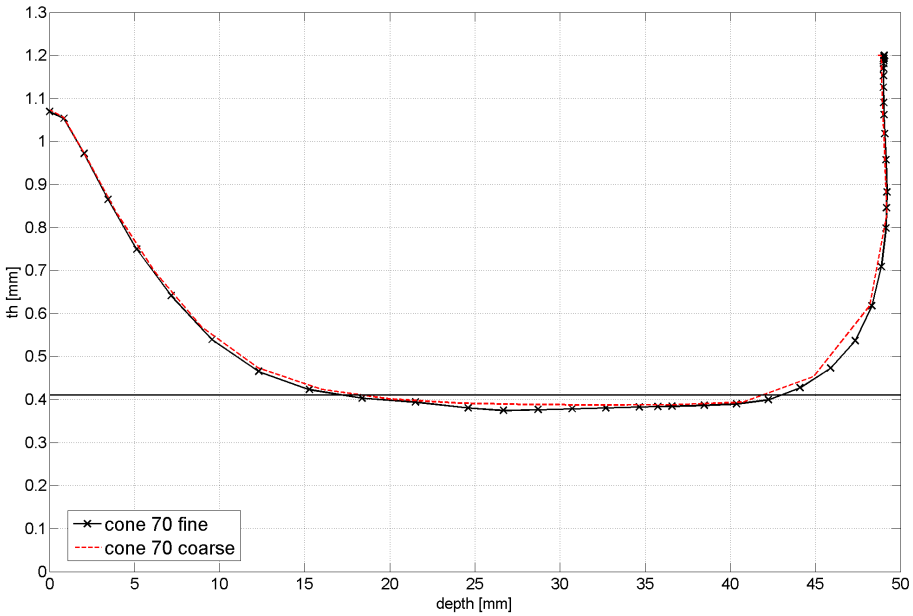


Figure 9.9: Thickness of the 70-degree cone after 100 contours as a function of depth

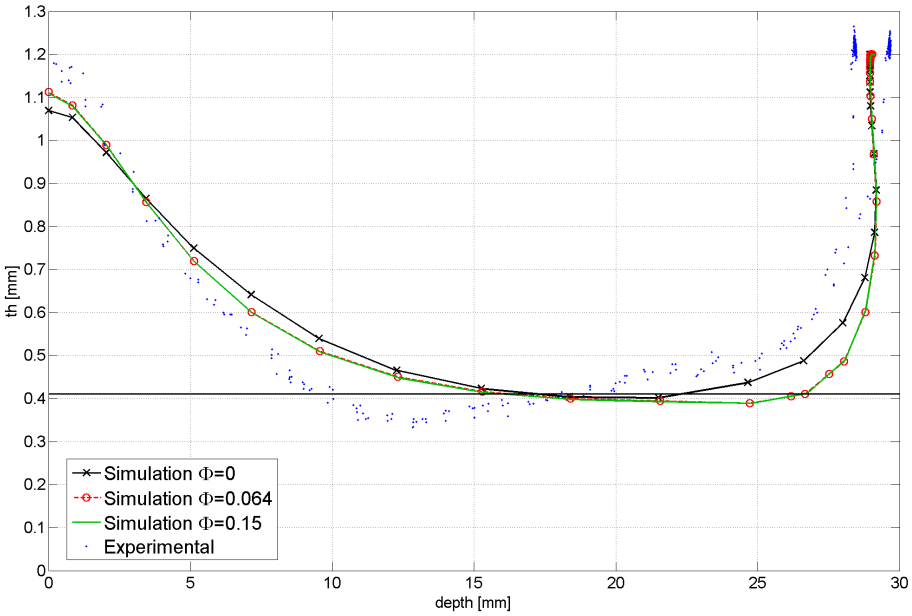


Figure 9.10: Influence of the friction coefficient on the thickness of the 70-degree cone after 60 contours as a function of depth

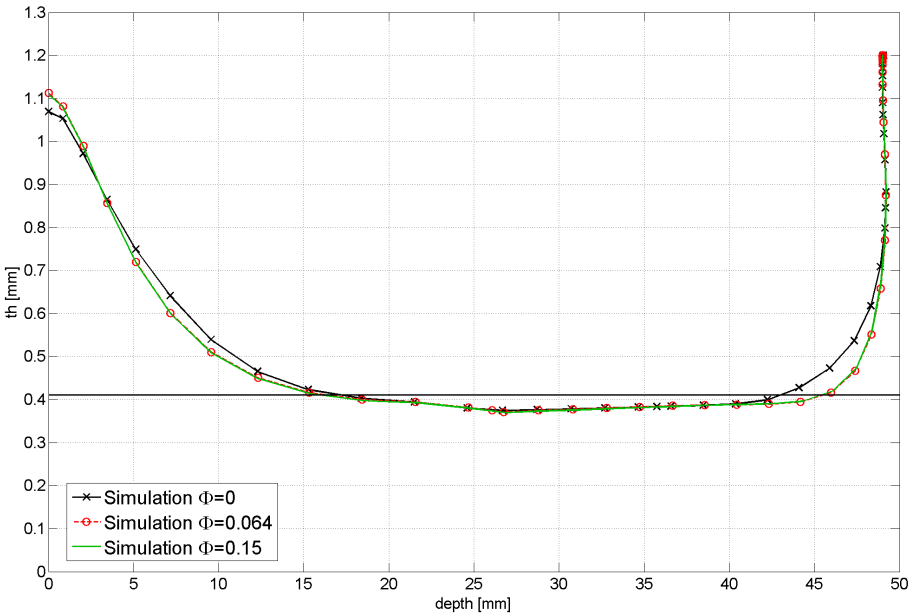


Figure 9.11: Influence of the friction coefficient on the thickness of the 70-degree cone after 100 contours as a function of depth

- the transitions between contours do not occur at the exact same time during the simulations and the experimental results;
- the simulation goes up to 100 contours, whereas the experimental test stops around contour 80.

For these reasons, Figure 9.13 represents the force during each contour by a point, whose value was obtained by averaging the norm of the force during that contour.[1] In this figure, it can be observed that both mesh densities predict a similar tool force on average, even though the coarser mesh introduces more oscillations than the finer one. On the other hand, the simulation results and the experimental measurements differ considerably.

Firstly, the general evolution of the curves are different. In the experimental result, the force rises until it reaches a peak. Then it drops to a minimum value before finally rising and stabilizing at its steady state value. In the simulations, the drop and the minimum are not clearly visible. The minimum of the force in the experimental result occurs during contour 33, at a depth of 16.5 mm. This is close to the experimental location of the minimum thickness, which is around a depth of 13 mm, as explained in Section 9.2.2.

Secondly, the levels of the forces are not at all comparable. The steady state force predicted by the simulations ($\approx$ 1000 N) is about twice as large as the measured value ($\approx$ 500 N). Several sources of error can explain this difference in force level. As was shown in an article by Flores et al. [FLO07], the tool movement triggers some cyclic effects. Therefore, the force can be reduced by 25 % if a mixed isotropic-kinematic hardening law is used instead of the simple isotropic hardening law. In the article, the authors performed the computation using three layers of BWD3D brick elements, but the conclusion should be similar when using shell elements.

With this in mind, the simulation was run again with a finer mesh and Ziegler's kinematic hardening law (the parameters are given in Table C.8 in Appendices). The result is also plotted in Figure 9.13. In that case, the steady state value of the norm of the force is around 780 N, which corresponds to a drop of 20 %.

In addition to the material law, Bouffioux et al. [BOU08b] performed a thorough investigation of all the other possible sources of error that could explain the difference between the simulations and the experimental measurements, namely:

- the clamping of the sheet is not perfect (triggering slight sliding which decreases the force);
- the fact that the sheet metal is not perfectly flat in its initial state;
- the thickness of the sheet is not exactly equal to the product specifications;
- the in-plane dimensions of the sheet could be slightly different;
- the rigidity of the machine on which the experimental test was performed is not infinite;
- the diameter, the initial position, and the rigidity of the forming tool might be slightly different than their expected values;

[1] Instead of taking the average of the force during the whole contour, only the central third of the contour was taken into account, in order to ensure that the force had reached a steady-state value.

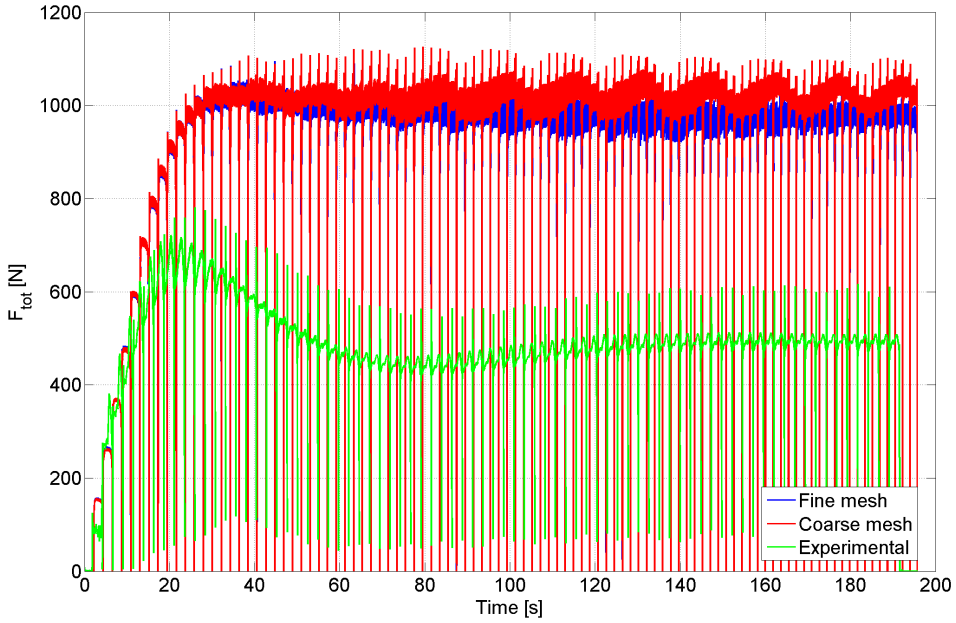


Figure 9.12: Norm of the tool force for the 70-degree cone

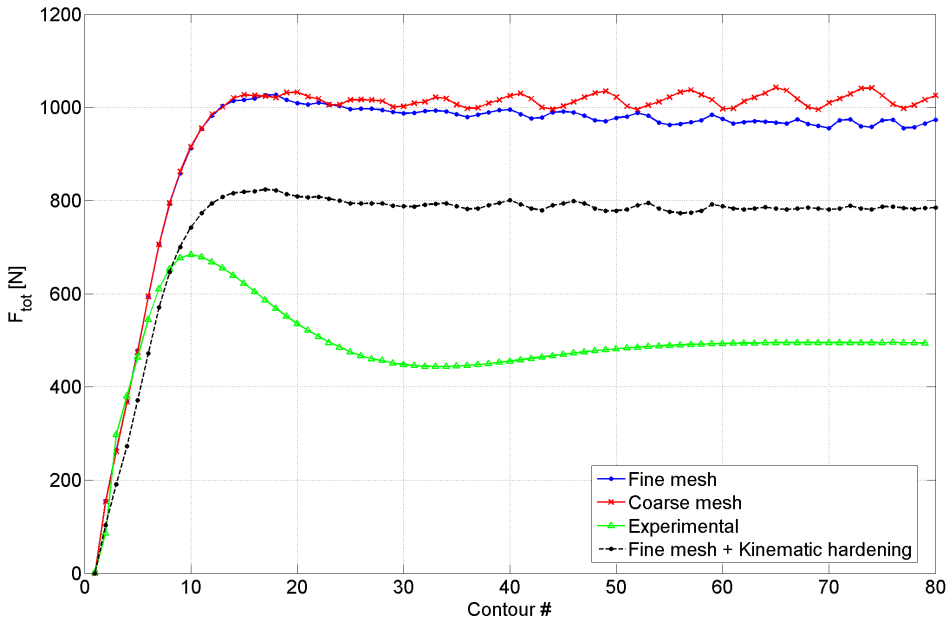


Figure 9.13: Average of the norm of the tool force for the 70-degree cone for the first 80 contours

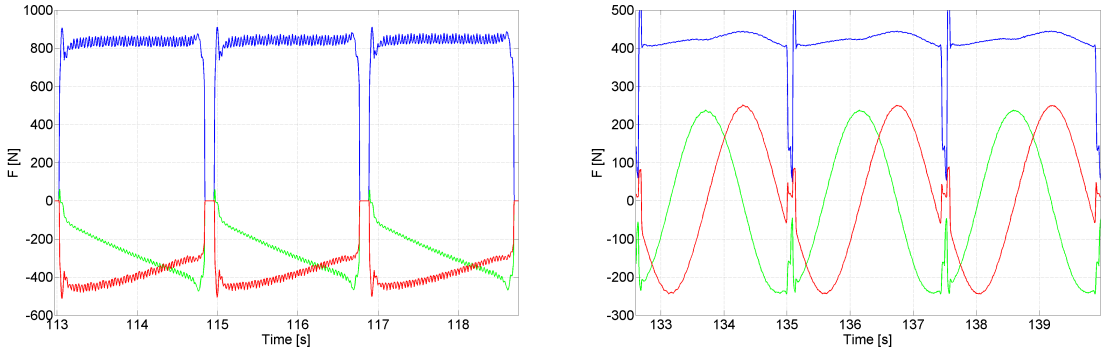
- the friction between the tool and the sheet is unknown.

The authors of the article concluded that, separately or together, none of these sources of error is able to explain the gap between the simulation and the experimental tool force.

Another source of error is the stiffness of the shell element[2]. and the fact that this element can only use a relatively simple plane-stress law. Moreover, the wall-angle of the cone is 70 degrees. As mentioned in the description of the part at the beginning of this chapter, this value is extremely close to the failure angle for this material (71 degrees). Therefore, it is very likely that some damage already occurs when forming this part, causing a drop in the force.

To conclude on the evolution of the norm of the tool force, it is likely necessary to use a solid-shell element with an advanced 3D mechanical law taking damage into account in order to reach a more accurate prediction.

The remaining part of this section focuses on the individual components of the tool force. Figure 9.14*a* shows the evolution of the three components (F_x,F_y,F_z) of the simulated tool force (with the finer mesh) during three successive contours (56 to 58), while Figure 9.14*b* shows the measured tool force during the same three contours. Since the tool is moving around the cone, the x- and y-components of the force should follow a sine function, but a comparison between the simulation and experiment is difficult to make because the simulation was performed with a 45-degree pie mesh whereas the measured force is for the whole cone.

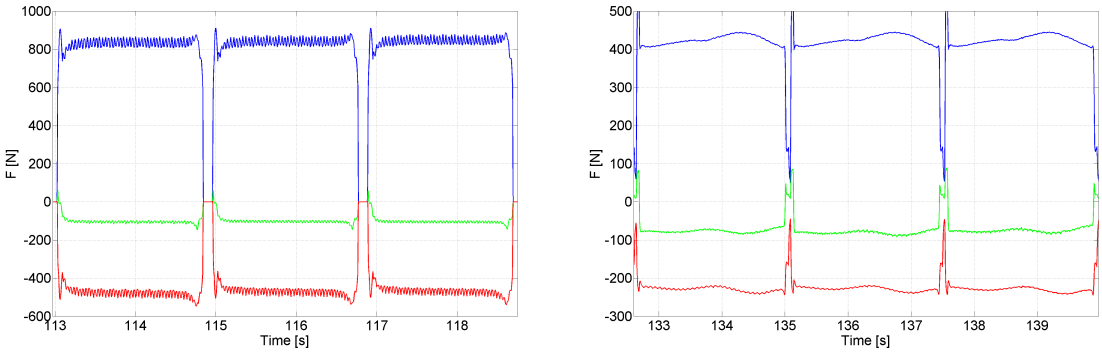


(*a*) Simulation (*b*) Experiment

Figure 9.14: Three components of the tool force during contours 56 to 58 in a cartesian coordinate system

In order to compare the relative intensity of the three components, it is easier to plot the force components in a polar coordinate system (r,θ,z) that follows the movement of the tool: the r- and θ-axis are in a horizontal plane, with the r-axis going from the center of the cone to the tool position and the θ-axis being perpendicular, and the z-axis is vertical, i.e., aligned with the tool's rotational axis. This is illustrated in Figures 9.15*a* and 9.15*b* for the simulation and the experimental measurement, respectively.

[2]An element is not supposed to have an internal stiffness. However, some energy terms are added to the element in order to reduce the various locking phenomena (e.g., shear and volumetric locking) and control the hourglass deformation modes, and adding these terms produces some internal stiffness



(*a*) Simulation (*b*) Experiment

Figure 9.15: Three components of the tool force during three successive contours in a polar coordinate system

Table 9.1 shows a comparison of the steady-state tool force components in a polar coordinate system. These values correspond to the average of force components during contours 60 to 80, contours during which the force does not change significantly, as seen in Figure 9.13. This table illustrates once more that the level of the tool force is not accurately predicted by the simulations.

	Exp.	Simulation				
Mesh	–	Coarse	Fine	Fine	Fine	Fine
Friction	–	Φ=0	Φ=0	Φ=5 %	Φ=6.4 %	Φ=15 %
F_r	-227 N	-489 N	-473 N	-473 N	-472 N	-469 N
F_θ	-78 N	-103 N	-108 N	-162 N	-177 N	-270 N
F_z	433 N	886 N	836 N	838 N	836 N	836 N

Table 9.1: Comparison of the measured and simulated steady-state tool force components in a polar coordinate system during the forming of the 70-degree cone

Table 9.2 shows the relative values of the components of the tool force. In this table, $F_{r\theta}$ represents the horizontal force component (perpendicular to the tool axis):

$$F_{r\theta} = \sqrt{F_r^2 + F_\theta^2}. \tag{9.2}$$

On the one hand, it can be seen that, without friction, the ratio between the radial component F_r and the vertical component F_z of the simulated force has the correct order of magnitude, compared to the experimental one. On the other hand, the ratio between the circumferential component F_θ and the vertical one is too small (12 % for the simulation, compared with 18 % for the experiment). This is probably due to the fact that the friction coefficient was chosen to be equal to zero.

The values of the simulated force in Table 9.1 can be used to estimate the experimental friction coefficient. Indeed, assuming that the normal to the contact surface is vertical and the tool movement is purely tangential, adding friction with

	Exp.	Simulation				
Mesh	–	Coarse	Fine	Fine	Fine	Fine
Friction	–	Φ=0	Φ=0	Φ=5 %	Φ=6.4 %	Φ=15 %
$F_{r\theta}/F_z$	55 %	56 %	58 %	60 %	60 %	65 %
F_r/F_z	-52 %	-55 %	-57 %	-56 %	-56 %	-56 %
F_θ/F_z	-18 %	-12 %	-13 %	-19 %	-21 %	-32 %

Table 9.2: Comparison of the measured and simulated relative values of the steady-state tool force components in a polar coordinate system during the forming of the 70-degree cone

a friction coefficient Φ would simply add $\Phi \cdot F_z$ to the tangential force component F_θ. Therefore, using the simulated force values obtained with the fine mesh, a friction coefficient of 5 % would increase the tangential force from -108 N to -150 N. In that case, the ratio between the circumferential component F_θ and the vertical one F_z would increase from -12 % to -18 %.

To check these rough estimates, the cone simulation was performed three more times with a value of the friction coefficient equal to 5 %, 6.4 %, and 15 %, respectively. The force results of these simulations are also included in both tables. It can be seen that a friction coefficient of around 5 % predicts a more accurate ratio between the circumferential and vertical components of the tool force.

To conclude, it can be said that, although the level of the tool force is not accurately predicted, the relative intensity of its components are quite accurate. In addition, the force prediction could be used to estimate the value of the experimental friction coefficient to about 5 %.

9.4 Detailed Analysis of the Material State

9.4.1 Introduction

For the remaining part of this chapter, the results of the 50-mm-deep cone (100 contours) will be analyzed in detail. With this cone, a steady state of deformation is reached during a significant portion of the wall region, which is not the case with the 30-mm-deep cone, as explained in Section 9.2.2.

In order to analyze the behavior of the material during the SPIF process, one element (number 1469) of the finer mesh was selected and its state variables were saved at each time increment during the whole process. As illustrated in Figure 9.16, this element is located approximately in the middle of the wall of the cone, more specifically at a radius of 75.55 mm and an angle of 22.5 degrees from the x-direction. Its initial size is approximately 1.236×0.688 mm^2, the longest dimension being in the circumferential direction, and the smallest one in the radial direction. As shown later in the text (see Figure 9.23, page 162), at the end of the simulation, this element is located, in the wall region at a depth of about 32 mm, which is approximately in the middle of the steady-state zone

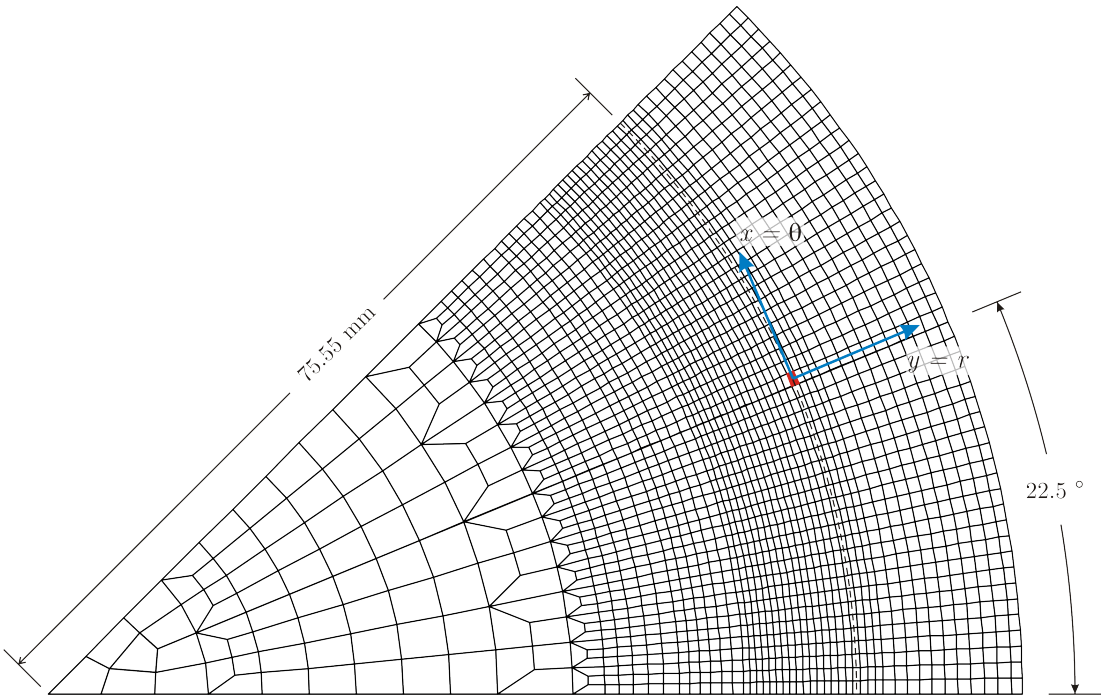


Figure 9.16: Element 1469 of the finer mesh of the 70-degree cone

of the wall.

The state variables of the element are expressed in a local reference frame attached to the element, as illustrated in Figure 9.16. The x- and y-axes of the element are in the mid-plane of the shell element. The x-axis is in the circumferential direction, i.e., in the direction of the tool movement during one contour. The y-axis is in the radial direction, but aligned with the wall of the cone. This definition is different than the radial axis defined in the previous sections. These two axes will be noted as the θ- and r-axes, respectively, in the remaining part of this chapter.

