THE UNIVERSITY OF QUEENSLAND

AUSTRALIA

Isogeometric shell analysis of incremental sheet forming

Jaro Hokkanen

B.Sc., M.Sc.

A thesis submitted for the degree of Doctor of Philosophy

at

The University of Queensland in 2019

School of Civil Engineering

Abstract

This thesis investigates the applicability and beneficiality of isogeometric analysis to the incremental sheet forming simulations. Incremental sheet forming (ISF) is a manufacturing process for sheet metals where the final shape is obtained incrementally using a blunt moving tool. ISF is a relatively slow process but does not need precise and expensive dies (which require a long lead time) in contrast to many other sheet forming processes. This makes it ideal for rapid prototyping, customized parts, and small-batch production.

However, the time-consuming nature of an incremental process in combination with strongly nonlinear material behavior makes obtaining accurate simulation results very challenging. The existing simulation technology is often slow or not sufficiently robust for complex ISF simulations. Therefore, isogeometric analysis is investigated in order to improve the performance of the simulations.

Isogeometric analysis (IGA) is a relatively new simulation method which uses spline-based interpolation in contrast to Lagrange polynomials often employed by conventional finite elements. This work discusses many important fundamental concepts related to IGA, namely, spline interpolation, Galerkin method, numerical interpolation, shell element formulations, and contact.

The most important contributions to the field are the introduction of multiple new isogeometric shell element formulations, demonstration of benefits of unclamped knot vectors, and evaluation of some recently proposed quadrature rules using a complex real-world application. However, these findings are not limited to sheet forming simulations but are beneficial for a vast variety of applications ranging from modeling of bioprosthetic heart valves to automobile crash simulations. Furthermore, some application-specific aspects for ISF simulations are discussed such as contact algorithms.

The implementation takes advantage of general-purpose computing on graphics processing units (GPGPU), which is found to be a very cost-efficient way to speed up the simulations. The simulations are observed to run significantly faster on a modern desktop PC in comparison to conventional CPU-based implementations. Finally, IGA is found to be an outstanding alternative for ISF simulations.

Declaration by author

This thesis is composed of my original work, and contains no material previously published or written by another person except where due reference has been made in the text. I have clearly stated the contribution by others to jointly-authored works that I have included in my thesis.

I have clearly stated the contribution of others to my thesis as a whole, including statistical assistance, survey design, data analysis, significant technical procedures, professional editorial advice, financial support, and any other original research work used or reported in my thesis. The content of my thesis is the result of work I have carried out since the commencement of my higher degree by research candidature and does not include a substantial part of work that has been submitted to qualify for the award of any other degree or diploma in any university or other tertiary institution. I have clearly stated which parts of my thesis, if any, have been submitted to qualify for another award.

I acknowledge that an electronic copy of my thesis must be lodged with the University Library and, subject to the policy and procedures of The University of Queensland, the thesis be made available for research and study in accordance with the Copyright Act 1968 unless a period of embargo has been approved by the Dean of the Graduate School.

I acknowledge that copyright of all material contained in my thesis resides with the copyright holder(s) of that material. Where appropriate I have obtained copyright permission from the copyright holder to reproduce material in this thesis and have sought permission from co-authors for any jointly authored works included in the thesis.

Publications included in this thesis

1. **Hokkanen, J.** and Pedroso, D. M. (2019). Isogeometric thickness stretchable shell: Efficient formulation for nonlinear dynamic problems. *International Journal for Numerical Methods in Engineering*, 119(2):105-127.

Submitted manuscripts included in this thesis

1. **Hokkanen, J.** and Pedroso, D. M. (2019). Quadrature rules for isogeometric shell formulations: study using a real-world application about metal forming. Submitted to *Computer Methods in Applied Mechanics and Engineering* on 25th March 2019.

Other publications during candidature

Conference proceedings

1. **Hokkanen, J.**, Pedroso, D. M., Elford, M. C., Stephan, A. J. E., and Zhang, Y. (2018). Efficient isogeometric shell element with through-thickness stretch: application to incremental sheet forming. *Journal of Physics: Conference Series*, 1063(1).

Contributions by others to the thesis

No contributions by others

Statement of parts of the thesis submitted to qualify for the award of another degree

No works submitted towards another degree have been included in this thesis.

Research involving human or animal subjects

No animal or human subjects were involved in this research.

Acknowledgments

Firstly, I would like to express my sincere gratitude to my principal advisor Dr. Dorival Pedroso for his precious support during my Ph.D. candidature. Dorival was always very thorough, had excellent ingenious ideas, and was always aware of the very latest innovations. I could not have imagined having a better advisor and mentor for my studies.

Besides my principal advisor, I would like to thank Mr. Michael Elford and Dr. Andrew Stephan from Boeing Research & Technology – Australia with whom I spent a significant amount of time perfecting the simulation methods and implementation. I likely never forget the countless hourslong arguments with Andrew where no stone was left unturned. Moreover, I would like to thank my associate advisor Professor José Torero for his contributions to my Ph.D. milestones.

My sincere thanks also go to the chair of the review panel, Dr. William Daniel, and the panel members Dr. Joe Gattas and Dr. Vinh Dao, for their insightful comments and encouragement, but also for the challenging questions which incented me to deepen my research from various perspectives.

Furthermore, I am grateful to many others involved in my research one way or another—I have had very interesting and motivating discussions with many other researchers during my candidature. Special thanks go to Dr. Yunpeng Zhang who implemented certain advanced material models and contributed to the isogeometric simulation code.

Finally, I would like to thank my family and friends for supporting me throughout my Ph.D. project.

Financial support

1. UQ International Scholarship (UQI)
2. Boeing Research & Technology – Australia Top-Up Scholarship
3. Merenkulun Säätiö Relocation Scholarship

Keywords

incremental sheet forming, isogeometric analysis, shell element, quadrature rules, large deformations, explicit dynamics, contact modeling, unclamped knot vectors

Australian and New Zealand Standard Research Classifications (ANZSRC)

ANZSRC code: 010301, Numerical Analysis, 40%
ANZSRC code: 080304, Concurrent Programming, 30%
ANZSRC code: 091308, Solid Mechanics, 30%

Fields of Research (FoR) Classification

FoR code: 0103, Numerical and Computational Mathematics, 40%
FoR code: 0803, Computer Software, 30%
FoR code: 0913, Mechanical Engineering, 30%

Contents

List of figures **x**

List of tables **xiii**

List of abbreviations **xv**

1 Introduction **1**

1.1 Incremental sheet forming (ISF) 1
1.2 Mechanics of ISF 3
1.3 Modeling of ISF 4
1.4 Isogeometric analysis (IGA) 5
1.5 Research questions 6
1.6 Thesis outline 6
1.7 Notational remarks 7

2 Geometry representation **9**

2.1 Bézier curves 10
2.2 B-spline curves 11
2.2.1 Knot vector 11
2.2.2 Basis functions 12
2.2.3 Derivatives of basis functions 13
2.3 Nonuniform rational b-splines (NURBS) 13
2.3.1 NURBS surface 15
2.3.2 NURBS solid 15
2.4 T-splines 16
2.5 Algorithms 19
2.5.1 Closest point projection onto a t-spline curve 19
2.5.2 Closest point projection onto a t-spline surface 19

3 Analysis **23**
3.1 Galerkin method . . . 23
3.1.1 Conservation of linear momentum . . . 24
3.1.2 Conservation of energy . . . 26
3.1.3 Linear thermoelastic coupling . . . 28
3.2 Numerical integration . . . 28
3.2.1 Explicit time integration . . . 29
3.2.2 Implicit time integration . . . 30
3.2.3 Domain and boundary integrals . . . 31
3.3 Isogeometric shell technology . . . 34
3.3.1 The principle of virtual power . . . 35
3.3.2 Shell kinematics . . . 36
3.3.3 Control point normals . . . 38
3.3.4 Local coordinate system . . . 39
3.3.5 Internal virtual power . . . 41
3.3.6 External virtual power . . . 47
3.3.7 Inertial virtual power . . . 48
3.3.8 Time integration . . . 49
3.4 Stress integration . . . 50
3.4.1 Objective stress rate . . . 50
3.4.2 Plane stress condition . . . 54
3.4.3 Incremental constitutive update . . . 54
3.4.4 Implicit radial return algorithm (von Mises) . . . 56
3.4.5 General explicit return mapping algorithm . . . 58
3.5 Loads and boundary conditions . . . 59
3.5.1 Point and line loads . . . 59
3.5.2 Dirichlet boundary conditions . . . 60
3.6 Contact . . . 62
3.6.1 Contact contribution to virtual power . . . 64
3.6.2 Stylus contact algorithm . . . 65
3.6.3 Die contact algorithm . . . 68

4 Numerical testing **71**
4.1 Timoshenko beam (SS) . . . 72
4.2 Mindlin plate (SSSS) . . . 74
4.3 Straight cantilever plate strip . . . 75

4.4 Cantilever ring plate 76
4.5 Snap-through of a thick roof 78
4.6 Dynamic elasto-plastic response of a square plate 79
4.7 Single point incremental forming example 81
4.8 Two-point incremental forming example 88

5 Conclusions **95**

A Isogeometric thickness stretchable shell **101**

B Quadrature rules for isogeometric shell formulations **135**

C ISF simulation **155**

Bibliography **163**

List of figures

1.1 Single point incremental forming (SPIF). Only the forming tool contacts the sheet. 2
1.2 Two-point incremental forming (TPIF). In addition to the forming tool, a moving frame and a die are employed. 2
1.3 Double-sided incremental forming (DSIF). Two synchronously driven tools are used to form the final shape. 3

2.1 A b-spline curve comprised of seven control points. The dashed lines connecting the control points represent the control grid. 11
2.2 The index space for a biquadratic b-spline surface. The local knot vectors associated with the corresponding control point are denoted by the blue lines. The yellow shaded area represents the support of the basis function. 17
2.3 The index space for biquadratic (left-hand side) and bicubic (right-hand side) t-spline surfaces. The local knot vectors associated with the anchors s_a and s_b are denoted by the blue lines. The yellow shaded area represents the support of the basis function. 18

3.1 Two-dimensional Gauss-Legendre and Gauss-Lobatto quadratures. 32
3.2 The first-ring neighborhood in the context of bilinear finite elements (top view). The elements within the shaded area are connected to the current node. 68

4.1 Timoshenko beam under a uniformly distributed load. 73
4.2 Mindlin plate under a uniformly distributed load. 75
4.3 Cantilever plate strip under a tip line load. 76
4.4 Cantilever ring plate under a tip line load. 77
4.5 A cylindrical roof structure under a point load. 78

4.6 Dynamic elasto-plastic response of a square sheet under a uniform pressure and the influence of fiber mass scaling. 80
4.7 A photograph of the formed part (SPIF). 82
4.8 The experimental normalized thickness from the 3D scanning (SPIF). The contours vary linearly from 0.62 (red) to 1.0 (blue). 82
4.9 The final shape of the formed part along a slice $x = 0$ for the SPIF example. 82
4.10 The normalized thickness plot of the SPIF example for each element type (6mm grid). The contours vary linearly from 0.62 (red) to 1.0 (blue). . . 83
4.11 The normalized thickness plot of the SPIF example for each element type (4mm grid). The contours vary linearly from 0.62 (red) to 1.0 (blue). . . 84
4.12 The normalized thickness plot of the SPIF example for each element type (2mm grid). The contours vary linearly from 0.62 (red) to 1.0 (blue). . . 85
4.13 Forming forces in the global z-direction (vertical direction) for the SPIF example. 87
4.14 A photograph of the formed part (TPIF). 89
4.15 The experimental normalized thickness from the 3D scanning (TPIF). The contours vary linearly from 0.66 (red) to 1.0 (blue). 89
4.16 The final shape of the formed part along a slice $x = 0$ for the TPIF example. 89
4.17 The normalized thickness plot of the TPIF example for each element type (6mm grid). The contours vary linearly from 0.66 (red) to 1.0 (blue). . . 90
4.18 The normalized thickness plot of the TPIF example for each element type (4mm grid). The contours vary linearly from 0.66 (red) to 1.0 (blue). . . 91
4.19 The normalized thickness plot of the TPIF example for each element type (2mm grid). The contours vary linearly from 0.66 (red) to 1.0 (blue). . . 92
4.20 Forming forces in the global z-direction (vertical direction) for the TPIF example. 93

A.1 Timoshenko beam under a uniformly distributed load. 120
A.2 Mindlin plate under a uniformly distributed load. 122
A.3 Cantilever plate strip under a tip line load. 123
A.4 Cantilever ring plate under a tip line load. 124
A.5 Snap-through behavior of a cylindrical roof structure under a point load. 125
A.6 Dynamic elasto-plastic response of a square plate under a uniform pressure. 126
A.7 Dynamic elasto-plastic response of a straight cantilever plate strip under a tip line load. 128

B.1 In-plane quadrature point locations proposed for biquadratic splines. . . 139

B.2 In-plane quadrature point locations proposed for bicubic splines. 140
B.3 Mindlin plate under a uniformly distributed load. 141
B.4 Cantilever plate strip under a tip line load—the problem setup. 142
B.5 Cantilever plate strip under a tip line load—load factor versus the vertical tip displacement. 143
B.6 The target geometry dimensions. 145
B.7 The path followed by the indenting stylus (toolpath). 145
B.8 A photograph of the formed part. 146
B.9 The experimental normalized thickness from the 3D scanning. The contours vary linearly from 0.62 to 1.0. 146
B.10 The deformed geometry for each quadrature rule (biquadratic splines). . 147
B.11 The normalized thickness plot for each quadrature rule (biquadratic splines). The contours vary linearly from 0.62 to 1.0. 148
B.12 The deformed geometry for each quadrature rule (bicubic splines). 149
B.13 The normalized thickness plot for each quadrature rule (bicubic splines). The contours vary linearly from 0.62 to 1.0. 149
B.14 Forming forces in the global z-direction (vertical direction). 149

C.1 The dimensions of the die. 158
C.2 The toolpath. 158
C.3 A photograph of the formed part. 159
C.4 The 3D scanned normalized thickness. The contours vary linearly from 0.66 to 1.0. 159
C.5 The simulation result of the isogeometric shell formulation. 160
C.6 The simulation result of BWC shell formulation. 160
C.7 The normalized thickness plot of the isogeometric shell formulation. The contours vary linearly from 0.66 to 1.0. 160
C.8 The normalized thickness plot of BWC shell formulation. The contours vary linearly from 0.66 to 1.0. 160

List of tables

4.1 SPIF simulation runtime for each element type using Nvidia Titan V GPU. 87
4.2 TPIF simulation runtime for each element type using Nvidia Titan V GPU. 94

B.1 Quadrature rule cost comparison using Nvidia Titan V GPU. 151

List of abbreviations

Abbreviations	
3D	Three-Dimensional
BWC	Belytschko-Wong-Chiang (shell element)
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CB	Continuum-Based
CFL	Courant-Friedrichs-Lewy (stability condition)
CNC	Computer Numerical Control
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture (NVIDIA parallel computing platform)
DSIF	Double-Sided Incremental Forming
EAS	Enhanced Assumed Strain
FEM	Finite Element Method
GPGPU	General-Purpose computing on Graphics Processing Units
GPTS	Gauss-Point-To-Surface
GPU	Graphics Processing Unit
IGA	IsoGeometric Analysis
ISF	Incremental Sheet Forming
KL3	Kirchhoff-Love shell element w/ 3 DOFs
KL6	Kirchhoff-Love shell element w/ 6 DOFs
LBB	Ladyzhenskaya-Babuška-Brezzi (stability condition)
NTS	Node-To-Surface
NURBS	NonUniform Rational B-Splines
PC	Personal Computer
RM6	Reissner-Mindlin shell element w/ 6 DOFs
S7	Solid-like shell element w/ 7 DOFs
SCSC	Simply supported–Clamped–Simply supported–Clamped

SDC	Scaled Director Conditioning
SPIF	Single Point Incremental Forming
SS	Simply supported–Simply supported
SSSS	Simply supported–Simply supported–Simply supported–Simply supported
STC	Scaled Thickness Conditioning
STS	Surface-To-Surface
TPIF	Two-Point Incremental Forming

Chapter 1

Introduction

1.1 Incremental sheet forming (ISF)

Incremental sheet forming (ISF) is a manufacturing process in which a blunt moving tool is used to deform a sheet into the desired shape. The tool tip creates a local plastic deformation while moving throughout the predefined tool path. The final form of the sheet is obtained by a series of small incremental deformations entirely from the original sheet thickness. The concept of ISF can be tracked to the patents filed by Roux (1960) and Leszak (1967).

Many widely adopted and cost effective sheet forming processes used for mass production (e.g., deep drawing) require a die. Nevertheless, manufacturing precise dies is usually costly and requires a long lead time (Altan and Tekkaya, 2012). This is not economical for rapid prototyping, small-batch production, or highly customized parts. Although ISF is a relatively slow process, it requires either no die or often only an inexpensive simple die, which makes it ideal for small-batch production (Jeswiet et al., 2005; Bambach, 2008). ISF can be divided in different variants depending on the type of the process.

Single point incremental forming (SPIF) is considered dieless method as it uses no die on the other side of the sheet (see Figure 1.1). However, often a simple backplate is utilized to prevent widely spread deformations at the flange area. The edges of the blank are clamped to a specific blank holder which remains at a constant height during the process. Even though using a backplate does not result in a single point contact, the acronym “SPIF” is well-established and shall be employed in this

context (Bambach, 2008).

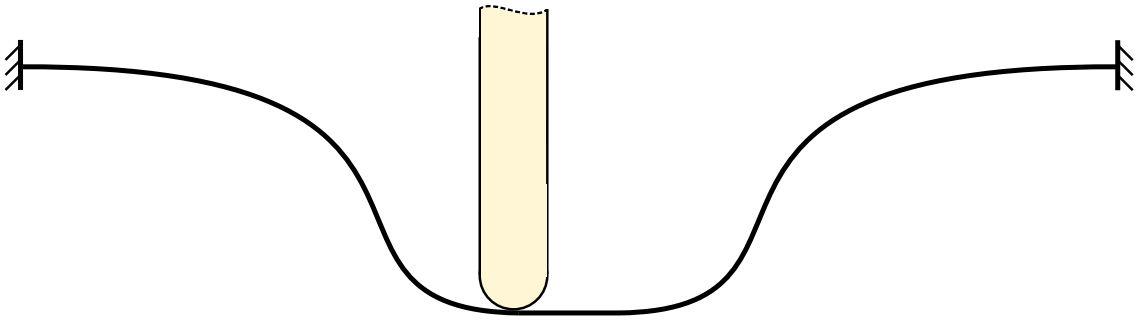


Figure 1.1: Single point incremental forming (SPIF). Only the forming tool contacts the sheet.

Two-point incremental forming (TPIF) uses a die to improve the accuracy of the process (Altan and Tekkaya, 2012). The die is placed on the opposite side of the blank compared to the tool (see Figure 1.2). Depending on the geometry requirements, a full or partial die can be used. The edges of the blank are still clamped, but the blank holder moves gradually to the direction of the die to allow proper shaping of the workpiece.

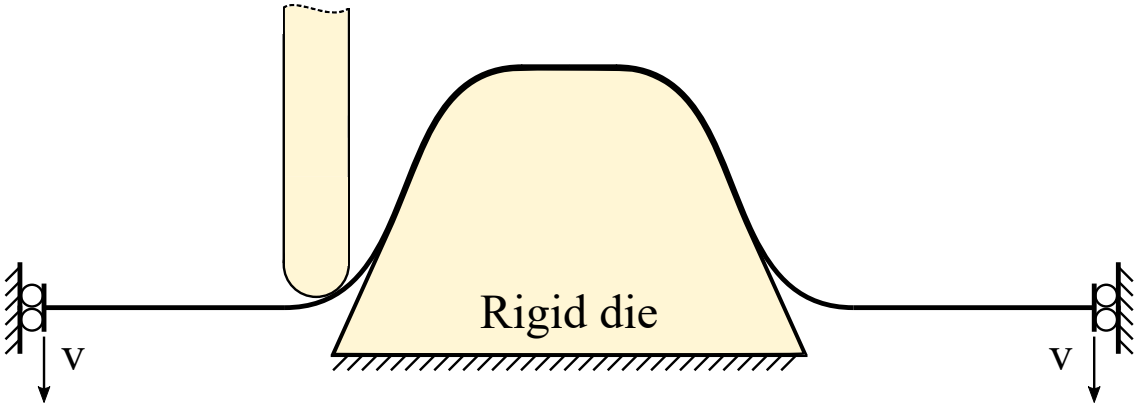


Figure 1.2: Two-point incremental forming (TPIF). In addition to the forming tool, a moving frame and a die are employed.

Double-sided incremental forming (DSIF) is a newer method first described in the Japanese patent filed by Shima et al. (1997). This approach relies on two forming tools—each at opposite sides of the blank (Altan and Tekkaya, 2012). The tools are driven synchronously such that the slave tool provides support to the master tool

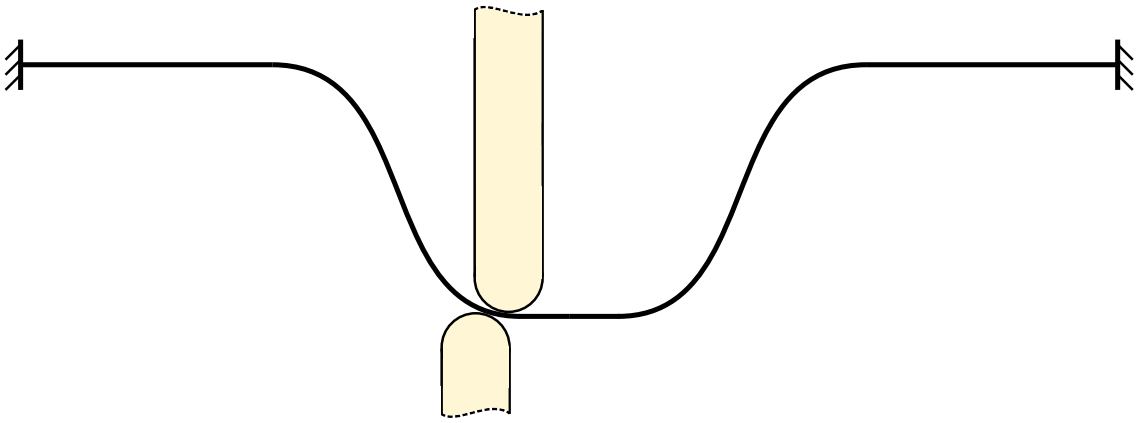


Figure 1.3: Double-sided incremental forming (DSIF). Two synchronously driven tools are used to form the final shape.

throughout the forming process (see Figure 1.3). The method is truly dieless and should result in better accuracy compared to SPIF.

Other ISF variants promising good accuracy such as methods based on a dynamic die (Franzen et al., 2008) or multiple parallel tools (Wang and Peng, 2015) can be found from the literature as well.

1.2 Mechanics of ISF

ISF requires a computer-controlled machine which is able to move the tool along the predefined path. Computer numerically controlled (CNC) milling machines and industrial robots are typical examples of utilized machines (Altan and Tekkaya, 2012). In general, the stiffness of the device should be high enough to prevent the tool deviating from the trajectory.

The typical forming tool has a hemispherical head with a diameter between 6 mm and 50 mm. The tool may or may not rotate during the ISF process. The plastic deformation area is limited to the contact zone of the tool and the blank, which makes the process' mechanics relatively complex (Altan and Tekkaya, 2012).

One of the most important limiting factors in ISF is the sheet thinning during the process. The sheet thinning can be approximated by the cosine of the wall angle, which assumes a projective behavior of the surface. Under this assumption, the

actual sheet thickness h is given by (Young and Jeswiet, 2004)

$$h = h_0 \cos(\psi) \tag{1.1}$$

where h_0 is the initial thickness, and ψ is the wall angle. This approximation is limited for single-pass processes only, as multi-pass forming strategies can better control the thinning regions and even produce a wall angle of 90° (Hirt et al., 2004; Altan and Tekkaya, 2012). However, according to the experiments conducted by Young and Jeswiet (2004), the thickness resulting from the single-pass ISF varies significantly even for a constant angled wall profile indicating poor accuracy of (1.1). According to Jeswiet et al. (2005), the maximum achievable draw angle using a single-pass forming strategy for typical applications is between 60° and 70°.

Another challenge in ISF is a springback effect caused by elastic strains. During the forming process, local and global springback effects can be recognized. Local springback occurs as the zone in contact with the tool moves along the path. At the same time, the part is always slightly inclined towards the position of the tool, which causes a global orbiting springback effect. Other springback effects may occur when the forming tool is removed, or the part is unclamped or trimmed (Altan and Tekkaya, 2012).

1.3 Modeling of ISF

As the industry has recognized the potential of ISF manufacturing process, there has also emerged a great interest for fast simulations—even for real-time analysis. The finite element method, or FEM, is a widely adopted modeling approach among the engineering industry. It has been successfully used to model metal forming problems including ISF. However, applying FEM to ISF is computationally costly as one simulation may easily take days depending on the required accuracy and the size of the problem (Elford et al., 2013).

In ISF, the final shape is achieved by a series of small local incremental deformations. As this very local deformation zone is a key element of the whole process, a fine space discretization is required in order to obtain accurate simulation results. On the other hand, the time-consuming nature of an incremental process also results in anticipation of a very large number of time steps, which is, indeed, often the case. Combining the demanding space and time discretization requirements with strongly nonlinear behavior results in an enormous computational cost.

If the simulation uses an explicit time integration method, the maximum time increment is limited by the Courant-Friedrichs-Lewy (CFL) stability condition (Courant et al., 1928). For the industry standard, four-noded, bilinear shell element, the CFL condition is proportional to the smallest element dimension and inversely proportional to the speed of sound of the material. On the other hand, the time increment size in implicit time integration methods is not necessarily penalized by the CFL condition. Although this would permit a use of larger time increments, convergence problems are often encountered due to the strong nonlinearities, i.e., contact, geometric changes, and material response.

1.4 Isogeometric analysis (IGA)

Isogeometric analysis (IGA) is a relatively new computational method first introduced by Hughes et al. (2005) that uses the exact geometry representation as a basis for analysis. The first motivation for the development of IGA was to eliminate the gap between computer aided design (CAD) and computer aided engineering (CAE) and avoid the often time-consuming step of mesh construction. In IGA, the interpolation functions are typically not tied to a single element-like entity as is the case in the context of standard FEM. Instead, the interpolation functions have support over multiple element-like entities that allows higher continuity. The increased smoothness have turned out to provide superior accuracy and robustness on a per-degree-of-freedom basis in comparison to conventional Lagrange finite elements (Cottrell et al., 2006, 2007; Evans et al., 2009; Großmann et al., 2012).

Improved accuracy implies that the number of degrees of freedom could be reduced, i.e., a coarser mesh could be employed to achieve the same level of accuracy comparable to FEM. As the maximum stable time increment size for explicit time integration methods is tied to the minimum element-like dimension, the applicable time increment size is simultaneously increased. However, according to Adam et al. (2015b) the maximum stable time step size could be increased even further due to the higher continuity. This could mean a significant reduction in computational time of explicit analysis. Furthermore, the increased smoothness could also improve the convergence of implicit methods, as the nonphysical jumps in contact forces are eliminated.

The higher continuity is also beneficial for the Galerkin method as the numerical integration need not be performed at most one element at a time (Hughes et al.,

2010; Schillinger et al., 2014; Adam et al., 2015b). Moreover, the smoothness of the interpolation functions makes collocation methods potentially a good alternative for solving the underlying partial differential equations. Collocation approach is based on the discretization of the strong form of equations and has recently been under intense research in the context of IGA (Auricchio et al., 2010, 2012a; Schillinger et al., 2013; De Lorenzis et al., 2015; Kiendl et al., 2015a; Kruse et al., 2015).

1.5 Research questions

The main emphasis of this research project is to investigate the applicability of IGA to ISF simulations and to determine if the simulations benefit from using isogeometric shell technology. The most desirable attributes of an isogeometric shell formulation are robustness and high computational efficiency. The following questions are studied among others:

- What type of shell formulation is ideal?
- What kind of quadrature rule to use?
- What is the maximum stable time step size?
- How are loads and boundary conditions treated?
- What are the requirements for the contact algorithm?

1.6 Thesis outline

The thesis is structured as follows. Chapter 2 reviews typical geometry representations used with IGA including a thorough description of Bézier curves, b-splines, NURBS, and t-splines. Chapter 3 introduces the Galerkin method and the numerical integration, which serve as the foundations for the analysis. Furthermore, several shell formulations based on Kirchhoff-Love and Reissner-Mindlin theories are presented, and stress integration, loads and boundary conditions, as well as contact modeling are discussed. Chapter 4 presents numerical results using several benchmark problems which range from simple static linear elastic examples to complex ISF simulations. Finally, conclusions are given in Chapter 5.

Three papers written during the research project are given as appendices: *Isogeometric thickness stretchable shell: efficient formulation for nonlinear dynamic problems*

(Appendix A), *Quadrature rules for isogeometric shell formulations: study using a real-world application about metal forming* (Appendix B), and *Efficient isogeometric shell element with through-thickness stretch: application to incremental sheet forming* (Appendix C).

1.7 Notational remarks

The following notation is used throughout the text. The order of a tensor variable is indicated by the number of dots added under the corresponding symbol. For instance, $\underset{\cdot}{\boldsymbol{a}}$ and $\underset{\cdot\cdot}{\boldsymbol{a}}$ are different tensors of first and second orders, respectively. However, the underdots are omitted whenever the matrix notation is employed. A similar dot on top of a variable indicates a material time derivative. Inner products are expressed with operator dots such as in $a = \underset{\cdot}{\boldsymbol{a}} \cdot \underset{\cdot}{\boldsymbol{a}}$ and $b = \underset{\cdot\cdot}{\boldsymbol{a}} : \underset{\cdot\cdot}{\boldsymbol{a}}$. A tensor product is expressed as $\underset{\cdot\cdot}{\boldsymbol{a}} = \underset{\cdot}{\boldsymbol{a}} \otimes \underset{\cdot}{\boldsymbol{a}}$. The symbol ($\otimes$) is also used when denoting a multi-dimensional tensor product space in terms of one-dimensional spaces. Furthermore, a partial derivative is denoted by a comma in the subscript. Lastly, although the concept of Bézier extraction is not employed, the subdomains bounded by the knots are still referred to as Bézier elements, similarly to the works of Schillinger et al. (2014), Adam et al. (2015c), and Hiemstra et al. (2017).

Chapter 2

Geometry representation

In the modern CAD technology, geometry is mostly represented by nonuniform rational b-splines, known as NURBS (Piegl and Tiller, 1996). However, a single NURBS patch is not capable of representing an arbitrary shape often required by an engineering design. This problem is solved by trimming and connecting multiple NURBS patches together. However, the resulting representation is not generally suitable as a basis for analysis (Scott et al., 2012). T-splines introduced by Sederberg et al. (2003) are a potentially superior alternative to NURBS geometry representation. T-splines can be combined with subdivision schemes such as Catmull-Clark (1978) surfaces, and, as a result, a geometry of any complexity can be represented as a single analysis suitable watertight patch.