9.4.2 Thickness

The change in the thickness of element 1469 is shown in Figure 9.17. It follows the classic trend during the SPIF process: no change can be observed when the tool is far away, i.e., at the beginning and the end of the tool path, whereas a step-wise decrease occurs in the contours during which the tool passes close to the element.

A close-up view of the evolution of the thickness during one contour (number 57) is shown in Figure 9.18. The vertical dotted lines, and more precisely the spaces between them, represent the 48 elements in the circumferential direction. These lines can be used to know where the tool center is at a given time compared to element 1469, which is painted red in the figure. This figure illustrates that the tool's zone of influence on a given element is very limited. Recalling that the circumferential length of element 1469 is equal to about 1.24 mm, the tool influences the thickness of this element when it is closer than approximately two times its circumferential length, i.e., 2.5 mm or half the tool radius.

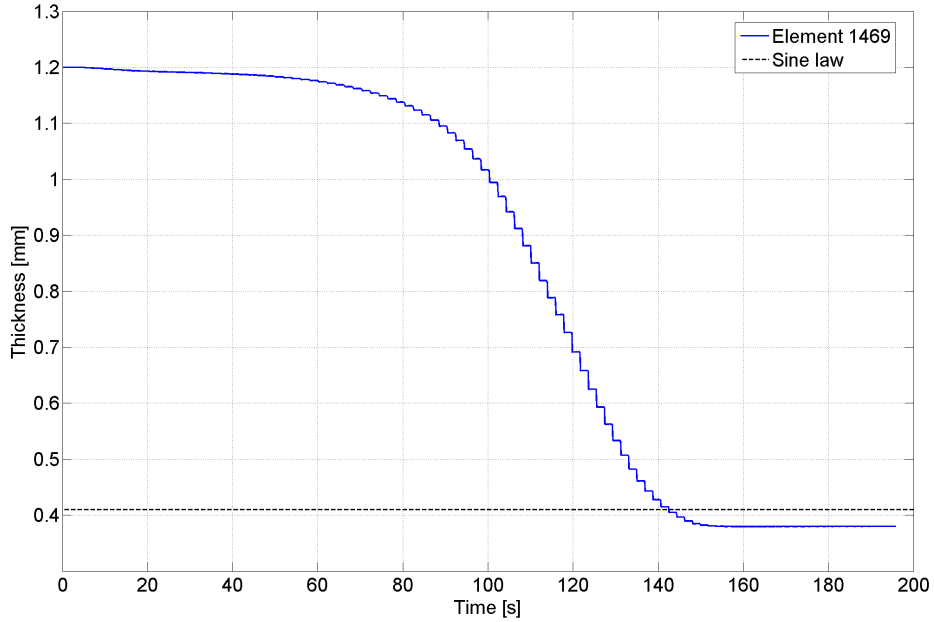


Figure 9.17: Thickness of element 1469 as a function of time

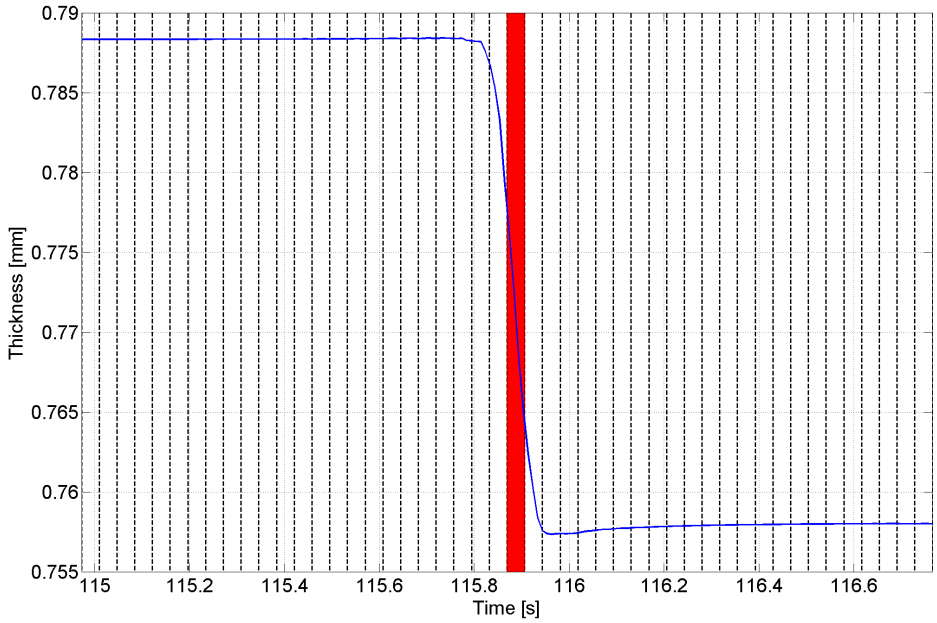


Figure 9.18: Thickness of element 1469 as a function of time during contour 57

9.4.3 Circumferential Strain

a. Evolution during the Whole Process

The evolution of the circumferential strain during the whole process is shown in Figure 9.19 for each of the five integration points across the thickness.

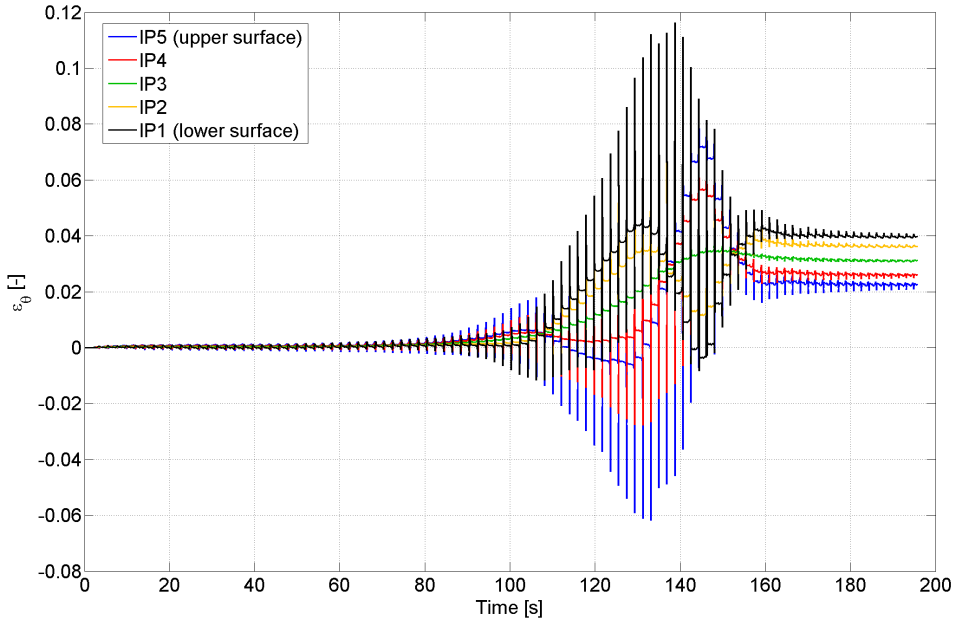


Figure 9.19: Circumferential strain of element 1469 as a function of time for the 70-degree cone

The curves in Figure 9.19 show large peaks, which are due to the bending and unbending occurring during each individual contour, as will be explained in the next section. In order to see the general trend of the curves, it is necessary to use an averaging method. The choice made here was to display only one point at the end of each contour, when the tool is lifted and does not touch the sheet metal anymore.[3] The result is shown in Figure 9.20, where the horizontal axis represents the contour number instead of the time. Seven specific contours have been identified by vertical lines and are denoted by the letters A to G.[4]

In this figure, the membrane deformation can be analyzed by looking at the strain of the third integration point because it is located in the mid-plane of the element and, since the element's curvature is assumed to be small, the mid-plane coincides with the neutral surface. Next, the bending deformation of the element can be analyzed by comparing the evolution of the strains of the first and fifth integration points (located on the lower and upper surfaces) to that of the third one (mid-plane).

Membrane Deformation The membrane deformation, which can be analyzed by looking at the strain of the third integration point, goes from zero to about

[3] As explained in Section 9.1.2, page 144, only a 45-degree pie of the metal sheet was meshed and the tool path was transformed accordingly. Therefore, the tool rises at the end of each contour, moves horizontally to a position where it can go down and start the next contour.

[4] In Figures 9.20, 9.21, and 9.23 to 9.26, the letters A to G correspond to contours 49, 52, 63, 68, 72, 76, and 80, respectively.

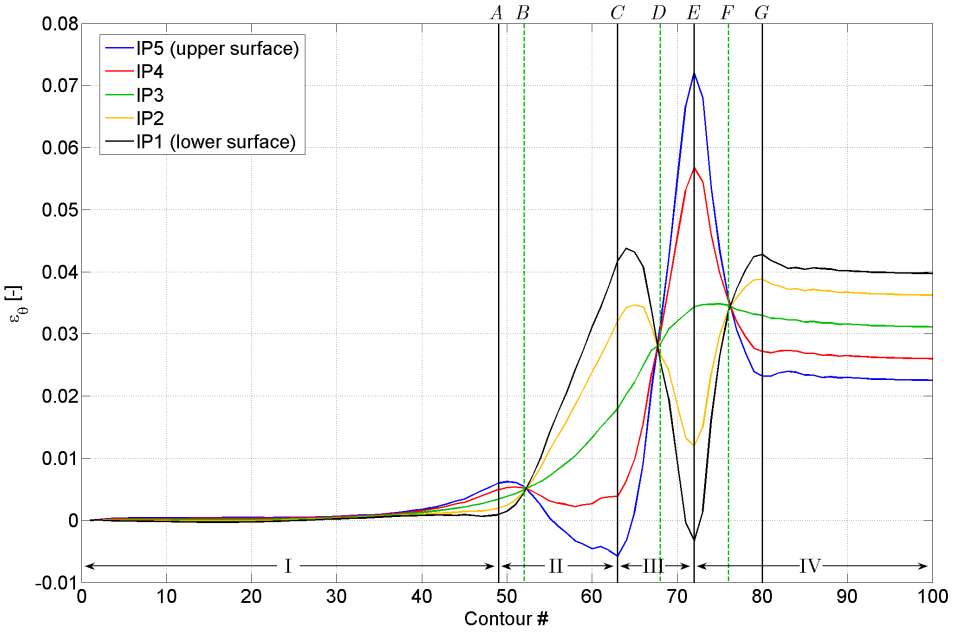


Figure 9.20: Circumferential strain of element 1469 for each contour of the 70-degree cone

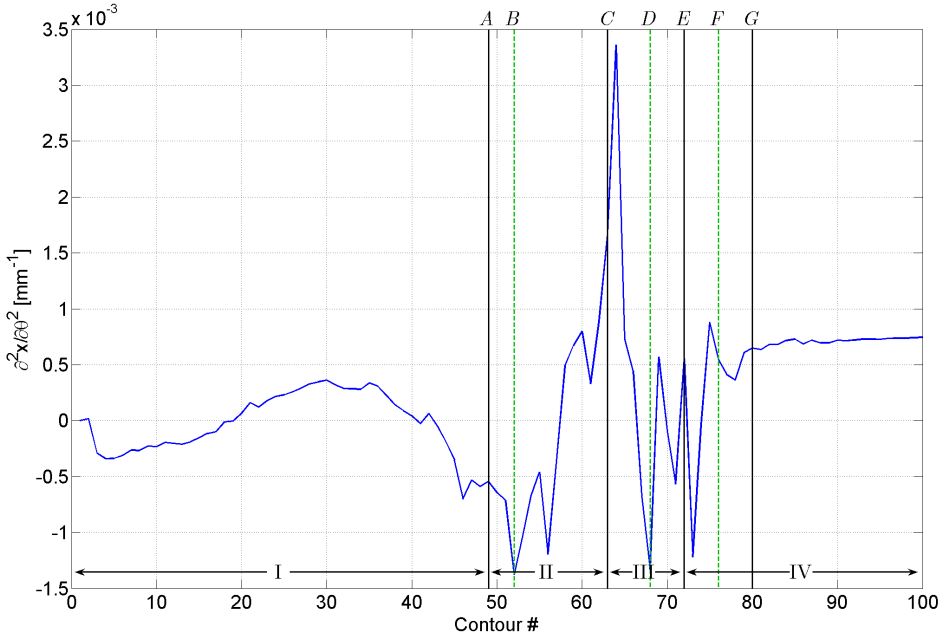


Figure 9.21: Curvature of element 1469 in the circumferential direction for each contour of the 70-degree cone

3.1 %. Since the angle between both edges of the pie mesh is fixed at 45 degrees with the boundary conditions, its 48 elements each form an approximately constant angle in the circumferential direction of

$$\theta = \frac{\pi/4}{48} = 0.016 \text{ rad} \tag{9.3}$$

during the whole process. Therefore, it can be assumed that the length of an element in the circumferential direction is approximately equal to its arc length, i.e., its radial position multiplied by its fixed angle length in radians. For this reason, the membrane circumferential strain can be estimated using the following formula:

$$\epsilon_\theta = \frac{1}{2}\left[\left(\frac{L_f}{L_0}\right)^2 - 1\right] \approx \frac{1}{2}\left[\left(\frac{R_f\ \theta}{R_0\ \theta}\right)^2 - 1\right] = \frac{1}{2}\left[\left(\frac{77.8}{75.5}\right)^2 - 1\right] = 3.1\ \%, \tag{9.4}$$

where L_0 and L_f are the initial and final lengths of the element in the circumferential direction, R_0 and R_f are the initial and final radial positions of the center of the element, and θ is the angle in the circumferential direction of the element. This estimate matches the value found in Figure 9.20.

Bending Deformation The curvature of the element in the circumferential direction is proportional to the norm of the second derivative of its position vector with respect to the circumferential θ-axis. Since this axis is always approximately in the same direction as the local x-axis of element 1469, this derivative can be computed by the following formula:

$$\frac{\partial^2 \underline{X}}{\partial \theta^2} \approx \frac{\partial^2 \underline{X}}{\partial x^2} = \frac{\partial}{\partial \xi}\left(\frac{\partial \underline{X}}{\partial x}\right)\frac{\partial \xi}{\partial x} = \frac{\partial^2 \underline{X}}{\partial \xi^2}\left(\frac{\partial \xi}{\partial x}\right)^2, \tag{9.5}$$

where ξ is one of the local coordinates in the reference element and $\partial^2 \underline{X}/\partial \xi^2$ is computed using the interpolation functions of the element and the nodal coordinates.[5] The evolution of the curvature during the successive contours is plotted in Figure 9.21. This curvature is relatively minor. Indeed, Figure 9.22*b* shows a cross section of element 1469 at the end of contour 63, during which the curvature is at its maximum. This section is taken along line A–A illustrated in Figure 9.22*a* and the vertical position z is plotted as a function of the circumferential coordinate θ.[6] In Figure 9.22*b*, the red dots are located along the curved surface of the element, whereas the dashed blue line is a straight line going from the first to the last red dot. The curvature is not visible. As a reminder, there are 48 elements for an arc of 45 degrees.

Even though the curvature is insignificant in the circumferential direction, several comments can be made about Figures 9.20 and 9.21. First of all, four

[5] More information about the interpolation functions and the local coordinates of the reference element can be found in Section 6.2.1, and in particular in Figure 6.2 page 86 and Equation 6.17 page 88

[6] The angular position is computed using the following formula:

$$\theta = \text{atan}\left(\frac{y}{x}\right) \cdot \frac{180}{\pi}.$$

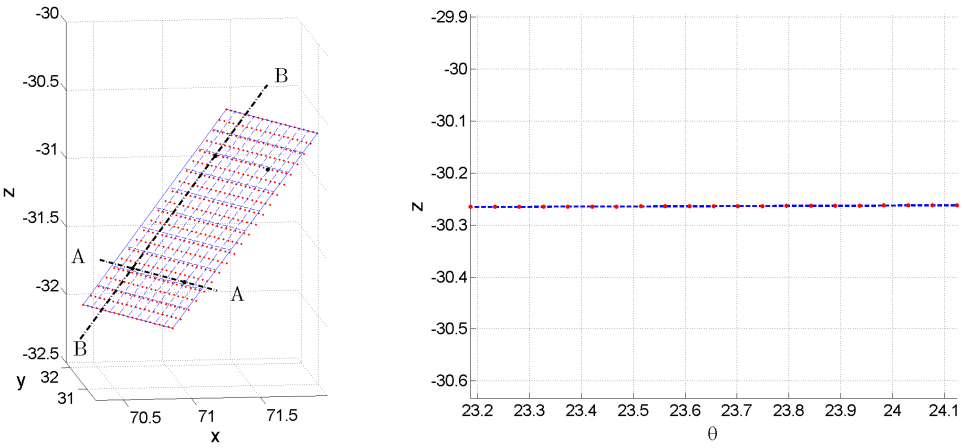


Figure 9.22: Cross section in element 1469 illustrating minor curvature in the circumferential direction

distinct stages (I to IV) can be identified. Figure 9.23 shows the shape of the cone in a cross section in the middle of the pie at the transition between these stages. (Element 1469 is highlighted in red.) The circumferential curvature is located in a plane perpendicular to the plane of the figure. Thus, this (small) curvature can not be visualized there. However, the purpose of this figure is not to show the curvature but to compare the tool's position during its successive contours with the element's position.

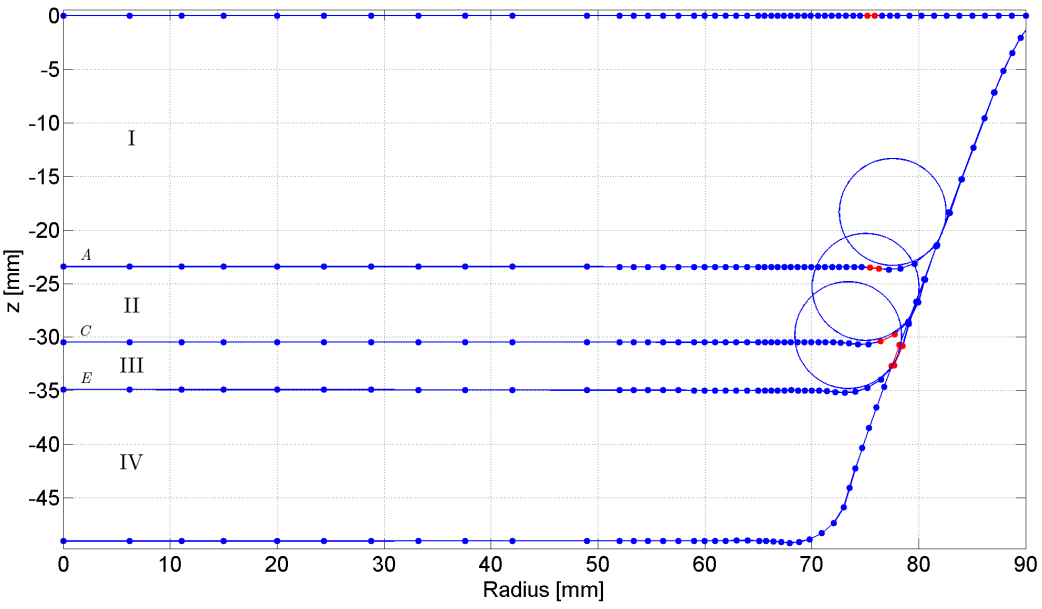


Figure 9.23: Position of the tool with respect to element 1469 in the four stages of the evolution of the circumferential strain for the 70-degree wall-angle cone

In the beginning of stage I, the tool passes, during its successive contours, at a large distance from the element which is still in the region of the bottom of the cone. The curvature of the element (in the circumferential direction) is then

close to zero or slightly negative. When the contours approach, the element is bent downwards (the lower surface being more under compression than the upper one) and the curvature in the circumferential direction decreases until it reaches a minimum at the end of the stage.

During stage II, the tool touches the element, whose curvature increases. Figure 9.24 shows the position of the tool during three different contours: A, B, and C. The curvature is minimal during contour A (beginning of the stage), passes through zero during contour B and reaches a maximum during contour C (end of the stage).

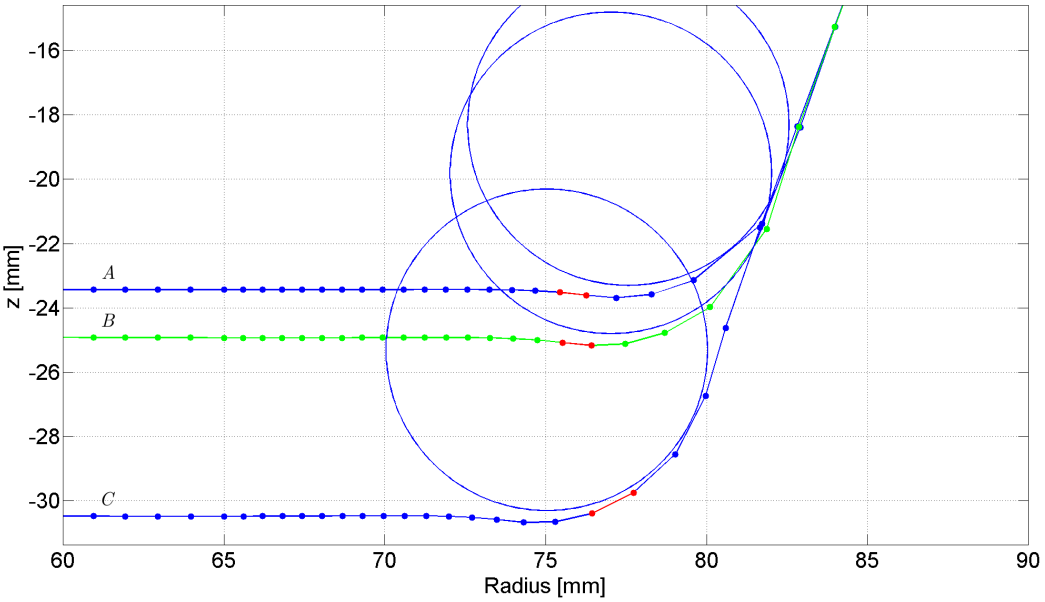


Figure 9.24: Position of the tool with respect to element 1469 during stage II of the evolution in the circumferential strain for the 70-degree wall-angle cone

During stage III, the element curvature starts to decrease again. Figure 9.25 shows again the position of the tool during three contours, C, D, and E, when the element's curvature is maximum, zero and minimum, respectively.

Finally, in stage IV, the element's curvature increases up to contour G and then reaches a steady state value, which corresponds to the curvature of the cone wall. The radius of the curvature is equal to the inverse of the curvature. This gives a radius of more than 1.3 m. This value is more than ten times larger than the radius of the cone, which is 90 mm. The reason for this error is probably the fact that the curvature is relatively minor (order of magnitude of 10^{-3}) and computed by taking the difference between the small rotations in the element's nodes. A slight error in the rotations produces a larger error in their difference, hence in the curvature. Figure 9.26 shows the position of the tool during contours E, F, and G, when the element curvature is minimum, zero and maximum, respectively.