In the CAE technology, the classical finite elements use Lagrange polynomials to approximate the exact CAD geometry. These polynomials are often of first or second order, and the same polynomials are typically also used to interpolate displacements. This approximated geometry requires an often computationally costly step of mesh generation and results in loss of precision even before the analysis has started. To overcome these drawbacks, the concept of isogeometric analysis proposed by Hughes et al. (2005) aims to use the exact CAD geometry as a basis for analysis. In order to improve the accuracy of the analysis, further refinement of the CAD representation is still possible without changing the initial CAD geometry.

To fully understand the concept of IGA, a thorough understanding of spline technology is crucial. Starting with Bézier curves (Section 2.1) seems reasonable as it is a simple special case, i.e., a subset, of b-splines (Section 2.2), NURBS (Section 2.3), and

t-splines (Section 2.4). Lastly, some geometric algorithms are given in Section 2.5.

2.1 Bézier curves

A Bézier curve is a parametric polynomial curve which is a function of one parameter. The parameter represents a straight line segment in the parametric space, which, if mapped into the three-dimensional Euclidean space is given as follows:

$$\boldsymbol{c}(\xi) = \begin{bmatrix} x(\xi) \\ y(\xi) \\ z(\xi) \end{bmatrix} \tag{2.1}$$

The parametric space is often normalized such that $0 \leq \xi \leq 1$. A Bézier curve is defined as (Piegl and Tiller, 1996)

$$\underset{\cdot}{\boldsymbol{c}}(\xi) = \sum_{i=0}^{p} N_i^p(\xi) \underset{\cdot}{\boldsymbol{x}}_i \tag{2.2}$$

where $N_i^p(\xi)$ are the Bernstein (1912) polynomials of degree p given by

$$N_i^p(\xi) = \frac{p!}{i!(p-i)!} \xi^i (1-\xi)^{p-i} \tag{2.3}$$

and $\underset{\cdot}{\boldsymbol{x}}_i$ are called control points. Bernstein polynomials $N_i^p(\xi)$ are nonnegative over the entire domain,

$$N_i^p(\xi) \geq 0, \quad \forall \xi \tag{2.4}$$

and they satisfy the partition of unity condition, i.e., (Piegl and Tiller, 1996)

$$\sum_{i=0}^{p} N_i^p(\xi) = 1, \quad \forall \xi \tag{2.5}$$

The derivatives of a Bézier curve are obtained by (Piegl and Tiller, 1996; Farouki, 2012)

$$\begin{aligned} \underset{\cdot}{\boldsymbol{c}}(\xi)_{,\xi} &= \sum_{i=0}^{p} N_i^p(\xi)_{,\xi} \underset{\cdot}{\boldsymbol{x}}_i \\ &= \sum_{i=0}^{p} p\big(N_{i-1}^{p-1}(\xi) - N_i^{p-1}(\xi)\big) \underset{\cdot}{\boldsymbol{x}}_i \\ &= p \sum_{i=0}^{p-1} N_i^{p-1}(\xi) (\underset{\cdot}{\boldsymbol{x}}_{i+1} - \underset{\cdot}{\boldsymbol{x}}_i) \end{aligned} \tag{2.6}$$

where

$$N_{-1}^{p-1}(\xi) \equiv N_p^{p-1}(\xi) \equiv 0 \tag{2.7}$$

2.2 B-spline curves

Bézier curves consist of only one polynomial or rational segment. This means that in order to accurately represent a complex shape or to satisfy a large number of constraints, high degree polynomials are required. However, curves of high degree are inefficient to process and numerically unstable (Piegl and Tiller, 1996). Therefore, b-splines have been introduced to overcome these drawbacks.

B-spline, also known as basis spline, is a piecewise polynomial or piecewise rational type of curve. B-spline curve of degree p is given by

$$\underset{\cdot}{\boldsymbol{c}}(\xi) = \sum_{i=1}^{n} N_i^p(\xi)\underset{\cdot}{\boldsymbol{x}}_i \tag{2.8}$$

where $N_i^p(\xi)$ are pth degree b-spline basis functions and $\underset{\cdot}{\boldsymbol{x}}_i$ are n control points. A b-spline curve comprised of seven control points is shown in Figure 2.1.

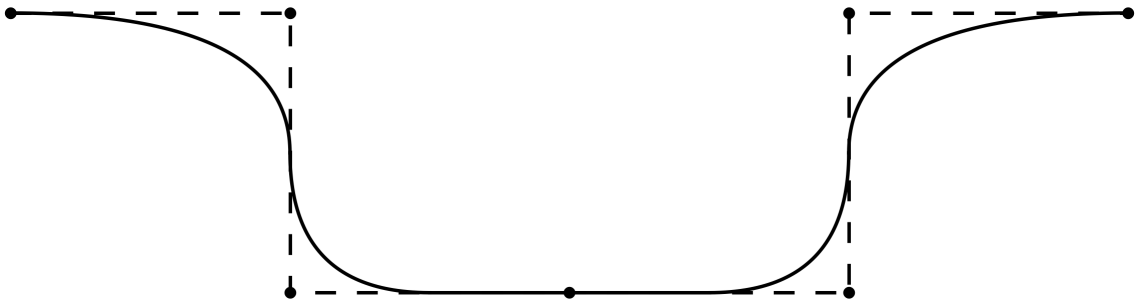


Figure 2.1: A b-spline curve comprised of seven control points. The dashed lines connecting the control points represent the control grid.

2.2.1 Knot vector

The breakpoints, i.e., the knots, between the curve segments are defined by a nondecreasing set of coordinates called a knot vector:

$$\Xi = \{\xi_1, \xi_2 \ldots, \xi_{n_\Xi}\}, \quad \xi_i \in \mathbb{R} \tag{2.9}$$

Knot vectors can be split into groups of clamped and unclamped, and further categorized as uniform and nonuniform. A clamped knot vector requires its first and last knot values to be repeated $p + 1$ times. This causes the curve to coincide with its first and last control points. The number of knots for a knot vector can be determined from the relation

$$n_\Xi = n + p + 1 \tag{2.10}$$

where p is the degree of the polynomial and n is the number of control points used to construct the b-spline curve (Piegl and Tiller, 1996; Cottrell et al., 2009). Furthermore, a clamped knot vector of the form

$$\Xi = \{\underbrace{\xi_1, \ldots, \xi_1}_{p+1}, \underbrace{\xi_{p+2}, \ldots, \xi_{p+2}}_{p+1}\} \tag{2.11}$$

produces a b-spline curve equal to a pth degree Bézier curve (Piegl and Tiller, 1996). Therefore b-spline curves are a generalization of Bézier curves.

If the knot spacing is constant within the parametric space, the knot vector is referred to as uniform, and otherwise nonuniform. However, a clamped knot vector may still be considered uniform although its first and last knots have a multiplicity of $p+1$ (Piegl and Tiller, 1996). Examples of knot vectors of quadratic b-splines are given below:

$$\begin{aligned}
\Xi_{cu} &= \{0,0,0,1,2,3,4,4,4\} \quad \text{(clamped, uniform)} \\
\Xi_{cn} &= \{-3,-3,-3,-2,6,6,9,9,9\} \quad \text{(clamped, nonuniform)} \\
\Xi_{uu} &= \{-2,-1,0,1,2,3,4,5,6\} \quad \text{(unclamped, uniform)} \\
\Xi_{un} &= \{-5,-4,0,1,1,2,5,7,9\} \quad \text{(unclamped, nonuniform)}
\end{aligned} \tag{2.12}$$

A b-spline curve is defined C^k continuous at a knot ξ_i if the derivatives up to the kth derivative of the adjacent curve segments are equal at the joint, i.e.,

$$\lim_{\xi \to \xi_i^-} \underset{\cdot}{\boldsymbol{c}}(\xi)_{,\xi^j} = \underset{\cdot}{\boldsymbol{c}}(\xi)_{,\xi^j} = \lim_{\xi \to \xi_i^+} \underset{\cdot}{\boldsymbol{c}}(\xi)_{,\xi^j}, \quad \forall \{j \in \mathbb{Z} \mid 0 \leq j \leq k\} \tag{2.13}$$

However, the continuity can be directly determined by

$$C^{p-m_i} \tag{2.14}$$

where p is the degree of the polynomial and m_i is the multiplicity of the corresponding knot. A knot at ξ_i satisfying the equation

$$\xi_{i+m_i-1} - \xi_i = 0 \tag{2.15}$$

has a multiplicity of at least m_i which means that the knots $\{\xi_i, \ldots, \xi_{i+m_i-1}\}$ are located at the same point in the parametric space.

2.2.2 Basis functions

After the knot vectors are defined, the b-spline basis functions can be calculated. B-splines were first introduced by defining the basis functions by divided differences of

truncated power functions (Schoenberg, 1946; Curry and Schoenberg, 1947). Later, Cox (1972) and de Boor (1972, 1978) introduced a recurrence formula that has become widely adopted. Furthermore, a method based on blossoming principle was proposed by Ramshaw (1987). The second one is known as Cox-de Boor recursion formula (Cottrell et al., 2009) and defines b-spline basis functions N_i^p of degree p by

$$\begin{aligned} N_i^0(\xi) &= \begin{cases} 1 & \text{if } \xi_i \leq \xi \leq \xi_{i+1} \\ 0 & \text{otherwise} \end{cases} \\ N_i^p(\xi) &= \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_i^{p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1}^{p-1}(\xi), \quad \forall \{p \in \mathbb{Z} \mid p > 0\} \end{aligned} \tag{2.16}$$

where $\frac{0}{0} \equiv 0$ by convention. Similarly to Bernstein polynomials, b-spline basis functions $N_i^p(\xi)$ are nonnegative over the entire domain,

$$N_i^p(\xi) \geq 0, \quad \forall \xi \tag{2.17}$$

and the basis satisfies the partition of unity condition, i.e., (Cottrell et al., 2009)

$$\sum_{i=1}^{n} N_i^p(\xi) = 1, \quad \forall \xi \tag{2.18}$$

Furthermore, the support of a basis function is always $p+1$ knot spans, which results in the continuity property given by (2.14) (Cottrell et al., 2009).

2.2.3 Derivatives of basis functions

The first derivative of a basis function can be calculated by (Piegl and Tiller, 1996; Cottrell et al., 2009)

$$N_i^p(\xi)_{,\xi} = \frac{p}{\xi_{i+p} - \xi_i} N_i^{p-1}(\xi) - \frac{p}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1}^{p-1}(\xi) \tag{2.19}$$

which can be generalized for kth order derivatives as follows:

$$N_i^p(\xi)_{,\xi^k} = \frac{p}{\xi_{i+p} - \xi_i} N_i^{p-1}(\xi)_{,\xi^{k-1}} - \frac{p}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1}^{p-1}(\xi)_{,\xi^{k-1}} \tag{2.20}$$

2.3 Nonuniform rational b-splines (NURBS)

However, not all shapes can be represented by polynomials. Representation of conic shapes, such as circles or ellipses requires rational functions. Therefore, NURBS have been introduced to support such shapes. A rational function is defined as a

ratio of two polynomials. A rational b-spline curve of degree p is defined by (Piegl and Tiller, 1996)

$$\underset{\cdot}{\boldsymbol{c}}(\xi) = \frac{\sum_{i=1}^{n} N_i^p(\xi) w_i \underset{\cdot}{\boldsymbol{x}}_i}{\sum_{j=1}^{n} N_j^p(\xi) w_j} = \sum_{i=1}^{n} R_i^p(\xi) \underset{\cdot}{\boldsymbol{x}}_i \tag{2.21}$$

where $R_i^p(\xi)$ are the rational basis functions given by

$$R_i^p(\xi) = \frac{N_i^p(\xi) w_i}{\sum_{j=1}^{n} N_j^p(\xi) w_j} \tag{2.22}$$

and w_i are scalar weights for n control points $\underset{\cdot}{\boldsymbol{x}}_i$. This rational curve in $\mathbb{R}^d$ can be efficiently represented as a polynomial curve in $\mathbb{R}^{d+1}$ using a projective transformation (Piegl and Tiller, 1996; Cottrell et al., 2009). A nonrational projective b-spline curve in $\mathbb{R}^4$ is given as

$$\underset{\cdot}{\boldsymbol{c}}^w(\xi) = \sum_{i=1}^{n} N_i^p(\xi) \underset{\cdot}{\boldsymbol{x}}_i^w \tag{2.23}$$

where the projective control points are

$$\boldsymbol{x}_i^w = \begin{bmatrix} w_i x_i \\ w_i y_i \\ w_i z_i \\ w_i \end{bmatrix} \tag{2.24}$$

Applying the projective transformation is equivalent to dividing the projective control points by the corresponding weights (Piegl and Tiller, 1996; Cottrell et al., 2009). Equation (2.23) of a polynomial b-spline curve in $\mathbb{R}^4$ can be written as

$$\begin{bmatrix} x^w(\xi) \\ y^w(\xi) \\ z^w(\xi) \\ W(\xi) \end{bmatrix} = \sum_{i=1}^{n} N_i^p(\xi) w_i \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \tag{2.25}$$

where

$$W(\xi) = \sum_{i=1}^{n} N_i^p(\xi) w_i \tag{2.26}$$

The projective transformation $\mathbb{R}^4 \to \mathbb{R}^3$ is obtained by mapping $\underset{\cdot}{\boldsymbol{c}}^w(\xi)$ onto a hyperplane $W(\xi) = 1$:

$$\begin{bmatrix} x^w(\xi) \\ y^w(\xi) \\ z^w(\xi) \\ W(\xi) \end{bmatrix} \to \frac{1}{W(\xi)} \begin{bmatrix} x^w(\xi) \\ y^w(\xi) \\ z^w(\xi) \\ W(\xi) \end{bmatrix} = \begin{bmatrix} x(\xi) \\ y(\xi) \\ z(\xi) \\ 1 \end{bmatrix} \tag{2.27}$$

The geometry can now be represented in $\mathbb{R}^3$. Using index notation, one obtains

$$c_k(\xi) = \frac{c_k^w(\xi)}{W(\xi)} = \frac{\sum_{i=1}^{n} N_i^p(\xi) x_{ki}^w}{\sum_{j=1}^{n} N_j^p(\xi) w_j} = \frac{\sum_{i=1}^{n} N_i^p(\xi) w_i x_{ki}}{\sum_{j=1}^{n} N_j^p(\xi) w_j}, \quad \forall\{k \in \mathbb{Z} \mid 1 \leq k \leq 3\} \tag{2.28}$$

which is equivalent to (2.21).

2.3.1 NURBS surface

A NURBS surface can be constructed from two parameters that represent a plane in the parametric space. Mapping into the three-dimensional Euclidean space results in

$$\boldsymbol{s}(\xi, \eta) = \begin{bmatrix} x(\xi, \eta) \\ y(\xi, \eta) \\ z(\xi, \eta) \end{bmatrix} \tag{2.29}$$

where the parametric space is often normalized such that $0 \leq \xi, \eta \leq 1$. A rational b-spline surface defined by the tensor product method is given by (Piegl and Tiller, 1996)

$$\dot{\boldsymbol{s}}(\xi, \eta) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} N_i^p(\xi) N_j^q(\eta) w_{ij} \dot{\boldsymbol{x}}_{ij}}{\sum_{k=1}^{n} \sum_{l=1}^{m} N_k^p(\xi) N_l^q(\eta) w_{kl}} = \sum_{i=1}^{n} \sum_{j=1}^{m} R_{ij}^{pq}(\xi, \eta) \dot{\boldsymbol{x}}_{ij} \tag{2.30}$$

where n and m denote the number of control points in the parametric directions ξ and η, respectively. Moreover, p and q are the degrees of the univariate basis functions $N_i^p(\xi)$ and $N_j^q(\eta)$. The bivariate rational basis functions R_{ij}^{pq} are given as follows:

$$R_{ij}^{pq}(\xi, \eta) = \frac{N_i^p(\xi) N_j^q(\eta) w_{ij}}{\sum_{k=1}^{n} \sum_{l=1}^{m} N_k^p(\xi) N_l^q(\eta) w_{kl}} \tag{2.31}$$

2.3.2 NURBS solid

A solid, which is a function of three parameters, is derived similarly. Mapping into the three-dimensional Euclidean space results in

$$\boldsymbol{v}(\xi, \eta, \zeta) = \begin{bmatrix} x(\xi, \eta, \zeta) \\ y(\xi, \eta, \zeta) \\ z(\xi, \eta, \zeta) \end{bmatrix} \tag{2.32}$$

where the parametric space is often normalized such that $0 \leq \xi, \eta, \zeta \leq 1$. A rational NURBS solid is defined analogously to the NURBS surface using the tensor product method by

$$\begin{aligned} \dot{\boldsymbol{y}}(\xi,\eta,\zeta) &= \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} N_i^p(\xi)N_j^q(\eta)N_k^r(\zeta)w_{ijk}\dot{\boldsymbol{x}}_{ijk}}{\sum_{f=1}^{n}\sum_{g=1}^{m}\sum_{h=1}^{l} N_f^p(\xi)N_g^q(\eta)N_h^r(\zeta)w_{fgh}} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} R_{ijk}^{pqr}(\xi,\eta,\zeta)\dot{\boldsymbol{x}}_{ijk} \end{aligned} \tag{2.33}$$

where n, m and l denote the number of control points in the parametric directions ξ, η and ζ, respectively. Similarly to the NURBS surface, p, q and r are the degrees of the univariate basis functions $N_i^p(\xi)$, $N_j^q(\eta)$ and $N_k^r(\zeta)$. The trivariate rational basis functions R_{ijk}^{pqr} for NURBS solid are given below:

$$R_{ijk}^{pqr}(\xi,\eta,\zeta) = \frac{N_i^p(\xi)N_j^q(\eta)N_k^r(\zeta)w_{ijk}}{\sum_{f=1}^{n}\sum_{g=1}^{m}\sum_{h=1}^{l} N_f^p(\xi)N_g^q(\eta)N_h^r(\zeta)w_{fgh}} \tag{2.34}$$

2.4 T-splines

T-splines are a generalization of b-splines and NURBS first proposed by Sederberg et al. (2003) for bicubic surfaces and later generalized to three dimensions and arbitrary degree by Bazilevs et al. (2010). T-splines support T-junctions such that a line of control points in the parametric space is allowed to terminate in the middle of the patch. Therefore, t-splines do not preserve the tensor product nature of b-splines.

T-splines rely on locally defined knot vectors. As the support of a b-spline basis function $N_i^p(\xi)$ is always contained within $[\xi_i, \xi_{i+p+1}]$, it suffices to determine a local knot vector consisting of $p+2$ knots (see Figure 2.2). This local knot vector can now be used by the Cox-de Boor recursion formula (2.16) to calculate the corresponding basis function without altering the geometry in any way. As a result, locally defined knot vectors permit an addition of a single control point into the patch in contrast to a line of control points associated with a global knot vector.

Bazilevs et al. (2010) define t-splines in an index space, where the spacing between the knots is equal regardless of their actual spacing, and the knot lines are labeled with their index value. For a b-spline surface, the index space is a uniform grid of squares, whereas, for a t-spline surface, it consists of rectangles but not necessarily squares. The edges of the rectangles represent a knot location. Bazilevs et al. (2010) introduced an anchor $\dot{\boldsymbol{s}}$ associated with each control point. The anchor is used for

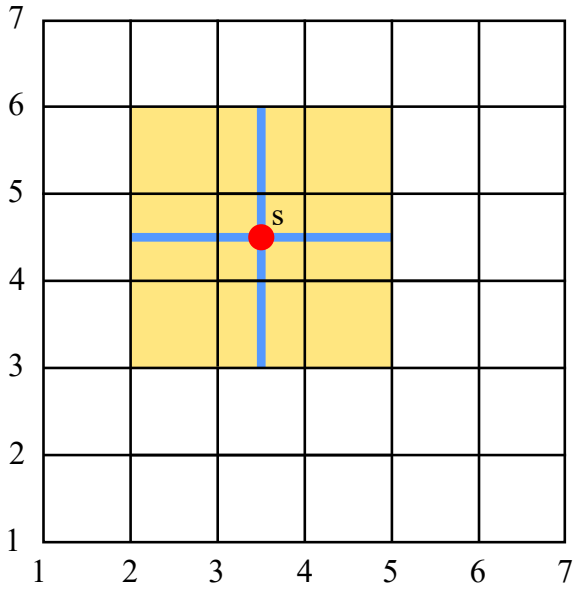


Figure 2.2: The index space for a biquadratic b-spline surface. The local knot vectors associated with the corresponding control point are denoted by the blue lines. The yellow shaded area represents the support of the basis function.

defining local knot vectors to be able to calculate the basis functions, or blending functions, as referred by Bazilevs et al. (2010) in the context of t-splines. For even degrees, the anchors are placed in the center of each rectangle. The local horizontally oriented knot vector Ξ_a is defined by shooting a ray left and right from the anchor, and choosing the first $p/2+1$ knots encountered separately from the both directions. The vertical knot vector $\mathcal{H}_a$ is derived analogously.

Following the aforementioned procedure for the anchor $\dot{\boldsymbol{s}}_a$ shown in Figure 2.3, the local knot vectors Ξ_a and $\mathcal{H}_a$ associated with the anchor are given for quadratic t-splines as

$$\begin{aligned} \Xi_a &= \{\xi_3, \xi_4, \xi_6, \xi_7\} \\ \mathcal{H}_a &= \{\eta_2, \eta_3, \eta_4, \eta_7\} \end{aligned} \tag{2.35}$$

For odd degrees, the anchors are placed at each vertex and $(p+1)/2$ knots are collected from each direction in addition to the knot coincident with the anchor. The local knot vectors associated with the anchor $\dot{\boldsymbol{s}}_b$ are given for cubic t-splines by (see Figure 2.3)

$$\begin{aligned} \Xi_b &= \{\xi_1, \xi_2, \xi_3, \xi_4, \xi_6\} \\ \mathcal{H}_b &= \{\eta_2, \eta_3, \eta_4, \eta_5, \eta_6\} \end{aligned} \tag{2.36}$$

Associating the control point $\dot{\boldsymbol{x}}_A$ and its weight w_A with each anchor location $\dot{\boldsymbol{s}}_A$, the multivariate rational t-spline basis functions for an arbitrary number (d) of

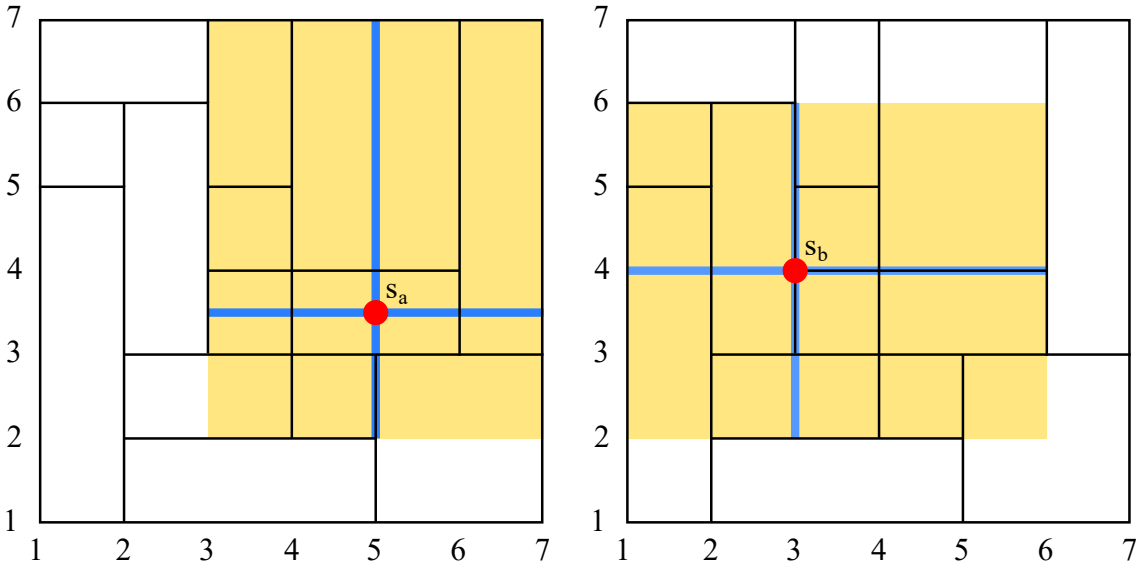


Figure 2.3: The index space for biquadratic (left-hand side) and bicubic (right-hand side) t-spline surfaces. The local knot vectors associated with the anchors s_a and s_b are denoted by the blue lines. The yellow shaded area represents the support of the basis function.

parametric dimensions are defined by (Bazilevs et al., 2010)

$$R_A(\boldsymbol{\xi}) = \frac{w_A \prod_{l=1}^{d} N_A^l(\xi^l)}{\sum_{B=1}^{n_T} w_B \prod_{l=1}^{d} N_B^l(\xi^l)} \tag{2.37}$$

where $N_A^l(\xi^l)$ is the univariate basis function of degree p_A^l constructed from a knot vector of length $p_A^l + 2$. It is associated with the anchor $\boldsymbol{s}_A$ and the parametric dimension l. Furthermore, n_T is the total number of control points and $\boldsymbol{\xi}$ is a vector representing all parametric dimensions. Finally, the t-spline in the physical space is given by

$$\boldsymbol{t}(\boldsymbol{\xi}) = \sum_{A=1}^{n_T} R_A(\boldsymbol{\xi}) \boldsymbol{x}_A \tag{2.38}$$

Unfortunately, Buffa et al. (2010) showed that not all t-spline basis functions are guaranteed to be linearly independent. Although this is not necessary for the most CAD applications, it is essential for analysis (Li et al., 2012). Therefore, Li et al. (2012) introduced a concept of analysis suitable t-splines, which is a slightly restricted subset of general t-splines. A local refinement algorithm for analysis suitable t-splines was developed by Scott et al. (2012). Later, Wang et al. (2014) pointed out that only t-splines consisting of repeated knots may be linearly dependent. Moreover, weighted t-splines (Liu et al., 2015b) and truncated t-splines (Wei et al., 2016) have been proposed to guarantee analysis suitability.

T-splines, as well as b-splines, can be combined with subdivision schemes such as Catmull-Clark surfaces (Catmull and Clark, 1978; Sederberg et al., 2003). This allows even a b-spline surface to have vertices in the middle of a surface that have valence other than four (Stam, 1998), where the term valence indicates the number of edges that meet at the vertex. These vertices are called extraordinary points. Furthermore, Stam (1998) showed that the surface and its derivatives can be efficiently evaluated around extraordinary points using a set of specific eigenbasis functions. Riffnaller-Schiefer et al. (2016) demonstrated isogeometric shell analysis using surfaces with extraordinary points, whereas Liu et al. (2015a) and Wei et al. (2016) discussed extraordinary points in the context of weighted and truncated t-splines, respectively.

2.5 Algorithms

2.5.1 Closest point projection onto a t-spline curve

A closest point projection onto a NURBS geometry can be calculated using Newton-Raphson iterative method provided that the initial guess for the first iteration is close enough (Piegl and Tiller, 1996). The same approach can be utilized for a general t-spline geometry.

A closest point projection onto a curve satisfies the orthogonality condition $\dot{\boldsymbol{c}}(\xi)_{,\xi} \perp (\dot{\boldsymbol{c}}(\xi) - \dot{\boldsymbol{p}})$ assuming that the point $\dot{\boldsymbol{p}}$ does not lie outside the end points of the curve. Therefore, due to the orthogonality condition, the following equation must hold:

$$f(\xi) = \dot{\boldsymbol{c}}(\xi)_{,\xi} \cdot (\dot{\boldsymbol{c}}(\xi) - \dot{\boldsymbol{p}}) = 0 \tag{2.39}$$

The Newton-Raphson method calculates the parametric coordinate for the next iteration $i + 1$ by

$$\xi_{i+1} = \xi_i - \frac{f(\xi_i)}{f'(\xi_i)} = \xi_i - \frac{\dot{\boldsymbol{c}}(\xi_i)_{,\xi} \cdot (\dot{\boldsymbol{c}}(\xi_i) - \dot{\boldsymbol{p}})}{\dot{\boldsymbol{c}}(\xi_i)_{,\xi\xi} \cdot (\dot{\boldsymbol{c}}(\xi_i) - \dot{\boldsymbol{p}}) + \|\dot{\boldsymbol{c}}(\xi_i)_{,\xi}\|^2} \tag{2.40}$$

where the derivatives $\dot{\boldsymbol{c}}(\xi_i)_{,\xi}$ and $\dot{\boldsymbol{c}}(\xi_i)_{,\xi\xi}$ can be calculated following the procedure discussed in Section 2.5.2.