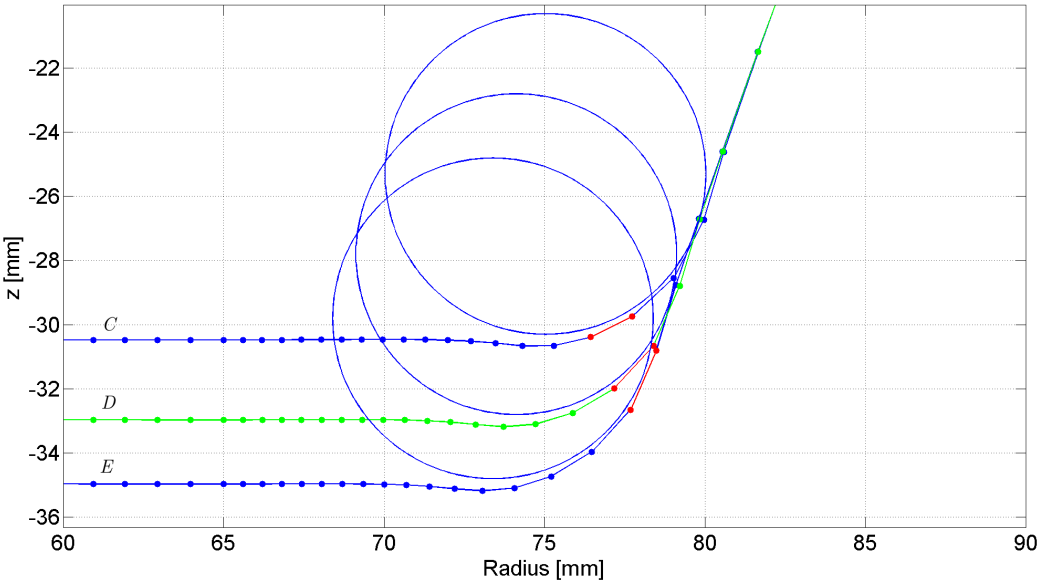


Figure 9.25: Position of the tool with respect to element 1469 during stage III of the evolution in the circumferential strain for the 70-degree wall-angle cone

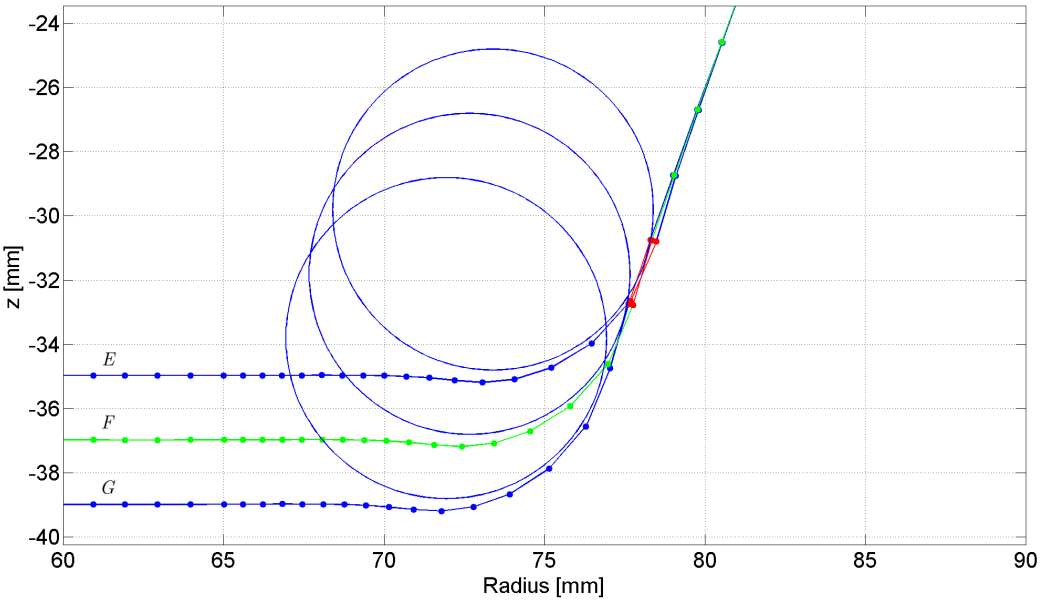


Figure 9.26: Position of the tool with respect to element 1469 during stage IV of the evolution in the circumferential strain for the 70-degree wall-angle cone

b. Evolution during One Contour

After analyzing the global behavior of the material during the whole process, this section focuses on an analysis of the evolution of circumferential strain during one contour.

Figures 9.27 and 9.28, respectively, show the evolution of the circumferential strain and the element's curvature in the circumferential direction during contour C. Just as the comment made for Figure 9.18, the vertical dotted black lines represent the 48 elements in the circumferential direction and the red zone, element 1469 itself.

Bending Deformation Both Figures 9.27 and 9.28 clearly show the bending and unbending effect that the material undergoes when the tool approaches, passes over the element, and moves away. This effect is illustrated in Figure 9.29. First, just before the tool touches the element, the element is bent downwards (with the lower surface under compression and negative curvature). Then, when the tool is above the element, it is bent upwards toward the tool (with the lower surface under tension and positive curvature). Finally, when the tool moves away, the element is once again bent (slightly) downwards before coming back to a nearly flat state.

Membrane Deformation During a given contour, the membrane strain does not vary greatly, as can be seen by looking at the strain of the third integration point in Figure 9.27. However, a close look at the curve shows that there are two small oscillations. As a reminder, the contact element associated with the COQJ4 shell element has four (two times two) in-plane integration points. Since the tool is moving in the circumferential direction, these oscillations are due to the fact that tool passes twice right above an integration point.

9.4.4 Radial Strain

a. Evolution during the Whole Process

The evolution of the radial strain of element 1469 is shown in Figure 9.30. In order to smooth out the curves, only one point per contour was printed in Figure 9.32, where the horizontal axis represents the contour number instead of the time.

In addition to this, the curvature of element 1469 in the radial direction (aligned with the wall of the cone) was also computed. Since the local y-axis of the element is always approximately in this radial direction, the curvature was computed using the following formula:

$$\frac{\partial^2 \underline{X}}{\partial r^2} \approx \frac{\partial^2 \underline{X}}{\partial y^2} = \frac{\partial^2 \underline{X}}{\partial \eta^2} \left(\frac{\partial \eta}{\partial y} \right)^2 , \tag{9.6}$$

where η is the local coordinate in the reference element corresponding to the y-axis and $\partial^2 \underline{X} / \partial \eta^2$ is computed using the interpolation functions of the element. The evolution of this curvature is plotted in Figure 9.33.

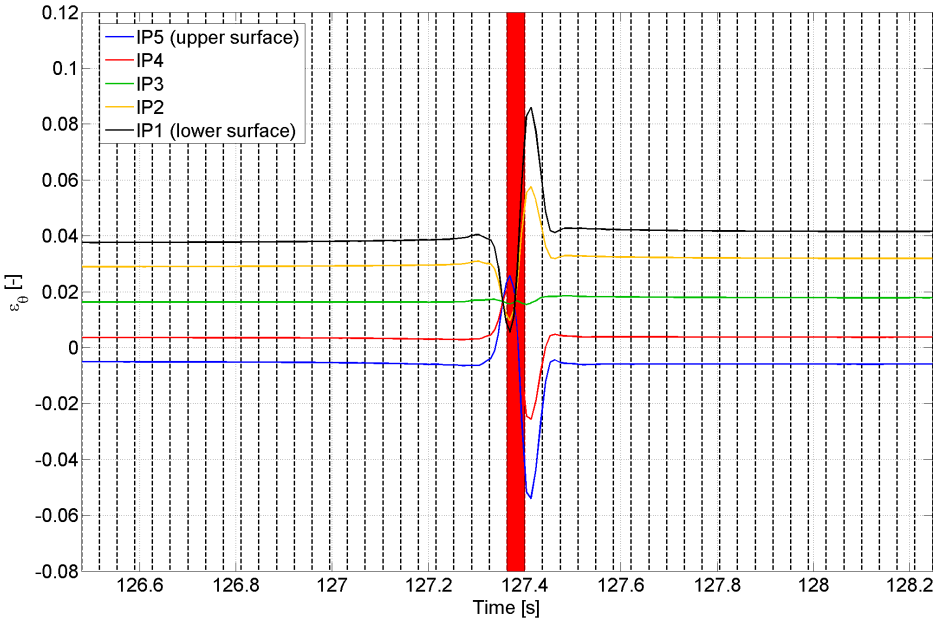


Figure 9.27: Circumferential strain of element 1469 during contour C of the forming of the 70-degree cone

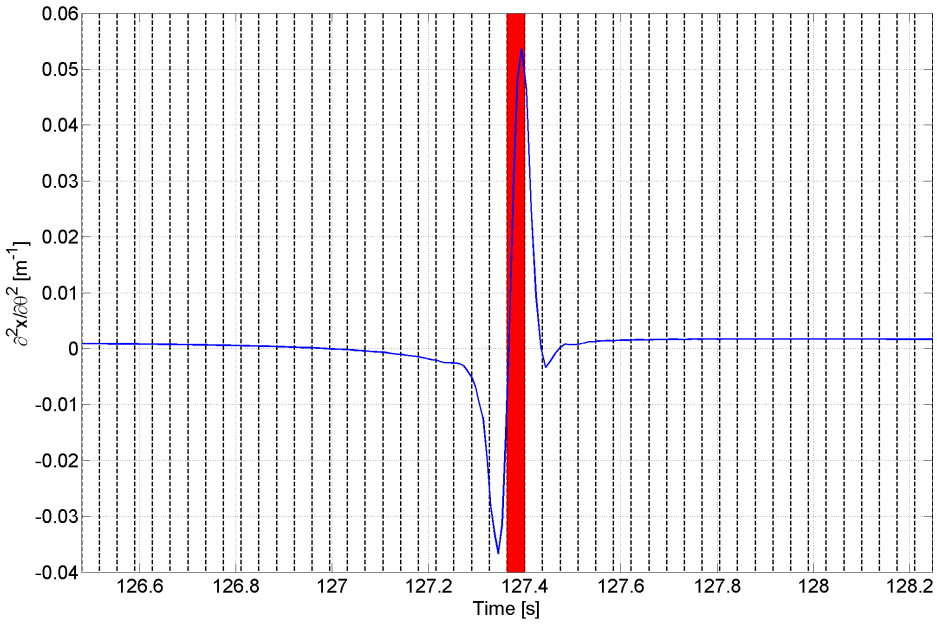


Figure 9.28: Curvature of element 1469 in the circumferential direction during contour C of the forming of the 70-degree cone

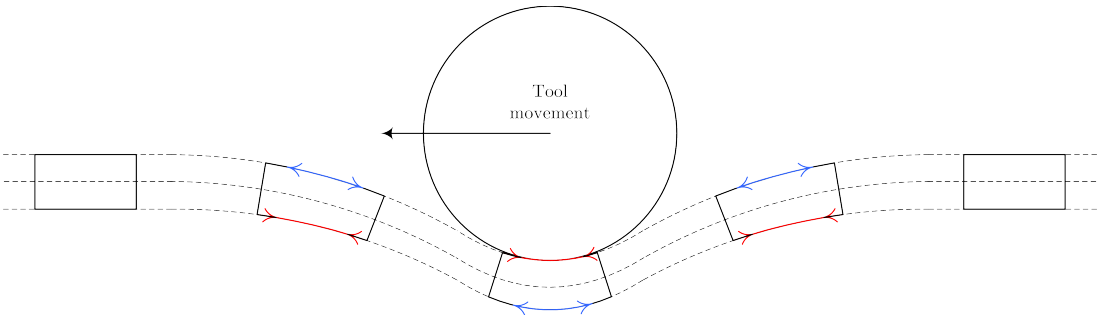


Figure 9.29: Bending and unbending effect during one contour of the 70-degree cone; cross section along the circumferential direction

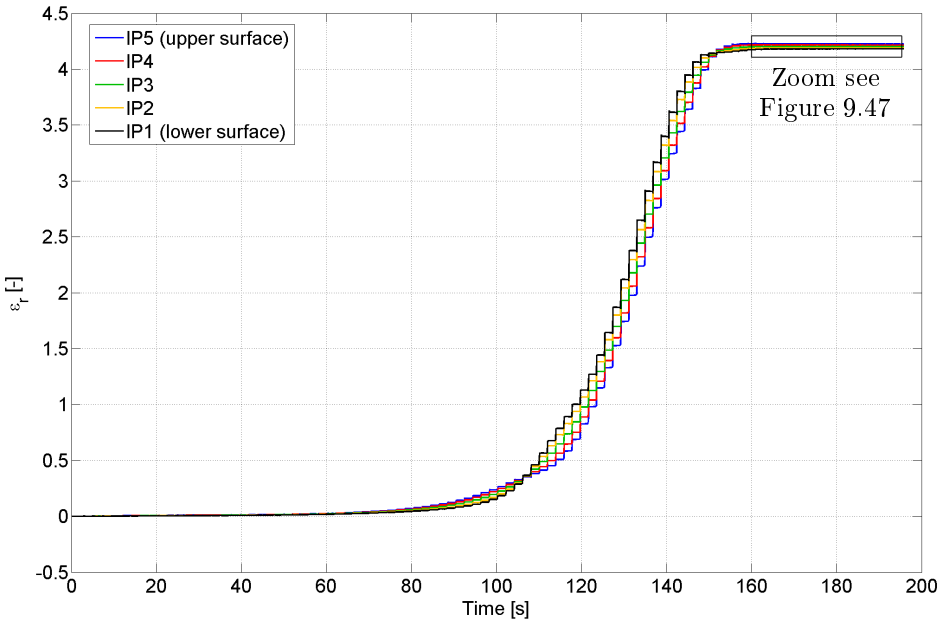


Figure 9.30: Radial strain of element 1469 as a function of time for the 70-degree wall-angle cone

Bending Deformation The four stages identified during the evolution in the circumferential strain have also been highlighted in Figures 9.32 and 9.33. The flexural strain is described schematically in Figure 9.31. As can be seen, in stage I, the element's curvature is zero or slightly negative. The element is bent downwards (lower surface in compression).

In stages II and III, the element's curvature is positive and the element is bent toward the tool. The maximum curvature occurs during contour D and is approximately equal to 0.2 mm^{-1}. The radius of curvature is equal to the inverse of the curvature. This gives a radius of 5 mm, which is the radius of the tool, as expected.

In stage IV, its curvature is once again negative. It decreases first until it reaches a minimum, then increases until the element is almost flat and aligned with the wall.

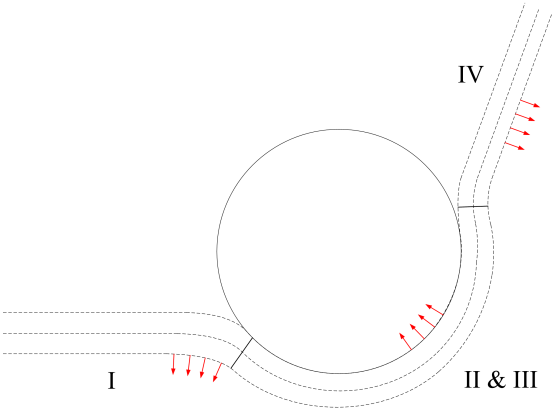


Figure 9.31: Schematic description of the bending and unbending of the material around the tool in the radial direction; cross section in the middle of the pie

Membrane Deformation The strain of the third integration point, located in the middle of the thickness, starts from zero and goes up to 420 %. The sine law predicts a radial strain of 377 % using Green's tensor, as explained in Figure 9.34. This theoretical value assumes that no change takes place in the circumferential direction (plane-strain state) and that the element is projected vertically onto the wall. Although these hypotheses are not completely true, there is only about a 10 % difference between this theoretical value and the simulated one. The global trend of the curve can then be predicted in that way.

The zone of the tool's influence on the radial strain is not very large. Indeed, if its influence is defined as the contours during which the strain is within 2 % of its extreme values, i.e., between 8 and 412 %, then its influence is limited to contours 42 to 75. The position of the tool during these contours is plotted in Figure 9.35. As expected, these correspond to the contours during which the tool is touching or is very close to the element.

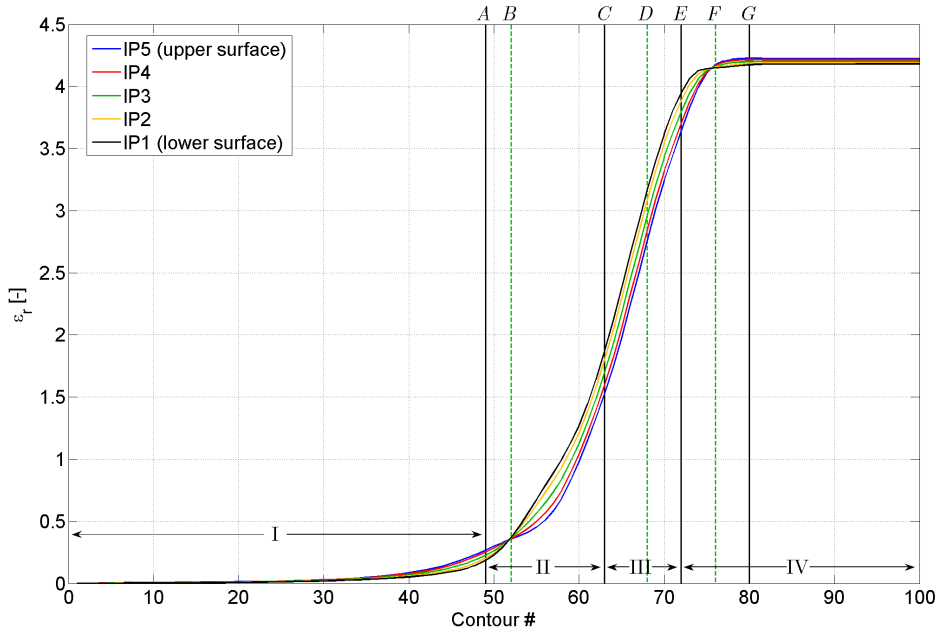


Figure 9.32: Radial strain of element 1469 for each contour of the 70-degree wall-angle cone

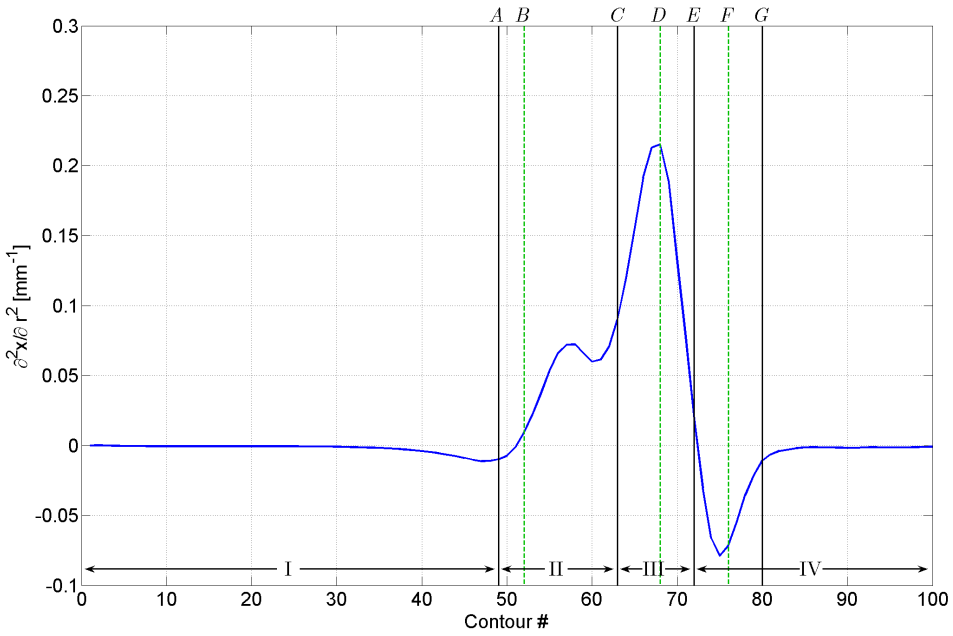
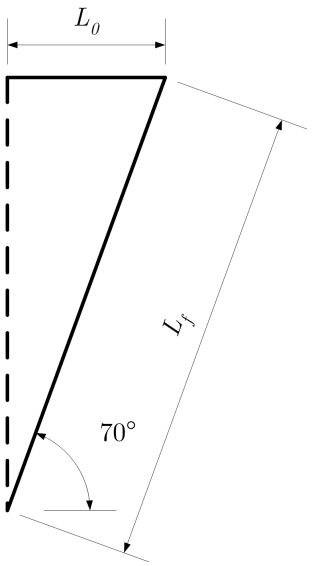


Figure 9.33: Curvature of element 1469 in the radial direction for each contour of the 70-degree wall-angle cone



$$\lambda = \frac{L_f}{L_0} = \frac{1}{\sin(20°)}$$

$$\epsilon_r = \frac{1}{2}\left(\lambda^2 - 1\right) = 3.77$$

Figure 9.34: Prediction of the radial strain using the sine law

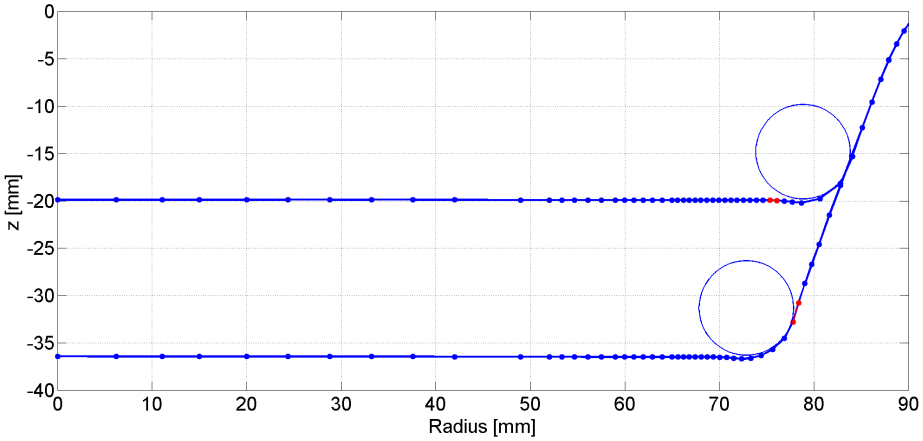


Figure 9.35: Position of the tool with respect to element 1469 in contours 42 and 75, which define the limits of the influence of the tool on the radial strain

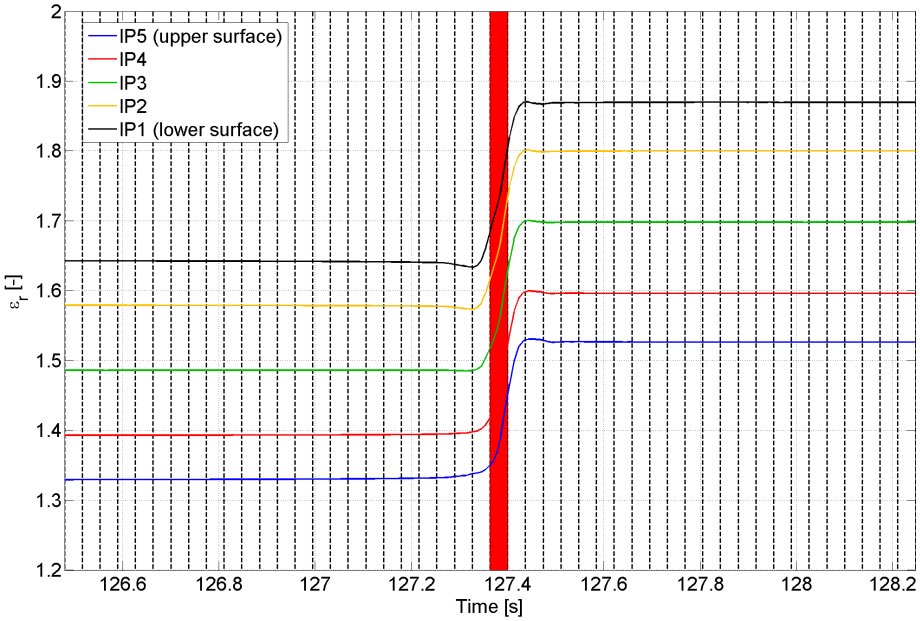


Figure 9.36: Radial strain of element 1469 during contour C of the forming of the 70-degree cone

b. Evolution during One Contour

The evolution of the radial strain during contour C is shown in Figure 9.36. It can be observed that the strains of all five integration points increase by approximately the same amount when the tool passes over the element. The zone of the tool's influence is very limited around the center of the element: it is approximately twice the circumferential length of the element, i.e., 2.5 mm, as was the case for the thickness evolution.

9.4.5 In-Plane Shear Strain

The evolution of the in-plane shear strain during the whole process is shown in Figure 9.37 for each of the five integration points across the thickness. Once again, in order to smooth out the curves, only one point per contour was printed in Figure 9.38, where the horizontal axis represents the contour number instead of the time. The in-plane shear strain is related to the in-plane shear angle of the element. For a general quadrilateral, this angle can be described as half the deviation of the angle between the two medians from a 90-degree angle.[7]

Figure 9.39 shows the evolution of the shear angle during the successive contours for three values of the friction coefficient. Until this point, all the graphs of this section were created with the results of the simulation without friction. The analysis of the evolution of the shear angle without friction shows that the shear angle decreases until contour 57. After that, it increases and stabilizes to a constant value in the last twenty contours. The reason for this is that, from contours 1 to 57, the radii of the tool contours are larger than the radius of the

[7]More information about the in-plane shear angle can be found in Appendix B, which describes the computation of the local axis of the COQJ4 shell element in detail. In particular, the shear angle is equal to angle α in Figure B.4 page 201.

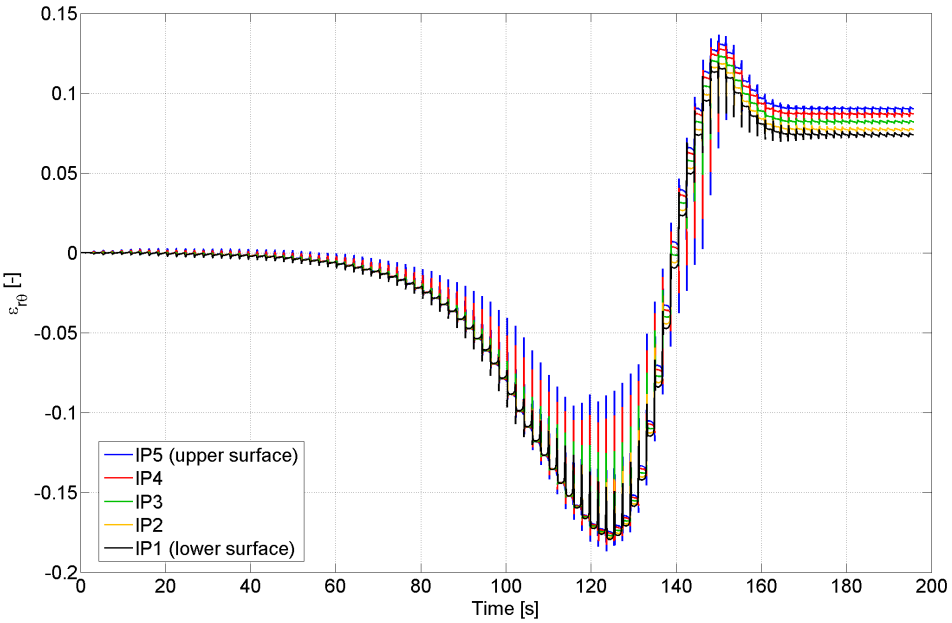


Figure 9.37: In-plane shear strain of element 1469 as a function of time for the 70-degree cone

center of element 1469, whereas from contour 58 until the end, the different radii of the contours are smaller than the radius of the center of element 1469. This is illustrated in Figure 9.40, which shows the position of the tool during contour 57, in a cross section in the middle of the pie.

When the radius of a contour is larger than the radius of the center of element 1469, the tool always shears the element in the same direction, which decreases the shear angle.[8] On the other hand, when the radius of a contour is smaller than the radius of the center of element 1469, the tool shears the element in the other direction, which increases the shear angle. A simple proof of this phenomenon is that increasing the friction between the tool and the sheet metal increases the variations in the shear angle, as illustrated in Figure 9.39.

The last phenomenon related to shear is the so-called twist effect. This effect is due to the fact that the successive contours of the tool are always performed with the tool moving in the same direction. Since the outside radius of the cone is clamped (represented by fixed nodes), the twist effect results in a global twisting of the cone in the region of the wall, as illustrated in Figure 9.41. This figure first shows a top-view of the initial mesh, along with the initial positions, in blue, of the nodes located in three vertical planes at 0, 22.5, and 45 degrees from the horizontal x-direction. In addition, the positions of the same nodes are superimposed on the figure, in green, at the end of the simulation, i.e., after 100 contours. Finally, the nodes belonging to element 1469 are colored in red, in both their initial and final positions. A zoom of the region around element 1469 is also included.

[8] Conventionally, a positive shear angle α means that the medians of the element $\underline{R}_{1P}$ and $\underline{R}_{2P}$ form an angle less than 90 degrees, using the notation defined in Figure B.4 page 201. Similarly, a negative shear angle means that the medians form an angle larger than 90 degrees.

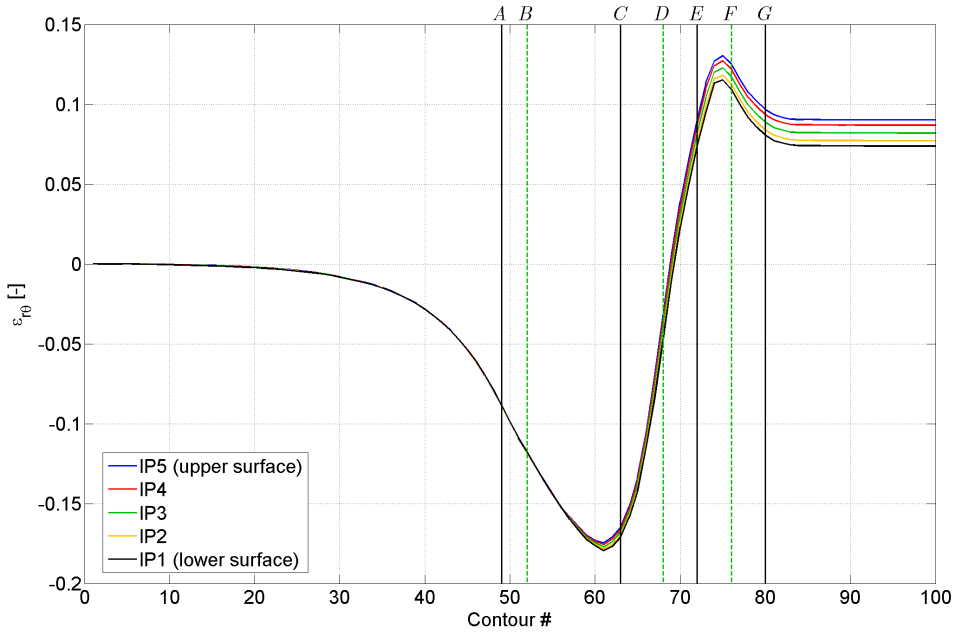


Figure 9.38: In-plane shear strain of element 1469 for each contour of the 70-degree cone

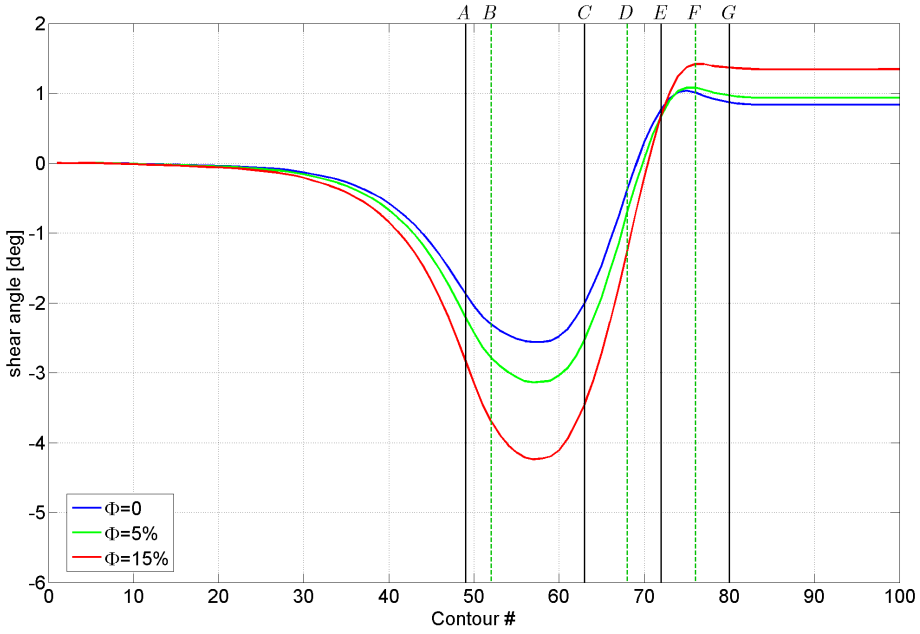


Figure 9.39: Shear angle, in degrees, of element 1469 for each contour of the 70-degree cone for different values of the friction coefficient

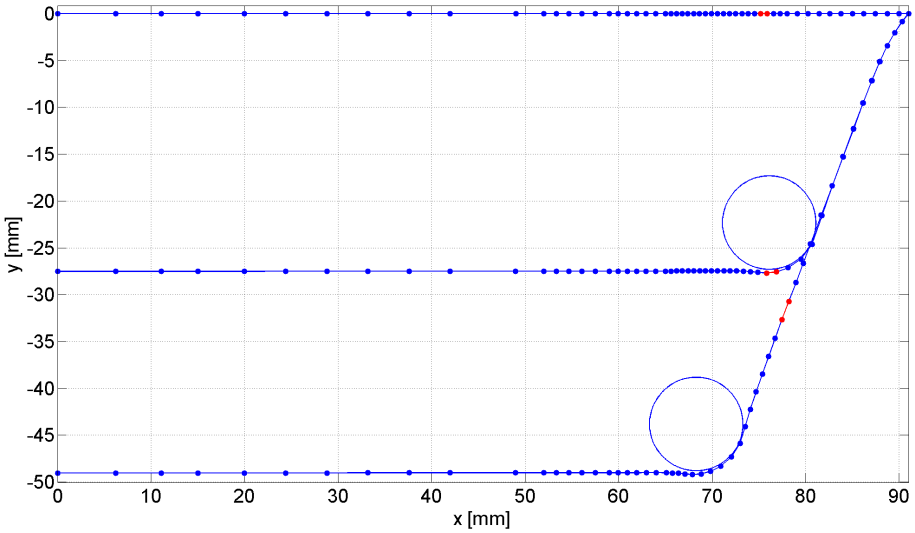


Figure 9.40: Position of the tool with respect to element 1469 in contours 0, 57, and 100

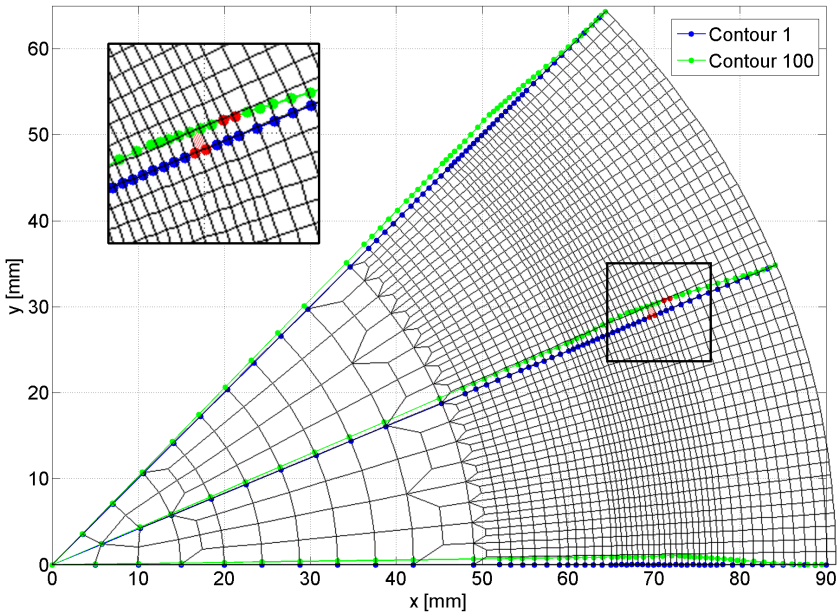


Figure 9.41: Twist effect of the 70-degree cone after 100 contours

9.4.6 Strain Path

After having analyzed the evolution of circumferential, radial, and in-plane shear strain components, this section will be dedicated to the analysis of the strain path. Figures 9.42, 9.43, and 9.44 show the strain path of element 1469 for the middle, lowest, and highest integration points, respectively, across the thickness. The horizontal axis represents the circumferential strain (minor strain) and the vertical one, the radial strain (major strain). In each figure, three contours were highlighted in red (contours 49, 60, and 72) and a zoom on the strain path during these contours was added on the right-hand side. These contours were chosen because, according to Figure 9.23, contours 49 (=A) and 72 (=E) delimit the zone in which the tool touches element 1469, and contour 60 is approximately in the center of that zone.

It can be seen in the three figures that the strain paths follow the typical zig-zag patterns encountered during SPIF, as explained in the literature review in Section 2.2. These patterns are due, on the one hand, to the almost monotonically increasing radial deformation (major strain) during the whole process, and, on the other hand, to the oscillations in the circumferential deformation (minor strain) during each contour.

The difference between the strain paths of the lowest, middle, and highest integration points comes from bending and unbending in the circumferential direction. The strain paths of the lowest and highest integration points are quite similar at first glance. A closer look shows, however, that during each contour the minor strain on one figure does exactly the opposite of the minor strain on the other one.

Regarding the strain path of the middle integration point, a final comment can be made on Figure 9.42. For almost all the contours between 49 and 72, i.e., the contours during which the tool passes over element 1469, two oscillations can be observed in the minor strain. This is especially visible in the zoom on the strain path during contour 60. As was already mentioned in the analysis of the membrane strain during one contour, these oscillations are due to the discrete locations of the contact integration points.

9.4.7 Equivalent Stress Integrated over the Thickness

Figure 9.45 shows the evolution of the equivalent stress (integrated over the thickness) as a function of time. Once again, in order to obtain a smoother curve, Figure 9.46 shows only one point for each contour.

First, in this last figure, the equivalent stress increases from one contour to the next until it reaches a first local maximum around contour C. Then it decreases slightly and increases again to reach its global maximum around contour E. As a reminder, both the circumferential and the radial strains have their maximum around contour E. However, there is no steady-state region at the end of the process, as was the case for the thickness, the tool force, and the circumferential, radial, and in-plane shear strains.

To understand this phenomenon, it is necessary to look at the evolution of the strain in detail. Figure 9.47 shows a zoom on the evolution of the radial strain of element 1469 as a function of time around the end of the process, an evolution

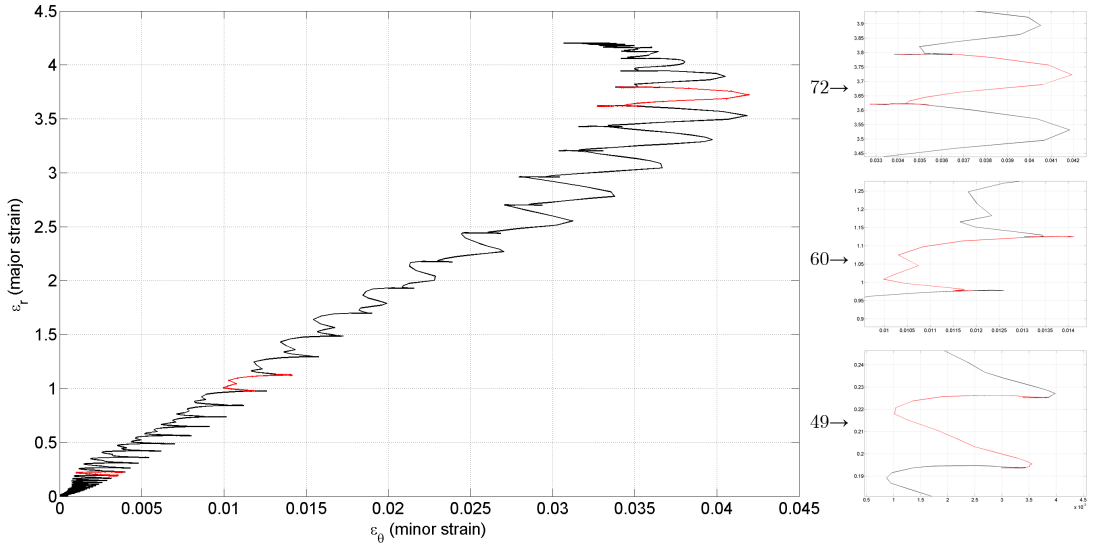


Figure 9.42: Strain path of the middle integration point of element 1469 for the 70-degree cone, with zooms on contours 49, 60, and 72

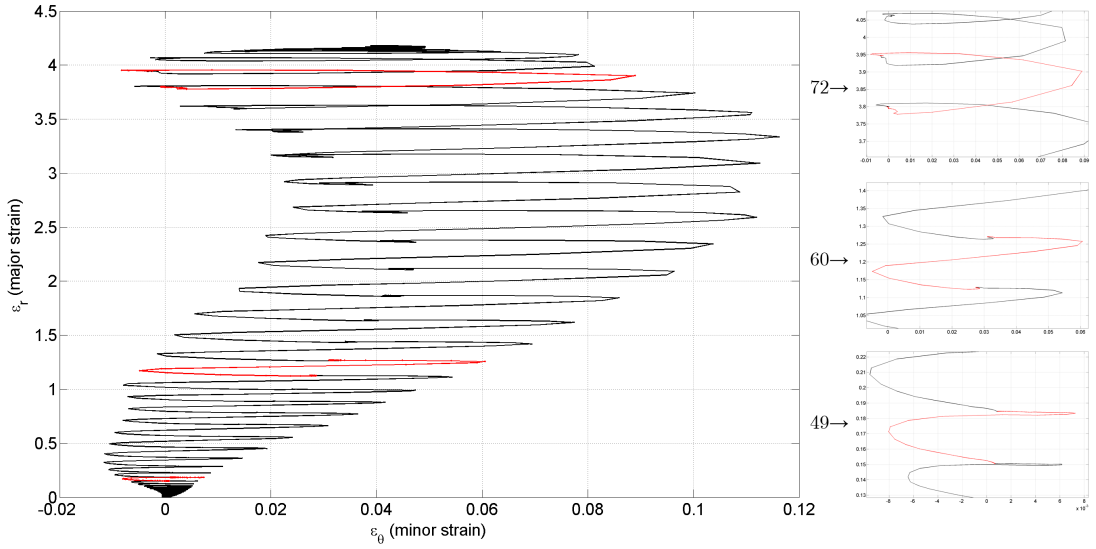


Figure 9.43: Strain path of the lowest integration point of element 1469 for the 70-degree cone, with zooms on contours 49, 60, and 72

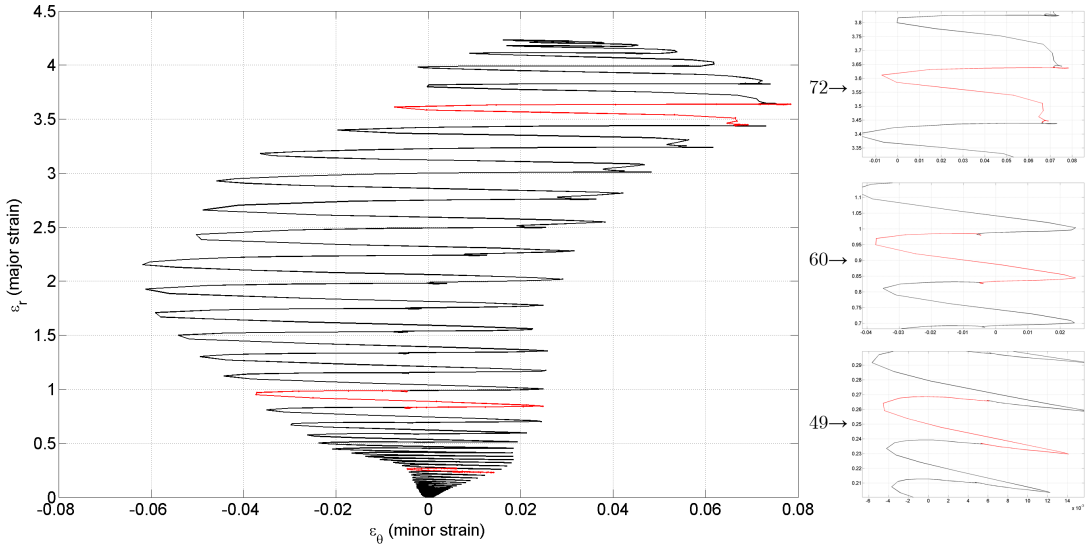


Figure 9.44: Strain path of the highest integration point of element 1469 for the 70-degree cone, with zooms on contours 49, 60, and 72

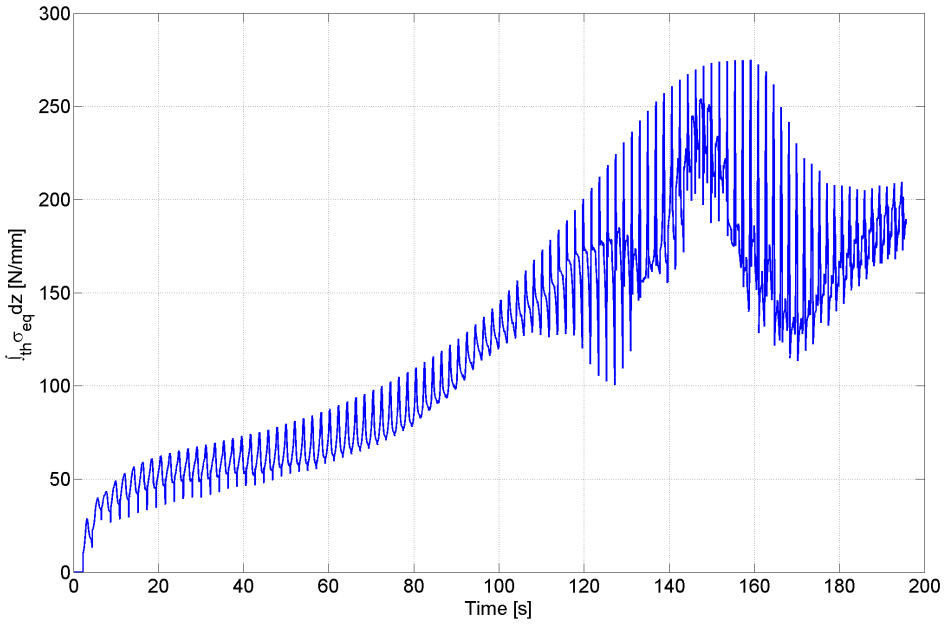


Figure 9.45: Equivalent Von Mises's stress of element 1469 as a function of time for the 70-degree cone

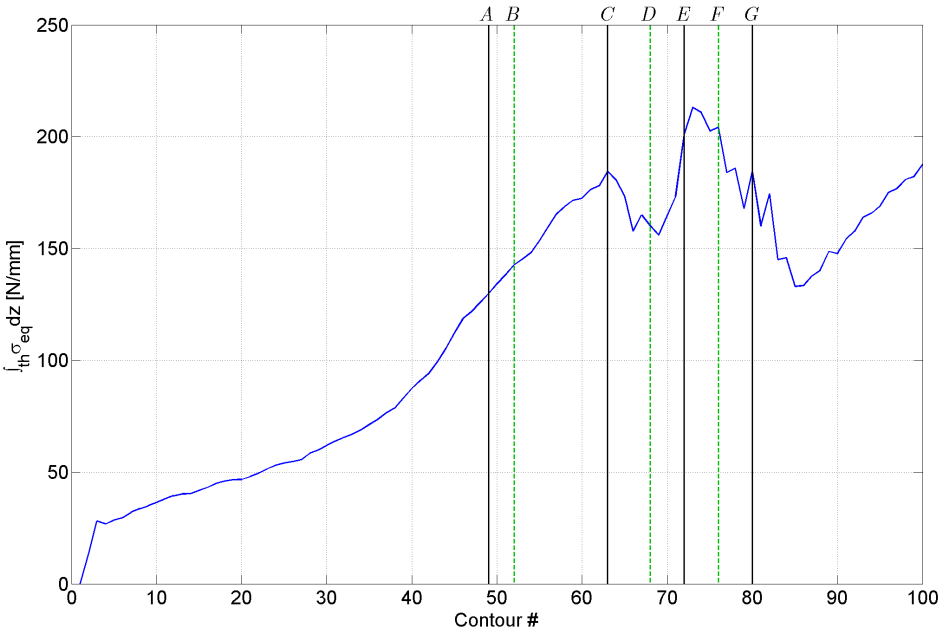


Figure 9.46: Equivalent Von Mises's stress of element 1469 for each contour of the 70-degree cone

which was shown in Figure 9.30 for the whole simulation. Even though the radial strain seemed stabilized at the end of the process, the zoom shows that there are some small variations: the lower surface is stretched and the upper surface is compressed slightly. This corresponds to a small unbending of the element.