2.5.2 Closest point projection onto a t-spline surface

A closest point projection onto a surface is computed analogously to Section 2.5.1 (Piegl and Tiller, 1996). Let $\dot{\boldsymbol{r}}$ be a vector between the point $\dot{\boldsymbol{p}}$ and a t-spline surface

$\dot{\boldsymbol{s}}(\xi,\eta)$:

$$\dot{\boldsymbol{r}}(\xi,\eta) = \dot{\boldsymbol{s}}(\xi,\eta) - \dot{\boldsymbol{p}} \tag{2.41}$$

The orthogonality condition must be satisfied for the both local coordinates giving

$$\boldsymbol{\kappa}_i = \begin{bmatrix} \dot{\boldsymbol{r}}(\xi_i,\eta_i) \cdot \dot{\boldsymbol{s}}(\xi_i,\eta_i)_{,\xi} \\ \dot{\boldsymbol{r}}(\xi_i,\eta_i) \cdot \dot{\boldsymbol{s}}(\xi_i,\eta_i)_{,\eta} \end{bmatrix} \tag{2.42}$$

for which the derivatives are given by

$$\boldsymbol{J}_i = \begin{bmatrix} \|\dot{\boldsymbol{s}}_{,\xi}\|^2 + \dot{\boldsymbol{r}} \cdot \dot{\boldsymbol{s}}_{,\xi\xi} & \dot{\boldsymbol{s}}_{,\xi} \cdot \dot{\boldsymbol{s}}_{,\eta} + \dot{\boldsymbol{r}} \cdot \dot{\boldsymbol{s}}_{,\xi\eta} \\ \dot{\boldsymbol{s}}_{,\eta} \cdot \dot{\boldsymbol{s}}_{,\xi} + \dot{\boldsymbol{r}} \cdot \dot{\boldsymbol{s}}_{,\eta\xi} & \|\dot{\boldsymbol{s}}_{,\eta}\|^2 + \dot{\boldsymbol{r}} \cdot \dot{\boldsymbol{s}}_{,\eta\eta} \end{bmatrix} \tag{2.43}$$

In (2.43), the dependencies of $\dot{\boldsymbol{r}}(\xi_i,\eta_i)$ and $\dot{\boldsymbol{s}}(\xi_i,\eta_i)$ are omitted for readability. The change of the coordinates from one consecutive iteration to another is denoted by $\boldsymbol{\delta}_i$:

$$\boldsymbol{\delta}_i = \begin{bmatrix} \Delta\xi_i \\ \Delta\eta_i \end{bmatrix} = \begin{bmatrix} \xi_{i+1} - \xi_i \\ \eta_{i+1} - \eta_i \end{bmatrix} \tag{2.44}$$

The problem forms a 2×2 system of linear equations of the form $\boldsymbol{J}_i\boldsymbol{\delta}_i = -\boldsymbol{\kappa}_i$. However, the procedure requires the first and the second derivatives of the t-spline surface to be calculated. As the control points are not functions of the parametric coordinates, the first derivatives of a t-spline surface can be expressed in terms of derivatives of the bivariate rational basis functions as

$$\dot{\boldsymbol{s}}(\xi,\eta)_{,\xi} = \sum_{A=1}^{n_T} R_A(\xi,\eta)_{,\xi}\dot{\boldsymbol{x}}_A \tag{2.45}$$

and

$$\dot{\boldsymbol{s}}(\xi,\eta)_{,\eta} = \sum_{A=1}^{n_T} R_A(\xi,\eta)_{,\eta}\dot{\boldsymbol{x}}_A \tag{2.46}$$

A bivariate rational basis function of a t-spline surface can be written as

$$R_A(\xi,\eta) = \frac{w_A N_A^{\xi}(\xi) N_A^{\eta}(\eta)}{W(\xi,\eta)} \tag{2.47}$$

where the denominator of (2.37) is denoted by $W(\xi,\eta)$. The ξ-derivative of a rational basis function is derived by applying the quotient rule:

$$R_A(\xi,\eta)_{,\xi} = \frac{w_A N_A^{\eta}(\eta) N_A^{\xi}(\xi)_{,\xi} - R_A(\xi,\eta) W(\xi,\eta)_{,\xi}}{W(\xi,\eta)} \tag{2.48}$$

After applying the product and the quotient rule to the first derivative, the second derivatives can be written as

$$R_A(\xi,\eta)_{,\xi\xi} = \frac{w_A N_A^{\eta}(\eta) N_A^{\xi}(\xi)_{,\xi\xi} - W(\xi,\eta)_{,\xi\xi} R_A(\xi,\eta) - 2W(\xi,\eta)_{,\xi} R_A(\xi,\eta)_{,\xi}}{W(\xi,\eta)} \tag{2.49}$$

and

$$\begin{aligned} R_A(\xi,\eta)_{,\xi\eta} = & \frac{w_A N_A^{\xi}(\xi)_{,\xi} N_A^{\eta}(\eta)_{,\eta} - W(\xi,\eta)_{,\xi\eta} R_A(\xi,\eta)}{W(\xi,\eta)} \\ & - \frac{w_A N_A^{\eta}(\eta) N_A^{\xi}(\xi)_{,\xi} W(\xi,\eta)_{,\eta} + w_A N_A^{\xi}(\xi) N_A^{\eta}(\eta)_{,\eta} W(\xi,\eta)_{,\xi}}{W(\xi,\eta)^2} \\ & + \frac{2W(\xi,\eta)_{,\xi} W(\xi,\eta)_{,\eta} R_A(\xi,\eta)}{W(\xi,\eta)^2} \end{aligned} \tag{2.50}$$

The derivatives $R_A(\xi,\eta)_{,\eta}$, $R_A(\xi,\eta)_{,\eta\eta}$ and $R_A(\xi,\eta)_{,\eta\xi}$ are derived similarly. However, the cross partial derivative only needs to be calculated once since $R_A(\xi,\eta)_{,\eta\xi} = R_A(\xi,\eta)_{,\xi\eta}$.

Chapter 3

Analysis

This chapter presents the foundations for the analysis. In solid mechanics, balance equations such as the conservation of mass, linear momentum, and energy are typically used as a basis for the analysis. Section 3.1 demonstrates the Galerkin method, which is often used for approximating the solution of the underlying balance equations. Section 3.2 discusses numerical integration. Explicit and implicit approaches for time integration are demonstrated by common time integration schemes, and the importance of the quadrature rule in solving the domain and boundary integrals is weighed.

Section 3.3 presents three closely related but different isogeometric shell elements—two of which are shear deformable. The shell elements adopt certain fundamental ideas from the fastest and most robust classical shell formulations, but pursue to solve certain IGA-related complexities by applying the latest technological innovations. The stress integration is discussed separately in Section 3.4.

Section 3.5 considers the application of loads and the enforcement of boundary conditions at an arbitrary location within a patch. Finally, Section 3.6 discusses contact treatment in the context of IGA and proposes some application-specific techniques for contact modeling in ISF simulations.

3.1 Galerkin method

In this section, the balance equations for the conservation of linear momentum as well as energy are used as the basis for a linear thermomechanical model. The strong

form of the balance equations is converted into a weak form by introducing a test function and integrating over the domain. In Bubnov-Galerkin method, the test function and the primary variable are both approximated by the same interpolation functions which often leads to symmetric matrices. Furthermore, the test function is herein obtained as a variation of the primary variable.

3.1.1 Conservation of linear momentum

The balance of linear momentum can be written as (Belytschko et al., 2013)

$$\rho \ddot{\boldsymbol{u}} = \operatorname{div} \boldsymbol{\sigma} + \boldsymbol{b} \tag{3.1}$$

where ρ is the material density, $\ddot{\boldsymbol{u}}$ is the acceleration, i.e., the second derivative of the displacement $\boldsymbol{u}$ with respect to time, and $\boldsymbol{b}$ contains all body forces per unit volume. In the context of linear thermoelasticity and isotropic media, the Cauchy stress $\boldsymbol{\sigma}$ is given by

$$\boldsymbol{\sigma} = 2\mu\boldsymbol{\varepsilon} + \lambda\big(\operatorname{tr}(\boldsymbol{\varepsilon}) - \alpha_\theta\theta\big)\boldsymbol{I} \tag{3.2}$$

where λ and μ are Lamé's first and second parameters, the temperature change $\theta = T - T_0$ is defined in terms of the current and the constant initial absolute temperatures T and T_0, respectively, α_θ is the volumetric coefficient of thermal expansion, and $\boldsymbol{I}$ is the identity tensor. The infinitesimal strain $\boldsymbol{\varepsilon}$ is a function of displacement $\boldsymbol{u}$ and is given by

$$\boldsymbol{\varepsilon} = \frac{1}{2}\big(\boldsymbol{u}_{,\boldsymbol{x}} + \boldsymbol{u}_{,\boldsymbol{x}}^T\big) \tag{3.3}$$

where $\boldsymbol{x}$ is the position vector.

The weak form can be obtained by introducing a variation $\delta\boldsymbol{u}$ known as a virtual displacement, and integrating over the domain Ω. This is known as the principle of virtual work. The weak form of (3.1) is given below:

$$\int_\Omega \delta\boldsymbol{u} \cdot \rho\ddot{\boldsymbol{u}}\,\mathrm{d}\Omega - \int_\Omega \delta\boldsymbol{u} \cdot \operatorname{div}\boldsymbol{\sigma}\,\mathrm{d}\Omega = \int_\Omega \delta\boldsymbol{u} \cdot \boldsymbol{b}\,\mathrm{d}\Omega \tag{3.4}$$

Considering the divergence theorem of Gauss, one may express the term consisting of the divergence of the stress $\boldsymbol{\sigma}$ as

$$\int_\Omega \delta\boldsymbol{u} \cdot \operatorname{div}\boldsymbol{\sigma}\,\mathrm{d}\Omega = -\int_\Omega \delta\boldsymbol{\varepsilon} : \boldsymbol{\sigma}\,\mathrm{d}\Omega + \int_\Gamma \delta\boldsymbol{u} \cdot \boldsymbol{t}\,\mathrm{d}\Gamma \tag{3.5}$$

where $\boldsymbol{u}_{,\boldsymbol{x}}$ is decomposed into its symmetric and skew-symmetric parts after which it is noted that a double dot product between a skew-symmetric and symmetric tensor is zero, i.e.,

$$\delta\boldsymbol{u}_{,\boldsymbol{x}} : \boldsymbol{\sigma} = \frac{1}{2}\left(\delta\boldsymbol{u}_{,\boldsymbol{x}} + \delta\boldsymbol{u}_{,\boldsymbol{x}}^T\right) : \boldsymbol{\sigma} + \underbrace{\frac{1}{2}\left(\delta\boldsymbol{u}_{,\boldsymbol{x}} - \delta\boldsymbol{u}_{,\boldsymbol{x}}^T\right) : \boldsymbol{\sigma}}_{=0} = \delta\boldsymbol{\varepsilon} : \boldsymbol{\sigma} \tag{3.6}$$

Furthermore, the last term of (3.5) represents a traction boundary condition along the boundary Γ (Zienkiewicz et al., 2013). The traction $\boldsymbol{t}$ is defined as

$$\boldsymbol{t} = \boldsymbol{n}_\Gamma \cdot \boldsymbol{\sigma} \tag{3.7}$$

where $\boldsymbol{n}_\Gamma$ is the unit normal vector of the boundary. The equation of the weak form can now be written as

$$\underbrace{\int_\Omega \delta\boldsymbol{u} \cdot \rho\ddot{\boldsymbol{u}}\,\mathrm{d}\Omega}_{W_{iner}} + \underbrace{\int_\Omega \delta\boldsymbol{\varepsilon} : \boldsymbol{\sigma}\,\mathrm{d}\Omega}_{W_{int}} = \underbrace{\int_\Omega \delta\boldsymbol{u} \cdot \boldsymbol{b}\,\mathrm{d}\Omega + \int_\Gamma \delta\boldsymbol{u} \cdot \boldsymbol{t}\,\mathrm{d}\Gamma}_{W_{ext}} \tag{3.8}$$

where the inertial, internal, and external virtual work contributions are denoted by W_{iner}, W_{int}, and W_{ext}, respectively.

The space domain is discretized by approximating the displacements $\delta\boldsymbol{u}$ and $\boldsymbol{u}$ by the shape functions N_A^u and control point displacements $\delta\boldsymbol{u}_A$ and $\boldsymbol{u}_A$, i.e.,

$$\delta\boldsymbol{u} \approx \sum_A N_A^u \delta\boldsymbol{u}_A, \quad \boldsymbol{u} \approx \sum_A N_A^u \boldsymbol{u}_A \tag{3.9}$$

The discretized weak form of the equation becomes

$$\begin{aligned}\sum_A \delta\boldsymbol{u}_A \cdot \underbrace{\int_\Omega \textstyle\sum_B \rho N_A^u N_B^u \ddot{\boldsymbol{u}}_B\,\mathrm{d}\Omega}_{\boldsymbol{f}^u_{inerA}} + \sum_A \delta\boldsymbol{u}_A \cdot \underbrace{\int_\Omega N^u_{A,\boldsymbol{x}} \cdot \boldsymbol{\sigma}\,\mathrm{d}\Omega}_{\boldsymbol{f}^u_{intA} - \boldsymbol{f}^\theta_{intA}} \\ = \sum_A \delta\boldsymbol{u}_A \cdot \underbrace{\int_\Omega N_A^u \boldsymbol{b}\,\mathrm{d}\Omega}_{\boldsymbol{f}^{ub}_{extA}} + \sum_A \delta\boldsymbol{u}_A \cdot \underbrace{\int_\Gamma N_A^u \boldsymbol{t}\,\mathrm{d}\Gamma}_{\boldsymbol{f}^{ut}_{extA}}\end{aligned} \tag{3.10}$$

where the control point virtual displacements $\delta\boldsymbol{u}_A$ are separated such that the integrals can now be perceived as control point forces. The second term of (3.10) is decomposed into displacement and temperature change dependent terms. The control point internal forces $\boldsymbol{f}^u_{intA}$ and $\boldsymbol{f}^\theta_{intA}$ for an isotropic linear thermoelastic material are now given by

$$\boldsymbol{f}^u_{intA} = \int_\Omega N^u_{A,\boldsymbol{x}} \cdot \left(\boldsymbol{C} : \boldsymbol{\varepsilon}\right)\mathrm{d}\Omega, \quad \boldsymbol{f}^\theta_{intA} = \int_\Omega \textstyle\sum_B \lambda\alpha_\theta N_B^\theta \theta_B N^u_{A,\boldsymbol{x}}\,\mathrm{d}\Omega \tag{3.11}$$

where the temperature change θ is approximated by the shape functions N_A^θ and control point temperature changes θ_A as $\theta \approx \sum_A N_A^\theta \theta_A$. The material tensor $\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}$ as well as the infinitesimal strain $\underset{\cdot\cdot}{\boldsymbol{\varepsilon}}$ are given in the index notation by

$$\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}} := C_{ijkl} = \lambda \delta_{ij}\delta_{kl} + \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) \tag{3.12}$$

and

$$\underset{\cdot\cdot}{\boldsymbol{\varepsilon}} := \varepsilon_{ij} = \sum_B \frac{1}{2}\big(N_{B,j}u_{iB} + N_{B,i}u_{jB}\big) \tag{3.13}$$

respectively, where δ_{ij} is the Kronecker delta function.

3.1.2 Conservation of energy

According to Belytschko et al. (2013), the energy balance equation for a thermomechanical process where the sources of energy consist of either mechanical work or heat, is given by

$$p_{int} = \underset{\cdot\cdot}{\boldsymbol{D}} : \underset{\cdot\cdot}{\boldsymbol{\sigma}} - \operatorname{div} \underset{\cdot}{\boldsymbol{w}} + r \tag{3.14}$$

where p_{int} is the internal power per unit volume, $\underset{\cdot}{\boldsymbol{w}}$ is the heat flow, r is the heat production per unit volume, and $\underset{\cdot\cdot}{\boldsymbol{D}}$ is the rate-of-deformation, or velocity strain, given by

$$\underset{\cdot\cdot}{\boldsymbol{D}} = \frac{1}{2}\big(\dot{\underset{\cdot}{\boldsymbol{u}}}_{,\underset{\cdot}{\boldsymbol{x}}} + \dot{\underset{\cdot}{\boldsymbol{u}}}^T_{,\underset{\cdot}{\boldsymbol{x}}}\big) \tag{3.15}$$

For a reversible process where the temperature change is small, Nicholson (2008) as well as Hetnarski and Eslami (2009) end up with the energy balance equation of the form

$$\rho c_p \dot{\theta} = -\lambda \alpha_\theta T_0 \operatorname{tr}(\underset{\cdot\cdot}{\boldsymbol{D}}) - \operatorname{div} \underset{\cdot}{\boldsymbol{w}} + r \tag{3.16}$$

where c_p is the specific heat capacity and $\dot{\theta}$ is the first time derivative of the temperature change $\theta = T - T_0$. If the response follows Fourier's law, the heat flow for an isotropic media can be written as

$$\underset{\cdot}{\boldsymbol{w}} = -k_\theta \, \theta_{,\underset{\cdot}{\boldsymbol{x}}} \tag{3.17}$$

where k_θ is the material conductivity coefficient. Equation (3.16) is a strong form representation of the energy balance equation. The weak form can be obtained

similarly to (3.4) by introducing a temperature variation $\delta\theta$ and integrating over the domain. The weak form is given as follows:

$$\int_\Omega \delta\theta \rho c_p \dot{\theta}\,\mathrm{d}\Omega = -\int_\Omega \delta\theta \lambda \alpha_\theta T_0 \,\mathrm{tr}(\dot{\boldsymbol{D}})\,\mathrm{d}\Omega - \int_\Omega \delta\theta \,\mathrm{div}\,\dot{\boldsymbol{w}}\,\mathrm{d}\Omega + \int_\Omega \delta\theta r\,\mathrm{d}\Omega \tag{3.18}$$

Considering the divergence theorem of Gauss similarly to (3.5), the term consisting of divergence of the heat flow can be written as

$$\int_\Omega \delta\theta \,\mathrm{div}\,\dot{\boldsymbol{w}}\,\mathrm{d}\Omega = -\int_\Omega \delta\theta_{,\boldsymbol{x}} \cdot \dot{\boldsymbol{w}}\,\mathrm{d}\Omega + \int_\Gamma \delta\theta \dot{\boldsymbol{n}}_\Gamma \cdot \dot{\boldsymbol{w}}\,\mathrm{d}\Gamma \tag{3.19}$$

where the last term represents a heat flow boundary condition along the boundary Γ. The heat flow boundary condition is often separated into a prescribed heat flow w, convection, and radiation yielding (Lewis et al., 2004; Dhondt, 2004)

$$\begin{aligned}\int_\Gamma \delta\theta \dot{\boldsymbol{n}}_\Gamma \cdot \dot{\boldsymbol{w}}\,\mathrm{d}\Gamma = \int_\Gamma \delta\theta w\,\mathrm{d}\Gamma &- \underbrace{\int_\Gamma \delta\theta h_\theta(\theta - \theta_{env})\,\mathrm{d}\Gamma}_{convection} \\ &- \underbrace{\int_\Gamma \delta\theta \epsilon_\theta \sigma_\theta \big((\theta + T_0)^4 - (\theta_{env} + T_0)^4\big)\,\mathrm{d}\Gamma}_{radiation}\end{aligned} \tag{3.20}$$

where h_θ is the convection heat transfer coefficient and θ_{env} is the environmental temperature change given by $\theta_{env} = T_{env} - T_0$ in which T_{env} is the absolute environmental temperature. Moreover, ϵ_θ and σ_θ in the radiation term are the surface emissivity and the Stefan-Boltzmann constant, respectively. However, the radiation boundary condition is quartic in θ causing a nonlinearity and is therefore not further considered for the linear thermoelastic model.

The temperature variation $\delta\theta$ and the temperature change θ are approximated by the shape functions N_A^θ to discretize the space domain similarly to (3.9):

$$\delta\theta \approx \sum_A N_A^\theta \delta\theta_A, \quad \theta \approx \sum_A N_A^\theta \theta_A \tag{3.21}$$

After substituting (3.20) into (3.19), and then (3.19) into (3.18), and neglecting the nonlinear radiation contribution, the discretized weak form of the equation becomes

$$\begin{aligned}&\sum_A \delta\theta_A \underbrace{\int_\Omega \textstyle\sum_B \rho c_p N_A^\theta N_B^\theta \dot{\theta}_B\,\mathrm{d}\Omega}_{f^\theta_{inerA}} + \sum_A \delta\theta_A \underbrace{\int_\Omega -N^\theta_{A,\boldsymbol{x}} \cdot \dot{\boldsymbol{w}}\,\mathrm{d}\Omega}_{f^\theta_{intA}} + \sum_A \delta\theta_A \underbrace{\int_\Omega \lambda\alpha_\theta T_0 N_A^\theta \,\mathrm{tr}(\dot{\boldsymbol{D}})\,\mathrm{d}\Omega}_{f^u_{intA}} \\ &= \sum_A \delta\theta_A \underbrace{\int_\Omega N_A^\theta r\,\mathrm{d}\Omega}_{f^{\theta r}_{extA}} + \sum_A \delta\theta_A \underbrace{\int_\Gamma N_A^\theta w\,\mathrm{d}\Gamma}_{f^{\theta w}_{extA}} - \sum_A \delta\theta_A \underbrace{\int_\Gamma \textstyle\sum_B N_A^\theta h_\theta (N_B^\theta \theta_B - \theta_{env})\,\mathrm{d}\Gamma}_{f^{\theta h}_{extA}}\end{aligned}$$

(3.22)

where

$$\dot{\boldsymbol{w}} = \sum_B -k_\theta N^\theta_{B,\dot{\boldsymbol{x}}}\theta_B, \quad \mathrm{tr}(\dot{\boldsymbol{D}}) = \sum_B N_{B,\dot{\boldsymbol{x}}} \cdot \dot{\boldsymbol{u}}_B \tag{3.23}$$

3.1.3 Linear thermoelastic coupling

The equations (3.10) and (3.22) form the following system of equations:

$$\begin{aligned}
\sum_A \delta\dot{\boldsymbol{u}}_A \cdot \Big(\underbrace{\textstyle\sum_B M^u_{AB}\ddot{\boldsymbol{u}}_B}_{\dot{\boldsymbol{f}}^u_{inerA}} + \underbrace{\textstyle\sum_B \dot{\boldsymbol{K}}^u_{AB} \cdot \boldsymbol{u}_B}_{\dot{\boldsymbol{f}}^u_{intA}} - \underbrace{\textstyle\sum_B \dot{\boldsymbol{\Omega}}^u_{AB}\theta_B}_{\dot{\boldsymbol{f}}^\theta_{intA}} \Big) &= \sum_A \delta\dot{\boldsymbol{u}}_A \cdot \dot{\boldsymbol{f}}^u_{extA} \\
\sum_A \delta\theta_A \Big(\underbrace{\textstyle\sum_B M^\theta_{AB}\dot{\theta}_B}_{f^\theta_{inerA}} + \underbrace{\textstyle\sum_B K^\theta_{AB}\theta_B}_{f^{\theta h}_{intA}} + \underbrace{\textstyle\sum_B \dot{\boldsymbol{\Omega}}^\theta_{AB} \cdot \dot{\boldsymbol{u}}_B}_{f^u_{intA}} \Big) &= \sum_A \delta\theta_A f^\theta_{extA}
\end{aligned} \tag{3.24}$$

The components of the mechanical part are given in the index form by

$$\begin{aligned}
M^u_{AB} &= \int_\Omega \rho N^u_A N^u_B \,\mathrm{d}\Omega \\
K^u_{ijAB} &= \int_\Omega \sum_{k,l,h} \frac{1}{2} N^u_{A,k} C_{iklh} \big(\delta_{jl} N^u_{B,h} + \delta_{jh} N^u_{B,l}\big) \,\mathrm{d}\Omega \\
\Omega^u_{iAB} &= \int_\Omega \lambda\alpha_\theta N^\theta_B N^u_{A,i} \,\mathrm{d}\Omega \\
f^u_{ext\,iA} &= \int_\Omega N^u_A b_i \,\mathrm{d}\Omega + \int_\Gamma N^u_A t_i \,\mathrm{d}\Gamma
\end{aligned} \tag{3.25}$$

and the components of the thermal part are written similarly as

$$\begin{aligned}
M^\theta_{AB} &= \int_\Omega \rho c_p N^\theta_A N^\theta_B \,\mathrm{d}\Omega \\
K^\theta_{AB} &= \int_\Omega \sum_i k_\theta N^\theta_{A,i} N^\theta_{B,i} \,\mathrm{d}\Omega + \int_\Gamma h_\theta N^\theta_A N^\theta_B \,\mathrm{d}\Gamma \\
\Omega^\theta_{iAB} &= \int_\Omega \lambda\alpha_\theta T_0 N^\theta_A N^u_{B,i} \,\mathrm{d}\Omega \\
f^\theta_{extA} &= \int_\Omega N^\theta_A r \,\mathrm{d}\Omega + \int_\Gamma N^\theta_A w \,\mathrm{d}\Gamma + \int_\Gamma h_\theta N^\theta_A \theta_{env} \,\mathrm{d}\Gamma
\end{aligned} \tag{3.26}$$

3.2 Numerical integration

This section discusses explicit and implicit numerical time integration schemes by considering the linear thermomechanical coupling derived in Section 3.1. Furthermore, evaluation of the domain and boundary integrals using a numerical quadrature is considered.

3.2.1 Explicit time integration

In the explicit time integration methods, the state of a system at a later time is calculated considering only the state of the system at the current time. In the thermomechanical coupling system (3.24), the first equation is a second-order partial differential equation whereas the second equation is a first-order partial differential equation. In the context of explicit time integration, these type of equations are often solved using central and forward difference methods (Koric et al., 2009).

Furthermore, the mass matrices M^u_{AB} and M^θ_{AB} are usually lumped, or diagonalized, to make the cost associated with the matrix inversion insignificant. Typical simple lumping techniques are row or column summing, but also other methods are used (Wu and Gu, 2012). The physical implication of a lumped mass matrix is that all the mass is located at the control points. This simplification affects the dynamic response of the system to small extent, but due to massive computational savings this procedure is widely adopted in the context of explicit methods.

The primary variables $\underset{\cdot}{\boldsymbol{u}}_B$ and θ_B in the thermoelastic coupling are updated by first arranging the unknown terms representing mechanical and thermal inertias to the left-hand side, and all known terms to the right-hand side as follows:

$$\begin{aligned} \sum_B M^u_{AB} \ddot{\underset{\cdot}{\boldsymbol{u}}}_B &= \underset{\cdot}{\boldsymbol{f}}^u_{extA} - \underset{\cdot}{\boldsymbol{f}}^u_{intA} + \underset{\cdot}{\boldsymbol{f}}^\theta_{intA} \\ \sum_B M^\theta_{AB} \dot{\theta}_B &= f^\theta_{extA} - f^{\theta h}_{intA} - f^u_{intA} \end{aligned} \tag{3.27}$$

Equation (3.27) is solved by finding the inverses of M^u_{AB} and M^θ_{AB}; this is trivial if these matrices are diagonal. Using the aforementioned central and forward differences, one obtains the updated displacement and temperature change by

$$\begin{aligned} \dot{\underset{\cdot}{\boldsymbol{u}}}_B^{t+\frac{1}{2}} &= \dot{\underset{\cdot}{\boldsymbol{u}}}_B^{t-\frac{1}{2}} + \Delta t^t \ddot{\underset{\cdot}{\boldsymbol{u}}}_B^t \\ \underset{\cdot}{\boldsymbol{u}}_B^{t+1} &= \underset{\cdot}{\boldsymbol{u}}_B^t \ + \Delta t^{t+\frac{1}{2}} \dot{\underset{\cdot}{\boldsymbol{u}}}_B^{t+\frac{1}{2}} \\ \theta_B^{t+1} &= \theta_B^t + \Delta t^{t+\frac{1}{2}} \dot{\theta}_B^t \end{aligned} \tag{3.28}$$

where Δt is the time increment, t and $t+1$ in the superscript denote the current and the next time step, and $t \pm \frac{1}{2}$ is used for the mid-point values. The central difference time integration scheme is discussed in more detail in the context of shell elements in Section 3.3.8.

Knowing $\underset{\cdot}{\boldsymbol{u}}_B^{t+1}$ and θ_B^{t+1} allows calculating the internal and external forces for the corresponding time step, i.e., the right-hand side of (3.27), after which the acceleration

and the rate of change of temperature can be solved again to start repeating the procedure.

However, the time increment size for explicit time integration methods is limited by the Courant-Friedrichs-Lewy (CFL) stability condition (Courant et al., 1928) leading to very small time increments in most applications. The time increment size for a purely mechanical response is related to the maximum natural frequency of the system ω by

$$\Delta t < \frac{2}{\omega} \tag{3.29}$$

if the central difference time integration scheme is used. The time step requirements for the thermal response are usually less restrictive, and, in most cases, permit many orders of magnitude larger stable time steps.

It is further noted that the diagonalization of a mass matrix typically lowers the maximum natural frequency of the system, and therefore, often slightly increases the maximum stable time increment size (Adam et al., 2015b).

3.2.2 Implicit time integration

In contrast to explicit methods, implicit time integration methods update the state of a system such that the unknown later state is taken into consideration when forming the equilibrium equations. A common choice in the context of implicit time integration schemes is Newmark's (1959) method which approximates the differential equations as follows:

$$\begin{aligned} \dot{\boldsymbol{u}}^{t+1} &\approx \dot{\boldsymbol{u}}^{t} + (1-\gamma)\ddot{\boldsymbol{u}}^{t}\Delta t + \gamma\ddot{\boldsymbol{u}}^{t+1}\Delta t \\ \boldsymbol{u}^{t+1} &\approx \boldsymbol{u}^{t} + \dot{\boldsymbol{u}}^{t}\Delta t + (\frac{1}{2}-\beta)\ddot{\boldsymbol{u}}^{t}\Delta t^2 + \beta\ddot{\boldsymbol{u}}^{t+1}\Delta t^2 \end{aligned} \tag{3.30}$$

The parameters β and γ determine the behavior of the method (de Borst et al., 2012). Unconditional stability is achieved for linear systems if

$$2\beta \geq \gamma \geq \frac{1}{2} \tag{3.31}$$

in which case the method remains stable for any time step size. The unknown acceleration $\ddot{\boldsymbol{u}}^{t+1}$ and velocity $\dot{\boldsymbol{u}}^{t+1}$ are functions of the unknown displacement $\boldsymbol{u}^{t+1}$, as well as the known current acceleration $\ddot{\boldsymbol{u}}^{t}$, velocity $\dot{\boldsymbol{u}}^{t}$, and displacement $\boldsymbol{u}^{t}$. Rearranging (3.30_2), one obtains the acceleration $\ddot{\boldsymbol{u}}^{t+1}$:

$$\ddot{\boldsymbol{u}}^{t+1} = \underbrace{\frac{1}{\beta\Delta t^2}}_{\bar{\beta}}\boldsymbol{u}^{t+1} \underbrace{-\frac{1}{\beta\Delta t^2}\boldsymbol{u}^{t} - \frac{1}{\beta\Delta t}\dot{\boldsymbol{u}}^{t} - \left(\frac{1}{2\beta}-1\right)\ddot{\boldsymbol{u}}^{t}}_{\boldsymbol{\zeta}} \tag{3.32}$$

The velocity $\dot{\boldsymbol{u}}^{t+1}$ is obtained by inserting (3.32) into (3.30_1):

$$\dot{\boldsymbol{u}}^{t+1} = \underbrace{\frac{\gamma}{\beta\Delta t}}_{\bar{\beta}} \boldsymbol{u}^{t+1} \underbrace{-\frac{\gamma}{\beta\Delta t}\boldsymbol{u}^{t} - \left(\frac{\gamma}{\beta} - 1\right)\dot{\boldsymbol{u}}^{t} - \left(\frac{\gamma}{2\beta} - 1\right)\ddot{\boldsymbol{u}}^{t}\Delta t}_{\boldsymbol{\chi}} \tag{3.33}$$

Denoting the terms known at the current time by $\boldsymbol{\zeta}$ and $\boldsymbol{\chi}$, one may express the linear thermomechanical coupling system (3.24) in the following form:

$$\begin{aligned} \boldsymbol{r}^{u}_{A} &= \textstyle\sum_B M^{u}_{AB}(\bar{\bar{\beta}}\boldsymbol{u}_B + \boldsymbol{\zeta}^{u}_{B}) + \sum_B \boldsymbol{K}^{u}_{AB} \cdot \boldsymbol{u}_B - \sum_B \boldsymbol{\Omega}^{u}_{AB}\theta_B - \boldsymbol{f}^{u}_{extA} = 0 \\ r^{\theta}_{A} &= \textstyle\sum_B M^{\theta}_{AB}(\bar{\beta}\theta_B + \chi^{\theta}_{B}) + \sum_B K^{\theta}_{AB}\theta_B + \sum_B \boldsymbol{\Omega}^{\theta}_{AB} \cdot (\bar{\beta}\boldsymbol{u}_B + \boldsymbol{\chi}^{u}_{B}) - f^{\theta}_{extA} = 0 \end{aligned} \tag{3.34}$$

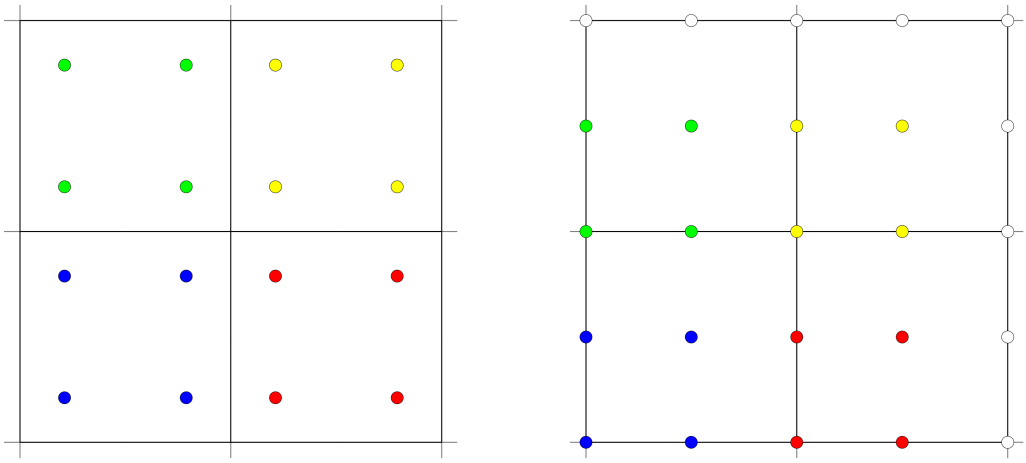
This system of equations is often solved by the iterative Newton-Raphson method which requires finding all partial derivatives of the force residuals $\boldsymbol{r}^{u}_{A}$ and r^{θ}_{A} with respect to the primary variables $\boldsymbol{u}_B$ and θ_B. For the linear thermoelastic coupling these derivatives are constant and simply given in the index notation by

$$\begin{aligned} r^{u}_{iA,u_{jB}} &= \bar{\bar{\beta}} M^{u}_{AB} + K^{u}_{ijAB} \\ r^{u}_{iA,\theta_B} &= -\Omega^{u}_{iAB} \\ r^{\theta}_{A,u_{iB}} &= \bar{\beta}\Omega^{\theta}_{iAB} \\ r^{\theta}_{A,\theta_B} &= \bar{\beta} M^{\theta}_{AB} + K^{\theta}_{AB} \end{aligned} \tag{3.35}$$

Therefore, only one iteration is required for the underlying system of linear equations. However, the Newton-Raphson method is still applicable even if nonlinearities need be considered, which is usually the case with large displacements or temperature changes, advanced material models, and certain types of boundary conditions such as contact or radiation.