These minor variations in the radial strain induce quite large variations in the radial stress, shown in Figure 9.48, because of the large value of Young's modulus. This explains the variations observed in the equivalent stress integrated over the thickness.

9.4.8 Summary of the Four Stages

The evolution of the components of the strain tensor and of the equivalent stress (integrated over the thickness) of element 1469 have been summarized in Table 9.3.

During stage I, there is almost no variation because the tool is far from element 1469 during its successive contours.

Next, during stage II, the tool starts to touch the element and influences all the variables.

Then, in stage III, the influence of the tool is even larger. The size of the contact zone reaches its maximum around contour *D*, where there is a peak in the flexural radial strain.

Finally, during stage IV, the tool moves away from the element. In particular, contour *E* is the last one during which there is a contact between the tool and the element. At the end of the simulation, after contour *G*, all components are almost stable, except some small elastic variations in the strain components, which induce quite large variations in the stress.

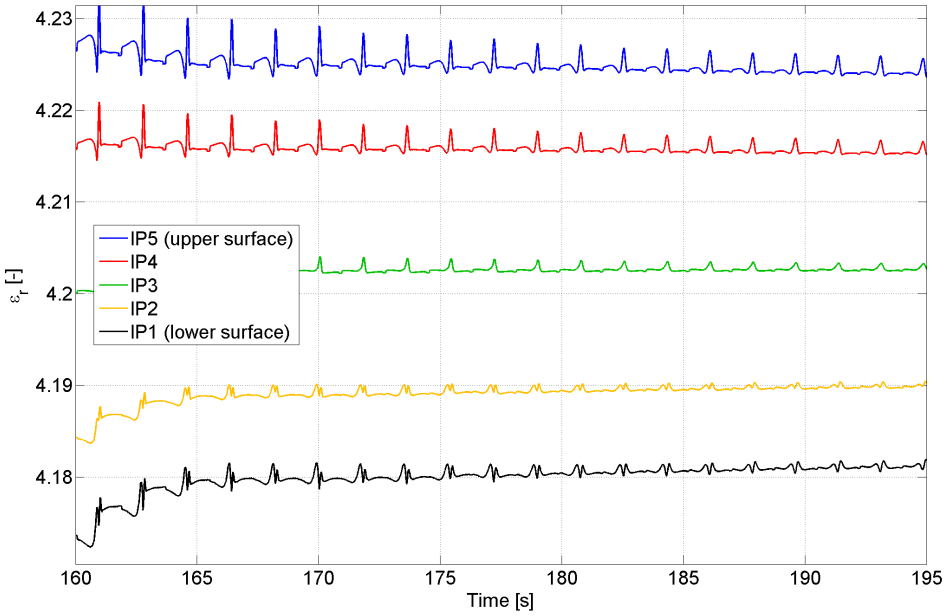


Figure 9.47: Zoom around the end of the process on the evolution of the radial strain of element 1469 as a function of time for each of the five integration points across the thickness

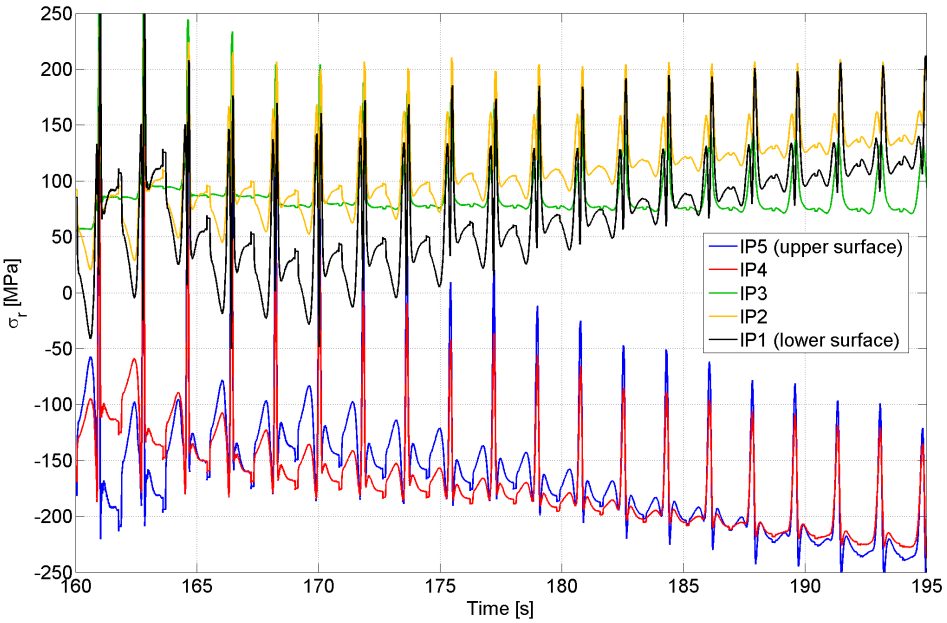


Figure 9.48: Zoom around the end of the process on the evolution of the radial stress of element 1469 as a function of time for each of the five integration points across the thickness

Stage	I		II				III				IV				
Contour		*A*		*B*		*C*		*D*		*E*		*F*		*G*	
ϵ_θ membrane	↗		↗		↗		↗		↗	max	↘		↘	⟶	3.1 %
ϵ_θ flexural	↘	min	↗	0	↗	max	↘	0	↘	min	↗	0	↗	⟶	cone radius
ϵ_r membrane	↗		↗		↗		↗		↗		↗		↗	⟶	420 %
ϵ_r flexural	↘	min	↗	0	↗		↗	max	↘	0	↘	min	↗	⟶	flat
$\epsilon_{r\theta}$	↘		↘		↘	min	↗		↗		↗	max	↘	⟶	8 %
$\int_{th} \sigma_{eq} \mathrm{d}z$	↗		↗		↗	max	↘	min	↗	max	↘		↘	↘	↗

Table 9.3: Summary of the evolution of the strain components of element 1469 during the forming of the 70-degree cone

9.5 Conclusions

To conclude this chapter on the analysis of the deformation mechanism during SPIF, several comments should be made. As a reminder, this analysis was performed on a 70-degree wall-angle cone with a depth of 30 or 50 mm. These parts were simulated using a 45-degree pie mesh made of shell elements.

Firstly, it should be noted that this partial mesh was able to reproduce the experimental shape accurately. In addition, the use of rotational-symmetry boundary conditions at the edges of the pie proved to be useful in the prediction of the twist effect.

The prediction of the thickness, however, was slightly less accurate, although acceptable in most regions. In particular, the location of the minimum thickness could not be predicted, even when friction was taken into account. The main reason for this is the use of a shell element which requires a plane-stress mechanical law.

Secondly, concerning the tool force, neither its general evolution nor its level was accurately predicted. Multiple reasons for this phenomenon were tested in this thesis and in literature by Bouffioux. The conclusion of this study is that a mixed-type isotropic-kinematic hardening law with accurate material parameters is required. In addition, a more advanced solid-shell element using a 3D mechanical law and being able to take through-thickness shear into account should be used. Indeed, the plane stress hypothesis of classic shell elements induces an error in the force prediction. Finally, when simulating a part whose wall angle is close to the failure angle, the mechanical law should be able to take damage into account in order to predict the large drop in the force observed experimentally.

Such an element was unfortunately not yet available in Lagamine at the time of this study. Still, the simulated tool force could be used to estimate the experimental friction coefficient.

In addition to these global predictions, a detailed analysis of the behavior of a single element was performed. Four stages in its behavior were identified and the evolution of the strain components and in the equivalent stress was analyzed in each of these stages. It was found that the radial strain is the quantity with the most influence on the global behavior of the element. Its value at the end of the simulation is more than 400 % (using Green's tensor). On the other hand, the circumferential component remains small (around 3 %) and oscillates considerably. However, this component should not be ignored because it is responsible for the zig-zag pattern observed in the strain path evolution, which plays a role in the prediction of failure during SPIF.

In addition, the evolution of the strain and stress was studied for one contour, effectively showing the bending and unbending effect when the tool approaches and moves away from an element.

Conclusions and Perspectives

Modeling the incremental forming process is an ambitious task. Indeed, the level of deformation reaches considerable proportions for some elements (much more so than in a more classic forming process such as deep drawing). In addition, a large area of the part is never touched by the forming tool, so the quality of its overall shape prediction is highly dependent on the quality of the element used.

In light of these obstacles, adapting a finite element code to model the process in a fast and accurate manner is even more ambitious. Nevertheless, this was and continues to be the aim of the project behind the present work.

The first step toward modeling the incremental forming process was to gain in-depth knowledge of the process itself. Therefore, this thesis began with a bibliographic review of the work previously undertaken by other researchers in this field, from both experimental and numerical points of view. Within the SeMPeR project, the monthly meetings and fruitful discussions with its scientific and industrial partners helped greatly in the search for a better understanding of the process.

Original syntheses emerged from this bibliographic study. In particular, the limitations of the incremental forming process and possible means for pushing these boundaries further were evaluated. Moreover, a thorough investigation of the enhanced formability observed in incremental forming was performed, providing an overview of the latest explanations for this highly-debated phenomenon.

With respect to the state of the art in finite element modeling of the incremental forming process, the main observation drawn from this review was that, in order to produce more accurate results and reduce the computation time, most authors chose to employ shell elements and a dynamic explicit integration scheme. Given the objective of this thesis, namely to produce a rapid simulation tool, this approach was also taken by the author and a significant amount of time was spent on these topics in the second part of this thesis.

At the beginning of the study, only one shell element was available within the FE code Lagamine. This element, however, suffered from numerous, severe problems and limitations. The first task was, therefore, to assess its performance and correct its problems in order to make it functional. This element was required, not only for this study but also for others in progress, because brick elements are

not well adapted to simulations of sheet metal forming. Thanks to the author's work on this element in the code, other researchers requiring this type of element have been able to benefit from its advantages in their own work.

Once this was accomplished, the use of a dynamic explicit integration scheme to simulate incremental forming required the development of a new strategy to take into account the contact between the tool and the metal sheet. Once again, in the search for lower computation time and higher efficiency, this contact strategy was designed especially with the distinctive features of the incremental forming process in mind. The solution adopted with respect to the tool's position and spherical shape simplified the contact search, resulting in more efficient and more accurate results.

Following its implementation, this newly-developed technique was validated thanks to a large series of simulations. The contact proved indeed to be more accurate and the computation time dedicated to it was minimal. Unfortunately, on the whole, these simulations were not as accurate as those performed with an implicit strategy, nor did they bring about a significant reduction in overall computation time as expected.

After further research, it was found that the main reason for this setback in the results was the lack of an efficient numerical damping algorithm in the finite element code. Therefore, the undesirable inertia in the equilibrium equation could not be reduced to an acceptable level; in addition, this lack prevented the use of large mass-scaling factors, at the cost of increased computation time.

At that time, it was decided not to continue investigating the dynamic explicit strategy, the reason for which was twofold. First, the project was originally funded on a four-year basis, which meant that the work necessary for modifying the finite element code to correct the accuracy and mass-scaling problems would have been excessively time-consuming. Secondly, given the large amount of experimental data available through the SeMPeR project, it was considered wiser to dedicate the remaining time in the thesis to detailed validations of the simulation tool using an implicit strategy. Thus, instead of concentrating on purely numerical aspects of simulating incremental forming, this thesis benefited from the valuable experimental contributions from the project's team.

Because of this decision, the third part of this thesis focused on the use of classic implicit simulations to study the material behavior during incremental forming. However, this choice was not without its own complications: this study required a very detailed mesh, whereas the computation time of implicit simulations increases rapidly with the number of elements. For this reason, a simplification technique was developed to simulate the forming of parts with rotational symmetry. A relatively small portion of the metal sheet was meshed with purpose-built boundary conditions replacing the missing material. Then, a simulation incorporating this simplification was compared to a complete simulation. This technique proved to be highly efficient: the computation time was reduced by a large factor of more than seventy, with only a slight decrease in accuracy, so long as the maximum level of deformation reached by the elements was limited.

This technique was finally applied to the simulation of a cone, whose results were compared to experimental measurements in terms of shape, thickness, and predicted tool force. In addition, the evolution of multiple variables, such as the

strain and stress tensor components, was studied during the simulation in order to gain a better understanding of the deformation mechanism.

In these simulations, the tool force appeared to be highly sensitive to the quality of the material law. Therefore, the law's parameters need to be identified with care in order to reach acceptable force predictions. Moreover, the use of a mixed-type isotropic-kinematic hardening law seems to be required, given the serrated strain path to which the material is subjected.

Significant progress has been achieved in understanding as well as modeling the incremental forming process itself. However, there are still a number of areas that have yet to be resolved or investigated, as shown by these latest results.

On the one hand, the dynamic explicit strategy seems to be a very promising solution to the problems of accuracy and lengthy computation time. However, using this strategy would require, in turn, the development of a more advanced damping technique, for two reasons. First, damping would make the use of larger mass-scaling factors possible, thereby reducing the computation time. Then, it would decrease the parasitic oscillations due to large inertia terms added to the system, which would help increase the accuracy of these simulations.

On the other hand, the existing finite element code does not contain a more advanced shell element, which is a major drawback. Indeed, the extensive use of the existing shell element using an implicit strategy showed that this element is not perfectly adapted to incremental forming. As a result, it would be necessary to develop a solid-shell element, which is (unlike classic shell elements) not limited to plane-stress laws. Such an element could be used with much more advanced mechanical laws including damage modeling, could take the through-thickness shear into account, and would most likely solve the problem of the inaccurate force prediction.

Finally, a remeshing technique has been recently developed for the finite element code. This technique, which seems very promising in the reduction in computation time, should be tested for modeling incremental forming.

Appendices

Appendix A

New Lagamine Routines

A.1 Main Routines

The program is composed of several routines.

`MINCFO`	Description of the module containing the shared variables used by the algorithm.
`MST_MAIN`	Main routine containing the structure of the program.
`MST_CORNER`	Routine that gives the list of nodes of one surface of a given element.
`MST_NODF`	Routine that computes the acceleration of the nodes of the top surface of an element for three different unit contact forces.
`LPROXEL`	Routine that computes the list of elements close to the tool at a given time.
`PROXIM`	Proximity criterion used by `LPROXEL`.
`MST_PCONT`	Routine that computes, for each element, the location of the unique potential contact point [HEN05a, page 81].
`MST_GROUP`	Routine that extracts, from the set of contact points, the potential contact points [HEN05a, page 81].
`MST_SYSTEM`	Routine that writes the system of equations linking the contact forces to the distances and relative velocities between the contact points and the tool.
`MST_DISVEL`	Routine, used in `MST_SYSTEM`, that computes the right-hand side of the system of equations.
`MST_SENSIB`	Routine, used in `MST_SYSTEM`, that computes the left-hand side of the system of equations.
`MST_SOLVE`	Routine, downloaded from Lapack [LAPACK], that solves a non-symmetric linear system of equations.
`INIT_INCFO`	Routine used in `INITB` in order to initialize the module and some variables.
`FCONEL`	Preprocessor routine that computes the connectivity matrix.

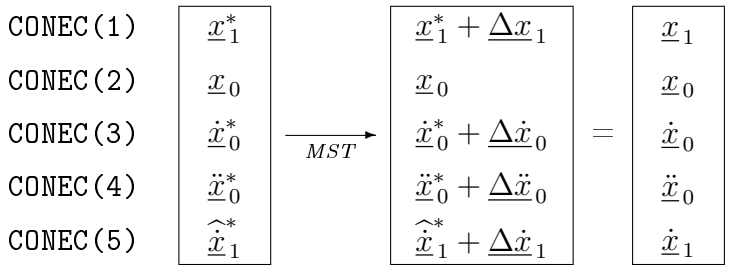


Figure A.1: Time step in a dynamic explicit strategy

A.2 Shared Variables

The program uses several shared variables, which are accessible whenever the module `MINCFO` is used. A short description is presented in Table A.1. A more thorough description can be found in the module itself, within the sources of the LAGAMINE code.

A.3 Program Structure

A.3.1 General Considerations

The goal of the program is to adjust the intensity of the contact forces at the beginning of the time step in order to achieve the desired conditions — as far as the distance and relative velocity with respect to the tool are concerned — at the end of the time step.

The program is called at the beginning of the time step, more precisely in `LAMIN2` between the subroutines `INIDDL` and `NORME1`. At this time, the end-of-step positions have already been calculated, but based on the initial accelerations without taking the contact forces into account. The content of the variable `CONEC`, for a given node, is presented in Figure A.1.

Throughout the rest of this chapter, a distinction will be made between the *new potential contact points* and the *old contact points*. The new potential contact points are the potential contact points at the end of the current step. The old contact points are the contact points at the end of the previous step for which there was a real contact. In other words, if there is no interpenetration with the tool on some new potential contact points without adding any contact force to the corresponding old contact point, those points will be ignored for the next step.

Provided that there is an equal number of old and new contact points, the program has to adjust the initial acceleration of the old contact point — three components for each point — in order to adjust geometric conditions at the new contact points — three geometric quantities.

The flow chart presenting the general structure of the Moving Spherical Tool algorithm is shown on page 197. The different parts will be discussed in detail in the next sections of this chapter.

Variable	Description
MAXELEM	Maximum number of elements in the neighborhood of the tool
MAXPC	Maximum number of contact points
MAXPC3	Maximum number of unknowns ($3 \times$ MAXPC $+ 1$)
DEBUG_MST	Flag for printing debug messages
INCFO	Indicator of whether (1) or not (0) the simulation uses the Incremental Forming method
MAXELC	Maximum number of elements connected to a given element
NBIFOR	Number of integers in vector IFORM
NBRFOR	Number of reals in vector RFORM
MCONEL(MAXELC,NELEM)	Connectivity matrix
IFORM(NBIFOR)	Vector of integer parameters
RFORM(NBRFOR)	Vector of real parameters
IENEAR(MAXELEM)	Vector of elements near the tool
INVELE(NELEM)	Vector indicating the position of all the elements of the mesh in the vector IENEAR (if $\neq 0$)
ITISSHELL	Indicator of whether (1) or not (0) the simulation uses shell elements (COQJ4) for the simulation
NDIM	Number of DOF per node
NNODE_FACE	Number of nodes per element face (4)
DATA_ELEM(56,MAXELEM)	Matrix containing the real numbers that need to be stored regarding the elements in the neighborhood
INFO_ELEM(3,MAXELEM)	Matrix containing the integer numbers that need to be stored regarding the elements in the neighborhood
LISTE_PCONT(MAXPC)	Vector containing the list of element numbers that had contact at the end of the previous step
THICKNESS_SHELL	Thickness of the shell elements
XYZ_PCONT(3,MAXPC)	Matrix containing the contact positions of the elements in LISTE_PCONT
RST_PCONT(9,MAXPC)	Matrix containing the local reference frame at the contact position for the elements in LISTE_PCONT
FORCE_MST(3)	Vector containing the total force on the Moving Spherical Tool
NUM_C	Number of contact points in the list LISTE_PCONT
XYZO(3)	Vector containing the coordinates of the tool at the previous step
VITO(3)	Vector containing the velocity of the center of the tool at the previous step
INI	Indicator of whether (1) or not (0) this is the first time the simulation calls LPROXEL
IWRK	Flag for printing the dynamic and internal energy in the file NTWRK (performed in PRISUM and LAMIN2)

Table A.1: Shared variables

A.3.2 Nodal Forces

The first step to perform is the computation of the nodal forces. For every old contact point, the routine `MST_NODF` computes the change in acceleration of the four nodes of the element if three unitary contact forces are added at the contact point. These unitary contact forces correspond to one normal and two tangential forces.

A.3.3 New Neighborhood

The next step deals with updating the neighborhood. The routine `LPROXEL` searches for the list of elements which are close to the new tool position, i.e., satisfying a given proximity criterion (stored in routine `PROXIM`). More details can be found in the present author's Masters thesis [HEN05a, page 79].

A.3.4 New Theoretical Contact Points

For each element in the new neighborhood, the routine `MST_PCONT` searches for one and only one theoretical contact point. This point is defined as the point from which the perpendicular to the element surface goes through the tool center. The search is done iteratively. Starting from an approximate location (the previous contact point if applicable, the center of the element surface otherwise), the solution progressively converges towards the correct point. If this point should happen to be outside the element surface, the point's location is limited to the closest edge or corner point.

A.3.5 New Potential Contact Points

When this information is available, the routine `MST_GROUP` identifies the real potential contact points according to three different possibilities:

- if a theoretical contact point is inside the element surface, it is considered a real contact point;
- if a theoretical contact point is located on an edge, it is considered a real contact point if and only both elements sharing this edge indicate contact there;
- if a theoretical contact point is located on a corner, it is considered a real contact point if and only if all of the elements sharing this corner indicate contact there.

In all other cases, the theoretical contact point is ignored. More details about this section and the previous one can be found in Section 5.3.1.

A.3.6 Selection of Some Potential Contact Points

In order to be able to solve the contact conditions, the algorithm must ensure that there is the same number of old contact points (three unknowns for each) and new contact points (three equations for each).

In most cases, the number of new contact points is larger or equal to the number of old ones. This is due to the fact that the old contact point list contains only the points for which there was a real contact at the previous time step, whereas the new contact point list contains all the possible points. It is most likely that the old contact points' locations will have moved according to the tool displacement during the time step. Knowing that, the algorithm will select, for each old contact point, the new contact point which is closest to the old location shifted by the tool displacement. The other new potential contact points will be kept in order to check if those points should also have been considered real contact points, i.e., if there is interpenetration with the tool. If this is the case, those points will be kept in the list of old contact points for the next step. This mechanism allows the number of contact points to increase, which is required for the algorithm to start.