3.2.3 Domain and boundary integrals

The definite domain and boundary integrals in the problem formulation arising from the Galerkin method are solved numerically by calculating the integrand value at the specific locations called integration points. The placement and quantity of these points are crucial in determining the accuracy of the solution. The integration points are specified by the quadrature rule which therefore defines the resulting integration point scheme for the corresponding domain. For the Lagrange finite elements, Gauss-Legendre quadrature rules are often used to determine the integration points. A one-dimensional Gauss-Legendre quadrature with N integration points provides



(a) Four-point Gauss-Legendre quadrature.

(b) Four-point Gauss-Lobatto quadrature.

Figure 3.1: Two-dimensional Gauss-Legendre and Gauss-Lobatto quadratures.

the exact solution for polynomials up to the degree $2N - 1$. A two-dimensional Gauss-Legendre quadrature with 2 points per parametric direction is shown in Figure 3.1a.

In the context of solid mechanics, if Lagrange elements are used, the definite integrals need be numerically evaluated for each element separately as the integrand function is discontinuous across the elements. Therefore, the quadratures of adjacent Lagrange elements are always independent of each other. On the other hand, in isogeometric analysis, the integrand function may be continuous throughout the whole patch. A patch consists of a number of Bézier elements which are determined by the knot locations defined in the parametric space. Furthermore, the definite integrals typically need not be evaluated separately for each Bézier element and one integration point may contribute to multiple elements.

A good example to demonstrate the advantages originating from a continuous integrand function is the Gauss-Lobatto quadrature shown in Figure 3.1b. For Lagrange finite elements, this rule requires nine integration point evaluations per element as the points located at the edges must be evaluated separately for each adjacent element. However, in IGA, the value calculated at the edge can often be shared between the neighboring elements in which case the number of integration point evaluations approaches only four for a large mesh.

Quadratures are often categorized into full and reduced integration rules depending on their accuracy. Full integration refers to exact integration in the case of a

polynomial integrand function (most quadratures are not able to exactly integrate rational functions). In the context of Lagrange elements, the integrand is typically polynomial if the transformation from the parametric domain to the physical domain is affine.

In IGA, the basis functions may be rational instead of polynomial. However, a quadrature resulting in an approximate integration of the corresponding rational function is still considered a full integration rule, provided that it exactly integrates the corresponding polynomial numerator of the respective rational function. This interpretation rests on the fact that the denominator of a rational spline basis function changes slowly compared to the numerator (Hughes et al., 2010).

If the quadrature does not result in full integration, it is referred to as a reduced integration rule. In the context of explicit analyses, the main advantage of reduced integration is the reduced computational cost. Another advantage is avoiding locking problems often encountered with full integration (Belytschko et al., 2013). However, reduced integration often leads to appearance of spurious modes (also called hourglassing).

A third term—selective integration—also often comes up. It refers to a procedure where the calculation is split, and two distinct quadrature rules are used. In the context of solid mechanics, selective integration typically implies that the stress associated with the Lamé's first parameter is evaluated using a reduced quadrature whereas the stress associated with Lamé's second parameter remains fully integrated (Adam et al., 2015c). The motivation to use selective integration is to alleviate numerical locking without introducing spurious modes.

The main interest in terms of computational efficiency in ISF simulations is the calculation of the stiffness integral derived in Section 3.1. If the solution space of a one-dimensional polynomial shape function of degree p and regularity q is denoted by S_q^p, the target space of the stiffness integral in two parametric dimensions is $S_{q-1}^{2p} \otimes S_{q-1}^{2p}$. Therefore, the target space requirements for the commonly employed biquadratic and bicubic splines are $S_0^4 \otimes S_0^4$ and $S_1^6 \otimes S_1^6$, respectively. In 2010, Hughes et al. initiated a study on optimal quadrature rules for isogeometric analysis and proposed a global algorithm to solve for the minimum number of integration points to exactly integrate any target space. However, the algorithm is costly for large meshes and convergence for the nonlinear problem is not guaranteed. Later, improved techniques have been suggested by many authors including Auricchio et al.

(2012b), Adam et al. (2015c), Ait-Haddou et al. (2015), Barto and Calo (2016), Barto et al. (2017), Johannessen (2017), and Hiemstra et al. (2017).

However, full integration is somewhat costly and often leads to strong numerical locking. Therefore, reduced quadratures have been proposed by several authors (Schillinger et al., 2014; Adam et al., 2015c; Hiemstra et al., 2017). A common practice is to determine the optimal quadrature for an approximation space that differs from the corresponding target space, and use that quadrature as a reduced integration rule for the actual target space. Adam et al. (2015c) suggested using the optimal quadratures for $S_0^2 \otimes S_0^2$ and $S_1^3 \otimes S_1^3$ spaces for biquadratic and bicubic splines, respectively, whereas Hiemstra et al. (2017) instead suggested the optimal quadratures for $S_0^3 \otimes S_0^3$ and $S_1^5 \otimes S_1^5$ spaces for the same spline interpolations.

A comprehensive overview of several quadrature rules proposed for isogeometric analysis in the literature was conducted by Hokkanen and Pedroso (2019b), who used an advanced ISF simulation as a benchmark problem (see Appendix B).

3.3 Isogeometric shell technology

As the main emphasis of this study is on the computational efficiency and robustness, a natural starting point for the shell formulation is the current state of fast explicit solver codes. According to Hallquist (2006) and Belytschko et al. (2013), one of the most efficient and robust explicit shell formulation is the widely adopted Belytschko-Tsay (Belytschko et al., 1984) element based on the Reissner-Mindlin theory. The main weakness of this element is its inability to correctly model twist, i.e., warped element configuration. Belytschko-Wong-Chiang (BWC) element (Belytschko et al., 1992) corrects this flaw but slightly increases the computational cost. These four-noded elements use bilinear interpolation functions and reduced integration. Hourglassing problem resulting from reduced integration is cured by introducing an artificial stiffness for each zero energy mode.

Concepts such as using the principle of virtual power instead of the principle of virtual work, evaluating calculations in appropriate local coordinate systems, and neglecting certain higher order terms for efficiency are adopted from the aforementioned classical formulations. Based on these ideas, an isogeometric solid-like shell element was developed by Hokkanen and Pedroso (2019a) and is given as Appendix A. The purpose of this section is, however, to develop shell formulations based on Kirchhoff-Love

and Reissner-Mindlin theories.

Several Kirchhoff-Love formulations are proposed in the literature. Kiendl et al. (2009, 2015b) developed an isogeometric Kirchhoff-Love formulation where the displacements are used as the primary variables and the second derivatives of the shape functions are explicitly calculated to model the bending response. An application to fluid-structure interaction simulations of bioprosthetic heart valves was demonstrated by Hsu et al. (2015) and the extension to elasto-plastic materials was discussed by Ambati et al. (2018). Echter et al. (2013) proposed a hierarchic family of isogeometric shell elements where a Kirchhoff-Love type shell model is used as a basis and additional degrees of freedom are added to a represent shear and thickness deformations. This concept was later further developed by Oesterle et al. (2016, 2017).

A Kirchhoff-Love formulation where the primary variables represent a velocity was proposed by Benson et al. (2011). This formulation constructs representative control point normals, and as a consequence, the second derivatives of the shape functions need not be explicitly calculated. However, the authors indicate that the formulation is susceptible to shear locking. Later, Benson et al. (2013) proposed a blended shell formulation where the Kirchhoff-Love shell is used for smooth regions and the Reissner-Mindlin assumption for sharp corners.

Furthermore, an isogeometric Reissner-Mindlin shell element for explicit dynamic simulations was proposed by Benson et al. (2010). The Reissner-Mindlin elements were also discussed in the context of IGA by Dornisch et al. (2013) and Adam et al. (2015a), among others.

3.3.1 The principle of virtual power

The balance of linear momentum is enforced following the Galerkin method presented in Section 3.1. However, the variation introduced herein represents a virtual velocity instead of a virtual displacement. This leads to the principle of virtual power (cf. virtual work). Analogously to the expression derived in Section 3.1 for the virtual work, the expression for the virtual power including inertial, internal, and external virtual power contributions takes the form

$$\underbrace{\int_\Omega \rho \dot{\boldsymbol{a}} \cdot \delta \dot{\boldsymbol{v}} \, \mathrm{d}\Omega}_{\delta P_{iner}} + \underbrace{\int_\Omega \dot{\boldsymbol{\sigma}} : \delta \dot{\boldsymbol{D}} \, \mathrm{d}\Omega}_{\delta P_{int}} = \underbrace{\int_\Omega \dot{\boldsymbol{b}} \cdot \delta \dot{\boldsymbol{v}} \, \mathrm{d}\Omega + \int_\Gamma \dot{\boldsymbol{t}} \cdot \delta \dot{\boldsymbol{v}} \, \mathrm{d}\Gamma}_{\delta P_{ext}} \tag{3.36}$$

where $\delta\dot{\boldsymbol{v}}$ is the virtual velocity and $\delta\dot{\boldsymbol{D}}$ is the virtual rate-of-deformation defined as the symmetric part of the spatial virtual velocity gradient $\delta\dot{\boldsymbol{v}}_{,\dot{\boldsymbol{x}}}$:

$$\delta\dot{\boldsymbol{D}} = \frac{1}{2}(\delta\dot{\boldsymbol{v}}_{,\dot{\boldsymbol{x}}} + \delta\dot{\boldsymbol{v}}_{,\dot{\boldsymbol{x}}}^T) \tag{3.37}$$

3.3.2 Shell kinematics

The kinematics of the proposed shell formulation are derived by first considering the continuum-based (CB) approach, also called degenerated continuum approach. The shell geometry is described by the reference surface coordinate $\dot{\boldsymbol{r}}(\xi,\eta,t)$ and the shell fiber vector $\dot{\boldsymbol{d}}(\xi,\eta,t)$ which remains straight during the deformation. This interpretation results in

$$\dot{\boldsymbol{x}}(\xi,\eta,\zeta,t) = \underbrace{\tfrac{1}{2}\big(\dot{\boldsymbol{x}}_{top}(\xi,\eta,t) + \dot{\boldsymbol{x}}_{bot}(\xi,\eta,t)\big)}_{\dot{\boldsymbol{r}}(\xi,\eta,t)} + \tfrac{1}{2}\zeta\underbrace{\big(\dot{\boldsymbol{x}}_{top}(\xi,\eta,t) - \dot{\boldsymbol{x}}_{bot}(\xi,\eta,t)\big)}_{\dot{\boldsymbol{d}}(\xi,\eta,t)} \tag{3.38}$$

where $\dot{\boldsymbol{x}}_{top}(\xi,\eta,t)$ represents the top surface and $\dot{\boldsymbol{x}}_{bot}(\xi,\eta,t)$ the bottom surface. Lines along the ζ axis are called fibers and a unit vector along a fiber is called a director (Belytschko et al., 2013).

However, in the Kirchhoff-Love shell theory, the fiber vector is required to remain normal to the reference surface of the shell, i.e., the surface formed by setting $\zeta = 0$. The shell geometry can now be written as

$$\dot{\boldsymbol{x}}(\xi,\eta,\zeta,t) = \dot{\boldsymbol{r}}(\xi,\eta,t) + \zeta\frac{h(\xi,\eta,t)}{2}\dot{\boldsymbol{n}}(\dot{\boldsymbol{r}}(\xi,\eta,t)) \tag{3.39}$$

where $h(\xi,\eta,t)$ is the shell thickness and $\dot{\boldsymbol{n}}(\dot{\boldsymbol{r}}(\xi,\eta,t))$ is the shell director which remains normal to the reference surface, i.e., the mid-surface unit normal. Taking the time derivative of (3.39), one ends up with the following expression for the velocity

$$\dot{\boldsymbol{v}} = \dot{\boldsymbol{r}} + \zeta\Big(\frac{h}{2}\dot{\boldsymbol{n}} + \frac{\dot{h}}{2}\boldsymbol{n}\Big) \tag{3.40}$$

where the function dependencies are omitted. The last term contributing to the thickness stretch is generally neglected in shell formulations enforcing the plane stress condition (Belytschko et al., 2013). Instead, the thickness change is calculated from the accumulated total strains. The resulting shell velocity definition for the Kirchhoff-Love shell element is referred to as "KL3" and simply given by

$$\dot{\boldsymbol{v}} = \dot{\boldsymbol{r}} + \zeta\frac{h}{2}\dot{\boldsymbol{n}} \tag{3.41}$$

One drawback of this definition is its inability to model transverse shear effects. Following the concept of hierarchical shells and the shear rotation introduced by Echter et al. (2013), the definition (3.41) is augmented by an angular shear velocity field to enhance the Kirchhoff-Love formulation (3.41) as

$$\dot{\boldsymbol{v}} = \dot{\boldsymbol{r}} + \zeta \frac{h}{2} \Big(\dot{\boldsymbol{n}} + \dot{\boldsymbol{\Psi}} \times \boldsymbol{n} \Big) \tag{3.42}$$

where $\dot{\boldsymbol{\Psi}}$ is the angular shear velocity contributing to the transverse shear effects. This shear enhanced Kirchhoff-Love formulation is referred to as "KL6". However, it is also possible to represent both, the velocity of the shell unit normal as well as the shear angle by an angular velocity field. This choice leads to a classical Reissner-Mindlin type shell formulation referred to as "RM6" and given by

$$\dot{\boldsymbol{v}} = \dot{\boldsymbol{r}} + \zeta \frac{h}{2} \dot{\boldsymbol{\omega}} \times \boldsymbol{n} \tag{3.43}$$

The equations in this section are derived for the KL6 shell formulation, i.e., Equation (3.42). However, switching to the KL3 or RM6 formulation is trivial and can be accomplished by simply setting $\dot{\boldsymbol{\Psi}}_A = \dot{\boldsymbol{f}}^{\Psi}_{intA} = \dot{\boldsymbol{f}}^{\Psi}_{extA} = 0$ or $\dot{\boldsymbol{n}}_A = \dot{\boldsymbol{f}}^{r1}_{intA} = \dot{\boldsymbol{f}}^{r1}_{extA} = 0$, respectively.

Although the velocity definitions (3.42) and (3.43) are inconsistent with the shell geometry definition (3.39), many shear deformable shell elements approximate the actual shell director of the deformed geometry by the unit normal to simplify the formulation and to increase its robustness (Benson et al., 2013; Belytschko et al., 2013). It should be noted that this approximation does not prevent transverse shear stresses from appearing as the velocity field still accumulates the transverse shear strains. Moreover, good accuracy is preserved for most applications, particularly, if the transverse shear strains are small. In fact, more accurate results may be obtained in some instances where the rotational inertias are scaled up and the response of the shell fiber vectors is slowed down as this could deteriorate the accuracy of the updated fiber vectors (Benson et al., 2010).

Furthermore, the developed solid-like shell formulation (Hokkanen and Pedroso, 2019a) which is capable of directly representing thickness changes and supports complete three-dimensional constitutive laws is referred to as "S7". This formulation does not require plane stress enforcement nor it assumes normality of the shell fiber vector to the reference surface.

3.3.3 Control point normals

In isogeometric analysis, the control points are not interpolatory, i.e., there are no unique material points associated with the control points. Therefore, the determination of representative control point normals is not trivial. The normals required by the developed formulations are constructed following the lifting operator method introduced by Benson et al. (2011). The normal of the control point A is defined as

$$\underset{\cdot}{\boldsymbol{n}}_A = \frac{\underset{\cdot}{\boldsymbol{p}}_A}{\|\underset{\cdot}{\boldsymbol{p}}_A\|} \tag{3.44}$$

where

$$\underset{\cdot}{\boldsymbol{p}}_A = \underset{\cdot}{\boldsymbol{t}}^{\xi}_A \times \underset{\cdot}{\boldsymbol{t}}^{\eta}_A \tag{3.45}$$

The vectors $\underset{\cdot}{\boldsymbol{t}}^{\xi}_A$ and $\underset{\cdot}{\boldsymbol{t}}^{\xi}_A$ are obtained from

$$\underset{\cdot}{\boldsymbol{t}}^{\xi}_A = \sum_B c^{\xi}_{AB} \underset{\cdot}{\boldsymbol{r}}_B, \qquad \underset{\cdot}{\boldsymbol{t}}^{\eta}_A = \sum_B c^{\eta}_{AB} \underset{\cdot}{\boldsymbol{r}}_B \tag{3.46}$$

Taking the time derivative of $\underset{\cdot}{\boldsymbol{n}}_A$, one ends up with the following definition for the normal velocity,

$$\dot{\underset{\cdot}{\boldsymbol{n}}}_A = \frac{1}{\|\underset{\cdot}{\boldsymbol{p}}_A\|}(\underset{\cdot\cdot}{\boldsymbol{I}} - \underset{\cdot}{\boldsymbol{n}}_A \otimes \underset{\cdot}{\boldsymbol{n}}_A) \cdot \dot{\underset{\cdot}{\boldsymbol{p}}}_A \tag{3.47}$$

where

$$\dot{\underset{\cdot}{\boldsymbol{p}}}_A = \dot{\underset{\cdot}{\boldsymbol{t}}}^{\xi}_A \times \underset{\cdot}{\boldsymbol{t}}^{\eta}_A + \underset{\cdot}{\boldsymbol{t}}^{\xi}_A \times \dot{\underset{\cdot}{\boldsymbol{t}}}^{\eta}_A \tag{3.48}$$

and

$$\dot{\underset{\cdot}{\boldsymbol{t}}}^{\xi}_A = \sum_B c^{\xi}_{AB} \dot{\underset{\cdot}{\boldsymbol{r}}}_B, \qquad \dot{\underset{\cdot}{\boldsymbol{t}}}^{\eta}_A = \sum_B c^{\eta}_{AB} \dot{\underset{\cdot}{\boldsymbol{r}}}_B \tag{3.49}$$

Furthermore, the virtual normal velocity is defined similarly to (3.47), i.e.,

$$\delta\dot{\underset{\cdot}{\boldsymbol{n}}}_A = \frac{1}{\|\underset{\cdot}{\boldsymbol{p}}_A\|}(\underset{\cdot\cdot}{\boldsymbol{I}} - \underset{\cdot}{\boldsymbol{n}}_A \otimes \underset{\cdot}{\boldsymbol{n}}_A) \cdot \delta\dot{\underset{\cdot}{\boldsymbol{p}}}_A \tag{3.50}$$

where

$$\delta\dot{\underset{\cdot}{\boldsymbol{p}}}_A = \delta\dot{\underset{\cdot}{\boldsymbol{t}}}^{\xi}_A \times \underset{\cdot}{\boldsymbol{t}}^{\eta}_A + \underset{\cdot}{\boldsymbol{t}}^{\xi}_A \times \delta\dot{\underset{\cdot}{\boldsymbol{t}}}^{\eta}_A \tag{3.51}$$

and

$$\delta\dot{\underset{\cdot}{\boldsymbol{t}}}^{\xi}_A = \sum_B c^{\xi}_{AB} \delta\dot{\underset{\cdot}{\boldsymbol{r}}}_B, \qquad \delta\dot{\underset{\cdot}{\boldsymbol{t}}}^{\eta}_A = \sum_B c^{\eta}_{AB} \delta\dot{\underset{\cdot}{\boldsymbol{r}}}_B \tag{3.52}$$

The coefficients c_{AB}^{ξ} and c_{AB}^{η} are determined such that the exact tangent vectors at the integration points given by

$$\dot{\boldsymbol{t}}^{\xi}(\dot{\boldsymbol{\xi}}_g) = \sum_A N_{A,\xi}(\dot{\boldsymbol{\xi}}_g)\dot{\boldsymbol{r}}_A, \qquad \dot{\boldsymbol{t}}^{\eta}(\dot{\boldsymbol{\xi}}_g) = \sum_A N_{A,\eta}(\dot{\boldsymbol{\xi}}_g)\dot{\boldsymbol{r}}_A \tag{3.53}$$

are required to be interpolants of the tangent vectors associated with the control points, i.e.,

$$\dot{\boldsymbol{t}}^{\xi}(\dot{\boldsymbol{\xi}}_g) = \sum_A N_A(\dot{\boldsymbol{\xi}}_g)\dot{\boldsymbol{t}}_A^{\xi}, \qquad \dot{\boldsymbol{t}}^{\eta}(\dot{\boldsymbol{\xi}}_g) = \sum_A N_A(\dot{\boldsymbol{\xi}}_g)\dot{\boldsymbol{t}}_A^{\xi} \tag{3.54}$$

The coefficients c_{AB}^{ξ} and c_{AB}^{η} are constant throughout the analysis and can be determined in advance by selecting $(p+1)^2$ points from the corresponding Bézier element and solving the following systems of linear equations,

$$\sum_A N_A(\dot{\boldsymbol{\xi}}_g)c_{AB}^{\xi} = N_{B,\xi}(\dot{\boldsymbol{\xi}}_g), \qquad \sum_A N_A(\dot{\boldsymbol{\xi}}_g)c_{AB}^{\eta} = N_{B,\eta}(\dot{\boldsymbol{\xi}}_g) \tag{3.55}$$

where $\dot{\boldsymbol{\xi}}_g$ represents the location of the selected point g. It is noted that the number of in-plane integration points is typically smaller than the number of points required to solve (3.55), in which case additional points must be chosen. Furthermore, the coefficients c_{AB}^{ξ} and c_{AB}^{η} are specific to the corresponding Bézier element, and therefore also $\dot{\boldsymbol{n}}_A$, $\dot{\boldsymbol{n}}_A$, and $\delta\dot{\boldsymbol{n}}_A$ vary from one element to another, or more specifically, are step functions of parametric coordinates ξ and η such that the value may only change across the element boundaries.

Other methods to determine the control point tangent vectors are also possible. The chosen lifting operator method was compared by Benson et al. (2011) to two other methods which extract the control point tangent vectors either from the point determined by the closest point projection, or from the point where the shape function reaches its maximum. However, the lifting operator method was shown to result in superior accuracy in comparison with the two other methods. Unfortunately, as the chosen lifting operator method makes the control point normals and normal velocities dependent on the Bézier element, the numerical integration can be performed at most one element at a time, i.e., the integration points cannot be shared between the adjacent elements. Therefore, the patchwise quadratures discussed in Appendix B are not directly applicable with this method.

3.3.4 Local coordinate system

A local orthonormal coordinate system is defined for each mid-surface integration point such that it approximately follows the material rotation. This type of coordinate

system is often referred to as a corotational coordinate system. The exact material rotation can be obtained from the deformation gradient $\underset{\cdot\cdot}{\boldsymbol{F}}$ which relates the current configuration $\underset{\cdot}{\boldsymbol{x}}$ to the initial configuration $\underset{\cdot}{\boldsymbol{x}}_0$:

$$\underset{\cdot\cdot}{\boldsymbol{F}} = \underset{\cdot}{\boldsymbol{x}}_{,\underset{\cdot}{\boldsymbol{x}}_0} = \underset{\cdot\cdot}{\boldsymbol{V}} \cdot \underset{\cdot\cdot}{\boldsymbol{R}} = \underset{\cdot\cdot}{\boldsymbol{R}} \cdot \underset{\cdot\cdot}{\boldsymbol{U}} \tag{3.56}$$

The polar decomposition of the deformation gradient gives the left or the right stretch tensor, $\underset{\cdot\cdot}{\boldsymbol{V}}$ or $\underset{\cdot\cdot}{\boldsymbol{U}}$, respectively, and the material rotation $\underset{\cdot\cdot}{\boldsymbol{R}}$ as follows:

$$\underset{\cdot\cdot}{\boldsymbol{V}} = \sqrt{\underset{\cdot\cdot}{\boldsymbol{F}} \cdot \underset{\cdot\cdot}{\boldsymbol{F}}^T}\,, \quad \underset{\cdot\cdot}{\boldsymbol{U}} = \sqrt{\underset{\cdot\cdot}{\boldsymbol{F}}^T \cdot \underset{\cdot\cdot}{\boldsymbol{F}}}\,, \quad \underset{\cdot\cdot}{\boldsymbol{R}} = \underset{\cdot\cdot}{\boldsymbol{V}}^{-1} \cdot \underset{\cdot\cdot}{\boldsymbol{F}} = \underset{\cdot\cdot}{\boldsymbol{F}} \cdot \underset{\cdot\cdot}{\boldsymbol{U}}^{-1} \tag{3.57}$$

However, the polar decomposition is somewhat costly and none of the bases of the resulting coordinate system are guaranteed to follow the surface unit normal which is undesirable for the plane stress enforcement. The original formulations proposed by Belytschko et al. (1984, 1992) construct the local coordinate system using Hughes-Liu's (1981) scheme which defines the orthonormal basis vectors $\underset{\cdot}{\boldsymbol{e}}_1$, $\underset{\cdot}{\boldsymbol{e}}_2$, and $\underset{\cdot}{\boldsymbol{e}}_3$ such that $\underset{\cdot}{\boldsymbol{e}}_1$ is aligned along ξ, i.e.,

$$\underset{\cdot}{\boldsymbol{e}}_1 = \frac{\underset{\cdot}{\boldsymbol{t}}^\xi}{\|\underset{\cdot}{\boldsymbol{t}}^\xi\|} \tag{3.58}$$

Then, $\underset{\cdot}{\boldsymbol{e}}_3$ is defined to be normal to the surface,

$$\underset{\cdot}{\boldsymbol{e}}_3 = \frac{\underset{\cdot}{\boldsymbol{e}}_1 \times \underset{\cdot}{\boldsymbol{t}}^\eta}{\|\underset{\cdot}{\boldsymbol{e}}_1 \times \underset{\cdot}{\boldsymbol{t}}^\eta\|} \tag{3.59}$$

and $\underset{\cdot}{\boldsymbol{e}}_2$ is required to be orthogonal to $\underset{\cdot}{\boldsymbol{e}}_1$ and $\underset{\cdot}{\boldsymbol{e}}_3$:

$$\underset{\cdot}{\boldsymbol{e}}_2 = \underset{\cdot}{\boldsymbol{e}}_3 \times \underset{\cdot}{\boldsymbol{e}}_1 \tag{3.60}$$

However, in the aforementioned scheme, $\underset{\cdot}{\boldsymbol{e}}_2$ is not generally aligned along $\underset{\cdot}{\boldsymbol{t}}^\eta$. To align both $\underset{\cdot}{\boldsymbol{e}}_1$ and $\underset{\cdot}{\boldsymbol{e}}_2$ to match with the tangent vector directions $\underset{\cdot}{\boldsymbol{t}}^\xi$ and $\underset{\cdot}{\boldsymbol{t}}^\eta$ as closely as possible, and to consequently improve the accuracy of the approximated material rotation, Hughes (2012) proposed another definition for the basis vectors. The basis vector normal to the surface does not change:

$$\underset{\cdot}{\boldsymbol{e}}_3 = \frac{\underset{\cdot}{\boldsymbol{t}}^\xi \times \underset{\cdot}{\boldsymbol{t}}^\eta}{\|\underset{\cdot}{\boldsymbol{t}}^\xi \times \underset{\cdot}{\boldsymbol{t}}^\eta\|} \tag{3.61}$$

The tangent vectors are normalized, i.e.,

$$\begin{aligned} \bar{\underset{\cdot}{\boldsymbol{t}}}^\xi &= \frac{\underset{\cdot}{\boldsymbol{t}}^\xi}{\|\underset{\cdot}{\boldsymbol{t}}^\xi\|} \\ \bar{\underset{\cdot}{\boldsymbol{t}}}^\eta &= \frac{\underset{\cdot}{\boldsymbol{t}}^\xi}{\|\underset{\cdot}{\boldsymbol{t}}^\xi\|} \end{aligned} \tag{3.62}$$

and auxiliary vectors are defined as

$$\begin{aligned} \underset{\cdot}{\boldsymbol{t}}_\alpha &= \frac{\bar{\underset{\cdot}{\boldsymbol{t}}}^\xi + \bar{\underset{\cdot}{\boldsymbol{t}}}^\eta}{\|\bar{\underset{\cdot}{\boldsymbol{t}}}^\xi + \bar{\underset{\cdot}{\boldsymbol{t}}}^\eta\|} \\ \underset{\cdot}{\boldsymbol{t}}_\beta &= \underset{\cdot}{\boldsymbol{e}}_3 \times \underset{\cdot}{\boldsymbol{t}}_\alpha \end{aligned} \tag{3.63}$$

Finally the basis vectors $\underset{\cdot}{\boldsymbol{e}}_1$ and $\underset{\cdot}{\boldsymbol{e}}_2$ are given by

$$\begin{aligned} \underset{\cdot}{\boldsymbol{e}}_1 &= \frac{\sqrt{2}}{2}(\underset{\cdot}{\boldsymbol{t}}_\alpha - \underset{\cdot}{\boldsymbol{t}}_\beta) \\ \underset{\cdot}{\boldsymbol{e}}_2 &= \frac{\sqrt{2}}{2}(\underset{\cdot}{\boldsymbol{t}}_\alpha + \underset{\cdot}{\boldsymbol{t}}_\beta) \end{aligned} \tag{3.64}$$

The material rotation is now approximated by

$$\boldsymbol{R} \approx [\boldsymbol{e}_1 \ \boldsymbol{e}_2 \ \boldsymbol{e}_3] \tag{3.65}$$

and the transformation from the global to the local coordinate system is given by

$$\hat{\underset{\cdot}{\boldsymbol{x}}} = \underset{\cdot\cdot}{\boldsymbol{R}}^T \cdot \underset{\cdot}{\boldsymbol{x}} \tag{3.66}$$

where the "hat" ($\hat{\bullet}$) is used to denote variables expressed in the local coordinate system.