In the (rare) cases where the number of new potential contact points is strictly smaller than the number of old ones[1], the algorithm will work backwards. For each *new* contact point, it will select the *old* contact point whose location, shifted by the tool displacement, is closest to the new contact point's location. Finally, the contact forces on the other old contact points will be set to zero.

A.3.7 System Solution

There is now a one-to-one relation between the old contact points and the new potential contact points. The algorithm now has to write the equations linking the desired geometric quantities to the contact forces and to solve the system. This is done in the routine `MST_SYSTEM`. The general equations are described in the author's Masters thesis [HEN05a, page 82]. For a given potential contact point, the three corresponding terms on the right-hand side are computed in the routine `MST_DISVEL`. For a given potential contact point and a given old contact point to which the contact forces are applied, the corresponding nine terms on the left-hand side are computed in the routine `MST_SENSIB`.

When the global system is written, a hypothesis is made on the type of contact of each contact points. Given this hypothesis, the general equations are adapted as described in Section 5.3.4. The system of equations is solved by the routine `MST_SOLVE`. If any of the hypotheses are not satisfied, it will be necessary to iterate on the type of contact.

Finally, when all the conditions are satisfied, the program must check if some of the new contact points that were ignored are in contact with the tool. In that case, these points will be considered old contact points for the next iteration.

A.3.8 Updating of `CONEC`

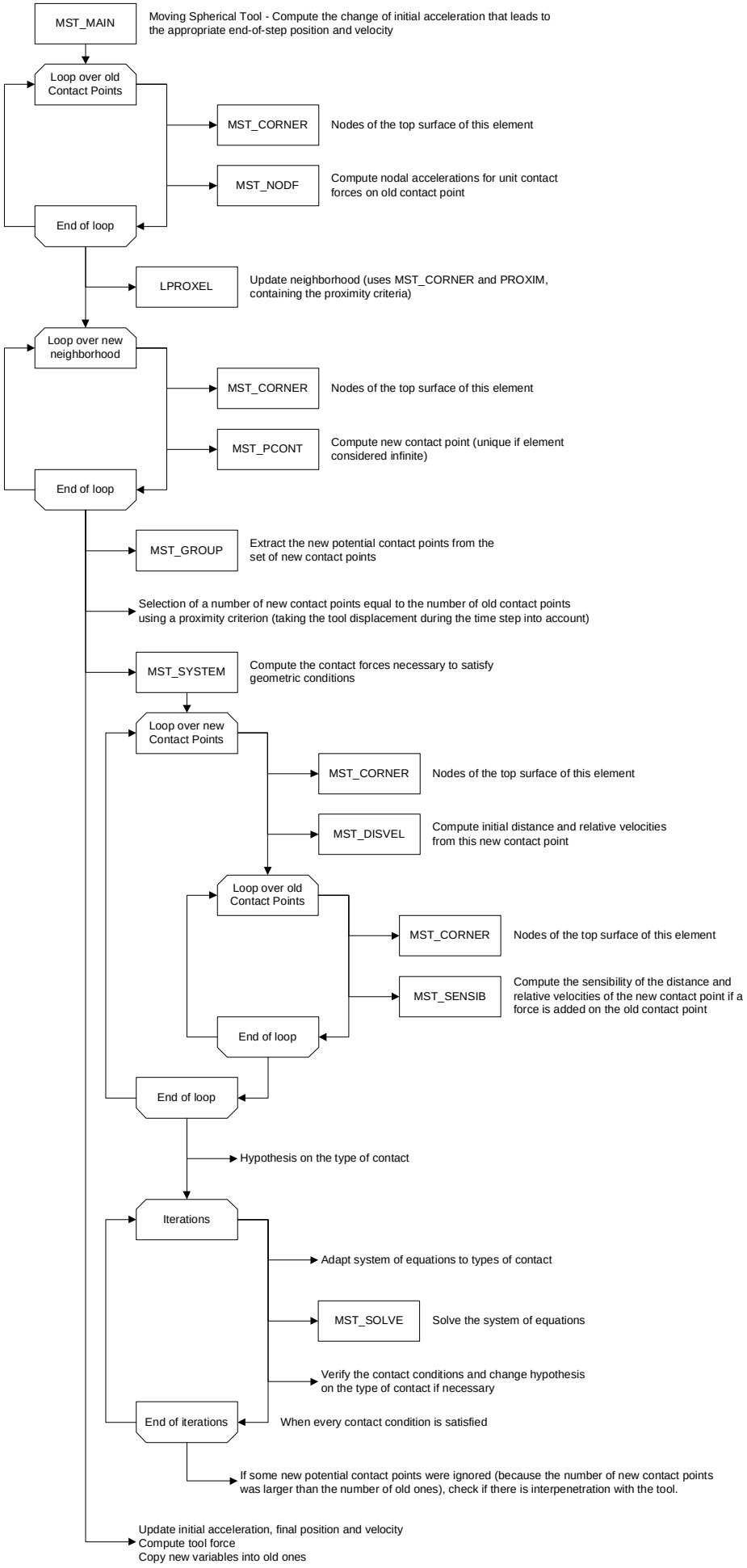
The last step, performed at the end of the routine `MST_MAIN`, is to update the initial acceleration, initial velocity and final position in the variable `CONEC`. In addition, all the contact forces are summed into one global tool reaction. Finally,

[1] This happens, for instance, if two old contact points were significantly close to each other, located on either sides an edge, at the beginning of one step, and they became only one point on the edge at the end of the time step.

the data concerning the new contact points are copied into the old ones, which are part of the module `MINCFO` and, so, will be kept for the next step.

A.3.9 Flow Chart of the MST Method

The flow chart of the MST method is presented in the next page.


MST_MAIN
Moving Spherical Tool - Compute the change of initial acceleration that leads to the appropriate end-of-step position and velocity
Loop over old Contact Points
MST_CORNER
Nodes of the top surface of this element
MST_NODF
Compute nodal accelerations for unit contact forces on old contact point
End of loop
LPROXEL
Update neighborhood (uses MST_CORNER and PROXIM, containing the proximity criteria)
Loop over new neighborhood
MST_CORNER
Nodes of the top surface of this element
MST_PCONT
Compute new contact point (unique if element considered infinite)
End of loop
MST_GROUP
Extract the new potential contact points from the set of new contact points
Selection of a number of new contact points equal to the number of old contact points using a proximity criterion (taking the tool displacement during the time step into account)
MST_SYSTEM
Compute the contact forces necessary to satisfy geometric conditions
Loop over new Contact Points
MST_CORNER
Nodes of the top surface of this element
MST_DISVEL
Compute initial distance and relative velocities from this new contact point
Loop over old Contact Points
MST_CORNER
Nodes of the top surface of this element
MST_SENSIB
Compute the sensibility of the distance and relative velocities of the new contact point if a force is added on the old contact point
End of loop
End of loop
Hypothesis on the type of contact
Iterations
Adapt system of equations to types of contact
MST_SOLVE
Solve the system of equations
Verify the contact conditions and change hypothesis on the type of contact if necessary
End of iterations
When every contact condition is satisfied
If some new potential contact points were ignored (because the number of new contact points was larger than the number of old ones), check if there is interpenetration with the tool.
Update initial acceleration, final position and velocity
Compute tool force
Copy new variables into old ones

Appendix B

Local Axes of the COQJ4 Element

The orientation of the local axis of the element does not depend on the curvature of the element. Its definition is given below.

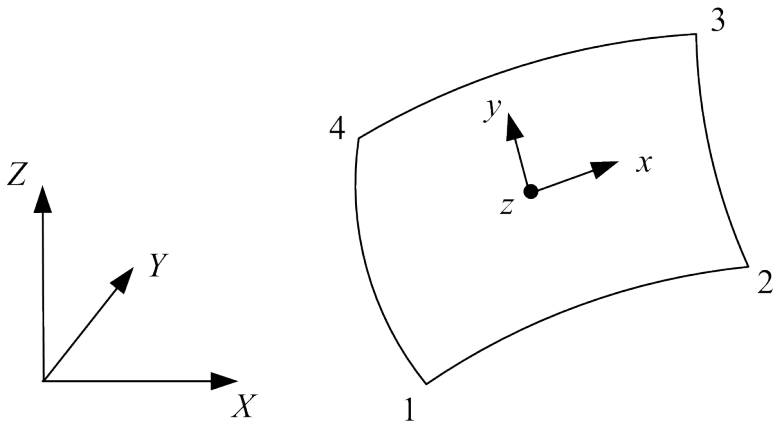


Figure B.1: Global and local reference frame of an element

The element's center of gravity is computed by averaging the positions of the four nodes:

$$\underline{G} = \left(\underline{X}_1 + \underline{X}_2 + \underline{X}_3 + \underline{X}_4\right)/4. \tag{B.1}$$

Since the four nodes of the element may not be co-planar, the mid-plane of the element is defined by one point, the center of gravity $\underline{G}$, and two unit vectors aligned with the medians of the quadrilateral:

$$\underline{R}_1 = \frac{\underline{X}_2 + \underline{X}_3}{2} - \frac{\underline{X}_1 + \underline{X}_4}{2};\ \underline{R}_{1P} = \frac{\underline{R}_1}{\|\underline{R}_1\|} \tag{B.2}$$

and

$$\underline{R}_2 = \frac{\underline{X}_3 + \underline{X}_4}{2} - \frac{\underline{X}_1 + \underline{X}_2}{2};\ \underline{R}_{2P} = \frac{\underline{R}_2}{\|\underline{R}_2\|}. \tag{B.3}$$

The normal to the mid-plane is computed by taking the cross product of those two vectors:

$$\underline{R}_3 = \underline{R}_{1P} \wedge \underline{R}_{2P};\ \underline{R}_{3B} = \frac{\underline{R}_3}{\|\underline{R}_3\|}. \tag{B.4}$$

Figure B.2 shows the three vectors $\underline{R}_{1P}$, $\underline{R}_{2P}$ and $\underline{R}_{3B}$. Figure B.3 illustrates the possibility that the mid-plane may not contain the nodes of the element.

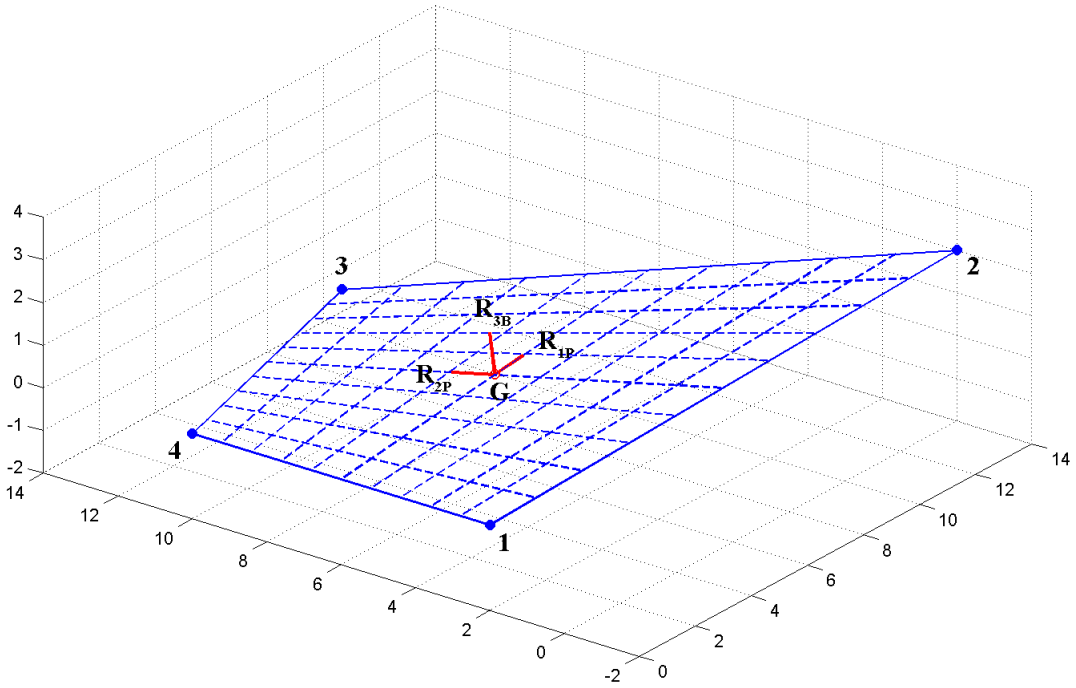


Figure B.2: Local axis of the element

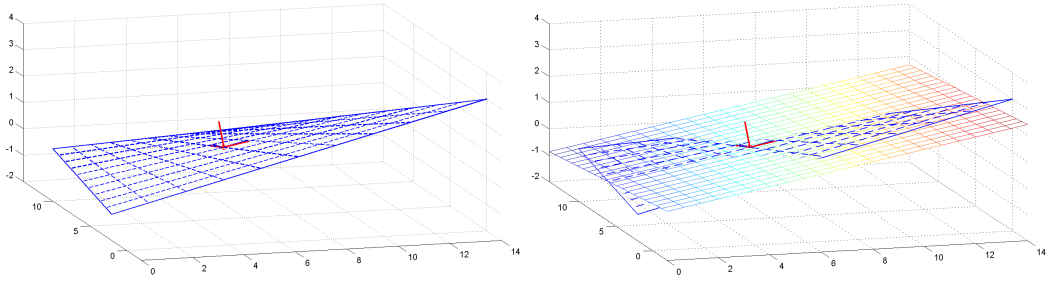


Figure B.3: Mid-plane of the element

Since the element is a general quadrilateral, its medians $\underline{R}_{1P}$ and $\underline{R}_{2P}$ are not necessarily perpendicular to each other. In order to define the local axis of the element, this must be corrected. The first idea would be to keep one of the vectors, for example, $\underline{R}_{2P}$, and choose another one in the mid-plane of the element by taking the following cross product:

$$\underline{R}_{2P} \wedge \underline{R}_{3B}. \tag{B.5}$$

However, this would break the symmetry of the vectors. So, the solution chosen in the element is the following. First, it should be noted that the vector

$$\underline{R}_{1PP} = (\underline{R}_{2P} \wedge \underline{R}_{3B}) + \underline{R}_{1P} \tag{B.6}$$

is the bisector of the two vectors $\underline{R}_{1P}$ and $\underline{R}_{2P} \wedge \underline{R}_{3B}$ but is not normalized. Therefore, the two vectors

$$\underline{R}_{1B} = \frac{\underline{R}_{1PP}}{\|\underline{R}_{1PP}\|} \tag{B.7}$$

and

$$\underline{R}_{2B} = \underline{R}_{3B} \wedge \underline{R}_{1B} \tag{B.8}$$

are orthogonal to each other and form the same angles α with respect to $\underline{R}_{1P}$ and $\underline{R}_{2P}$, as illustrated in Figure B.4. The local axis is then defined by the three vectors $\underline{R}_{1B}$, $\underline{R}_{2B}$, and $\underline{R}_{3B}$.

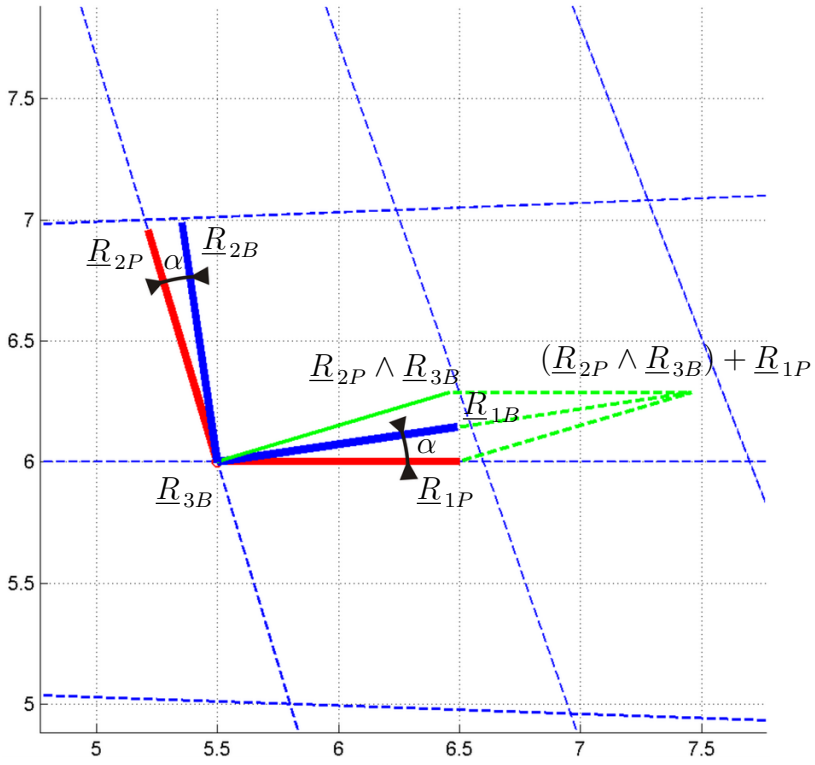


Figure B.4: Orthogonalization of the local axis

The rotation matrix to transform global coordinates into local ones is defined by

$$\underline{\underline{R}} = \begin{pmatrix} R_{1x} & R_{2x} & R_{3x} \\ R_{1y} & R_{2y} & R_{3y} \\ R_{1z} & R_{2z} & R_{3z} \end{pmatrix}. \tag{B.9}$$

Appendix C

Material Parameters

The material parameters presented in this chapter correspond to the AA3003-O aluminum alloy. This material is a fully annealed aluminum that was supplied by Corus. In addition to aluminum, this alloy contains traces of other chemical elements. Its chemical composition is presented in Table C.1.

Si	Fe	Cu	Mn	Zn
0.2	0.6	0.08	1	0.01

Table C.1: Chemical composition (in weight %) of the AA3003-O aluminum alloy

C.1 Hill's Yield Locus

The material parameters of Hill's 1948 yield locus were determined at the MTM Department of the Catholic University of Leuven (KUL) using Lankford's coefficients, also called anisotropy coefficients. These coefficients are defined as the ratio between the plastic strain rate in the width direction ϵ_w^{pl} and in the thickness direction ϵ_{th}^{pl} during uniaxial tensile tests performed at a given angle from the rolling direction of the metal sheet:

$$r_L = \frac{\dot{\epsilon}_w^{pl}}{\dot{\epsilon}_{th}^{pl}}. \tag{C.1}$$

Therefore, three tensile tests were performed at 0, 45, and 90° from the rolling direction of the sheet and the three Lankford's coefficients were measured:

$$\begin{cases} r_0 &= 0.68 \\ r_{45} &= 0.73 \\ r_{90} &= 0.66. \end{cases}$$

Lankford's coefficients can be written as functions of the six Hill parameters:

$$r_0 = \frac{H}{G}, \quad r_{90} = \frac{H}{F} \quad \text{and} \quad r_{45} = \frac{2N - F - G}{2\,(F + G)}. \tag{C.2}$$

These equations can be inversed if three additional constraints are added $H+G=2$ and $N=L=M$. In that case, Hill's parameters can be computed using the following equations:

$$\begin{cases} G = \dfrac{2}{1+r_0}, \quad H = 2-G, \quad F = \dfrac{H}{r_{90}}, \\ N = L = M = (F+G)\left(r_{45} + \dfrac{1}{2}\right). \end{cases} \tag{C.3}$$

The values of Hill's coefficients are given in Table C.2.

Parameter	Value
F	1.224
G	1.193
H	0.807
$N=L=M$	2.977

Table C.2: Hill's 1948 yield locus parameters for AA3003 identified with three tensile tests only

During his PhD at the University of Liège, Flores developed a biaxial testing machine [FLO06]. With this machine, it is now possible to perform a large tensile test and a shear test separately, consecutively or simultaneously.

Using this machine, a simple shear test was performed on the same aluminum alloy. The same shear test was also simulated with LAGAMINE using the parameters identified using only Lankford's coefficients, as presented above. The numerical results were quite different from the experimental ones. For this reason, Bouffioux used an inverse method to recalibrate the parameter N of Hill's yield locus, which is the only one to have an influence on the shear results [BOU08b]. This modification of the N parameter is related to the texture evolution during the shear test, as demonstrated by Flores in his thesis [FLO06] and in an article by Flores et al. [FLO07]. Indeed, the authors proved that changing only the hardening parameters when identifying a material law was not enough to obtain an accurate prediction of both a tensile test and a shear test. The reason for this is that when a material undergoes large deformations, the shape of the yield surface evolves with the change of texture. Since Hill's yield locus is a simple phenomenological law and its parameters are kept constant during deformation, they should be chosen in such a way that they represent the average value of the shape of the yield surface for the different material states reached during the process to be simulated with the model. With this aim in view, Bouffioux focused her research on the development of the best identification procedure for the material models used to simulate SPIF. In particular, she analyzed the influence of the choice of tests used to identify the material parameters on the accuracy of the forming forces and part shapes during SPIF [BOU08b, BOU08a, FLO07, BOU07].

The new values of Hill's parameters are given in Table C.3. These parameters were used throughout this thesis in all the simulations.

Parameter	Value
F	1.224
G	1.193
H	0.807
$N = L = M$	*4.06*

Table C.3: Hill's 1948 yield locus parameters for AA3003 identified with three tensile tests and a shear test

C.2 Isotropic Hardening

In order to identify the parameters of the hardening law, the simplest solution was to fit Swift's isotropic law (power law) on a tensile test result. The stress–strain curve obtained is shown in Figure C.1, along with the experimental measurements. The material parameters are given in Table C.4, with σ_F^0 being the initial yield stress. The elastic parameters, E and ν, were identified at the VUB using an acoustic method.

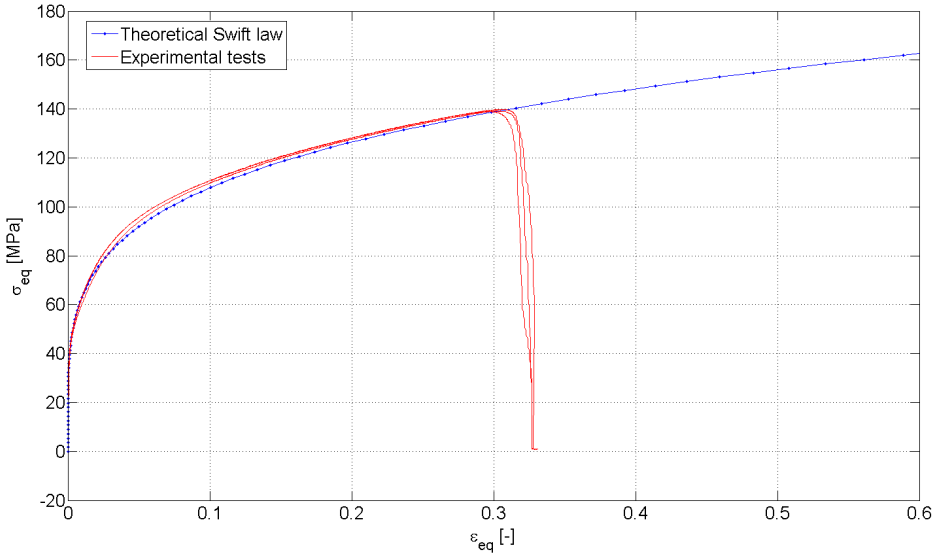


Figure C.1: Stress–strain curve for AA3003-O

Parameter	Value	[Unit]
E	72600	[MPa]
ν	0.36	[-]
σ_F^0	42.97	[MPa]
K	180	[MPa]
ϵ_0	0.00109	[-]
n	0.21	[-]

Table C.4: Elastic and isotropic hardening parameters for AA3003

C.3 Mixed Isotropic-Kinematic Hardening

C.3.1 Identification for the BWD3D Brick Element

A more advanced hardening law can be identified by inverse modeling. Using the experimental results of a tensile test, a Baushinger test (shear and reverse shear) and an indent test (first part of the line test), Bouffioux identified the parameters of both Armstrong-Frederick's and Ziegler's kinematic hardening laws using an inverse modeling technique [BOU08b].

By definition, a material law is supposed to represent the behavior of the material and be independent of the FE type for which it will be used. In practice, however, this is not the case since the element itself has a small internal stiffness which is due to the anti-hourglass energy modes. For this reason, Bouffioux performed two identifications: one using simulations performed with brick (BWD3D) elements and one with shell (COQJ4) elements. This section only gives the material parameters obtained for the brick elements, as shown in Tables C.5 and C.6.

It can be seen that for Armstrong-Frederick's kinematic hardening parameters, the yield surface used was Hill's, whereas for Ziegler's, Von Mises's simple law was used. Even though Hill's yield locus is more accurate since it was measured for the actual material, Ziegler's hardening is the only kinematic hardening law available in Abaqus and it can only be used with Von Mises's yield locus. Since it was interesting to be able to compare simulation results obtained with both LAGAMINE and Abaqus FE codes with the same constitutive law, this choice seemed logical.

Yield surface coefficients	Swift's law parameters	Back-stress data
$F = 1.224$	$K = 120.2$	C_X=29.7
$G = 1.193$	$\epsilon_0 = 9.26\ 10^{-4}$	X_{sat}=26
$H = 0.8067$	$n = 0.288$	
$N = L = M = 4.06$		

Table C.5: Armstrong-Frederick's parameters identified with brick elements

Yield surface coefficients	Swift's law parameters	Back-stress data
$F = G = H = 1$	$K = 175.0$	$C_A = 800$
$N = L = M = 3$	$\epsilon_0 = 1.5\ 10^{-4}$	$G_A = 45.9$
	$n = 0.328$	

Table C.6: Ziegler's parameters identified with brick elements

C.3.2 Identification for the COQJ4 Shell Element

This section gives the material parameters identified by Bouffioux using an inverse modeling technique and simulations performed with shell (COQJ4) elements [BOU08b]. The parameters of both Armstrong-Frederick's and Ziegler's laws are given in Tables C.7 and C.8.

Yield surface coefficients	Swift's law parameters	Back-stress data
$F = 1.224$	$K = 111.0$	C_X=51.4
$G = 1.193$	$\epsilon_0 = 5.0\ 10^{-4}$	X_{sat}=46
$H = 0.8067$	$n = 0.266$	
$N = L = M = 4.06$		

Table C.7: Armstrong-Frederick's parameters identified with shell elements

Yield surface coefficients	Swift's law parameters	Back-stress data
$F = G = H = 1$	$K = 94.0$	$C_A = 3000$
$N = L = M = 3$	$\epsilon_0 = 8.3\ 10^{-5}$	$G_A = 56.9$
	$n = 0.266$	

Table C.8: Ziegler's parameters identified with shell elements

Bibliography

[ABA03] Abaqus. *User's Manual, version 6.4.* Hibbitt, Karlsson and Sorensen, Inc., Pawtucket, USA, 2003.

[AER09] R. Aerens, J. R. Duflou, P. Eyckens, and A. Van Bael. Force Prediction for Single Point Incremental Forming deduced from experimental and FEM observations. *International Journal of Machine Tools and Manufacture* (2009). To be published.

[ALL05] J. Allwood, N. E. Houghton, and K. P. Jackson. The Design of an Incremental Sheet Forming Machine. *Advanced Materials Research 6–8* (2005), pp. 471–478.

[ALL07] J. M. Allwood, D. R. Shouler, and A. E. Tekkaya. The Increased Forming Limits of Incremental Sheet Forming Processes. *Key Engineering Materials 344* (2007), pp. 621–628.

[ALV05] R. J. Alves de Sousa, R. P. R. Cardoso, R. A. Fontes Valente, J.-W. Yoon, J. J. Gracio, and R. M. Natal Jorge. A New One-Point Quadrature Enhanced Assumed Strain (EAS) Solid-Shell Element With Multiple Integration Points Along Thickness: Part I—Geometrically Linear Applications. *International Journal for Numerical Methods in Engineering 62* (2005), pp. 952–977. doi: 10.1002/nme.1226.

[ALV06] R. J. Alves de Sousa, R. P. R. Cardoso, R. A. Fontes Valente, J.-W. Yoon, J. J. Gracio, and R. M. Natal Jorge. A New One-Point Quadrature Enhanced Assumed Strain (EAS) Solid-Shell Element With Multiple Integration Points Along Thickness—Part II: Nonlinear Applications. *International Journal for Numerical Methods in Engineering 67* (2006), pp. 160–188. doi: 10.1002/nme.1609.

[AMB04a] G. Ambrogio, I. Costantino, L. De Napoli, L. Filice, L. Fratini, and M. Muzzupappa. Influence of some relevant process parameters on the dimensional accuracy in incremental forming: a numerical and experimental investigation. *Journal of Ma-*

terials Processing Technology 153–154 (2004), pp. 501–507. doi: 10.1016/j.jmatprotec.2004.04.139.

[AMB04b] G. AMBROGIO, L. FILICE, L. FRATINI, AND F. MICARI. Process Mechanics Analysis in Single Point Incremental Forming. In S. Ghosh, J. C. Castro, and J. K. Lee, editors, *Proc. of the 8th Numiform conference*, vol. 712 of *AIP Conf. Proc.*, pp. 922–927, Columbus, OH, USA, 2004. The Ohio State University. doi:10.1063/1.1766645.

[AMB05a] G. AMBROGIO, L. DE NAPOLI, L. FILICE, F. GAGLIARDI, AND M. MUZZUPAPPA. Application of incremental forming process for high customized medical product manufacturing. *Journal of Materials Processing Technology 162–163* (2005), pp. 156–162. doi: 10.1016/j.jmatprotec.2005.02.148.

[AMB05b] G. AMBROGIO, L. FILICE, F. GAGLIARDI, AND F. MICARI. Sheet Thinning Prediction in Single Point Incremental Forming. *Advanced Materials Research 6–8* (2005), pp. 479–486.

[AMB06] G. AMBROGIO, L. FILICE, AND F. MICARI. A force measuring based strategy for failure prevention in incremental forming. *Journal of Materials Processing Technology 177*, 1-3 (2006), pp. 413–416. doi: 10.1016/j.jmatprotec.2006.04.076.

[ARM66] P. J. ARMSTRONG AND C. O. FREDERICK. *A mathematical representation of the multiaxial Bauschinger effect.* C.e.g.b. report rd/b/n731, Berkeley Nuclear Laboratories, Berkeley, UK, 1966.

[ATT08] A. ATTANASIO, A. FIORENTINO, L. MAZZONI, E. CERETTI, AND C. GIARDINI. Design of an Equipment for Forces Measurement in TPIF Process. In *Proc. of the 9th ICTP conference*, pp. 1783–1788, Gyeongju, Korea, 2008. Tue Korean Society for Technology of Plasticity.

[BAM04] M. BAMBACH, G. HIRT, AND J. AMES. Modeling of Optimization Strategies in the Incremental CNC Sheet Metal Forming Process. In S. Ghosh, J. C. Castro, and J. K. Lee, editors, *Proc. of the 8th Numiform conference*, vol. 712 of *AIP Conf. Proc.*, pp. 1969–1974, Columbus, OH, USA, 2004. The Ohio State University. doi: 10.1063/1.1766822.

[BAM05a] M. BAMBACH, G. HIRT, AND J. AMES. Quantitative Validation of FEM Simulations for Incremental Sheet Forming Using Optical Deformation Measurement. *Advanced Materials Research 6-8* (2005), pp. 509–516.

[BAM05b] M. BAMBACH, J. AMES, M. AZAOUZI, L. CAMPAGNE, G. HIRT, AND J. L. BATOZ. Initial experimental and numerical investigations into a class of new strategies for single point incremental sheet forming (SPIF). In D. Banabic, editor, *Proc. of the 8th Esaform conference*, vol. 2, pp. 671–674, Bucharest, 2005.

[BAM05c] M. BAMBACH AND G. HIRT. Performance Assessment of Element Formulations and Constitutive Laws for the Simulation of Incremental Sheet Forming(ISF). In E. Oñate and D. R. J. Owen, editors, *Proc. of the 8th Complas conference*, Barcelona, 2005. CIMNE.

[BAM07] M. BAMBACH, M. TODOROVA, AND G. HIRT. Experimental and Numerical Analysis of Forming Limits in CNC Incremental Sheet Forming. *Key Engineering Materials 344* (2007), pp. 511–518.

[BAN00] D. BANABIC, H.-J. BUNGE, K. PÖHLANDT, AND A. E. TEKKAYA. *Formability of Metallic Materials – Plastic Anisotropy, Formability Testing, Forming Limits*. Engineering Materials. Springer, 2000.

[BATH76] K.-J. BATHE AND E. L. WILSON. *Numerical Methods in Finite Element Analysis*. Prentice–Hall, Inc., Englewood Cliffs, New Jersey, 1976.

[BATO82] J.-L. BATOZ AND M. B. TAHAR. Evaluation of a New Quadrilateral Thin Plate Bending Element. *International Journal for Numerical Methods in Engineering 18*, 11 (1982), pp. 1655–1677. doi: 10.1002/nme.1620181106.

[BATO92] J.-L. BATOZ AND G. DHATT. *Modélisation des structures par elements finis : Coques*, vol. 3. Hermes, Paris, 1992.

[BEL91] T. BELYTSCHKO AND L. P. BINDEMAN. Assumed Strain Stabilization of the 4-Node Quadrilateral With 1-Point Quadrature for Nonlinear Problems. *Computer Methods in Applied Mechanics and Engineering 88*, 3 (1991), pp. 311–340. doi:10.1016/0045-7825(91)90093-L.

[BOL05] O. BOLOGA, V. OLEKSIK, AND G. RACZ. Experimental research for determining the forces on incremental sheet forming process. In D. Banabic, editor, *Proc. of the 8th Esaform conference*, vol. 1, pp. 317–320, Bucharest, 2005.

[BOU07] C. BOUFFIOUX, C. HENRARD, J. GU, J. R. DUFLOU, A. M. HABRAKEN, AND H. SOL. Development of an Inverse Method for Identification of Materials Parameters in the Single Point Incremental Forming Process. In M. Tisza, editor, *Proc. of the IDDRG 2007 conference*, Győr, Hungary, 2007.

[BOU08a] C. BOUFFIOUX, P. EYCKENS, C. HENRARD, R. AERENS, A. VAN BAEL, H. SOL, J. DUFLOU, AND A. HABRAKEN. Identification of material parameters to predict Single Point Incremental Forming forces. *International Journal of Material Forming* (2008). doi:10.1007/s12289-008-0183-0.

[BOU08b] C. BOUFFIOUX, C. HENRARD, P. EYCKENS, R. AERENS, A. VAN BAEL, H. SOL, J. R. DUFLOU, AND A. M. HABRAKEN. Comparison of the Tests Chosen for Material Parameters Identification

to Predict Single Point Incremental Forming Forces. In *Proc. of the IDDRG 2008 conference*, vol. 1, pp. 133–144, Olofström, Sweden, 2008.

[BUT30] S. BUTTERWORTH. On the Theory of Filter Amplifiers. *Experimental Wireless and the Radio Engineer 7* (1930), pp. 536–541.

[CASO05] S. CASOTTO, F. PASCON, A. M. HABRAKEN, AND S. BRUSCHI. Thermo-Mechanical-Metallurgical Model to Predict Geometrical Distortions of Rings During Cooling Phase After Ring Rolling Operations. *International Journal of Machine Tools and Manufacture 45*, 6 (2005), pp. 657–664. doi:10.1016/j.ijmachtools.2004.10.007.

[CAST03] S. CASTAGNE, A. M. HABRAKEN, AND S. CESCOTTO. Application of a Damage Model to an Aluminum Alloy. *International Journal of Damage Mechanics 12*, 1 (2003), pp. 5–30. doi: 10.1177/1056789503012001001.

[CAST04] S. CASTAGNE, F. PASCON, G. BLES, AND A. M. HABRAKEN. Developments in Finite Element Simulations of Continuous Casting. *Journal de Physique IV 120* (2004), pp. 447–455. doi: 10.1051/jp4:2004120051.

[CES85] S. CESCOTTO AND H. GROBER. Calibration and Application of an Elastic Viscoplastic Constitutive Equation for Steels in Hot-Rolling Conditions. *Engineering Computations 2*, 2 (1985), pp. 101–106. doi: 10.1108/eb023607.

[CES93] S. CESCOTTO AND R. CHARLIER. Frictional Contact Finite Element Based on Mixed Variational Principles. *International Journal of Numerical Methods in Engineering 36*, 10 (1993), pp. 1681–1701. doi:10.1002/nme.1620361005.

[CHA87] R. CHARLIER. *Approche unifiée de quelques problèmes non linéaires de mécanique des milieux continus par la méthode des éléments finis (grandes déformations des métaux et des sols, contact unilatéral de solides, conduction thermique et écoulements en milieu poreux)*. PhD thesis, University of Liège, Belgium, 1987. In French.

[DEJ08a] S. DEJARDIN, J. C. GELIN, AND S. THIBAU. Finite element analysis and experimental investigations for improving knowledge in single point incremental sheet forming process. In *Proc. of the 9th ICTP conference*, pp. 2002–2007, Gyeongju, Korea, 2008. Tue Korean Society for Technology of Plasticity.

[DEJ08b] S. DEJARDIN, J. C. GELIN, AND S. THIBAUD. Finite element analysis and experimental investigations for precision im-proving in single point incremental sheet forming. In *Proc. of the 12th Metal Forming conference*, vol. 1 of *Steel Research International*, pp. 616–623, Kraków, Poland, 2008.

[DEM03] P. DE MONTLEAU. *Isotropic and Kinematic Hardening - law HILL3D_KI*. Internal report, University of Liège, 2003.

[DET04] W. DETTMER AND S. REESE. On the theoretical and numerical modelling of Armstrong-Frederick kinematic hardening in the finite strain regime. *Computer Methods in Applied Mechanics and Engineering 193*, 1-2 (2004), pp. 87–116. doi:10.1016/j.cma.2003.09.005.

[DOG00] I. DOGHRI. *Mechanics of Deformable Solids, Linear and Nonlinear, Analytical and Computational Aspects*. Springer, 2000.

[DUC02] L. DUCHÊNE, A. GODINAS, S. CESCOTTO, AND A. M. HABRAKEN. Texture Evolution During Deep-Drawing Processes. *Journal of Materials Processing Technology 125–126* (2002), pp. 110–118. doi: 10.1016/S0924-0136(02)00331-X.

[DUC03] L. DUCHÊNE. *FEM Study of Metal Sheets With a Texture Based, Local Description of the Yield Locus*. PhD thesis, University of Liège, Belgium, 2003.

[DUC05a] L. DUCHÊNE, P. DE MONTLEAU, F. EL HOUDAIGUI, S. BOUVIER, AND A. M. HABRAKEN. Analysis of Texture Evolution and Hardening Behavior During Deep Drawing With an Improved Mixed Type FEM Element. In L. M. Smith, F. Pourboghrat, J.-W. Yoon, and T. B. Stoughton, editors, *Proc. of the 6th Numisheet Conference*, vol. 778, pp. 409–414, Detroit, MI, USA, 2005. AIP Conf. Proc.

[DUC05b] L. DUCHÊNE AND A. M. HABRAKEN. Analysis of the Sensitivity of FEM Predictions to Numerical Parameters in Deep Drawing Simulations. *European Journal of Mechanics A/Solids 24*, 4 (2005), pp. 614–629. doi:10.1016/j.euromechsol.2005.04.007.

[DUC06] L. DUCHÊNE, P. DE MONTLEAU, AND A. M. HABRAKEN. Development and Performance Assessment of an Improved 8-Node Mixed Type FEM Element. In *Proc. of the 9th Esaform conference*, pp. 135–138, Glasgow, UK, 2006.

[DUC07] L. DUCHÊNE, F. EL HOUDAIGUI, AND A. M. HABRAKEN. Length Changes and Texture Prediction During Free End Torsion Test of Copper Bars With FEM and Remeshing Techniques. *International Journal of Plasticity 23* (2007), pp. 1417–1438. doi: 10.1016/j.ijplas.2007.01.008.

[DUC08] L. DUCHÊNE, T. LELOTTE, P. FLORES, S. BOUVIER, AND A. M. HABRAKEN. Rotation of Axes for Anisotropic Metal in FEM Simulations. *International Journal of Plasticity 24* (2008), pp. 397–427. doi:10.1016/j.ijplas.2007.03.015.

[DUF03] J. R. DUFLOU, H. SOL, A. VAN BAEL, AND A. M. HABRAKEN. *Description of the SeMPeR projet (Sheet Metal oriented Prototyping and Rapid manufacturing), SBO-project financed by the IWT institute*, 2003.

[DUF05a] J. R. DUFLOU, B. LAUWERS, J. VERBERT, Y. TUNCKOL, AND H. DE BAERDEMAKER. Achievable Accuracy in Single Point Incremental Forming: Case Studies. In D. Banabic, editor, *Proc. of the 8th Esaform conference*, vol. 2, pp. 675–678, Bucharest, 2005.

[DUF05b] J. R. DUFLOU, A. SZEKERES, AND P. VANHERCK. Force Measurements for Single Point Incremental Forming: An Experimental Study. *Advanced Materials Research 6-8* (2005), pp. 441–448.

[DUF06a] J. R. DUFLOU, B. CALLEBAUT, AND J. VERBERT. *Asymmetric Incremental Sheet Forming System*. Patent WO/2006/110962, 2006.

[DUF06b] J. R. DUFLOU AND Y. TUNÇKOL. Force modelling for single point incremental forming. In *Proc. of the 9th Esaform conference*, pp. 287–290, Glasgow, UK, 2006.

[DUF07a] J. R. DUFLOU, B. CALLEBAUT, J. VERBERT, AND H. DE BAERDEMAEKER. Laser Assisted Incremental Forming: Formability and Accuracy Improvement. *CIRP Annals – Manufacturing Technology 56*, 1 (2007), pp. 273–276. doi:10.1016/j.cirp.2007.05.063.

[DUF07b] J. R. DUFLOU, Y. TUNÇKOL, AND R. AERENS. Force Analysis for Single Point Incremental Forming. *Key Engineering Materials 344* (2007), pp. 543–550.

[DUF08a] J. R. DUFLOU, B. CALLEBAUT, J. VERBERT, AND H. DE BAERDEMAEKER. Improved SPIF performance through dynamic local heating. *International Journal of Machine Tools and Manufacture 48*, 5 (2008), pp. 543–549. doi:10.1016/j.ijmachtools.2007.08.010.

[DUF08b] J. R. DUFLOU, J. VERBERT, B. BELKASSEM, J. GU, H. SOL, C. HENRARD, AND A. M. HABRAKEN. Process Window Enhancement for Single Point Incremental Forming through Multi-Step Toolpaths. *CIRP Annals – Manufacturing Technology 57* (2008), pp. 253–256. doi:10.1016/j.cirp.2008.03.030.

[EMM07] W. C. EMMENS AND A. H. VAN DEN BOOGAARD. Strain in Shear, and Material Behaviour in Incremental Forming. *Key Engineering Materials 344* (2007), pp. 519–526.

[EMM08a] W. C. EMMENS AND A. H. VAN DEN BOOGAARD. Incremental Forming Studied by Tensile Tests with Bending. In *Proc. of the 9th ICTP conference*, pp. 508–513, Gyeongju, Korea, 2008. Tue Korean Society for Technology of Plasticity.

[EMM08b] W. C. EMMENS AND A. H. VAN DEN BOOGAARD. Tensile tests with bending: a mechanism for incremental forming. *International Journal of Material Forming* (2008). doi:10.1007/s12289-008-0185-y.

[EYC07] P. EYCKENS, S. HE, A. VAN BAEL, P. VAN HOUTTE, AND J. DUFLOU. Forming Limit Predictions for the Serrated Strain Paths in

Single Point Incremental Sheet Forming. In *Proc. of the 9th Numiform conference*, vol. 908 of *AIP Conf. Proc.*, pp. 141–146, 2007. doi: 10.1063/1.2740802.

[EYC08a] P. EYCKENS, A. VAN BAEL, R. AERENS, J. R. DUFLOU, AND P. VAN HOUTTE. Small-scale Finite Element Modelling of the Plastic Deformation Zone in the Incremental Forming Process. *International Journal of Material Forming* (2008). doi:10.1007/s12289-008-0186-x.

[EYC08b] P. EYCKENS, A. VAN BAEL, AND P. VAN HOUTTE. An Extended Marciniak-Kuczynski Forming Limit Model to Assess the Influence of Through-Thickness Shear on Formability. In P. Hora, editor, *Proc. of the 7th Numisheet conference*, vol. A, pp. 193–198, Interlaken, Switzerland, 2008.

[FIL02] L. FILICE, L. FRATINI, AND F. MICARI. Analysis of Material Formability in Incremental Forming. *CIRP Annals – Manufacturing Technology 51*, 1 (2002), pp. 199–202. doi:10.1016/S0007-8506(07)61499-1.

[FLO05] P. FLORES, L. DUCHÊNE, T. LELOTTE, C. BOUFFIOUX, F. EL HOUDAIGUI, A. VAN BAEL, S. HE, J. R. DUFLOU, AND A. M. HABRAKEN. Model Identification and FE Simulations: Effect of Different Yield Loci and Hardening Laws. In L. M. Smith, F. Pourboghrat, J.-W. Yoon, and T. B. Stoughton, editors, *Proc. of the 6th Numisheet Conference*, vol. 778, pp. 371–381, Detroit, MI, USA, 2005. AIP Conf. Proc. doi:10.1063/1.2011248.

[FLO06] P. FLORES. *Development of Experimental Equipment and Identification Procedures for Sheet Metal Constitutive Laws.* PhD thesis, University of Liège, Belgium, 2006.

[FLO07] P. FLORES, L. DUCHÊNE, C. BOUFFIOUX, T. LELOTTE, C. HENRARD, N. PERNIN, A. VAN BAEL, S. HE, J. DUFLOU, AND A. M. HABRAKEN. Model Identification and FE Simulations: Effect of Different Yield Loci and Hardening Laws in Sheet Forming. *International Journal of Plasticity 23* (2007), pp. 420–449. doi: 10.1016/j.ijplas.2006.05.006.

[FRA08] V. FRANZEN, L. KWIATKOWSKI, G. SEBASTIANI, R. SHANKAR, A. TEKKAYA, AND M. KLEINER. Dyna-Die: Towards Full Kinematic Incremental Forming. *International Journal of Material Forming* (2008). doi:10.1007/s12289-008-0187-9.

[GOM] Website of GOM, manufacturer of the optical measuring system Aramis: http://www.gom.com. Last visited in September 2008.

[HAB90] A. M. HABRAKEN AND S. CESCOTTO. An Automatic Remeshing Technique for Finite Element Simulation of Forging Processes. *International Journal for Numerical Methods in Engineering 30*, 8 (1990), pp. 1503–1525. doi:10.1002/nme.1620300811.