3.3.5 Internal virtual power

To evaluate the internal virtual power of (3.36), i.e.,

$$\delta P_{int} = \int_\Omega \underset{\cdot\cdot}{\boldsymbol{\sigma}} : \delta \underset{\cdot\cdot}{\boldsymbol{D}} \, \mathrm{d}\Omega \tag{3.67}$$

one must first calculate the Cauchy stress $\underset{\cdot\cdot}{\boldsymbol{\sigma}}$. The Cauchy stress is obtained as a time integral of the Cauchy stress material derivative, which in turn is a function of the spatial velocity gradient $\underset{\cdot}{\boldsymbol{v}}_{,\underset{\cdot}{\boldsymbol{x}}}$. The spatial velocity gradient given by

$$\underset{\cdot}{\boldsymbol{v}}_{,\underset{\cdot}{\boldsymbol{x}}} = \dot{\underset{\cdot}{\boldsymbol{r}}}_{,\underset{\cdot}{\boldsymbol{x}}} + \Big(\zeta\frac{h}{2}\Big)_{,\underset{\cdot}{\boldsymbol{x}}}\Big(\dot{\underset{\cdot}{\boldsymbol{n}}} + \underset{\cdot}{\boldsymbol{\Psi}} \times \underset{\cdot}{\boldsymbol{n}}\Big) + \zeta\frac{h}{2}\Big(\dot{\underset{\cdot}{\boldsymbol{n}}} + \underset{\cdot}{\boldsymbol{\Psi}} \times \underset{\cdot}{\boldsymbol{n}}\Big)_{,\underset{\cdot}{\boldsymbol{x}}} \tag{3.68}$$

is calculated in the local coordinate system (3.66) where $\hat{\underset{\cdot}{\boldsymbol{x}}} = \hat{x}\hat{\underset{\cdot}{\boldsymbol{e}}}_1 + \hat{y}\hat{\underset{\cdot}{\boldsymbol{e}}}_2 + \hat{z}\hat{\underset{\cdot}{\boldsymbol{e}}}_3$. Noting that $\underset{\cdot}{\boldsymbol{r}}$, $\underset{\cdot}{\boldsymbol{n}}$, and $\underset{\cdot}{\boldsymbol{\Psi}}$ are independent of $\hat{z}$, and since ζ is not a function of $\hat{x}$ or $\hat{y}$, the components of the spatial velocity gradient in the local coordinate system are given as

$$\begin{aligned} \hat{\underset{\cdot}{\boldsymbol{v}}}_{,\chi} &= \dot{\hat{\underset{\cdot}{\boldsymbol{r}}}}_{,\chi} + \zeta\frac{h}{2}\Big(\dot{\hat{\underset{\cdot}{\boldsymbol{n}}}} + \hat{\underset{\cdot}{\boldsymbol{\Psi}}} \times \hat{\underset{\cdot}{\boldsymbol{n}}}\Big)_{,\chi} \\ \hat{\underset{\cdot}{\boldsymbol{v}}}_{,\hat{z}} &= \zeta_{,\hat{z}}\frac{h}{2}\Big(\dot{\hat{\underset{\cdot}{\boldsymbol{n}}}} + \hat{\underset{\cdot}{\boldsymbol{\Psi}}} \times \hat{\underset{\cdot}{\boldsymbol{n}}}\Big) \end{aligned} \tag{3.69}$$

where the derivatives of h are neglected (Belytschko et al., 2013) and χ represents either $\hat{x}$ or $\hat{y}$. The derivative $\zeta_{,\hat{z}}$ is obtained from (3.39) by first calculating the component along $\underset{\cdot}{\boldsymbol{e}}_3$ in the local coordinate system:

$$\hat{z} = \left(\hat{\underset{\cdot}{\boldsymbol{r}}} + \zeta\frac{h}{2}\hat{\underset{\cdot}{\boldsymbol{n}}}\right) \cdot \hat{\underset{\cdot}{\boldsymbol{e}}}_3 \tag{3.70}$$

Differentiating $\hat{z}$ with respect to ζ and noting that $\hat{\underset{\cdot}{\boldsymbol{n}}} \cdot \hat{\underset{\cdot}{\boldsymbol{e}}}_3 = 1$ yields

$$\hat{z}_{,\zeta} = \frac{h}{2} \tag{3.71}$$

hence,

$$\zeta_{,\hat{z}} = \frac{2}{h} \tag{3.72}$$

The spatial velocity gradient and the spatial virtual velocity gradient can now be written by

$$\begin{aligned} \hat{\underset{\cdot}{\boldsymbol{v}}}_{,\chi} &= \dot{\hat{\underset{\cdot}{\boldsymbol{r}}}}_{,\chi} + \zeta\frac{h}{2}\left(\dot{\hat{\underset{\cdot}{\boldsymbol{n}}}} + \hat{\underset{\cdot}{\boldsymbol{\Psi}}} \times \hat{\underset{\cdot}{\boldsymbol{n}}}\right)_{,\chi} \\ \hat{\underset{\cdot}{\boldsymbol{v}}}_{,\hat{z}} &= \dot{\hat{\underset{\cdot}{\boldsymbol{n}}}} + \hat{\underset{\cdot}{\boldsymbol{\Psi}}} \times \hat{\underset{\cdot}{\boldsymbol{n}}} \end{aligned} \tag{3.73}$$

and

$$\begin{aligned} \delta\hat{\underset{\cdot}{\boldsymbol{v}}}_{,\chi} &= \delta\dot{\hat{\underset{\cdot}{\boldsymbol{r}}}}_{,\chi} + \zeta\frac{h}{2}\left(\delta\dot{\hat{\underset{\cdot}{\boldsymbol{n}}}} + \delta\hat{\underset{\cdot}{\boldsymbol{\Psi}}} \times \hat{\underset{\cdot}{\boldsymbol{n}}}\right)_{,\chi} \\ \delta\hat{\underset{\cdot}{\boldsymbol{v}}}_{,\hat{z}} &= \delta\dot{\hat{\underset{\cdot}{\boldsymbol{n}}}} + \delta\hat{\underset{\cdot}{\boldsymbol{\Psi}}} \times \hat{\underset{\cdot}{\boldsymbol{n}}} \end{aligned} \tag{3.74}$$

respectively. The components for the symmetric rate-of-deformation tensor $\hat{\underset{\cdot\cdot}{\boldsymbol{D}}} = \frac{1}{2}(\hat{\underset{\cdot}{\boldsymbol{v}}}_{\underset{\cdot}{\boldsymbol{x}}} + \hat{\underset{\cdot}{\boldsymbol{v}}}^T_{\underset{\cdot}{\boldsymbol{x}}})$ are given in the local coordinate system by

$$\begin{aligned} D_{\hat{x}\hat{x}} &= \dot{r}_{\hat{x},\hat{x}} + \zeta\frac{h}{2}\left(\dot{n}_{\hat{x},\hat{x}} + (\Psi_{\hat{y}}n_{\hat{z}})_{,\hat{x}} - (\Psi_{\hat{z}}n_{\hat{y}})_{,\hat{x}}\right) \\ D_{\hat{x}\hat{y}} &= \frac{1}{2}\left(\dot{r}_{\hat{x},\hat{y}} + \dot{r}_{\hat{y},\hat{x}}\right) + \zeta\frac{h}{4}\left(\dot{n}_{\hat{x},\hat{y}} + (\Psi_{\hat{y}}n_{\hat{z}})_{,\hat{y}} - (\Psi_{\hat{z}}n_{\hat{y}})_{,\hat{y}} + \dot{n}_{\hat{y},\hat{x}} + (\Psi_{\hat{z}}n_{\hat{x}})_{,\hat{x}} - (\Psi_{\hat{x}}n_{\hat{z}})_{,\hat{x}}\right) \\ D_{\hat{x}\hat{z}} &= \frac{1}{2}\left(\underline{\dot{r}_{\hat{z},\hat{x}}} + \underline{\dot{n}_{\hat{x}}} + \Psi_{\hat{y}}n_{\hat{z}} - \Psi_{\hat{z}}n_{\hat{y}}\right) + \zeta\frac{h}{4}\left(\dot{n}_{\hat{z},\hat{x}} + (\Psi_{\hat{x}}n_{\hat{y}})_{,\hat{x}} - (\Psi_{\hat{y}}n_{\hat{x}})_{,\hat{x}}\right) \\ D_{\hat{y}\hat{y}} &= \dot{r}_{\hat{y},\hat{y}} + \zeta\frac{h}{2}\left(\dot{n}_{\hat{y},\hat{y}} + (\Psi_{\hat{z}}n_{\hat{x}})_{,\hat{y}} - (\Psi_{\hat{x}}n_{\hat{z}})_{,\hat{y}}\right) \\ D_{\hat{y}\hat{z}} &= \frac{1}{2}\left(\underline{\dot{r}_{\hat{z},\hat{y}}} + \underline{\dot{n}_{\hat{y}}} + \Psi_{\hat{z}}n_{\hat{x}} - \Psi_{\hat{x}}n_{\hat{z}}\right) + \zeta\frac{h}{4}\left(\dot{n}_{\hat{z},\hat{y}} + (\Psi_{\hat{x}}n_{\hat{y}})_{,\hat{y}} - (\Psi_{\hat{y}}n_{\hat{x}})_{,\hat{y}}\right) \end{aligned} \tag{3.75}$$

and $D_{\hat{z}\hat{z}}$ is determined from the plane stress condition. The discretized form of (3.75) is given below:

$$
\begin{aligned}
D_{\hat{x}\hat{x}} &= \sum_A N^0_{A,\hat{x}}\dot{\boldsymbol{r}}_{\hat{x}A} + \zeta\frac{h}{2}\sum_A\Big(N^1_{A,\hat{x}}\dot{\boldsymbol{r}}_{\hat{x}A} + N^0_{A,\hat{x}}(\dot{n}_{\hat{x}A} + \Psi_{\hat{y}A}n_{\hat{z}A} - \Psi_{\hat{z}A}n_{\hat{y}A})\Big)\\
D_{\hat{x}\hat{y}} &= \frac{1}{2}\sum_A\Big(N^0_{A,\hat{y}}\dot{\boldsymbol{r}}_{\hat{x}A} + N^0_{A,\hat{x}}\dot{\boldsymbol{r}}_{\hat{y}A}\Big) + \zeta\frac{h}{4}\sum_A\Big(N^1_{A,\hat{y}}\dot{\boldsymbol{r}}_{\hat{x}A} + N^1_{A,\hat{x}}\dot{\boldsymbol{r}}_{\hat{y}A} +\\
&\quad N^0_{A,\hat{y}}(\dot{n}_{\hat{x}A} + \Psi_{\hat{y}A}n_{\hat{z}A} - \Psi_{\hat{z}A}n_{\hat{y}A}) + N^0_{A,\hat{x}}(\dot{n}_{\hat{y}A} + \Psi_{\hat{z}A}n_{\hat{x}A} - \Psi_{\hat{x}A}n_{\hat{z}A})\Big)\\
D_{\hat{x}\hat{z}} &= \frac{1}{2}\sum_A\Big(\underline{N^0_{A,\hat{x}}\dot{\boldsymbol{r}}_{\hat{z}A}} + \underline{N_A\dot{n}_{\hat{x}A}} + N_A(\Psi_{\hat{y}A}n_{\hat{z}A} - \Psi_{\hat{z}A}n_{\hat{y}A})\Big) +\\
&\quad \zeta\frac{h}{4}\sum_A\Big(N^1_{A,\hat{x}}\dot{\boldsymbol{r}}_{\hat{z}A} + N^0_{A,\hat{x}}(\dot{n}_{\hat{z}A} + \Psi_{\hat{x}A}n_{\hat{y}A} - \Psi_{\hat{y}A}n_{\hat{x}A})\Big)\\
D_{\hat{y}\hat{y}} &= \sum_A N^0_{A,\hat{y}}\dot{\boldsymbol{r}}_{\hat{y}A} + \zeta\frac{h}{2}\sum_A\Big(N^1_{A,\hat{y}}\dot{\boldsymbol{r}}_{\hat{y}A} + N^0_{A,\hat{y}}(\dot{n}_{\hat{y}A} + \Psi_{\hat{z}A}n_{\hat{x}A} - \Psi_{\hat{x}A}n_{\hat{z}A})\Big)\\
D_{\hat{y}\hat{z}} &= \frac{1}{2}\sum_A\Big(\underline{N^0_{A,\hat{y}}\dot{\boldsymbol{r}}_{\hat{z}A}} + \underline{N_A\dot{n}_{\hat{y}A}} + N_A(\Psi_{\hat{z}A}n_{\hat{x}A} - \Psi_{\hat{x}A}n_{\hat{z}A})\Big) +\\
&\quad \zeta\frac{h}{4}\sum_A\Big(N^1_{A,\hat{y}}\dot{\boldsymbol{r}}_{\hat{z}A} + N^0_{A,\hat{y}}(\dot{n}_{\hat{z}A} + \Psi_{\hat{x}A}n_{\hat{y}A} - \Psi_{\hat{y}A}n_{\hat{x}A})\Big)
\end{aligned}
\tag{3.76}
$$

The expression for the virtual rate-of-deformation $\delta\hat{\dot{\boldsymbol{D}}} = \frac{1}{2}(\delta\hat{\dot{\boldsymbol{v}}}_{\hat{\boldsymbol{x}}} + \delta\hat{\dot{\boldsymbol{v}}}^T_{\hat{\boldsymbol{x}}})$ is calculated similarly. The underlined terms in (3.75) cancel out at the mid-surface in the KL3 and KL6 formulations, see Lemma 1. However, in the discretized form, these terms do not generally cancel out which causes shear locking. Mesh refinement diminishes the problem as the situation approaches the continuum limit. A simple strategy of dropping the underlined terms in (3.76) is investigated in Chapter 4 to tackle the shear locking problem. However, it should be noted that this approach is not applicable to RM6 formulation.

Lemma 1. *Under the Kirchhoff-Love assumption, the transverse shear must vanish at the mid-surface. This implies that* $\dot{\boldsymbol{r}}_{\hat{x},\hat{z}} + \dot{n}_{\hat{x}} = 0$ *and* $\dot{\boldsymbol{r}}_{\hat{y},\hat{z}} + \dot{n}_{\hat{y}} = 0$ *in* (3.75).

Proof. In the local coordinate system defined in Section 3.3.4, where $\hat{\boldsymbol{n}} = \begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$, the expression for the normal velocity (3.47) can be written as

$$
\dot{\hat{\boldsymbol{n}}} = \frac{1}{\|\hat{\boldsymbol{p}}\|}\begin{bmatrix}\dot{p}_{\hat{x}}\\ \dot{p}_{\hat{y}}\\ 0\end{bmatrix} = \frac{1}{\|\hat{\boldsymbol{p}}\|}\begin{bmatrix}\dot{\boldsymbol{r}}_{\hat{y},\xi}r_{\hat{z},\eta} - \dot{\boldsymbol{r}}_{\hat{z},\xi}r_{\hat{y},\eta} + r_{\hat{y},\xi}\dot{\boldsymbol{r}}_{\hat{z},\eta} - r_{\hat{z},\xi}\dot{\boldsymbol{r}}_{\hat{y},\eta}\\ \dot{\boldsymbol{r}}_{\hat{z},\xi}r_{\hat{x},\eta} - \dot{\boldsymbol{r}}_{\hat{x},\xi}r_{\hat{z},\eta} + r_{\hat{z},\xi}\dot{\boldsymbol{r}}_{\hat{x},\eta} - r_{\hat{x},\xi}\dot{\boldsymbol{r}}_{\hat{z},\eta}\\ 0\end{bmatrix}
\tag{3.77}
$$

where

$$
\|\hat{\dot{\boldsymbol{p}}}\| = \sqrt{(r_{\hat{y},\xi}r_{\hat{z},\eta} - r_{\hat{z},\xi}r_{\hat{y},\eta})^2 + (r_{\hat{z},\xi}r_{\hat{x},\eta} - r_{\hat{x},\xi}r_{\hat{z},\eta})^2 + (r_{\hat{x},\xi}r_{\hat{y},\eta} - r_{\hat{y},\xi}r_{\hat{x},\eta})^2}
\tag{3.78}
$$

Noting that $r_{\hat{z},\xi} = r_{\hat{z},\eta} = 0$, one may write the $\hat{x}$ and $\hat{y}$ components of the normal velocity as

$$\dot{n}_{\hat{x}} = \frac{-\dot{r}_{\hat{z},\xi} r_{\hat{y},\eta} + r_{\hat{y},\xi} \dot{r}_{\hat{z},\eta}}{\sqrt{(r_{\hat{x},\xi} r_{\hat{y},\eta} - r_{\hat{y},\xi} r_{\hat{x},\eta})^2}}, \quad \dot{n}_{\hat{y}} = \frac{\dot{r}_{\hat{z},\xi} r_{\hat{x},\eta} - r_{\hat{x},\xi} \dot{r}_{\hat{z},\eta}}{\sqrt{(r_{\hat{x},\xi} r_{\hat{y},\eta} - r_{\hat{y},\xi} r_{\hat{x},\eta})^2}} \tag{3.79}$$

On the other hand, the terms $\dot{r}_{\hat{z},\hat{x}}$ and $\dot{r}_{\hat{z},\hat{y}}$ can be obtained by

$$\begin{bmatrix} \dot{r}_{\hat{z},\hat{x}} \\ \dot{r}_{\hat{z},\hat{y}} \end{bmatrix} = \frac{1}{J_S} \begin{bmatrix} r_{\hat{y},\eta} & -r_{\hat{y},\xi} \\ -r_{\hat{x},\eta} & r_{\hat{x},\xi} \end{bmatrix} \begin{bmatrix} \dot{r}_{\hat{z},\xi} \\ \dot{r}_{\hat{z},\eta} \end{bmatrix}, \quad J_S = \begin{vmatrix} r_{\hat{x},\xi} & r_{\hat{y},\xi} \\ r_{\hat{x},\eta} & r_{\hat{y},\eta} \end{vmatrix} \tag{3.80}$$

which can be written as

$$\dot{r}_{\hat{z},\hat{x}} = \frac{\dot{r}_{\hat{z},\xi} r_{\hat{y},\eta} - r_{\hat{y},\xi} \dot{r}_{\hat{z},\eta}}{r_{\hat{x},\xi} r_{\hat{y},\eta} - r_{\hat{y},\xi} r_{\hat{x},\eta}}, \quad \dot{r}_{\hat{z},\hat{y}} = \frac{-\dot{r}_{\hat{z},\xi} r_{\hat{x},\eta} + r_{\hat{x},\xi} \dot{r}_{\hat{z},\eta}}{r_{\hat{x},\xi} r_{\hat{y},\eta} - r_{\hat{y},\xi} r_{\hat{x},\eta}} \tag{3.81}$$

Therefore, from (3.79) and (3.81):

$$\dot{n}_{\hat{x}} = -\dot{r}_{\hat{z},\hat{x}}, \quad \dot{n}_{\hat{y}} = -\dot{r}_{\hat{z},\hat{y}}, \quad \forall\, J_S \in \mathbb{R}^+ \tag{3.82}$$

□

In (3.76), the derivatives of the shape functions $N_{A,\chi}$ are approximated by the constant and linear terms in ζ, denoted by $N^0_{A,\chi}$ and $N^1_{A,\chi}$, respectively. A similar approach is suggested by Belytschko et al. (1989) and was adopted in the developments of the S7 element (Hokkanen and Pedroso, 2019a). As the shell formulation does not consider quadratic variation in ζ for the velocity strains, $N^1_{A,\chi}$ is not considered for $\dot{\hat{\boldsymbol{n}}}_{,\chi}$ or $(\hat{\boldsymbol{\Psi}} \times \hat{\boldsymbol{n}})_{,\chi}$. The relations for the derivatives of the shape functions with respect to the local coordinates are derived from

$$\begin{bmatrix} N_{A,\xi} \\ N_{A,\eta} \\ N_{A,\zeta} \end{bmatrix} = \boldsymbol{J} \begin{bmatrix} N_{A,\hat{x}} \\ N_{A,\hat{y}} \\ N_{A,\hat{z}} \end{bmatrix} = \begin{bmatrix} \hat{x}_{,\xi} & \hat{y}_{,\xi} & \hat{z}_{,\xi} \\ \hat{x}_{,\eta} & \hat{y}_{,\eta} & \hat{z}_{,\eta} \\ \hat{x}_{,\zeta} & \hat{y}_{,\zeta} & \hat{z}_{,\zeta} \end{bmatrix} \begin{bmatrix} N_{A,\hat{x}} \\ N_{A,\hat{y}} \\ N_{A,\hat{z}} \end{bmatrix} \tag{3.83}$$

where $\boldsymbol{J}$ is the Jacobian matrix of the transformation from the Euclidean to the parametric space. The expression for the shape function derivatives $N_{A,\hat{x}}$ and $N_{A,\hat{y}}$ can be written as

$$\begin{bmatrix} N_{A,\hat{x}} \\ N_{A,\hat{y}} \end{bmatrix} = \frac{1}{J} \begin{bmatrix} \hat{y}_{,\eta}\hat{z}_{,\zeta} - \hat{y}_{,\zeta}\hat{z}_{,\eta} & -\hat{y}_{,\xi}\hat{z}_{,\zeta} + \hat{y}_{,\zeta}\hat{z}_{,\xi} \\ -\hat{x}_{,\eta}\hat{z}_{,\zeta} + \hat{x}_{,\zeta}\hat{z}_{,\eta} & \hat{x}_{,\xi}\hat{z}_{,\zeta} - \hat{x}_{,\zeta}\hat{z}_{,\xi} \end{bmatrix} \begin{bmatrix} N_{A,\xi} \\ N_{A,\eta} \end{bmatrix} \tag{3.84}$$

where

$$J = \hat{x}_{,\zeta} \begin{vmatrix} \hat{y}_{,\xi} & \hat{z}_{,\xi} \\ \hat{y}_{,\eta} & \hat{z}_{,\eta} \end{vmatrix} - \hat{y}_{,\zeta} \begin{vmatrix} \hat{x}_{,\xi} & \hat{z}_{,\xi} \\ \hat{x}_{,\eta} & \hat{z}_{,\eta} \end{vmatrix} + \hat{z}_{,\zeta} \begin{vmatrix} \hat{x}_{,\xi} & \hat{y}_{,\xi} \\ \hat{x}_{,\eta} & \hat{y}_{,\eta} \end{vmatrix} \tag{3.85}$$

is the Jacobian determinant.

As $\hat{z}$ is aligned along ζ, $\hat{x}_{,\zeta} = \hat{y}_{,\zeta} = 0$. Additionally, as the shape functions are not functions of ζ, they are not functions of $\hat{z}$ either, and therefore $N_{A,\hat{z}} = N_{A,\zeta} = 0$. Hence, (3.84) is simplified to

$$\begin{bmatrix} N_{A,\hat{x}} \\ N_{A,\hat{y}} \end{bmatrix} = \frac{1}{J} \begin{bmatrix} \hat{y}_{,\eta}\hat{z}_{,\zeta} & -\hat{y}_{,\xi}\hat{z}_{,\zeta} \\ -\hat{x}_{,\eta}\hat{z}_{,\zeta} & \hat{x}_{,\xi}\hat{z}_{,\zeta} \end{bmatrix} \begin{bmatrix} N_{A,\xi} \\ N_{A,\eta} \end{bmatrix} \tag{3.86}$$

where

$$J = \hat{z}_{,\zeta} \begin{vmatrix} \hat{x}_{,\xi} & \hat{y}_{,\xi} \\ \hat{x}_{,\eta} & \hat{y}_{,\eta} \end{vmatrix} \tag{3.87}$$

After carrying out a Taylor expansion about the mid-surface, only the constant and linear terms in ζ are preserved in (3.86). This can be written as

$$\begin{bmatrix} N_{A,\hat{x}} \\ N_{A,\hat{y}} \end{bmatrix} \approx \begin{bmatrix} N^0_{A,\hat{x}} \\ N^0_{A,\hat{y}} \end{bmatrix} + \frac{\zeta}{2} \begin{bmatrix} N^1_{A,\hat{x}} \\ N^1_{A,\hat{y}} \end{bmatrix} \tag{3.88}$$

The constant terms in ζ are given as

$$\begin{bmatrix} N^0_{A,\hat{x}} \\ N^0_{A,\hat{y}} \end{bmatrix} = \frac{1}{J_0} \begin{bmatrix} r_{\hat{y},\eta} & -r_{\hat{y},\xi} \\ -r_{\hat{x},\eta} & r_{\hat{x},\xi} \end{bmatrix} \begin{bmatrix} N_{A,\xi} \\ N_{A,\eta} \end{bmatrix} \tag{3.89}$$

and the linear terms by

$$\begin{bmatrix} N^1_{A,\hat{x}} \\ N^1_{A,\hat{y}} \end{bmatrix} = \frac{1}{J_0} \begin{bmatrix} n_{\hat{y},\eta} - \frac{J_1}{J_0} r_{\hat{y},\eta} & -n_{\hat{y},\xi} + \frac{J_1}{J_0} r_{\hat{y},\xi} \\ -n_{\hat{x},\eta} + \frac{J_1}{J_0} r_{\hat{x},\eta} & n_{\hat{x},\xi} - \frac{J_1}{J_0} r_{\hat{x},\xi} \end{bmatrix} \begin{bmatrix} N_{A,\xi} \\ N_{A,\eta} \end{bmatrix} \tag{3.90}$$

where

$$\begin{aligned} J_0 &= r_{\hat{x},\xi} r_{\hat{y},\eta} - r_{\hat{x},\eta} r_{\hat{y},\xi} \\ J_1 &= r_{\hat{x},\xi} n_{\hat{y},\eta} + r_{\hat{y},\eta} n_{\hat{x},\xi} - r_{\hat{x},\eta} n_{\hat{y},\xi} - r_{\hat{y},\xi} n_{\hat{x},\eta} \end{aligned} \tag{3.91}$$

As a consequence of the chosen lifting operator method to determine the control point normals and normal velocities, the numerical integration to obtain the control point forces must be performed at most one Bézier element at a time. The force contribution of the element e associated with $\delta\dot{\boldsymbol{r}}_A$ is given in the global coordinate system by

$${}^e\dot{\boldsymbol{f}}^{r_0}_{intA} = \int_{S^e} \dot{\boldsymbol{R}} \cdot \hat{\dot{\boldsymbol{r}}}^{r_0}_A \mathrm{d}\xi\mathrm{d}\eta \tag{3.92}$$

where S^e represents the mid-surface of the element. The resultant $\hat{\boldsymbol{r}}_A^{r_0} = [r_{\hat{x}A}^{r_0}\ r_{\hat{y}A}^{r_0}\ r_{\hat{z}A}^{r_0}]^T$ is given as follows:

$$\begin{aligned} r_{\hat{x}A}^{r_0} &= N_{A,\hat{x}}^0 r_{\hat{x}\hat{x}}^0 + N_{A,\hat{y}}^0 r_{\hat{x}\hat{y}}^0 + N_{A,\hat{x}}^1 r_{\hat{x}\hat{x}}^1 + N_{A,\hat{y}}^1 r_{\hat{x}\hat{y}}^1 \\ r_{\hat{y}A}^{r_0} &= N_{A,\hat{x}}^0 r_{\hat{x}\hat{y}}^0 + N_{A,\hat{y}}^0 r_{\hat{y}\hat{y}}^0 + N_{A,\hat{x}}^1 r_{\hat{x}\hat{y}}^1 + N_{A,\hat{y}}^1 r_{\hat{y}\hat{y}}^1 \\ r_{\hat{z}A}^{r_0} &= \underline{N_{A,\hat{x}}^0 r_{\hat{x}\hat{z}}^0} + \underline{N_{A,\hat{y}}^0 r_{\hat{y}\hat{z}}^0} + N_{A,\hat{x}}^1 r_{\hat{x}\hat{z}}^1 + N_{A,\hat{y}}^1 r_{\hat{y}\hat{z}}^1 \end{aligned} \tag{3.93}$$

The element force contribution associated with $\delta\dot{\boldsymbol{\eta}}_A$ is given by

$${}^e\dot{\boldsymbol{f}}_{intA}^n = \int_{S^e} \ddot{\boldsymbol{R}} \cdot \dot{\hat{\boldsymbol{r}}}_A^n \mathrm{d}\xi\mathrm{d}\eta \tag{3.94}$$

where $\hat{\boldsymbol{r}}_A^n = [r_{\hat{x}A}^n\ r_{\hat{y}A}^n\ r_{\hat{z}A}^n]^T$ and

$$\begin{aligned} r_{\hat{x}A}^n &= N_{A,\hat{x}}^0 r_{\hat{x}\hat{x}}^1 + N_{A,\hat{y}}^0 r_{\hat{x}\hat{y}}^1 + \underline{N_A r_{\hat{x}\hat{z}}^0} \\ r_{\hat{y}A}^n &= N_{A,\hat{x}}^0 r_{\hat{x}\hat{y}}^1 + N_{A,\hat{y}}^0 r_{\hat{y}\hat{y}}^1 + \underline{N_A r_{\hat{y}\hat{z}}^0} \\ r_{\hat{z}A}^n &= N_{A,\hat{x}}^0 r_{\hat{x}\hat{z}}^1 + N_{A,\hat{y}}^0 r_{\hat{y}\hat{z}}^1 \end{aligned} \tag{3.95}$$

Finally, the contribution to the transversal shear is given as

$${}^e\dot{\boldsymbol{f}}_{intA}^{\Psi} = \int_{S^e} \ddot{\boldsymbol{R}} \cdot \dot{\hat{\boldsymbol{r}}}_A^{\Psi} \mathrm{d}\xi\mathrm{d}\eta \tag{3.96}$$

where $\hat{\boldsymbol{r}}_A^{\Psi} = [r_{\hat{x}A}^{\Psi}\ r_{\hat{y}A}^{\Psi}\ r_{\hat{z}A}^{\Psi}]^T$ and

$$\begin{aligned} r_{\hat{x}A}^{\Psi} &= n_{\hat{y}A}(N_{A,\hat{x}}^0 r_{\hat{x}\hat{z}}^1 + N_{A,\hat{y}}^0 r_{\hat{y}\hat{z}}^1) - n_{\hat{z}A}(N_{A,\hat{x}}^0 r_{\hat{x}\hat{y}}^1 + N_{A,\hat{y}}^0 r_{\hat{y}\hat{y}}^1 + N_A r_{\hat{y}\hat{z}}^0) \\ r_{\hat{y}A}^{\Psi} &= n_{\hat{z}A}(N_{A,\hat{x}}^0 r_{\hat{x}\hat{x}}^1 + N_{A,\hat{y}}^0 r_{\hat{x}\hat{y}}^1 + N_A r_{\hat{x}\hat{z}}^0) - n_{\hat{x}A}(N_{A,\hat{x}}^0 r_{\hat{x}\hat{z}}^1 + N_{A,\hat{y}}^0 r_{\hat{y}\hat{z}}^1) \\ r_{\hat{z}A}^{\Psi} &= n_{\hat{x}A}(N_{A,\hat{x}}^0 r_{\hat{x}\hat{y}}^1 + N_{A,\hat{y}}^0 r_{\hat{y}\hat{y}}^1 + N_A r_{\hat{y}\hat{z}}^0) - n_{\hat{y}A}(N_{A,\hat{x}}^0 r_{\hat{x}\hat{x}}^1 + N_{A,\hat{y}}^0 r_{\hat{x}\hat{y}}^1 + N_A r_{\hat{x}\hat{z}}^0) \end{aligned} \tag{3.97}$$

The resultants $\hat{\ddot{\boldsymbol{r}}}^0$ and $\hat{\ddot{\boldsymbol{r}}}^1$ are obtained by integrating along the fiber as

$$\hat{\ddot{\boldsymbol{r}}}^0 = \int_{-1}^{1} \hat{\ddot{\boldsymbol{\sigma}}} J \mathrm{d}\zeta, \qquad \hat{\ddot{\boldsymbol{r}}}^1 = \int_{-1}^{1} \zeta \frac{h}{2} \hat{\ddot{\boldsymbol{\sigma}}} J \mathrm{d}\zeta \tag{3.98}$$

The constitutive relation between the Cauchy stress $\hat{\ddot{\boldsymbol{\sigma}}}$ and the material deformation is discussed in Section 3.4. Moreover, the underlined internal force contributions in (3.93) and (3.95) must be dropped if the underlined terms in (3.76) are dropped.

The discretized internal virtual power for the Bézier element e can now be expressed in the following form:

$$\delta^e P_{int} = \sum_A \delta\dot{\hat{\boldsymbol{r}}}_A \cdot {}^e\dot{\boldsymbol{f}}_{intA}^{r_0} + \sum_A \delta\dot{\boldsymbol{\eta}}_A \cdot {}^e\dot{\boldsymbol{f}}_{intA}^{n} + \sum_A \delta\dot{\boldsymbol{\Psi}}_A \cdot {}^e\dot{\boldsymbol{f}}_{intA}^{\Psi} \tag{3.99}$$

However, $\delta\dot{\boldsymbol{n}}_A$ is a function of $\delta\dot{\boldsymbol{r}}_B$ and is given by (3.50). After substituting (3.50) into (3.99), one may write the total discretized internal virtual power in terms of $\delta\dot{\boldsymbol{r}}_A$ and $\delta\dot{\boldsymbol{\Psi}}_A$ as

$$\delta P_{int} = \sum_A \delta\dot{\boldsymbol{r}}_A \cdot \underbrace{\sum_e ({}^e\dot{\boldsymbol{f}}^{r_0}_{intA} + {}^e\dot{\boldsymbol{f}}^{r_1}_{intA})}_{\dot{\boldsymbol{f}}^r_{intA}} + \sum_A \delta\dot{\boldsymbol{\Psi}}_A \cdot \underbrace{\sum_e {}^e\dot{\boldsymbol{f}}^{\Psi}_{intA}}_{\dot{\boldsymbol{f}}^{\Psi}_{intA}} \tag{3.100}$$

where

$${}^e\dot{\boldsymbol{f}}^{r1}_{intA} = \sum_B (c^{\xi}_{BA}\dot{\boldsymbol{t}}^{\eta}_B \times {}^e\dot{\boldsymbol{q}}^n_B + c^{\eta}_{BA}\, {}^e\dot{\boldsymbol{q}}^n_B \times \dot{\boldsymbol{t}}^{\xi}_B) \tag{3.101}$$

and

$${}^e\dot{\boldsymbol{q}}^n_A = \frac{1}{\|\dot{\boldsymbol{p}}_A\|}(\dot{\boldsymbol{I}} - \dot{\boldsymbol{n}}_A \otimes \dot{\boldsymbol{n}}_A) \cdot {}^e\dot{\boldsymbol{f}}^n_{intA} \tag{3.102}$$

3.3.6 External virtual power

The external body and traction force contributions to the virtual power are given by

$$\delta P^b_{ext} = \int_\Omega \dot{\boldsymbol{b}} \cdot \delta\dot{\boldsymbol{v}}\, \mathrm{d}\Omega, \qquad \delta P^t_{ext} = \int_\Gamma \dot{\boldsymbol{t}} \cdot \delta\dot{\boldsymbol{v}}\, \mathrm{d}\Gamma \tag{3.103}$$

After discretizing and substituting the virtual velocity

$$\delta\dot{\boldsymbol{v}} = \delta\dot{\boldsymbol{r}} + \zeta\frac{h}{2}\Big(\delta\dot{\boldsymbol{n}} + \delta\dot{\boldsymbol{\Psi}} \times \dot{\boldsymbol{n}}\Big) \tag{3.104}$$

into (3.103), one obtains

$$\delta^e P^b_{ext} = \sum_A \Big(\delta\dot{\boldsymbol{r}}_A \cdot \underbrace{\int_{\Omega^e} N_A\dot{\boldsymbol{b}}\, \mathrm{d}\Omega}_{{}^e\dot{\boldsymbol{f}}^{r_0}_{extA}} + \delta\dot{\boldsymbol{n}}_A \cdot \underbrace{\int_{\Omega^e} \zeta\frac{h}{2} N_A\dot{\boldsymbol{b}}\, \mathrm{d}\Omega}_{{}^e\dot{\boldsymbol{f}}^{n}_{extA}} + \delta\dot{\boldsymbol{\Psi}}_A \cdot \underbrace{\int_{\Omega^e} \zeta\frac{h}{2} N_A\dot{\boldsymbol{n}}_A \times \dot{\boldsymbol{b}}\, \mathrm{d}\Omega}_{{}^e\dot{\boldsymbol{f}}^{\Psi}_{extA}}\Big) \tag{3.105}$$

for the body force. The total external body forces associated with $\delta\dot{\boldsymbol{r}}_A$ and $\delta\dot{\boldsymbol{\Psi}}_A$ are obtained by substituting (3.50) into (3.105) which yields

$$\delta P^b_{ext} = \sum_A \delta\dot{\boldsymbol{r}}_A \cdot \underbrace{\sum_e ({}^e\dot{\boldsymbol{f}}^{r_0}_{extA} + {}^e\dot{\boldsymbol{f}}^{r_1}_{extA})}_{\dot{\boldsymbol{f}}^r_{extA}} + \sum_A \delta\dot{\boldsymbol{\Psi}}_A \cdot \underbrace{\sum_e {}^e\dot{\boldsymbol{f}}^{\Psi}_{extA}}_{\dot{\boldsymbol{f}}^{\Psi}_{extA}} \tag{3.106}$$

where ${}^e\dot{\boldsymbol{f}}^{r_1}_{extA}$ is determined analogously to (3.101). The external traction force contribution is obtained similarly, but the contribution is usually applied only to the top, $\zeta = 1$, or bottom surface, $\zeta = -1$.

3.3.7 Inertial virtual power

The inertial force contribution to the virtual power is given by

$$\delta P_{iner} = \int_\Omega \rho \dot{\boldsymbol{a}} \cdot \delta \dot{\boldsymbol{v}} \, \mathrm{d}\Omega \tag{3.107}$$

where the acceleration is

$$\dot{\boldsymbol{a}} = \ddot{\boldsymbol{r}} + \zeta \frac{h}{2} \Big(\ddot{\boldsymbol{n}} + \dot{\boldsymbol{\Psi}} \times \boldsymbol{n} + \boldsymbol{\Psi} \times \dot{\boldsymbol{n}} \Big) \tag{3.108}$$

Equation (3.107) takes the following discretized form:

$$\delta P_{iner} = \sum_{A,B} \int_\Omega \rho N_A \delta \dot{\boldsymbol{v}}_A \cdot N_B \dot{\boldsymbol{a}}_B \, \mathrm{d}\Omega \tag{3.109}$$

In the context of explicit analyses, a lumped mass matrix is preferred for the sake of computational efficiency. The mass lumping is accomplished by replacing the shape function associated with the acceleration by the Kronecker delta. This corresponds to the row sum method in the context of solid elements with three degrees of freedom per control point. Now, the expression can be written as

$$\delta P_{iner} \approx \sum_{A,B} \int_\Omega \rho N_A \delta \dot{\boldsymbol{v}}_A \cdot \delta_{AB} \dot{\boldsymbol{a}}_B \, \mathrm{d}\Omega \tag{3.110}$$

After discretizing and substituting (3.108) into (3.110), one obtains

$$\delta P_{iner} \approx \sum_{A} \int_\Omega \rho N_A \delta \dot{\boldsymbol{v}}_A \cdot \left(\ddot{\boldsymbol{r}}_A + \zeta \frac{h}{2} \Big(\ddot{\boldsymbol{n}}_A + \dot{\boldsymbol{\Psi}}_A \times \boldsymbol{n}_A + \boldsymbol{\Psi}_A \times \dot{\boldsymbol{n}}_A \Big) \right) \mathrm{d}\Omega \tag{3.111}$$

The terms involving derivatives of $\boldsymbol{n}_A$ are neglected similarly to the rotation-free shell element developed by Benson et al. (2011) to simplify the formulation and to avoid a coupling between $\ddot{\boldsymbol{r}}_A$ and $\dot{\boldsymbol{\Psi}}_A$. Although the situation is slightly different from the aforementioned shell element due to shear deformability, the angular shear acceleration $\dot{\boldsymbol{\Psi}}_A$ is expected to be usually small in comparison with the acceleration of the shell normal $\boldsymbol{n}_A$ and hence not have significant influence to $\ddot{\boldsymbol{r}}_A$. The physical mass associated with $\ddot{\boldsymbol{r}}_A$ is therefore given by

$$M_A^r = \int_\Omega \rho N_A d\Omega = \int_{\Omega_0} \rho_0 N_A \mathrm{d}\Omega_0 \tag{3.112}$$

where 0 in the subscript denotes the initial configuration. An isotropic rotational inertia is defined similarly to Benson et al. (2010) and is given by

$$M_A^\Psi = \frac{\alpha_\Psi M_A^r}{12} \int_\Omega h^2 \mathrm{d}\Omega \tag{3.113}$$

where α_Ψ is chosen such that the time step is not penalized by the rotational degrees of freedom. The total expression for the virtual power including internal and external forces as well as optional damping becomes

$$\begin{aligned}\sum_A \delta\dot{\boldsymbol{r}}_A \cdot M_A^r \ddot{\boldsymbol{r}}_A &= \sum_A \delta\dot{\boldsymbol{r}}_A \cdot \left(\boldsymbol{f}_{extA}^r - \boldsymbol{f}_{intA}^r - \boldsymbol{f}_{dampA}^r\right) \\ \sum_A \delta\boldsymbol{\Psi}_A \cdot M_A^\Psi \dot{\boldsymbol{\Psi}}_A &= \sum_A \delta\boldsymbol{\Psi}_A \cdot \left(\boldsymbol{f}_{extA}^\Psi - \boldsymbol{f}_{intA}^\Psi - \boldsymbol{f}_{dampA}^\Psi\right)\end{aligned} \tag{3.114}$$

3.3.8 Time integration

The time integration is based on the central difference method. The translational and angular accelerations $\ddot{\boldsymbol{r}}_A^t$ and $\dot{\boldsymbol{\Psi}}_A^t$ of the control point A are calculated by

$$\begin{aligned}\ddot{\boldsymbol{r}}_A^t &= \frac{\boldsymbol{f}_{extA}^{rt} - \boldsymbol{f}_{intA}^{rt} - \tilde{\gamma} M_A^r \dot{\boldsymbol{r}}_A^{t-\frac{1}{2}}}{M_A^r} \\ \dot{\boldsymbol{\Psi}}_A^t &= \frac{\boldsymbol{f}_{extA}^{\Psi t} - \boldsymbol{f}_{intA}^{\Psi t} - \tilde{\gamma} M_A^\Psi \boldsymbol{\Psi}_A^{t-\frac{1}{2}}}{M_A^\Psi}\end{aligned} \tag{3.115}$$

where the last term in the numerators represent the artificial damping forces and $\tilde{\gamma}$ is the damping coefficient. The translational and angular velocities $\dot{\boldsymbol{r}}_A^{t+\frac{1}{2}}$ and $\boldsymbol{\Psi}_A^{t+\frac{1}{2}}$ are calculated as

$$\begin{aligned}\dot{\boldsymbol{r}}_A^{t+\frac{1}{2}} &= \dot{\boldsymbol{r}}_A^{t-\frac{1}{2}} + \Delta t^t \ddot{\boldsymbol{r}}_A^t \\ \boldsymbol{\Psi}_A^{t+\frac{1}{2}} &= \boldsymbol{\Psi}_A^{t-\frac{1}{2}} + \Delta t^t \dot{\boldsymbol{\Psi}}_A^t\end{aligned} \tag{3.116}$$

and $\boldsymbol{r}_A$ for the next time step is finally given by

$$\boldsymbol{r}_A^{t+1} = \boldsymbol{r}_A^t + \Delta t^{t+\frac{1}{2}} \dot{\boldsymbol{r}}_A^{t+\frac{1}{2}} \tag{3.117}$$

The time increment Δt^t is defined as

$$\Delta t^t = \frac{\Delta t^{t-\frac{1}{2}} + \Delta t^{t+\frac{1}{2}}}{2} \tag{3.118}$$

where $\Delta t^{t\pm\frac{1}{2}}$ is obtained from the element time step estimate (Hokkanen and Pedroso, 2019a). Although not required if the normal assumption of the shell fiber is used, it is further noted that in the Reissner-Mindlin formulations where the angular velocity field is responsible for the total rotation of the shell fiber, the actual director vector can be updated using Rodrigues' (1840) formula as follows:

$$\frac{\boldsymbol{d}_A^{t+1}}{\|\boldsymbol{d}_A^{t+1}\|} = \boldsymbol{P}_A^{t+\frac{1}{2}} \cdot \frac{\boldsymbol{d}_A^t}{\|\boldsymbol{d}_A^t\|} \tag{3.119}$$

The rotation tensor $\underset{\cdot\cdot}{\boldsymbol{P}}_A^{t+\frac{1}{2}}$ is defined by

$$\underset{\cdot\cdot}{\boldsymbol{P}}_A^{t+\frac{1}{2}} = \underset{\cdot\cdot}{\boldsymbol{I}} + \frac{\sin\ \phi_A^{t+\frac{1}{2}}}{\|\boldsymbol{\omega}_A^{t+\frac{1}{2}}\|}\underset{\cdot\cdot}{\boldsymbol{\Lambda}}_A^{t+\frac{1}{2}} + \frac{1-\cos\ \phi_A^{t+\frac{1}{2}}}{\|\boldsymbol{\omega}_A^{t+\frac{1}{2}}\|^2}\underset{\cdot\cdot}{\boldsymbol{\Lambda}}_A^{t+\frac{1}{2}} \cdot \underset{\cdot\cdot}{\boldsymbol{\Lambda}}_A^{t+\frac{1}{2}} \tag{3.120}$$

where $\boldsymbol{\omega}_A^{t+\frac{1}{2}}$ is the total angular velocity, ϕ is the rotation angle defined as

$$\phi_A^{t+\frac{1}{2}} = \Delta t^{t+\frac{1}{2}} \|\boldsymbol{\omega}_A^{t+\frac{1}{2}}\| \tag{3.121}$$

and $\underset{\cdot\cdot}{\boldsymbol{\Lambda}}_A^{t+\frac{1}{2}}$ is given by

$$\boldsymbol{\Lambda}_A^{t+\frac{1}{2}} = \begin{bmatrix} 0 & -\omega_{zA}^{t+\frac{1}{2}} & \omega_{yA}^{t+\frac{1}{2}} \\ \omega_{zA}^{t+\frac{1}{2}} & 0 & -\omega_{xA}^{t+\frac{1}{2}} \\ -\omega_{yA}^{t+\frac{1}{2}} & \omega_{xA}^{t+\frac{1}{2}} & 0 \end{bmatrix} \tag{3.122}$$

3.4 Stress integration

To evaluate the internal force contribution to the weak form, a constitutive relation between material deformations and stresses is required. The shell formulation presented in Section 3.3 adopts the principle of virtual power in the updated Lagrangian framework, and therefore, a constitutive relation between the rate-of-deformation tensor and an objective stress rate is formed.

Furthermore, the material response is assumed to be hypoelastic-plastic, but the elastic response is not required to be hyperelastic. The grade zero hypoelastic materials often lack the existence of a hyperelastic potential, i.e., the energy is not conserved exactly for a purely elastic closed deformation cycle. However, the energy error is insignificant for materials such as metals which experience only small elastic strains (Belytschko et al., 2013).

3.4.1 Objective stress rate

The Cauchy stress is obtained as a time integral of the Cauchy stress material derivative, i.e.,

$$\underset{\cdot\cdot}{\boldsymbol{\sigma}} = \int \dot{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}\, \mathrm{d}t \tag{3.123}$$

However, if $\underset{\cdot\cdot}{\boldsymbol{\sigma}} \neq \underset{\cdot\cdot}{\boldsymbol{0}}$, then generally $\dot{\underset{\cdot\cdot}{\boldsymbol{\sigma}}} \neq \underset{\cdot\cdot}{\boldsymbol{0}}$ under a rigid body rotation even though $\underset{\cdot\cdot}{\boldsymbol{D}} = \underset{\cdot\cdot}{\boldsymbol{0}}$. Therefore, $\dot{\underset{\cdot\cdot}{\boldsymbol{\sigma}}} \neq \underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_\sigma : \underset{\cdot\cdot}{\boldsymbol{D}}$ for any material tensor $\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_\sigma$. For this reason, an objective stress rate must be employed, i.e., a stress rate which is invariant under rigid body rotations.

The exact power-conjugate to the Green-Lagrange strain tensor is the Truesdell rate of Kirchhoff stress, also called the convected rate of Kirchhoff stress. This noncorotational objective stress rate is obtained by taking the Lie derivative of the Kirchhoff stress, i.e., the push-forward of the time derivative of the pull-back of the Kirchhoff stress as (Belytschko et al., 2013)

$$\mathcal{L}_v \underset{\cdot\cdot}{\boldsymbol{\tau}} = \underset{\cdot\cdot}{\boldsymbol{F}} \cdot \frac{D}{Dt}(\underset{\cdot\cdot}{\boldsymbol{F}}^{-1} \cdot \underset{\cdot\cdot}{\boldsymbol{\tau}} \cdot \underset{\cdot\cdot}{\boldsymbol{F}}^{-T}) \cdot \underset{\cdot\cdot}{\boldsymbol{F}}^{T} = \dot{\underset{\cdot\cdot}{\boldsymbol{\tau}}} - \underset{\cdot}{\boldsymbol{v}}_{,\underset{\cdot}{\boldsymbol{x}}} \cdot \underset{\cdot\cdot}{\boldsymbol{\tau}} - \underset{\cdot\cdot}{\boldsymbol{\tau}} \cdot \underset{\cdot}{\boldsymbol{v}}^{T}_{,\underset{\cdot}{\boldsymbol{x}}} = \overset{\triangledown}{\underset{\cdot\cdot}{\boldsymbol{\tau}}}_{T} \tag{3.124}$$

where $\mathcal{L}_v$ denotes the Lie derivative. The constitutive relation between $\overset{\triangledown}{\underset{\cdot\cdot}{\boldsymbol{\tau}}}_{T}$ and the rate-of-deformation tensor $\underset{\cdot\cdot}{\boldsymbol{D}}$ is

$$\overset{\triangledown}{\underset{\cdot\cdot}{\boldsymbol{\tau}}}_{T} = \underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_{T} : \underset{\cdot\cdot}{\boldsymbol{D}} \tag{3.125}$$

where $\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_{T}$ is the material tensor corresponding to the Truesdell rate of Kirchhoff stress. For hyperelastic materials, $\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_{T}$ is known as the fourth elasticity tensor, i.e., the spatial form of the second elasticity tensor $\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_{SE}$:

$$\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_{T} = \underset{\cdot\cdot}{\boldsymbol{F}} \cdot \underset{\cdot\cdot}{\boldsymbol{F}} \cdot \underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_{SE} \cdot \underset{\cdot\cdot}{\boldsymbol{F}}^{T} \cdot \underset{\cdot\cdot}{\boldsymbol{F}}^{T} \tag{3.126}$$

However, for a hypoelastic-plastic material, Prager's (1961) yielding stationary criterion requires the objective stress rate to be corotational (Xiao et al., 2000) and hence to be expressed as

$$\overset{\triangledown}{\underset{\cdot\cdot}{\boldsymbol{\tau}}} = \dot{\underset{\cdot\cdot}{\boldsymbol{\tau}}} - \underset{\cdot\cdot}{\boldsymbol{\Omega}} \cdot \underset{\cdot\cdot}{\boldsymbol{\tau}} + \underset{\cdot\cdot}{\boldsymbol{\tau}} \cdot \underset{\cdot\cdot}{\boldsymbol{\Omega}} \tag{3.127}$$

where $\underset{\cdot\cdot}{\boldsymbol{\Omega}}$ is a skew-symmetric spin tensor.

The exact power-conjugate to the Hencky strain tensor $\underset{\cdot\cdot}{\boldsymbol{H}}$, i.e., the logarithmic strain or true strain, is the Logarithmic rate of Kirchhoff stress. This corotational objective stress rate is a relatively new discovery, first proposed by Lehmann et al. (1991), and later discovered by Reinhardt and Dubey (1995, 1996) and Xiao et al. (1997). The skew-symmetric spin tensor $\underset{\cdot\cdot}{\boldsymbol{\Omega}}_{log}$ for the Logarithmic rate is determined such that the exact relation between $\underset{\cdot\cdot}{\boldsymbol{D}}$ and $\dot{\underset{\cdot\cdot}{\boldsymbol{H}}}$ is

$$\dot{\underset{\cdot\cdot}{\boldsymbol{H}}} = \underset{\cdot\cdot}{\boldsymbol{D}} + \underset{\cdot\cdot}{\boldsymbol{\Omega}}_{log} \cdot \underset{\cdot\cdot}{\boldsymbol{H}} - \underset{\cdot\cdot}{\boldsymbol{H}} \cdot \underset{\cdot\cdot}{\boldsymbol{\Omega}}_{log} \tag{3.128}$$

which gives (Xiao et al., 1997)

$$\underset{\cdot\cdot}{\boldsymbol{\Omega}}_{log} = \underset{\cdot\cdot}{\boldsymbol{\omega}} + \sum_{\substack{i,j=1 \\ i \neq j}}^{m} \left(\frac{b_i + b_j}{b_i - b_j} + \frac{2}{\ln b_j - \ln b_i} \right) \underset{\cdot\cdot}{\boldsymbol{\beta}}_j \cdot \underset{\cdot\cdot}{\boldsymbol{D}} \cdot \underset{\cdot\cdot}{\boldsymbol{\beta}}_i \tag{3.129}$$

where b_i are the m distinct eigenvalues of the left Cauchy-Green deformation tensor $\underset{\cdot\cdot}{\boldsymbol{B}} = \underset{\cdot\cdot}{\boldsymbol{F}} \cdot \underset{\cdot\cdot}{\boldsymbol{F}}^T$ and $\underset{\cdot\cdot}{\boldsymbol{\omega}} = \frac{1}{2}(\underset{\cdot}{\boldsymbol{v}}_{,\underset{\cdot}{\boldsymbol{x}}} - \underset{\cdot}{\boldsymbol{v}}^T_{,\underset{\cdot}{\boldsymbol{x}}})$ is the skew-symmetric vorticity tensor. Furthermore, the eigenprojections $\underset{\cdot\cdot}{\boldsymbol{\beta}}_i$ are given by

$$\underset{\cdot\cdot}{\boldsymbol{\beta}}_i = \prod_{\substack{j=1 \\ j\neq i}}^{m} \frac{\underset{\cdot\cdot}{\boldsymbol{B}} - b_j \underset{\cdot\cdot}{\boldsymbol{I}}}{b_i - b_j} \quad \text{for} \quad m \geq 2; \quad \underset{\cdot\cdot}{\boldsymbol{\beta}}_i = \underset{\cdot\cdot}{\boldsymbol{I}} \quad \text{for} \quad m = 1 \tag{3.130}$$

The Logarithmic rate of Kirchhoff stress is now given by

$$\overset{\triangledown}{\underset{\cdot\cdot}{\boldsymbol{\tau}}}_{log} = \dot{\underset{\cdot\cdot}{\boldsymbol{\tau}}} - \underset{\cdot\cdot}{\boldsymbol{\Omega}}_{log} \cdot \underset{\cdot\cdot}{\boldsymbol{\tau}} + \underset{\cdot\cdot}{\boldsymbol{\tau}} \cdot \underset{\cdot\cdot}{\boldsymbol{\Omega}}_{log} \tag{3.131}$$

where the constitutive relation between $\overset{\triangledown}{\underset{\cdot\cdot}{\boldsymbol{\tau}}}_{log}$ and $\underset{\cdot\cdot}{\boldsymbol{D}}$ is

$$\overset{\triangledown}{\underset{\cdot\cdot}{\boldsymbol{\tau}}}_{log} = \underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_{log} : \underset{\cdot\cdot}{\boldsymbol{D}} \tag{3.132}$$

and $\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_{log}$ is the respective material tensor. Xiao et al. (1999) showed that for an isotropic grade zero hypoelastic material of the form

$$\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}_{log} := C_{ijkl} = \lambda \delta_{ij} \delta_{kl} + \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}) \tag{3.133}$$

the Hencky hyperelastic potential exist. Bruhns et al. (1999) proposed a self-consistency condition stating that an additive hypoelastic-plastic model should produce a dissipation-free hyperelastic response in the absence of plastic flow. However, Jiao and Fish (2017, 2018) showed that the self-consistency condition is violated when the material is subjected to elastic unloading following yielding. They proposed a modified Logarithmic rate (known as the Kinetic logarithmic rate) to correct this problem and showed the equivalence of multiplicative hyperelastic-plastic and additive hypoelastic-plastic material models. Further discussion on the Logarithmic rate is found from the works of Xiao et al. (1998), Xiao and Chen (2002), Xiao et al. (2006), Zhou and Tamma (2003), Freed (2014), and Naumenko and Altenbach (2018).

Other popular corotational objective stress rates associated with the Kirchhoff stress include the classical Jaumann and Green-Naghdi rates (Belytschko et al., 2013). The spin tensor for the Jaumann rate is simply the vorticity tensor

$$\underset{\cdot\cdot}{\boldsymbol{\Omega}}_J = \underset{\cdot\cdot}{\boldsymbol{\omega}} \tag{3.134}$$

For the Green-Naghdi rate, the spin tensor is independent of the material stretch (Zhou and Tamma, 2003):

$$\underset{\cdot\cdot}{\boldsymbol{\Omega}}_G = \dot{\underset{\cdot\cdot}{\boldsymbol{R}}} \cdot \underset{\cdot\cdot}{\boldsymbol{R}}^T = \underset{\cdot\cdot}{\boldsymbol{\omega}} + \sum_{\substack{i,j=1 \\ i\neq j}}^{m} \frac{\sqrt{b_i} - \sqrt{b_j}}{\sqrt{b_i} + \sqrt{b_j}} \underset{\cdot\cdot}{\boldsymbol{B}}_j \cdot \underset{\cdot\cdot}{\boldsymbol{D}} \cdot \underset{\cdot\cdot}{\boldsymbol{B}}_i \tag{3.135}$$

It is worth noting that all the presented objective rates can produce identical results, but their respective material tensors are different. Some relationships between the material tensors associated with different objective stress rates are given by Belytschko et al. (2013). However, if a grade zero hypoelastic material of the form (3.133) is employed with all rates, the results will differ and only the Logarithmic rate gives a hyperelastic response. Also, in the absence of shear deformations all the aforementioned corotational rates coincide, i.e., $\underset{\cdot\cdot}{\boldsymbol{\Omega}}_{log}=\underset{\cdot\cdot}{\boldsymbol{\Omega}}_{J}=\underset{\cdot\cdot}{\boldsymbol{\Omega}}_{G}$.

Evaluating an objective stress rate directly may introduce notable computational cost. Furthermore, if an anisotropic material model is used, and the stress update is performed in a noncorotational coordinate system, a transformation to the material tensor must be applied. A common practice is to use the rate of corotational Kirchhoff stress which is objective if the rotation tensor $\underset{\cdot\cdot}{\boldsymbol{Q}}$ defining the local coordinate system can be expressed as

$$\dot{\underset{\cdot\cdot}{\boldsymbol{Q}}} - \underset{\cdot\cdot}{\boldsymbol{\Omega}} \cdot \underset{\cdot\cdot}{\boldsymbol{Q}} = \underset{\cdot\cdot}{\boldsymbol{0}} \tag{3.136}$$

In case the local coordinate system follows the rotation from the polar decomposition of the deformation gradient, i.e., $\underset{\cdot\cdot}{\boldsymbol{Q}} = \underset{\cdot\cdot}{\boldsymbol{R}}$, this corresponds to the Green-Naghdi rate. If the corresponding rate is instead the Logarithmic rate, the Hencky strain can be obtained as a corotational time integral of the rate-of-deformation:

$$\underset{\cdot\cdot}{\hat{\boldsymbol{H}}} = \int \dot{\underset{\cdot\cdot}{\hat{\boldsymbol{H}}}}\, \mathrm{d}t = \int \underset{\cdot\cdot}{\hat{\boldsymbol{D}}}\, \mathrm{d}t \tag{3.137}$$

The rate of corotational Kirchhoff stress is given by

$$\dot{\underset{\cdot\cdot}{\hat{\boldsymbol{\tau}}}} = \underset{\cdot\cdot\cdot\cdot}{\hat{\boldsymbol{C}}} : \underset{\cdot\cdot}{\hat{\boldsymbol{D}}} \tag{3.138}$$

where $\underset{\cdot\cdot\cdot\cdot}{\hat{\boldsymbol{C}}}$ is the material tensor corresponding to the corotational rate which the local coordinate system follows. A relation for $\dot{\underset{\cdot\cdot}{\hat{\boldsymbol{\tau}}}}$ and $\dot{\underset{\cdot\cdot}{\hat{\boldsymbol{\sigma}}}}$ which preserves second-order accuracy in the internal power of the stress rate is given as follows (Bazant and Cedolin, 2010):

$$\dot{\underset{\cdot\cdot}{\hat{\boldsymbol{\sigma}}}} = \dot{\underset{\cdot\cdot}{\hat{\boldsymbol{\tau}}}} - \mathrm{tr}(\underset{\cdot\cdot}{\hat{\boldsymbol{D}}})\underset{\cdot\cdot}{\hat{\boldsymbol{\sigma}}} = \underset{\cdot\cdot\cdot\cdot}{\hat{\boldsymbol{C}}} : \underset{\cdot\cdot}{\hat{\boldsymbol{D}}} - \mathrm{tr}(\underset{\cdot\cdot}{\hat{\boldsymbol{D}}})\underset{\cdot\cdot}{\hat{\boldsymbol{\sigma}}} \tag{3.139}$$

Unfortunately, the local coordinate system definition (3.65) does not satisfy (3.136) exactly for any of the discussed rates, and the expression (3.139) follows the aforementioned corotational rates precisely only in the absence of shear deformations. However, the approximation is acceptable for many large deformation problems, particularly, if the shear strains are small (Belytschko et al., 2013).