[HAB92] A. M. HABRAKEN AND M. BOURDOUXHE. Coupled Thermo-Mechanical-Metallurgical Analysis During the Cooling of Steel Pieces. *European Journal of Mechanics A/Solids 11*, 3 (1992), pp. 381–402.

[HAB98a] A. M. HABRAKEN AND S. CESCOTTO. Contact Between Deformable Solids, the Fully Coupled Approach. *Mathematical and Computer Modelling 28*, 4–8 (1998), pp. 153–169. doi:10.1016/S0895-7177(98)00115-0.

[HAB98b] A. M. HABRAKEN, J. F. CHARLES, J. WEGRIA, AND S. CESCOTTO. Dynamic Recrystallization During Zinc Rolling. *International Journal of Forming Processes 1*, 1 (1998).

[HAB04a] A. M. HABRAKEN. Modelling the Plastic Anisotropy of Metals. *Archives of Computational Methods in Engineering 11*, 1 (2004), pp. 3–96. doi:10.1007/BF02736210.

[HAB04b] A. M. HABRAKEN AND L. DUCHÊNE. Anisotropic Elasto-Plastic Finite Element Analysis Using a Stress-Strain Interpolation Method Based on a Polycrystalline Model. *International Journal of Plasticity 20* (2004), pp. 1525–1560. doi:10.1016/j.ijplas.2003.11.006.

[HAD08] A. HADOUSH AND A. H. VAN DEN BOOGAARD. Time reduction in implicit single point incremental sheet forming simulation by refinement - derefinement. *International Journal of Material Forming* (2008). doi:10.1007/s12289-008-0188-8.

[HAG03] E. HAGAN AND J. JESWIET. A review of conventional and modern single point sheet metal forming methods. In *Proc. of the IMechE, Journal of Engineering Manufacture - Part B*, vol. 217, pp. 213–225, 2003. doi:10.1243/095440503321148858.

[HE05a] S. HE, A. VAN BAEL, P. VAN HOUTTE, A. SZEKERES, J. R. DUFLOU, C. HENRARD, AND A. M. HABRAKEN. Finite Element Modeling of Incremental Forming of Aluminum Sheets. *Advanced Materials Research 6-8* (2005), pp. 525–532.

[HE05b] S. HE, A. VAN BAEL, P. VAN HOUTTE, Y. TUNCKOL, J. R. DOUFLOU, C. HENRARD, C. BOUFFIOUX, AND A. M. HABRAKEN. Effect of FEM choices in the modelling of incremental forming of aluminium sheets. In D. Banabic, editor, *Proc. of the 8th Esaform conference*, vol. 2, pp. 711–714, Bucharest, 2005.

[HEN05a] C. HENRARD. *Development of a Contact Model Adapted to Incremental Forming.* Master's thesis, University of Liège, 2005. Supervisor: Dr. Anne Marie Habraken.

[HEN05b] C. HENRARD. *Moving Spherical Tool: Programming Report.* Internal report 2008_4, University of Liège, 2005.

[HEN05c] C. HENRARD, A. M. HABRAKEN, A. SZEKERES, J. R. DUFLOU, S. HE, A. VAN BAEL, AND P. VAN HOUTTE. Comparison of FEM Simulations for the Incremental Forming Process. *Advanced Materials Research 6-8* (2005), pp. 533–542.

[HEN07a] C. HENRARD, C. BOUFFIOUX, L. DUCHÊNE, J. R. DUFLOU, AND A. M. HABRAKEN. Validation of a New Finite Element for Incremental Forming Simulation Using a Dynamic Explicit Approach. *Key Engineering Materials 344* (2007), pp. 495–502.

[HEN07b] C. HENRARD. *Computation of the Consistent and Lumped Mass Matrices of the COQJ4 Shell Element.* Internal report 2008_5, University of Liège, 2007.

[HEN07c] C. HENRARD. *State Variables and Local Axis of the COQJ4 element.* Internal report 2008_3, University of Liège, 2007.

[HIL50] R. HILL. *The Mathematical Theory of Plasticity.* Oxford Engineering Science Series. Oxford at the Clarendon Press, 1950.

[HIR02] G. HIRT, S. JUNK, AND N. WITULSKI. Incremental Sheet Forming: Quality Evaluation and Process Simulation. In *Proc. of the 7th ICTP conference*, vol. 1, pp. 925–930, 2002.

[HIR04a] G. HIRT, J. AMES, M. BAMBACH, AND R. KOPP. Forming strategies and Process Modelling for CNC Incremental Sheet Forming. *CIRP Annals – Manufacturing Technology 53* (2004), pp. 203–206. doi:10.1016/S0007-8506(07)60679-9.

[HIR04b] G. HIRT. Tools and Equipment used in Incremental Forming. In *1st Incremental Forming Workshop.* University of Saarbrucken, 2004.

[HIR05] G. HIRT, J. AMES, AND M. BAMBACH. A new forming strategy to realize parts designed for deep-drawing by incremental CNC sheet forming. *Steel Research International 76*, 2-3 (2005), pp. 160–166.

[ISE92] H. ISEKI, K. KATO, AND S. SAKAMOTO. Flexible and Incremental Sheet Metal Bulging Using a Path-Controlled Spherical Roller. *Transactions of the JSME C 58*, 554 (1992), pp. 3147–3155.

[ISE93] H. ISEKI, K. KATO, AND S. SAKAMOTO. Forming limit of flexible and incremental sheet metal bulging using a spherical roller. In *Proc. of the 4th ICTP conference*, pp. 1635–1640, 1993.

[ISE94] H. ISEKI. Forming Limit of Incremental Sheet Metal Stretch Forming Using Spherical Rollers. *Journal of the Japan Society for Technology of Plasticity 35*, 406 (1994), pp. 1337–1342.

[ISE00] H. ISEKI. An experimental and theoretical study of a forming limit curve in incremental forming of sheet metal using spherical roller. In *Proc. of the 8th Metal Forming conference*, pp. 557–562, 2000.

[ISE01] H. ISEKI. An approximate deformation analysis and FEM analysis for the incremental bulging of sheet metal using a spherical roller. *Journal of Materials Processing Technology 111*, 1-3 (2001), pp. 150–154. doi:10.1016/S0924-0136(01)00500-3.

[ISE02] H. ISEKI AND T. NAGANAWA. Vertical wall surface forming of rectangular shell using multistage incremental forming with spherical and cylindrical rollers. *Journal of Materials Processing Technology 130-131* (2002), pp. 675–679. doi:10.1016/S0924-0136(02)00735-5.

[ISE08] H. ISEKI. A Strain Distribution Analysis in Multistage Incremental Forming of Cylindrical Cup with Flange. In *Proc. of the 9th ICTP conference*, pp. 1765–1770, Gyeongju, Korea, 2008. Tue Korean Society for Technology of Plasticity.

[JAA89] S. JAAMEI, F. FREY, AND P. JETTEUR. Nonlinear Thin Shell Finite Element With Six Degrees of Freedom per Node. *Computer Methods in Applied Mechanics and Engineering 75*, 1–3 (1989), pp. 251–266. doi:10.1016/0045-7825(89)90028-5.

[JES01a] J. JESWIET. Incremental Single Point Forming with a Tool Post. In *Proc. of the 9th International Conference on Sheet Metal*, pp. 37–42, Leuven, 2001.

[JES01b] J. JESWIET AND E. HAGAN. Rapid Proto-Typing of a Headlight with Sheet Metal. In *Proc. of the 9th International Conference on Sheet Metal*, pp. 165–170, Leuven, 2001.

[JES05a] J. JESWIET, J. R. DUFLOU, AND A. SZEKERES. Forces in Single Point and Two Point Incremental Forming. *Advanced Materials Research 6-8* (2005), pp. 449–456.

[JES05b] J. JESWIET, J. R. DUFLOU, A. SZEKERES, AND P. LEFEBVRE. Custom Manufacture a Solar Cooker - A Case Study. *Advanced Materials Research 6-8* (2005), pp. 487–492.

[JES05c] J. JESWIET, F. MICARI, G. HIRT, A. N. BRAMLEY, J. R. DUFLOU, AND J. ALLWOOD. Asymmetric Single Point Incremental Forming of Sheet Metal. *CIRP Annals – Manufacturing Technology 54*, 2 (2005), pp. 88–114. doi:10.1016/S0007-8506(07)60021-3.

[JES05d] J. JESWIET, D. YOUNG, AND M. HAM. Non-traditional Forming Limit Diagrams for Incremental Forming. *Advanced Materials Research 6-8* (2005), pp. 409–416.

[JET85] P. JETTEUR. *Nonlinear Shell Elements Based on Marguerre Theory.* Internal report 85/5, IREM, 1985.

[JET86a] P. JETTEUR. *A Shallow Shell Element With In-Plane Rotational Degrees of Freedom.* Internal report 86/3, IREM, 1986.

[JET86b] P. JETTEUR. Implicit Integration Algorithm for Elastoplasticity in Plane Stress Analysis. *Engineering Computations 3*, 3 (1986), pp. 251–253. doi:10.1108/eb023664.

[JET86c] P. JETTEUR AND F. FREY. A Four Node Marguerre Element for Non-Linear Shell Analysis. *Engineering Computations 3*, 4 (1986), pp. 276–282. doi:10.1108/eb023667.

[JET87a] P. JETTEUR. *Improvement and Large Rotations of the JET Shell Element in Nonlinear Analysis.* Internal report 87/4, IREM, 1987.

[JET87b] P. JETTEUR. *Improvement of the Quadrangular JET Shell Element for a Particular Class of Shell Problems.* Internal report 87/1, IREM, 1987.

[KAL95] S. KALPAKJIAN. *Manufacturing Engineering and Technology.* Addison – Wesley, 3rd edition, 1995.

[KHA95] A. S. KHAN AND S. HUANG. *Continuum Theory of Plasticity.* Wiley-Interscience, 1995.

[KIM00] T. J. KIM AND D. Y. YANG. Improvement of formability for the incremental sheet metal forming process. *International Journal of Mechanical Sciences 42* (2000), pp. 1271–1286. doi:10.1016/S0020-7403(99)00047-8.

[KIM08] S. W. KIM, Y. S. LEE, Y. N. KWON, AND J. H. LEE. Study on warm incremental formability of AZ31 alloy sheet. In *Proc. of the 12th Metal Forming conference*, vol. 1 of *Steel Research International*, pp. 691–698, Kraków, Poland, 2008.

[KISTL] Website of Kistler: http://www.kistler.com. Last visited in September 2008.

[KIT93] K. KITAZAWA. Incremental Sheet Metal Stretch-Expanding with CNC Machine Tools. In *Proc. of the 4th ICTP conference*, 1993.

[KIT97] K. KITAZAWA AND A. NAKAJIMA. Method for producing aluminum cylindrical shell having uniform wall-thickness by CNC incremental forming process. *Journal of Japan Institute of Light Metals 47* (1997), pp. 378–384. In Japanese.

[KIT99] K. KITAZAWA AND A. NAKAJIMA. Cylindrical Incremental Drawing of Sheet Metals by CNC Incremental Forming Process. In *Proc. of the 6th ITCP conference*, pp. 1495–1500, 1999.

[KOB61] S. KOBAYASHI, I. K. HALL, AND E. G. THOMSEN. A Theory of Shear Spinning of Cones. *Trans. ASME, series B, J. Engineering for Industry 83* (1961), pp. 485–495.

[LAM05] L. LAMMINEN. Incremental Sheet Forming with an Industrial Robot - Forming Limits and Their Effect on Component Design. *Advanced Materials Research 6-8* (2005), pp. 457–464.

[LAPACK] *LAPACK, developed by the University of Colorado at Boulder.* http://www.cs.colorado.edu/~jessup/lapack/.

[LEA01] D. LEACH, A. J. GREEN, AND A. N. BRAMLEY. A New Incremental Sheet Forming Process for Small Batch and Prototypes Parts. In *Proc. of the 9th International Conference on Sheet Metal*, pp. 211–218, Leuven, 2001.

[LEQ08] C. LEQUESNE, C. HENRARD, C. BOUFFIOUX, J. DUFLOU, AND A. HABRAKEN. Adaptive Remeshing for Incremental Forming Simulations. In P. Hora, editor, *Proc. of the 7th Numisheet conference*, vol. A, pp. 399–403, Interlaken, Switzerland, 2008.

[LES67] E. LESZAK. *Apparatus and Process for Incremental Dieless Forming.* Patent US3342051A1, 1967.

[LI95] K. LI. *Contribution to the Finite Element Simulation of Three-Dimensional Sheet Metal Forming.* PhD thesis, University of Liège, Belgium, 1995.

[LI96] K. LI AND S. CESCOTTO. Finite Element Method for Gradient Plasticity at Large Strains. *International Journal for Numerical Methods in Engineering 39*, 4 (1996), pp. 619–633. doi:10.1002/(SICI)1097-0207(19960229)39:4<619::AID-NME873>3.0.CO;2-1.

[LI97] K. LI AND S. CESCOTTO. An 8-Node Brick Element With Mixed Formulation for Large Deformation Analyses. *Computer Methods in Applied Mechanics and Engineering 141*, 1–2 (1997), pp. 157–204. doi:10.1016/S0045-7825(96)01071-7.

[LIMESS] Website of Limess: http://www.limess.de. Last visited in September 2008.

[MAI07] E. MAIDAGAN, J. ZETTLER, M. BAMBACH, P. P. RODRÍGUEZ, AND G. HIRT. A new incremental sheet forming process based on a flexible supporting die system. *Key Engineering Materials 344* (2007), pp. 607–614.

[MAR08] P. A. F. MARTINS, N. BAY, M. SKJOEDT, AND M. B. SILVA. Theory of single point incremental forming. *CIRP Annals - Manufacturing Technology 57*, 1 (2008), pp. 247–252. doi: 10.1016/j.cirp.2008.03.047.

[MAT94] S. MATSUBARA. Incremental Backward Bulge Forming of a Sheet Metal with a Hemispherical Head Tool. In *Journal of the Japan Society for Technology of Plasticity*, vol. 35/406, pp. 119–123, 1994.

[MAT01] S. MATSUBARA. A computer numerically controlled dieless incremental forming of a sheet metal. In *Proc. of the IMechE, Journal of Engineering Manufacture - Part B*, vol. 215, pp. 959–966, 2001. doi: 10.1243/0954405011518863.

[MEI05] H. MEIER, O. DEWALD, AND J. ZHANG. A New Robot-Based Sheet Metal Forming Process. *Advanced Materials Research 6-8* (2005), pp. 465–470.

[MEI07] H. MEIER, V. SMUKALA, O. DEWALD, AND J. ZHANG. Two Point Incremental Forming with Two Moving Forming Tools. *Key Engineering Materials 344* (2007), pp. 599–605.

[METRIS] Website of Metris: http://www.metris.com. Last visited in September 2008.

[MEY05] C. MEYER, L. W.AND GAHLERT AND F. HAHN. Influence of an incremental deformation on material behaviour and forming limit of Aluminium Al99,5 and QT-steel 42CrMo4. *Advanced Materials Research 6–8* (2005), pp. 417–424.

[MOS99] P. MOSBAH, S. SHIMA, A. M. HABRAKEN, AND R. CHARLIER. Numerical Simulation of Compacting Process of a Multi-Steeped Part with Comparisons to Experiments. *Journal of the Japan Society of Powder and Powder Metallurgy 46* (1999), pp. 696–704.

[PAR03] J.-J. PARK AND Y.-H. KIM. Fundamental studies on the incremental sheet metal forming technique. *Journal of Materials Processing Technology 140* (2003), pp. 447–453. doi:10.1016/S0924-0136(03)00768-4.

[POW92] N. POWELL AND C. ANDREW. Incremental forming of flanged sheet metal components without dedicated dies. In *Proc. of the IMechE, Journal of Engineering Manufacture - Part B*, vol. 206, pp. 41–47, 1992.

[PRA55] W. PRAGER. The theory of plasticity: A survey of recent achievements. *Proc. of the Institution of Mechanical Engineers 169* (1955), p. 41.

[QIN05] Q. QIN, E. S. MASUKU, A. N. BRAMLEY, A. R. MILEHAM, AND G. W. OWEN. Incremental Sheet Forming - Simulation And Accuracy. In P. F. Bariani, editor, *Proc. of the 8th ICTP conference*, Advanced Technology of Plasticity, p. 343, 2005.

[ROB08] C. ROBERT, P. DAL SANTO, A. DELAMÉZIÈRE, A. POTIRON, AND J.-L. BATOZ. On some computational aspects for incremental sheet metal forming simulations. *International Journal of Material Forming* (2008). doi:10.1007/s12289-008-0155-4.

[SAW99] T. SAWADA, S. MATSUBARA, M. SAKAMOTO, AND G. FUKUHARA. Deformation Analysis for Stretch Forming of Sheet Metal With CNC Machine Tools. In *Proc. of the 6th ICTP conference*, vol. 2, pp. 1501–1504, 1999.

[SCH04] T. SCHÄFER AND R. D. SCHRAFT. Incremental sheet metal forming by industrial robots using a hammering tool. In *10e Assises*

Européennes de Prototypage Rapide. Association Française de Prototypage Rapide, 2004.

[SHA08] R. SHANKAR. *Surface Reconstruction and Tool Path Strategies for incremental Sheet Metal Forming*. PhD thesis, Universität Dortmund, Germany, 2008.

[SHI01] M.-S. SHIM AND J.-J. PARK. The formability of aluminum sheet in incremental forming. *Journal of Materials Processing Technology 113*, 1-3 (2001), pp. 654–658. doi:10.1016/S0924-0136(01)00679-3.

[SIL08] M. B. SILVA, M. SKJOEDT, P. A. F. MARTINS, AND N. BAY. Revisiting the fundamentals of single point incremental forming by means of membrane analysis. *International Journal of Machine Tools and Manufacture 48*, 1 (2008), pp. 73–83. doi: 10.1016/j.ijmachtools.2007.07.004.

[SIM86] J. C. SIMO AND T. J. R. HUGHES. On the Variational Foundations of Assumed Strain Methods. *Journal of Applied Mechanics, ASME 53* (1986), pp. 51–54.

[SKJ08a] M. SKJOEDT, N. BAY, B. ENDELT, AND G. INGARAO. Multi Stage Strategies for Single Point Incremental Forming of a Cup. *International Journal of Material Forming* (2008). doi:10.1007/s12289-008-0156-3.

[SKJ08b] M. SKJOEDT, M. B. SILVA, P. A. F. MARTINS, AND N. BAY. Strain Paths and Fracture in Multi Stage Single Point Incremental Forming. In *Proc. of the 9th ICTP conference*, pp. 502–507, Gyeongju, Korea, 2008. The Korean Society for Technology of Plasticity.

[SMI02] S. W. SMITH. *Digital Signal Processing: A Practical Guide for Engineers and Scientists*. Newnes, 2002.

[SOR63] H. C. SORTAIS, S. KOBAYASHI, AND E. G. THOMSEN. Mechanics of Conventional Spinning. *Trans. ASME, series B, J. Engineering for Industry 85* (1963), pp. 346–350.

[TAL96] S. H. TALBERT AND B. AVITZUR. *Elementary Mechanics of Plastic Flow in Metal Forming*. John Wiley & Sons, 1996.

[TAN99] S. TANAKA, T. NAKAMURA, AND K. HAYAKAWA. Incremental Sheet Metal Forming Using Elastic Tools. In *Proc. of the 6th ICTP conference*, vol. 2, pp. 1477–1482, 1999.

[TEK07] A. TEKKAYA, R. SHANKAR, G. SEBASTIANI, W. HOMBERG, AND M. KLEINER. Surface reconstruction for incremental forming. *Production Engineering 1*, 1 (2007), pp. 71–78. doi:10.1007/s11740-007-0024-x.

[UDDEL] Website of Uddeholm (http://www.uddeholm.com), manufacturer of the Vanadis 23 tools. Last visited in September 2008.

[VBA07] A. VAN BAEL, P. EYCKENS, S. HE, P. VAN HOUTTE, C. BOUFFIOUX, C. HENRARD, A. M. HABRAKEN, AND J. R. DUFLOU. Forming Limit Predictions For Single-Point Incremental Sheet Metal Forming. In *Proc. of the 10th Esaform conference*, vol. 907 of *AIP Conf. Proc.*, pp. 309–314, 2007. doi:10.1063/1.2729530.

[VER08] J. VERBERT, B. BELKASSEM, C. HENRARD, A. HABRAKEN, J. GU, H. SOL, AND J. LAUWERS, B.AND DUFLOU. Multi-Step Toolpath Approach to Overcome Forming Limitations in Single Point Incremental Forming. *International Journal of Material Forming* (2008). doi:10.1007/s12289-008-0157-2.

[WAN04] J. WANG AND R. H. WAGONER. A New Hexahedral Solid Element for 3D FEM Simulation of Sheet Metal Forming. In S. Ghosh, J. C. Castro, and J. K. Lee, editors, *Proc. of the 8th Numiform conference*, vol. 712 of *AIP Conf. Proc.*, pp. 2181–2186, Columbus, OH, USA, 2004. The Ohio State University. doi:10.1063/1.1766858.

[WAT05] K. WATZEELS, Y. TUNCKOL, C. HENRARD, J. GU, H. SOL, J. R. DUFLOU, AND A. M. HABRAKEN. Experimental validation of the Finite Element Simulation of the First Stroke in Single Point Incremental Forming. In D. Banabic, editor, *Proc. of the 8th Esaform conference*, vol. 2, pp. 703–706, Bucharest, 2005.

[YOO03] S. J. YOON AND D. Y. YANG. Development of a Highly Flexible Incremental Roll Forming Process for the Manufacture of a Doubly Curved Sheet Metal. *CIRP Annals – Manufacturing Technology 52* (2003), pp. 201–204. doi:10.1016/S0007-8506(07)60565-4.

[YOU04] D. YOUNG AND J. JESWIET. Wall thickness variations in single-point incremental forming. In *Proc. of the IMechE, Journal of Engineering Manufacture - Part B*, vol. 218, pp. 1453–1459, 2004. doi:10.1243/0954405042418400.

[YOU05] D. YOUNG AND J. JESWIET. Forming Limit Diagrams for Single-Point Incremental Forming of Aluminium Sheet. In *Proc. of the IMechE, Journal of Engineering Manufacture - Part B*, vol. 219, pp. 359–364, 2005. doi:10.1243/095440505X32210.

[ZHU92] Y. Y. ZHU AND S. CESCOTTO. A Fully Coupled Elastoplastic Damage Modelling of Contact-Impact Between Two Deformable Bodies. In P. S. Bulson, editor, *Structures Under Shock and Impact II, Computational Mechanics Publications*, pp. 113–132, London, 1992. Thomas Telford.

[ZHU94] Y. Y. ZHU AND S. CESCOTTO. Transient Thermal and Thermomechanical Analysis by mixed FEM. *Computers & Structures 53*, 2 (1994), pp. 275–304. doi:10.1016/0045-7949(94)90202-X.

[ZHU95] Y. Y. ZHU AND S. CESCOTTO. A Fully Coupled Elasto-Visco-Plastic Damage Theory for Anisotropic Materials. *International Journal of*

Solids and Structures 32, 11 (1995), pp. 1607–1641. doi:10.1016/0020-7683(94)00195-3.

[ZIE59] H. ZIEGLER. A modification of Prager's hardening ruel. *Quarterly of Applied Mathematics 17* (1959), p. 55.

Index