3.4.2 Plane stress condition

Not all shell formulations define a transversal normal strain rate component $D_{\hat{z}\hat{z}}$. In these cases, $D_{\hat{z}\hat{z}}$ is usually determined from the plane stress requirement $\sigma_{\hat{z}\hat{z}} = 0$. If the material response is characterized by the material tensor of the form (3.133), the value for $D_{\hat{z}\hat{z}}$ that satisfies the plane stress condition for the corotational rate of Kirchhoff stress is given by

$$D_{\hat{z}\hat{z}} = -\frac{\lambda(D_{\hat{x}\hat{x}} + D_{\hat{y}\hat{y}})}{\lambda + 2\mu} \tag{3.140}$$

where the Lamé's parameters λ and μ can be expressed in terms of Young's modulus E and Poisson's ratio ν as follows:

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \qquad \mu = \frac{E}{2(1+\nu)} \tag{3.141}$$

However, if a nonlinear elasto-plastic material model is used, determining $D_{\hat{z}\hat{z}}$ becomes a nonlinear problem and is solved iteratively. Assuming that the elastic response follows (3.133), Equation (3.140) provides a good initial guess for the first iteration $n = 0$. If this value does not satisfy the plane stress condition, the value for the next iteration $n + 1$ is determined by (de Souza Neto et al., 2011)

$$D^{n+1}_{\hat{z}\hat{z}} = D^{n}_{\hat{z}\hat{z}} - \frac{\dot{\sigma}^{n}_{\hat{z}\hat{z}}}{E} \tag{3.142}$$

and is substituted into (3.139) to start a new iteration.

It is further noted that the underlying material model does not require any modifications whether the plane stress condition is enforced or not, and full three-dimensional material models can be used regardless of the element formulation and necessity for the plane stress enforcement. The stress update procedure for a nonlinear material may therefore often consists of two nested iterative loops: the outer loop for finding $D_{\hat{z}\hat{z}}$ and the inner loop for enforcing the yield condition.

3.4.3 Incremental constitutive update

The Cauchy stress tensor is updated incrementally in the local coordinate system. Although the time-continuum constitutive equations discussed in Section 3.4.1 ensure material objectivity, the time-discretized constitutive equations presented within this subsection are not incrementally objective. However, this is usually acceptable if the time steps are sufficiently small, which is typically the case with explicit methods (Hallquist, 2006; de Souza Neto et al., 2011).

The stress update procedure begins by considering (3.139) and calculating the so-called trial stress state

$$\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{tr} = \hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1} + \Delta\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{tr} \tag{3.143}$$

where the nonupdated Cauchy stress $\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1} = \hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{*}_{t-1}$ is obtained directly from the stress state $\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{*}_{t-1}$ calculated during the previous time step in the preceding local coordinate system. The elastic trial stress increment $\Delta\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{tr}$ associated with the underlying rate is given by

$$\Delta\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{tr} = \big(\hat{\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}}^{e} : \hat{\underset{\cdot\cdot}{\boldsymbol{D}}} - \mathrm{tr}(\hat{\underset{\cdot\cdot}{\boldsymbol{D}}})\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1}\big)\Delta t \tag{3.144}$$

where e in the superscript of $\hat{\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}}^{e}$ implies elastic material tensor.

However, if instead a corotational stress rate of the form (3.127) is applied directly in the local coordinate system, $\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1}$ is obtained by

$$\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1} = \widetilde{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1} + \big(\hat{\underset{\cdot\cdot}{\boldsymbol{\Omega}}} \cdot \widetilde{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1} - \widetilde{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1} \cdot \hat{\underset{\cdot\cdot}{\boldsymbol{\Omega}}}\big)\Delta t \tag{3.145}$$

where

$$\widetilde{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1} = (\underset{\cdot\cdot}{\boldsymbol{R}}^{T}_{t-1} \cdot \underset{\cdot\cdot}{\boldsymbol{R}}_{t})^{T} \cdot \hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{*}_{t-1} \cdot (\underset{\cdot\cdot}{\boldsymbol{R}}^{T}_{t-1} \cdot \underset{\cdot\cdot}{\boldsymbol{R}}_{t}) \tag{3.146}$$

It is worth noting that $\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1}$ and $\Delta\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{tr}$ may differ between different objective rates even if $\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{*}_{t-1}$ and $\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{tr}$ do not. Furthermore, $\hat{\underset{\cdot\cdot}{\boldsymbol{D}}}$, $\hat{\underset{\cdot\cdot}{\boldsymbol{\Omega}}}$, and Δt are evaluated at the mid-point between the current and the previous time step (which are denoted by t and $t-1$ in the subscript, respectively).

The rate-of-deformation $\hat{\underset{\cdot\cdot}{\boldsymbol{D}}}$ is further decomposed into elastic $\hat{\underset{\cdot\cdot}{\boldsymbol{D}}}^{e}$ and plastic $\hat{\underset{\cdot\cdot}{\boldsymbol{D}}}^{p}$ parts:

$$\hat{\underset{\cdot\cdot}{\boldsymbol{D}}} = \hat{\underset{\cdot\cdot}{\boldsymbol{D}}}^{e} + \hat{\underset{\cdot\cdot}{\boldsymbol{D}}}^{p} \tag{3.147}$$

The correct updated stress can be obtained by subtracting the plastic part from the elastic trial stress:

$$\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}} = \hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1} + \big(\hat{\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}}^{e} : (\hat{\underset{\cdot\cdot}{\boldsymbol{D}}} - \hat{\underset{\cdot\cdot}{\boldsymbol{D}}}^{p}) - \mathrm{tr}(\hat{\underset{\cdot\cdot}{\boldsymbol{D}}})\hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}_{t-1}\big)\Delta t = \hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}}^{tr} - \hat{\underset{\cdot\cdot\cdot\cdot}{\boldsymbol{C}}}^{e} : \Delta t\hat{\underset{\cdot\cdot}{\boldsymbol{D}}}^{p} \tag{3.148}$$

The plastic part of rate-of-deformation $\hat{\underset{\cdot\cdot}{\boldsymbol{D}}}^{p}$ is determined such that the yield condition is satisfied.

In the framework of isotropic hardening the yield condition takes the generic form

$$F(\underset{\cdot\cdot}{\boldsymbol{\sigma}}, \sigma_y) = \Phi(\underset{\cdot\cdot}{\boldsymbol{\sigma}}) - \sigma_y(\varepsilon^p) \leq 0 \tag{3.149}$$

where $\Phi(\underset{\sim}{\boldsymbol{\sigma}})$ is the yield function, $\sigma_y(\varepsilon^p)$ represents the yield stress, and ε^p is the effective plastic strain. Employing the associative plastic flow rule (de Souza Neto et al., 2011), one may represent the plastic rate-of-deformation increment in terms of the nonnegative effective plastic strain increment $\Delta\varepsilon^p$ and the flow direction $\underset{\sim}{\boldsymbol{n}}$ as follows:

$$\Delta t \underset{\sim}{\boldsymbol{D}}^p = \Delta\varepsilon^p \Phi_{,\underset{\sim}{\boldsymbol{\sigma}}} = \Delta\varepsilon^p \underset{\sim}{\boldsymbol{n}} \tag{3.150}$$

If the trial stress does not exceed the yield surface, the material response is elastic and the stress update is complete. Otherwise, a plastic corrector phase is required to bring the stress back onto the yield surface such that

$$F = \Phi(\underset{\sim}{\boldsymbol{\sigma}}) - \sigma_y(\varepsilon^p) = 0 \tag{3.151}$$

Two efficient return mapping methods are presented in the following subsections.

3.4.4 Implicit radial return algorithm (von Mises)

This subsection presents an efficient return mapping algorithm which is specific for the von Mises (1913) yield criterion. The algorithm is known as the implicit radial return method and was introduced by Krieg and Key (1976). The derivation considers isotropic hardening, but extension to mixed isotropic-kinematic hardening is straightforward and does not add extensive computational overhead. The von Mises yield function can be written as

$$\Phi(\underset{\sim}{\boldsymbol{\sigma}}) = \sqrt{\frac{3}{2}}\, \|\underset{\sim}{\boldsymbol{\sigma}}_{dev}\| \tag{3.152}$$

where the deviatoric stress $\underset{\sim}{\boldsymbol{\sigma}}_{dev}$ is obtained by subtracting the volumetric stress $\underset{\sim}{\boldsymbol{\sigma}}_{vol}$ from the total stress $\underset{\sim}{\boldsymbol{\sigma}}$ as follows:

$$\underset{\sim}{\boldsymbol{\sigma}}_{dev} = \underset{\sim}{\boldsymbol{\sigma}} - \underset{\sim}{\boldsymbol{\sigma}}_{vol}, \quad \underset{\sim}{\boldsymbol{\sigma}}_{vol} = \frac{\mathrm{tr}(\underset{\sim}{\boldsymbol{\sigma}})}{3} \underset{\sim}{\boldsymbol{I}} \tag{3.153}$$

Considering the associative plastic flow rule (3.150), one obtains the following expression for the plastic rate-of-deformation increment:

$$\Delta t \underset{\sim}{\boldsymbol{D}}^p = \Delta\varepsilon^p \Phi_{,\underset{\sim}{\boldsymbol{\sigma}}} = \Delta\varepsilon^p \underset{\sim}{\boldsymbol{n}} = \Delta\varepsilon^p \sqrt{\frac{3}{2}}\, \frac{\underset{\sim}{\boldsymbol{\sigma}}_{dev}}{\|\underset{\sim}{\boldsymbol{\sigma}}_{dev}\|} \tag{3.154}$$

From (3.148) and (3.154):

$$\underset{\sim}{\boldsymbol{\sigma}} = \underset{\sim}{\boldsymbol{\sigma}}^{tr} - \underset{\sim\sim}{\boldsymbol{C}}^e : \Delta\varepsilon^p \sqrt{\frac{3}{2}}\, \frac{\underset{\sim}{\boldsymbol{\sigma}}_{dev}}{\|\underset{\sim}{\boldsymbol{\sigma}}_{dev}\|} \tag{3.155}$$

On the other hand, as the flow tensor $\underset{\sim}{\boldsymbol{n}}$ is purely deviatoric, the volumetric part of the elastic trial stress is not affected by the plastic corrector phase. Therefore, the stress return mapping procedure can be decomposed into deviatoric and volumetric parts

$$\begin{aligned} \underset{\sim}{\boldsymbol{\sigma}}_{dev} &= \underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev} - 2\mu\Delta\varepsilon^p\sqrt{\frac{3}{2}}\,\frac{\underset{\sim}{\boldsymbol{\sigma}}_{dev}}{\|\underset{\sim}{\boldsymbol{\sigma}}_{dev}\|} \\ \underset{\sim}{\boldsymbol{\sigma}}_{vol} &= \underset{\sim}{\boldsymbol{\sigma}}^{tr}_{vol} \end{aligned} \tag{3.156}$$

where $\underset{\sim\sim}{\boldsymbol{C}}^e$ of the form (3.133) is assumed. Equation (3.156_1) implies colinearity of $\underset{\sim}{\boldsymbol{\sigma}}_{dev}$ and $\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev}$. This yields to the following relationship

$$\frac{\underset{\sim}{\boldsymbol{\sigma}}_{dev}}{\|\underset{\sim}{\boldsymbol{\sigma}}_{dev}\|} = \frac{\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev}}{\|\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev}\|} \tag{3.157}$$

Substituting (3.157) into (3.156_1), one obtains

$$\underset{\sim}{\boldsymbol{\sigma}}_{dev} = \left(1 - \sqrt{\frac{3}{2}}\,\frac{2\mu\Delta\varepsilon^p}{\|\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev}\|}\right)\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev} = \left(1 - \frac{3\mu\Delta\bar{\varepsilon}^p}{\Phi^{tr}}\right)\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev} \tag{3.158}$$

Considering (3.158), one may now write the von Mises yield function (3.152) as

$$\Phi = \sqrt{\frac{3}{2}}\,\|\left(1 - \frac{3\mu\Delta\varepsilon^p}{\Phi^{tr}}\right)\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev}\| \tag{3.159}$$

which can be further rearranged into

$$\Phi = \left(1 - \frac{3\mu\Delta\varepsilon^p}{\Phi^{tr}}\right)\underbrace{\sqrt{\frac{3}{2}}\,\|\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev}\|}_{\Phi^{tr}} \tag{3.160}$$

provided that $1 - 3\mu\Delta\varepsilon^p/\Phi^{tr} \geq 0$. Inserting (3.160) into the yield condition (3.151) produces a single scalar equation for the effective plastic strain increment:

$$F(\Delta\varepsilon^p) = \Phi^{tr} - 3\mu\Delta\varepsilon^p - \sigma_y(\varepsilon^p_{t-1} + \Delta\varepsilon^p) = 0 \tag{3.161}$$

This equation is generally nonlinear (due to σ_y being a nonlinear function of ε^p in most cases) and can be solved by the Newton-Raphson iterative method. The value of $\Delta\varepsilon_p$ for the next iteration $n+1$ is given by

$$\Delta\varepsilon^p_{n+1} = \Delta\varepsilon^p_n - \frac{F(\Delta\varepsilon^p_n)}{F'(\Delta\varepsilon^p_n)} \tag{3.162}$$

where

$$F'(\Delta\varepsilon^p_n) = -3\mu - H(\varepsilon^p_{t-1} + \Delta\varepsilon^p_n) \tag{3.163}$$

and H is the slope of the isotropic hardening curve at $(\varepsilon^p_{t-1} + \Delta\varepsilon^p_n)$. After the effective plastic strain increment $\Delta\varepsilon^p$ is solved, the stress and the effective plastic strain can be updated by

$$\begin{aligned} \underset{\sim}{\boldsymbol{\sigma}} &= \underset{\sim}{\boldsymbol{\sigma}}^{tr} - 2\mu\Delta\varepsilon^p\sqrt{\frac{3}{2}}\frac{\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev}}{\|\underset{\sim}{\boldsymbol{\sigma}}^{tr}_{dev}\|} \\ \varepsilon^p &= \varepsilon^p_{t-1} + \Delta\varepsilon^p \end{aligned} \tag{3.164}$$

3.4.5 General explicit return mapping algorithm

A general return mapping method proposed by Grilo et al. (2015) is discussed in this subsection. The algorithm belongs to forward Euler category and uses a subincrementation technique to improve the accuracy of the stress return mapping procedure. Similarly to Section 3.4.4, isotropic hardening is considered but extension to mixed isotropic-kinematic hardening is straightforward. The yield criterion (3.151) is satisfied incrementally assuming associative plastic flow for each subincrement n such that

$$\Delta F_n = \underset{\sim}{\boldsymbol{n}}_{n-1} : \Delta\Delta\underset{\sim}{\boldsymbol{\sigma}}_{n-1} - H_{n-1}\Delta\varepsilon^p_n = 0 \tag{3.165}$$

where $\underset{\sim}{\boldsymbol{n}} = \Phi_{,\underset{\sim}{\boldsymbol{\sigma}}}$ and $H = \sigma_{y,\varepsilon^p}$. Substituting (3.148) into (3.165), and considering (3.150), one ends up with the following equality:

$$\underset{\sim}{\boldsymbol{n}}_{n-1} : (\Delta\Delta\underset{\sim}{\boldsymbol{\sigma}}^{tr} - \underset{\sim\sim}{\boldsymbol{C}}^e : \Delta\varepsilon^p_n\underset{\sim}{\boldsymbol{n}}_{n-1}) - H_{n-1}\Delta\varepsilon^p_n = 0 \tag{3.166}$$

The trial stress subincrement $\Delta\Delta\underset{\sim}{\boldsymbol{\sigma}}^{tr}$ is given by

$$\Delta\Delta\underset{\sim}{\boldsymbol{\sigma}}^{tr} = \frac{r_\sigma}{m_\sigma}\Delta\underset{\sim}{\boldsymbol{\sigma}}^{tr} \tag{3.167}$$

where m_σ is the total number of subincrements and

$$r_\sigma = \frac{\Phi(\underset{\sim}{\boldsymbol{\sigma}}^{tr}) - \sigma_y(\varepsilon^p_{t-1})}{\Phi(\underset{\sim}{\boldsymbol{\sigma}}^{tr}) - \Phi(\underset{\sim}{\boldsymbol{\sigma}}_{t-1})} \tag{3.168}$$

Equation (3.166) can be further rearranged to obtain an expression for the effective plastic strain increment

$$\Delta\varepsilon^p_n = \frac{\underset{\sim}{\boldsymbol{n}}_{n-1} : \Delta\Delta\underset{\sim}{\boldsymbol{\sigma}}^{tr}}{\underset{\sim\sim}{\boldsymbol{C}}^e : \underset{\sim}{\boldsymbol{n}}_{n-1} : \underset{\sim}{\boldsymbol{n}}_{n-1} + H_{n-1}} \tag{3.169}$$

The effective plastic strain is updated to the current subincrement n by

$$\varepsilon^p_n = \varepsilon^p_{n-1} + \Delta\varepsilon^p_n \tag{3.170}$$

after which the stress can be updated by

$$\underset{\sim}{\boldsymbol{\sigma}}_n = \frac{\sigma_y(\varepsilon_n^p)}{\Phi(\bar{\underset{\sim}{\boldsymbol{\sigma}}}_n)} \bar{\underset{\sim}{\boldsymbol{\sigma}}}_n, \quad \bar{\underset{\sim}{\boldsymbol{\sigma}}}_n = \left(\underset{\sim}{\boldsymbol{\sigma}}_{n-1} + \Delta\Delta\underset{\sim}{\boldsymbol{\sigma}}^{tr} - \Delta\varepsilon_n^p \underset{\sim\sim}{\boldsymbol{C}}^e : \underset{\sim}{\boldsymbol{n}}_{n-1} \right) \tag{3.171}$$

where the term $\sigma_y(\varepsilon_n^p)/\Phi(\bar{\underset{\sim}{\boldsymbol{\sigma}}}_n)$ contributes to a proportioning procedure (Grilo et al., 2015). Finally, the elasto-plastic tangent modulus can be calculated by

$$\underset{\sim\sim}{\boldsymbol{C}} = \underset{\sim\sim}{\boldsymbol{C}}^e - \frac{\underset{\sim\sim}{\boldsymbol{C}}^e : \underset{\sim}{\boldsymbol{n}} \otimes \underset{\sim}{\boldsymbol{n}} : \underset{\sim\sim}{\boldsymbol{C}}^e}{\underset{\sim\sim}{\boldsymbol{C}}^e : \underset{\sim}{\boldsymbol{n}} : \underset{\sim}{\boldsymbol{n}} + H} \tag{3.172}$$

3.5 Loads and boundary conditions

In the context of classical finite elements, point loads are usually applied to nodal locations such that the nodal force is simply appended by the point load. Similarly, line loads are often applied along a line of nodes for which the line load is distributed. Furthermore, Dirichlet boundary conditions are also typically enforced by directly specifying the nodal values.

In IGA, the control points are generally not interpolatory and applying a load or a boundary condition directly to a noninterpolatory control point is rarely physically justifiable. Many authors circumvent the problem by introducing repeated knots such that the loads or boundary conditions can be directly applied to the control points that are now interpolatory. However, this approach lowers the patch continuity to C^0 which is undesirable as it may reduce the accuracy as well as the stable time step size and leads to a geometry definition that does not satisfy the C^1 continuity requirement of the Kirchhoff-Love shell formulations.

3.5.1 Point and line loads

Point and line loads are herein considered as external traction loads applied over an infinitesimal area similarly to the work of Hokkanen and Pedroso (2019a). The external traction force contribution to the weak form (3.103) now takes the following form for a point load applied to $\xi = \xi^{pl}$, $\eta = \eta^{pl}$, $\zeta = \zeta^{pl} = 0$:

$$\delta P_{ext}^t = \sum_i \sum_j \delta \dot{\underset{\sim}{\boldsymbol{r}}}_{ij} \cdot \underbrace{\lim_{\substack{\Delta\xi\to 0 \\ \Delta\eta\to 0}} \int_{\eta^{pl}-\Delta\eta}^{\eta^{pl}+\Delta\eta} \int_{\xi^{pl}-\Delta\xi}^{\xi^{pl}+\Delta\xi} N_i^\xi(\xi) N_j^\eta(\eta) \underset{\sim}{\boldsymbol{t}}^{pl} J_S \mathrm{d}\xi \mathrm{d}\eta}_{\underset{\sim}{\boldsymbol{f}}_{ij}^{pl}} \tag{3.173}$$

In (3.173), the multivariate shape function has been decoupled into its univariate spline basis functions (2.16), i and j are control point indices associated with the

parametric directions ξ and η, respectively, and J_S is the Jacobian determinant. As the integral is only evaluated over an infinitesimal area, the basis functions are considered to be constant within the integrated area and can thus be brought outside the integral expression. The control point forces due to the point load can be now written as

$$\dot{\boldsymbol{f}}_{ij}^{pl} = N_i^{\xi}(\xi^{pl}) N_j^{\eta}(\eta^{pl}) \underbrace{\lim_{\substack{\Delta\xi\to 0 \\ \Delta\eta\to 0}} \int_{\eta^{pl}-\Delta\eta}^{\eta^{pl}+\Delta\eta} \int_{\xi^{pl}-\Delta\xi}^{\xi^{pl}+\Delta\xi} \dot{\boldsymbol{t}}^{pl} J_S \mathrm{d}\xi \mathrm{d}\eta}_{\dot{\boldsymbol{f}}^{pl}(\xi^{pl},\eta^{pl},\zeta^{pl})} \tag{3.174}$$

where the integral expression is equal to the known point load $\dot{\boldsymbol{f}}^{pl}(\xi^{pl}, \eta^{pl}, \zeta^{pl})$.

Line loads are treated in a similar way. In the case of a line load along $\xi = \xi^{ll}$, $\zeta = \zeta^{ll} = 0$

$$\dot{\boldsymbol{f}}_{ij}^{ll} = N_i^{\xi}(\xi^{ll}) \int_{\eta} N_j^{\eta}(\eta) \underbrace{\lim_{\Delta\xi\to 0} \int_{\xi^{ll}-\Delta\xi}^{\xi^{ll}+\Delta\xi} \dot{\boldsymbol{t}}^{ll} J_S \mathrm{d}\xi}_{\bar{\dot{\boldsymbol{f}}}^{ll}(\xi^{ll},\eta,\ \zeta^{ll})} \mathrm{d}\eta \tag{3.175}$$

where $\bar{\dot{\boldsymbol{f}}}^{ll}(\xi^{ll}, \eta,\ \zeta^{ll})$ represents the line load in the parametric space and can be obtained from the line load $\dot{\boldsymbol{f}}^{ll}(\xi^{ll}, \eta,\ \zeta^{ll})$ given in the Euclidean space by

$$\bar{\dot{\boldsymbol{f}}}^{ll}(\xi^{ll}, \eta,\ \zeta^{ll}) = \sqrt{x_{,\eta}^2 + y_{,\eta}^2 + z_{,\eta}^2} \Big|_{\substack{\xi=\xi^{ll} \\ \zeta=\zeta^{ll}}} \dot{\boldsymbol{f}}^{ll}(\xi^{ll}, \eta,\ \zeta^{ll}) \tag{3.176}$$

3.5.2 Dirichlet boundary conditions

An efficient technique compatible with diagonal mass matrices was proposed by Hokkanen and Pedroso (2019a) to enforce Dirichlet boundary conditions at arbitrary locations within a patch. A boundary condition of a prescribed acceleration along $\xi = \xi^{bc}$, $\zeta = 0$ requires that

$$\sum_i \sum_j N_i^{\xi}(\xi^{bc}) N_j^{\eta}(\eta) \ddot{\dot{\boldsymbol{r}}}_{ij} = \ddot{\dot{\boldsymbol{r}}}^{prescribed}(\xi^{bc}, \eta) \tag{3.177}$$

where the multivariate shape function has been decoupled into its univariate spline basis functions similarly to (3.173). However, the boundary condition can be satisfied exactly only if the prescribed acceleration $\ddot{\dot{\boldsymbol{r}}}^{prescribed}(\xi^{bc}, \eta)$ is representable for all η by the underlying spline interpolation, i.e.,

$$\sum_j N_j^{\eta}(\eta) \ddot{\dot{\boldsymbol{r}}}_j^{prescribed} = \ddot{\dot{\boldsymbol{r}}}^{prescribed}(\xi^{bc}, \eta) \tag{3.178}$$

where $\ddot{\boldsymbol{r}}_j^{prescribed}$ are the control values for the prescribed acceleration $\ddot{\boldsymbol{r}}^{prescribed}(\xi^{bc}, \eta)$. The acceleration $\ddot{\boldsymbol{r}}_{ij}$ of (3.177) is decoupled into an uncorrected acceleration $\ddot{\boldsymbol{r}}_{ij}^{uc}$, i.e., the acceleration that is obtained without applying the boundary condition, and a change in the acceleration due to the boundary condition $\ddot{\boldsymbol{r}}_{ij}^{bc}$:

$$\sum_i \sum_j N_i^\xi(\xi^{bc}) N_j^\eta(\eta)(\ddot{\boldsymbol{r}}_{ij}^{uc} + \ddot{\boldsymbol{r}}_{ij}^{bc}) = \sum_k N_k^\eta(\eta) \ddot{\boldsymbol{r}}_k^{prescribed} \tag{3.179}$$

Now, it is sufficient to guarantee that

$$\sum_i N_i^\xi(\xi^{bc})(\ddot{\boldsymbol{r}}_{ij}^{uc} + \ddot{\boldsymbol{r}}_{ij}^{bc}) = \ddot{\boldsymbol{r}}_j^{prescribed} \tag{3.180}$$

for all j.

However, (3.180) is often underdetermined as there are generally multiple unknowns $\ddot{\boldsymbol{r}}_{ij}^{bc}$. Therefore, additional equations are required. Considering (3.175), the control point forces due to a line load that exactly satisfy the boundary condition are given by

$$\boldsymbol{f}_{ij}^{bc} = N_i^\xi(\xi^{bc}) \underbrace{\int_\eta N_j^\eta(\eta) \lim_{\Delta\xi \to 0} \int_{\xi^{bc}-\Delta\xi}^{\xi^{bc}+\Delta\xi} \boldsymbol{t}^{bc} J_S \mathrm{d}\xi \mathrm{d}\eta}_{\boldsymbol{\lambda}_j} \tag{3.181}$$

where $\boldsymbol{t}^{bc}$ is the unknown traction exerted by the boundary condition. From (3.181), the following relation is obtained:

$$\boldsymbol{f}_{ij}^{bc} = N_i^\xi(\xi^{bc}) \frac{\boldsymbol{f}_{kj}^{bc}}{N_k^\xi(\xi^{bc})} \tag{3.182}$$

Considering Newton's second law $\boldsymbol{f}_{ij}^{bc} = M_{ij}^{rr} \ddot{\boldsymbol{r}}_{ij}^{bc}$, i.e.,

$$M_{ij}^{rr} \ddot{\boldsymbol{r}}_{ij}^{bc} = N_i^\xi(\xi^{bc}) \frac{M_{kj}^{rr} \ddot{\boldsymbol{r}}_{kj}^{bc}}{N_k^\xi(\xi^{bc})} \tag{3.183}$$

and thus

$$\ddot{\boldsymbol{r}}_{ij}^{bc} = N_i^\xi(\xi^{bc}) \frac{M_{kj}^{rr} \ddot{\boldsymbol{r}}_{kj}^{bc}}{N_k^\xi(\xi^{bc}) M_{ij}^{rr}} \tag{3.184}$$

where M_{ij}^{rr} represents the mass of the control point ij which is obtained from a diagonal mass matrix. Substituting (3.184) into (3.180) gives the change in the acceleration due to the boundary condition:

$$\ddot{\boldsymbol{r}}_{kj}^{bc} = \frac{N_k^\xi(\xi^{bc}) \left(\ddot{\boldsymbol{r}}_j^{prescribed} - \sum_l N_l^\xi(\xi^{bc}) \ddot{\boldsymbol{r}}_{lj}^{uc} \right)}{\sum_i \frac{N_i^\xi(\xi^{bc})^2}{M_{ij}^{rr}} M_{kj}^{rr}} \tag{3.185}$$

The control point accelerations that exactly satisfy the boundary condition are now given by

$$\ddot{\boldsymbol{r}}_{ij} = \ddot{\boldsymbol{r}}^{uc}_{ij} + \ddot{\boldsymbol{r}}^{bc}_{ij} \tag{3.186}$$

The unknown control point forces $\boldsymbol{f}^{bc}_{ij}$ are only used to form the relation (3.184) between the accelerations $\ddot{\boldsymbol{r}}^{bc}_{ij}$ of the neighboring control points, and need not be explicitly calculated for the purposes of evaluating the boundary condition. Furthermore, the unknowns $\boldsymbol{\lambda}_j$ can be understood as Lagrange multipliers that exactly satisfy the boundary condition.

It is further noted that if the boundary condition is enforced along a parametric coordinate that is simultaneously a knot location, the summations in (3.185) are evaluated only over p control points, where p is the order of the shape function. Therefore, for a typical edge constraint with unclamped knot vectors, only two control points need be considered for each j assuming biquadratic interpolation. However, if two edge boundary conditions intersect at a patch corner, then a coupled problem of four control points must be solved at that location.

3.6 Contact

In the context of classical finite elements, contact constraints are often enforced between the independent surface (master) and the nodes of the dependent surface (slave). This approach—referred to as the node-to-surface (NTS) method—is relatively simple to implement and computationally inexpensive (De Lorenzis et al., 2014). Moreover, the NTS method is often considered a collocation approach; the concentrated nodal contact forces can be determined directly without solving for the contact pressure. The method satisfies the Ladyzhenskaya-Babuška-Brezzi (LBB) stability condition (Oden and Kikuchi, 1982), but does not pass the contact patch test. Furthermore, the accuracy is strongly dependent on the space discretization.

In comparison to the NTS approach, better accuracy can be obtained by evaluating the contact integral more precisely along both contacting surfaces, not just at the nodal locations of the slave surface. These type of formulations can be traced back to the work of Simo et al. (1985) and are often referred to as surface-to-surface (STS) formulations (De Lorenzis et al., 2014).

In IGA, the control points are not interpolatory, i.e., they do not generally coincide with the geometry. Therefore it is not physically sensible to enforce the contact

constraints between the master surface and the control points of the slave surface. A method which is referred to as isogeometric NTS by De Lorenzis et al. (2014) was introduced by Matzen et al. (2013), where the contact constraints are evaluated at certain collocation points within the surface. Matzen et al. (2013) investigated Greville abscissae as well as Botella points, both of which guarantee that the number of contact constraints is equal to the number of control points.

An approach belonging in the class of STS formulations and known as Gauss-point-to-surface (GPTS) method (De Lorenzis et al., 2014) has attracted attention in the context of classical finite elements (Fischer and Wriggers, 2005, 2006) and IGA (Lu, 2011; Temizer et al., 2011; Dimitri et al., 2014). In this method, the contact constraint is enforced at an arbitrary number of contact integration points. However, if the contact constraint is to be satisfied locally at each integration point (herein referred to as the nonmortar method), the problem may become overconstrained, and thus violate the LBB stability condition.

Although the fulfillment of the LBB stability condition is an important requirement for mixed methods such as Lagrange multiplier method, it is not a compulsory prerequisite for the penalty method which does not satisfy the constraint exactly (De Lorenzis et al., 2014). However, unphysical oscillations may appear with large penalty parameter values due to violation of the LBB stability condition—Temizer et al. (2011) demonstrated this in the context of IGA. Later, Sauer and De Lorenzis (2015) showed that a nonmortar GPTS formulation remains stable if the mesh density is increased appropriately with the increasing penalty parameter.

On the other hand, the so-called mortar method does not satisfy the contact constraint locally at each integration point, but enforces the constraints in an average sense. In a way, the contact pressures are smoothed such that the LBB stability condition is satisfied. However, the mortar method requires significantly higher computational cost in comparison with the nonmortar method due to the smoothing procedure. Early applications of mortar method to contact mechanics are demonstrated by Belgacem et al. (1998), Belgacem (2000), Hild (2000), as well as McDevitt and Laursen (2000), among others.

In this work, a nonmortar GPTS approach combined with the penalty method is considered. The formulation satisfies the contact constraint locally at each integration point leading to a piecewise constant approximation of the contact pressure. In the wake of the mortar method, an efficient application-specific contact pressure

smoothing procedure is proposed in Section 3.6.2 to avoid unphysical oscillations and to improve the robustness of the formulation.

3.6.1 Contact contribution to virtual power

The frictionless normal contact contribution to the virtual power can be written as (Wriggers, 2006)

$$\delta P_c = \int_\Gamma \delta \dot{g}_N \, p \, \mathrm{d}\Gamma \tag{3.187}$$

where p represents the contact pressure. The contact interface must satisfy the classical Kuhn-Tucker conditions

$$g_N \geq 0, \quad p \geq 0, \quad p g_N = 0 \tag{3.188}$$

which guarantee impenetrability between the contacting surfaces such that the contact pressure is always nonnegative, and positive only if the gap is zero. The gap g_N and the virtual gap velocity $\delta \dot{g}_N$ are given by

$$\begin{aligned} g_N &= (\boldsymbol{x}^S - \boldsymbol{x}^M) \cdot \boldsymbol{n}_\Gamma^M \\ \delta \dot{g}_N &= (\delta \boldsymbol{v}^S - \delta \boldsymbol{v}^M) \cdot \boldsymbol{n}_\Gamma^M + (\boldsymbol{x}^S - \boldsymbol{x}^M) \cdot \delta \dot{\boldsymbol{n}}_\Gamma^M \end{aligned} \tag{3.189}$$

where the terms $\boldsymbol{x}^S$ and $\boldsymbol{x}^M$ represent the geometry, and the terms $\delta \boldsymbol{v}^S$ and $\delta \boldsymbol{v}^M$ the virtual velocity of the slave and the master surface, respectively. Furthermore, $\boldsymbol{n}_\Gamma^M$ is the unit normal of the master surface. The contact contribution to the virtual power now takes the form

$$\delta P_c = \int_\Gamma p(\delta \boldsymbol{v}^S - \delta \boldsymbol{v}^M) \cdot \boldsymbol{n}_\Gamma^M \, \mathrm{d}\Gamma + \underbrace{\int_\Gamma p(\boldsymbol{x}^S - \boldsymbol{x}^M) \cdot \delta \dot{\boldsymbol{n}}_\Gamma^M \, \mathrm{d}\Gamma}_{=0} \tag{3.190}$$

where the last terms vanishes due to (3.188).

The contact constraint is typically enforced by the Lagrange multiplier method or the penalty method. In the former method, the Lagrange multiplier represents the unknown contact pressure p that exactly satisfies (3.188). On the other hand, the penalty method enforces the contact constraint only in an approximate sense depending on the choice of the penalty parameter. As the penalty parameter is increased, the impenetrability condition is satisfied more closely. For the penalty method, the pressure p is herein considered to be a function of the penalty parameter and the gap g_N.

Interpolating the virtual velocities of the slave and the master surface by the shape functions N_A^S and N_B^M, respectively, one may express the discretized virtual power contribution by

$$\delta P_c = \int_\Gamma p\Big(\sum_A N_A^S \delta \dot{\boldsymbol{y}}_A^S - \sum_B N_B^M \delta \dot{\boldsymbol{y}}_B^M\Big) \cdot \dot{\boldsymbol{n}}_\Gamma^M \, \mathrm{d}\Gamma \tag{3.191}$$

where A and B represent the control points of the slave and the master surfaces, respectively. Rearranging (3.191), one obtains

$$\delta P_c = \sum_A \delta \dot{\boldsymbol{y}}_A^S \cdot \underbrace{\int_\Gamma p N_A^S \dot{\boldsymbol{n}}_\Gamma^M \, \mathrm{d}\Gamma}_{\dot{\boldsymbol{f}}_{cA}^S} + \sum_B \delta \dot{\boldsymbol{y}}_B^M \cdot \underbrace{\int_\Gamma -p N_B^M \dot{\boldsymbol{n}}_\Gamma^M \, \mathrm{d}\Gamma}_{\dot{\boldsymbol{f}}_{cB}^M} \tag{3.192}$$

from which the slave and the master contact forces, $\dot{\boldsymbol{f}}_{cA}^S$ and $\dot{\boldsymbol{f}}_{cB}^M$, are identified. The contact virtual power contribution for the shell formulation of Section 3.3 is obtained by substituting the shell virtual velocity definition (3.104) into (3.192). For the slave surface, this is given by

$$\delta^e P_c = \sum_A \Big(\delta \dot{\boldsymbol{r}}_A \cdot \underbrace{\int_{\Gamma^e} p N_A \dot{\boldsymbol{n}}_\Gamma^M \mathrm{d}\Gamma}_{{}^e\dot{\boldsymbol{f}}_{cA}^r} + \delta \dot{\boldsymbol{n}}_A \cdot \underbrace{\int_{\Gamma^e} \zeta \frac{h}{2} p N_A \dot{\boldsymbol{n}}_\Gamma^M \mathrm{d}\Gamma}_{{}^e\dot{\boldsymbol{f}}_{cA}^n} + \delta \dot{\boldsymbol{\Psi}}_A \cdot \underbrace{\int_{\Gamma^e} \zeta \frac{h}{2} p N_A \dot{\boldsymbol{n}}_A \times \dot{\boldsymbol{n}}_\Gamma^M \mathrm{d}\Gamma}_{{}^e\dot{\boldsymbol{f}}_{cA}^\Psi}\Big) \tag{3.193}$$

where $\zeta = 1$ for the top, and $\zeta = -1$ for the bottom surface of the shell. However, as the master surface normal $\dot{\boldsymbol{n}}_\Gamma^M$ is typically close to parallel to the shell mid-surface normal $\dot{\boldsymbol{n}}$, i.e., $\dot{\boldsymbol{n}}_\Gamma^M \times \dot{\boldsymbol{n}} \approx \dot{\boldsymbol{0}}$, the terms $\delta \dot{\boldsymbol{n}}_A \cdot {}^e\dot{\boldsymbol{f}}_{cA}^n$ and $\delta \dot{\boldsymbol{\Psi}}_A \cdot {}^e\dot{\boldsymbol{f}}_{cA}^\Psi$ may be neglected.

3.6.2 Stylus contact algorithm

In the ISF simulation, a stylus contact algorithm is required for enforcing the contact constraint between a sheet and a hemispherical headed stylus. The algorithm shall accurately capture the deformation caused by the tool tip regardless of the tool diameter. Therefore, a contact integration scheme where the density of the points is inversely proportional to the stylus diameter is employed. This uniform integration scheme consists of equal weights for the whole patch such that the distance between two adjacent integration points is one tenth of the forming tool diameter. However, if this would result in less number of integration points than given by the uniformly weighted Gauss-Lobatto scheme (see Figure 3.1b), then the latter is used.

In the adopted nonmortar GPTS approach, the pressure at each integration point is a function of a global penalty parameter and the local gap g_N. For this type

of formulation, an appropriate choice of the penalty parameter in relation to the mesh density is important as shown by Sauer and De Lorenzis (2015). However, as the contact algorithm is explicit, also the time step size must be considered in the determination of the penalty parameter. If the penalty parameter is too large relative to the time step size, the overclosure (i.e., the negative gap) which develops between two consecutive time steps may create excessive contact pressure which pushes the surfaces too far away from each other (Belytschko et al., 2013). Therefore, the appropriate choice of the penalty parameter in relation to mesh density and time step size is crucial for the accuracy of the solution. A special type of piecewise-linear pressure-overclosure relationship given by

$$p = \begin{cases} \varepsilon_s(s_N - g_N) & \text{if } (s_N - g_N) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.194}$$

is suggested for the stylus contact, where ε_s is the stylus contact global penalty parameter and s_N is referred to as the shift. The penalty parameter ε_s is adjusted in time such that

$$\min(g_N) \gtrsim 0 \tag{3.195}$$

An explicit update procedure for the penalty parameter ε_s is given by

$$\varepsilon_s^t = \begin{cases} (1 + \beta_s)\varepsilon_s^{t-1} & \text{if } \min(g_N^{t-1}) \leq 0 \\ (1 - \beta_s)\varepsilon_s^{t-1} & \text{otherwise} \end{cases} \tag{3.196}$$

where β_s is referred to as the sensitivity coefficient. Furthermore, a minimum value for ε_s is recommended. Good results were obtained by Hokkanen et al. (2018) with $\beta_s = 10^{-3}$ and $\varepsilon_s \geq 1$, although a time step size dependent expression for β_s may be more adaptive for a general case. However, the algorithm is not very sensitive to the choice of the these values. The contact pressure is adjusted appropriately for each mesh density (cf. the mortar method) by the shift s_N, given by

$$s_N = \sqrt{r_s^2 + \frac{l_0^2}{4}} - r_s \tag{3.197}$$

where l_0 is the maximum in-plane edge length of a Bézier element in the undeformed configuration assuming uniform mesh and r_s is the stylus head radius. Furthermore, a time step dependent minimum value for the shift is required to ensure that the penalty parameter does not become too large in relation to the time step size. The relation between the penalty stiffness and the time step size is discussed in more detail by Hallquist (2006).

The distance d_I between the integration point $\dot{\boldsymbol{p}}_I$ on a sheet surface and the closest point on a spherical surface (e.g., the stylus head) is simply given by

$$d_I = \|\dot{\boldsymbol{p}}_I - \dot{\boldsymbol{x}}_s\| - r_s \tag{3.198}$$

where $\dot{\boldsymbol{x}}_s$ is the center of the sphere. To avoid comparing the distance between the stylus and every single point $\dot{\boldsymbol{p}}_I$ on a sheet, a neighborhood search method is suggested. As a preprocessing step, a subgrid of points is associated with each I such that the subgrid contains all other integration points within a certain radius from $\dot{\boldsymbol{p}}_I$. A recommended value for this radius is the stylus diameter. During the analysis, only one subgrid is active at a time, i.e., the stylus contact algorithm neglects all points that do not belong to this set. The active subgrid is updated by keeping track of the point $\dot{\boldsymbol{p}}_I$ nearest to the stylus. The algorithm relies on the assumption that if the nearest point changes from the previous time step, some of the points surrounding the previously stored point is now closer to the stylus, where the surrounding points form the first-ring neighborhood of the previously stored point (see Figure 3.2). If this is the case, the first-ring neighborhood of this new point is further checked. This recursive procedure is continued as long as the closest point is found. The neighborhood method for the stylus contact is illustrated by Algorithm 1.

```
for the surrounding points (of the stored point) do
    if the surrounding point is closer (than the stored point) then
        replace the stored point by the surrounding point;
        restart for-loop;
    end
end
update the active subgrid;
for the points within the active subgrid do
    calculate the closest point projection;
    if contact is detected then
        evaluate contact constraint;
    end
end
```

Algorithm 1: The stylus contact neighborhood method.

As the time step size in an explicit analysis is very small, it is very likely that the nearest point for the next time step is always contained within the first-ring neighborhood of the previously stored point. This makes the method very efficient for explicit analyses as the algorithm hardly ever needs to check first-ring neighborhoods more than twice. However, it is further acknowledged that in the ISF manufacturing

process, the closest point projection from the stylus onto the sheet is not guaranteed to follow a continuous path, and therefore, the proposed algorithm may fail in certain rare circumstances. If this is not acceptable, one could use a simple approach and check every contact integration point for penetration.

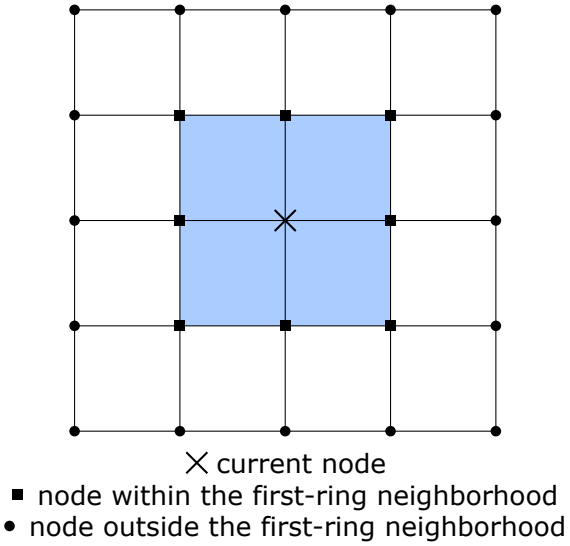


Figure 3.2: The first-ring neighborhood in the context of bilinear finite elements (top view). The elements within the shaded area are connected to the current node.

3.6.3 Die contact algorithm

In the case of a two-point incremental forming problem, a die contact algorithm is required for enforcing the contact constraints between a sheet and a rigid die. The integration scheme proposed in Section 3.6.2 is also suggested for the die contact, although other options are possible. The special pressure-overclosure relationship proposed for the stylus contact in Section 3.6.2 is not directly applicable to the die contact due to multiple possible contact regions. Therefore, a classical pressure-overclosure relationship given by

$$p = \begin{cases} -\varepsilon_d \, g_N & \text{if } g_N < 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.199}$$

is adopted, where ε_d is the user defined global die contact penalty parameter which is constant in time.

The die geometry is represented by a surface that is constructed of either Lagrange polynomials or splines. In the case of Lagrange polynomials, an analytic expression can be used for each element to calculate the closest point projection. To determine

the correct element, a neighborhood method similar to the stylus contact algorithm is employed. The algorithm keeps track of the node within the die horizontally nearest to the point $\dot{\boldsymbol{p}}_I$ on the sheet, where horizontal refers to a plane perpendicular to the longitudinal axis of the forming tool. This method requires that the boundary formed by an orthographic projection of the die geometry onto a horizontal projection plane is convex, and that each point within the die is characterized by a unique horizontal coordinates.

The procedure begins by determining and storing the horizontally nearest node on the die surface. This is accomplished by a global search as a preprocessing step; the computational efficiency is not important because this step is performed only once. During the analysis, a recursive neighborhood search similar to the procedure discussed in Section 3.6.2 is employed. However, instead of tracking the total distance, the horizontal distance is tracked. After the horizontally nearest node is updated, it suffices to check the elements connected to this node to determine if one them contains the closest point projection (see Figure 3.2). Similarly to Algorithm 1, the neighborhood method for the die contact, i.e., Algorithm 2, is very efficient for explicit analysis as it hardly ever needs to proceed further from the first-ring neighborhood of the previously converged location due to a small time step size.

```
for the surrounding nodes (of the stored node) do
    if the surrounding node is horizontally closer then
        replace the stored node by the surrounding node;
        restart for-loop;
    end
end
for the elements connected to the stored node do
    if the element contains the closest point projection then
        calculate the closest point projection;
        if contact is detected then
            evaluate contact constraint;
        end
    end
end
```

Algorithm 2: The die contact neighborhood method.

Another approach is based on the bucket sort method (Cormen et al., 2001). As a preprocessing step, a hypothetical uniform grid is laid on top of the die. Each rectangle within the grid represents a bucket for which the underlying elements are assigned. As the grid is uniform, knowing the horizontal position (i.e., a point on

a sheet) suffices to determine the right bucket. Increasing the number of buckets only affects the preprocessing step and does not increase the computational cost of determining the correct bucket. Therefore, the number of candidate elements can be reduced into only a few by using a small bucket size.

The die geometry may also be represented by a spline surface. The distance between a point on a sheet and the closest point on a spline surface can be calculated iteratively by the procedure given in Section 2.5.2. The algorithm requires relatively accurate initial guess for the Newton-Raphson iterative method. The initial guess can be based on the aforementioned methods; either assigning an arbitrary number of guess points (cf. nodes in the context of Lagrange elements) to the die geometry (e.g., one for each Bézier element) and tracking the horizontally closest guess point by the neighborhood method, or applying a bucket sort method and assigning a predefined initial guess for each bucket.

It is noted that in the context of two-point incremental forming, tracking the horizontal distance for the die contact instead of the closest point projection is more reliable as the closest point projection is not guaranteed to follow a continuous path. For a more general metal forming problem (e.g., a problem where not all points within the die have unique horizontal coordinates), the closest point projection approach may be more appropriate choice. However, one must keep in mind that—similarly to the stylus contact algorithm—this method may fail in some rare cases if the closest point projection does not follow a continuous path.

Chapter 4

Numerical testing

In this chapter, the shell formulations introduced in Section 3.3 are applied to linear elastic as well as geometrically nonlinear elastic and elasto-plastic problems. The thickness stretchable formulation proposed by Hokkanen and Pedroso (2019a) is also investigated. Sections 4.1 and 4.2 consider static linear elasticity, Sections 4.3–4.5 quasi-static geometrically nonlinear elasticity, and Sections 4.6–4.8 explicit dynamic elasto-plasticity. Sections 4.7 and 4.8 consider a single point and a two-point incremental forming problems, respectively, where the simulation results are compared to experimental data.

The straightforward idea of neglecting the underlined terms in (3.76), (3.93), and (3.95) to avoid shear locking is further investigated. The benchmarked formulations are referred to as KL3, KL3 full, KL6, KL6 full, RM6, and S7 following the definitions given in Section 3.3, and where the formulations for which the shear locking correction is not applied (i.e., the aforementioned underlined terms are not neglected) are referred to as full formulations.

Biquadratic b-splines with unclamped knot vectors are employed for the in-plane interpolation and a hypoelastic or hypoelastic-plastic material model is used. Each Bézier element is integrated with 2×2 reduced Gauss-Legendre quadrature rule in-plane and 9-point Simpson's rule through-thickness. In comparison with full integration, the in-plane 2×2 reduced Gauss-Legendre quadrature lowers the computational overhead and alleviates several locking phenomena while still preserving full rank of the stiffness matrix and providing good accuracy (Hokkanen and Pedroso, 2019b). Different types and orders of spline functions can be used as well, but the

patchwise quadratures discussed in Appendix B are not directly applicable with the KL3, KL6, and RM6 formulations, and the former two also require at least C^1 continuous geometry definition. The central difference scheme discussed in Section 3.3.8 is used for the time integration.

Different types of boundary conditions are treated as follows. A typical simply supported boundary condition is employed by constraining the degrees of freedom associated with the velocity of the mid-surface. A fixed boundary condition is enforced by constraining all degrees of freedom for the corresponding edges. However, for the KL3 and KL6 formulations, the shell normal velocity is constrained as well. If biquadratic interpolation is used, this is accomplished simply by constraining the degrees of freedom associated with the velocity of the mid-surface for two rows of control points, see the work of Kiendl et al. (2009). This approach enforces the physical fixed constraint exactly for the KL3 shell, but only approximately for the KL6 shell. The approximation results from the fact that both, the mid-surface normal velocity $\dot{\boldsymbol{\eta}}$ and the angular shear velocity $\dot{\boldsymbol{\Psi}}$ are constrained instead of the total angular velocity $\dot{\boldsymbol{\omega}}$, i.e., the constraint is more restrictive in comparison with the usual fixed boundary condition. However, the error is insignificant in many applications, particularly if the transversal shear strains remain small in the close proximity of the constrained location.

The implementation takes advantage of general-purpose computing on graphics processing units (GPGPU). The implemented GPU-based code was shown to be between one and two orders of magnitude faster for a representative incremental forming problem in comparison with traditional CPU-based implementations (e.g., Abaqus) on a modern desktop PC (Elford et al., 2018). Most calculations required by the benchmark problems of this chapter are distributed among 5120 CUDA cores located on a single Nvidia Titan V GPU.

4.1 Timoshenko beam (SS)

The first example considers the linear elastic response of a simply supported Timoshenko beam under a uniformly distributed load of magnitude q_0 (see Figure 4.1a). A beam of slenderness ratio $L/h = 100$ and Poisson's ratio $\nu = 0$ is considered, where L and h represent the length and the thickness of the beam, respectively. The analytical solution for the nondimensionalized deflection $\bar{w}_{exact}(x)$ of a simply

supported Timoshenko beam under a uniformly distributed load is given as follows:

$$\bar{w}_{exact}(x) = \frac{w(x)EI}{q_0 L^4} \tag{4.1}$$

The absolute deflection $w(x)$ is given by

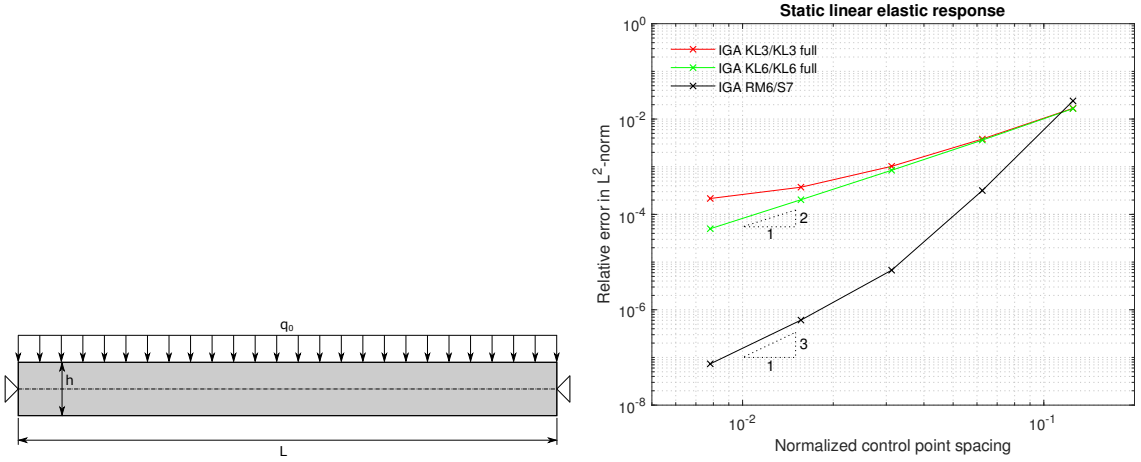
$$w(x) = \frac{q_0 L^4}{24EI}\left(\frac{x}{L} - 2\frac{x^3}{L^3} + \frac{x^4}{L^4}\right) + \frac{q_0 L^2}{2\kappa\mu A}\left(\frac{x}{L} - \frac{x^2}{L^2}\right) \tag{4.2}$$

where E is Young's modulus, I is the second moment of area, κ is the shear correction factor, μ is Lamé's second parameter, A is the cross-sectional area, and x is the coordinate along the length of the beam proceeding from 0 to L.

The relative error in the L^2-norm of the nondimensionalized deflection $\bar{w}_{iga}$ given by

$$\|e_{\bar{w}}\|_{L_2} = \sqrt{\frac{\int_\Omega (\bar{w}_{exact} - \bar{w}_{iga})^2 \, \mathrm{d}\Omega}{\int_\Omega \bar{w}_{exact}^2 \, \mathrm{d}\Omega}} \tag{4.3}$$

is plotted against the normalized control point spacing in Figure 4.1b for all element types. The shear corrected KL3 and KL6 formulations produce identical results with the respective full formulations which are prone to shear locking, and thus these identical result are represented by a single common line plot. The same observation applies to the RM6 and S7 formulations which produce practically indistinguishable results.



(a) The problem setup.

(b) The relative error in the L^2-norm of the nondimensionalized deflection.

Figure 4.1: Timoshenko beam under a uniformly distributed load.

The KL3 formulation can only reach a certain level of accuracy as it is unable to capture the transversal shear deformations. On the other hand, the shear deformable

KL6 formulation achieves the optimal 2 : 1 convergence rate for a formulation relying on a continuous normal definition (at least C^1 continuous geometry). The RM6 and S7 formulations achieve the optimal 3 : 1 convergence rate for these type of formulations, but show a clear shear locking tendency for coarser meshes.

4.2 Mindlin plate (SSSS)

The second example considers the linear elastic response of a simply supported Mindlin plate under a uniformly distributed load q_0 (see Figure 4.2a). The square plate of slenderness ratio $L/h = 100$ and Poisson's ratio $\nu = 0.3$ is considered, where L and h represent the edge length and the thickness of the plate, respectively. The analytical solution for the nondimensionalized deflection $\bar{w}_{exact}(x, y)$ of a simply supported Mindlin plate under a uniformly distributed load is given as follows:

$$\bar{w}_{exact}(x, y) = \frac{w(x, y)D}{q_0 a^4} \tag{4.4}$$

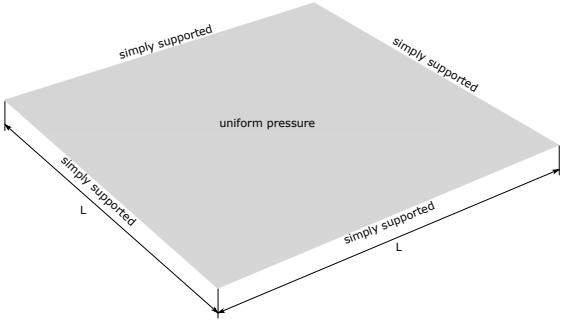
The absolute deflection $w(x, y)$ is given in terms of the Kirchhoff plate solution $w_k(x, y)$ and Marcus moment $M_k(x, y)$ by

$$\begin{aligned} w(x, y) &= w_k(x, y) + \frac{M_k(x, y)}{\kappa \mu h} \\ w_k(x, y) &= \frac{16 q_0}{\pi^6 D} \sum_{m=1,3,5\ldots}^{\infty} \sum_{n=1,3,5\ldots}^{\infty} \frac{\sin(\frac{m\pi x}{a}) \sin(\frac{n\pi y}{b})}{mn\left(\frac{m^2}{a^2} + \frac{n^2}{b^2}\right)^2}, \quad M_k(x, y) = -D\nabla^2 w_k(x, y) \end{aligned} \tag{4.5}$$

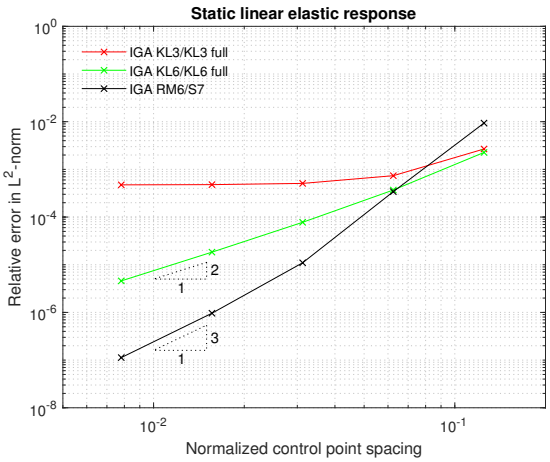
where a represents the lengths of the edges at $y = 0$ and $y = b$ and b represents the lengths of the edges at $x = 0$ and $x = a$. For a square plate $a = b$. Furthermore, the flexural rigidity is given as $D = \frac{Eh^3}{12(1-\nu^2)}$.

The relative error in the L^2-norm of the nondimensionalized deflection $\bar{w}_{iga}$ (4.3) is plotted against the normalized control point spacing in Figure 4.2b for all element types. Similar response compared to the Timoshenko beam problem is observed (cf. Figure 4.1b). The shear corrected KL3 and KL6 formulations still produce identical results with their non-shear-corrected counterparts. Furthermore, the RM6 and S7 formulations also still produce practically indistinguishable results.

The difference between the KL3 and the shear deformable KL6 formulations is now more pronounced compared to the Timoshenko beam problem. For the coarsest mesh, the KL3 and KL6 formulations provide better accuracy compared to the RM6



(a) The problem setup.



(b) The relative error in the L^2-norm of the nondimensionalized deflection.

Figure 4.2: Mindlin plate under a uniformly distributed load.

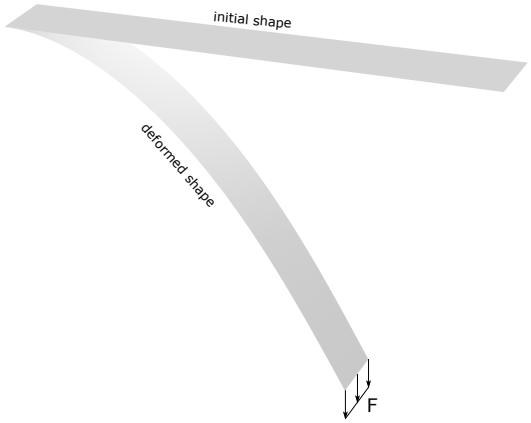
and S7 formulations regardless of lower convergence rates, as the latter formulations suffer from shear locking. Nevertheless, all shear deformable formulations converge to the correct solution and achieve their optimal convergence rates ($2:1$ or $3:1$).

4.3 Straight cantilever plate strip

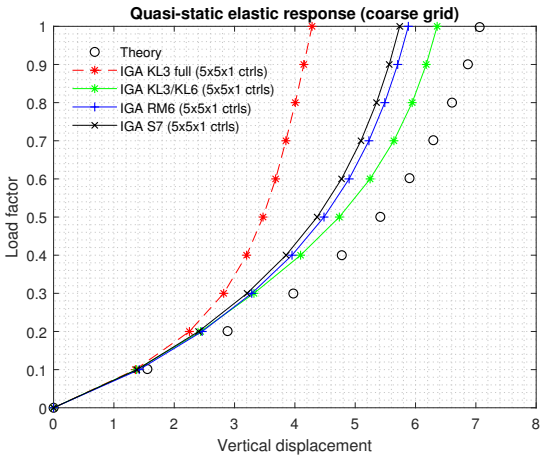
This example considers large deformations of the elastic straight cantilever plate strip shown in Figure 4.3a. The plate strip has the length $L = 10$, the width $w = 1.0$ and the thickness $h = 0.1$. Young's modulus and Poisson's ratio are $E = 10^7$ and $\nu = 0.3$, respectively. The applied line load is given as $F = 40\lambda$, where λ is the load factor ranging from 0.0 to 1.0.

The maximum vertical tip displacement is plotted against the load factor λ in Figures 4.3b, 4.3c, and 4.3d for three different discretizations—5, 9, and 17 control points along the length of the plate strip (referred to as the coarse, medium, and fine grid, respectively). The theoretical solution is given by Frisch-Fay (1962).

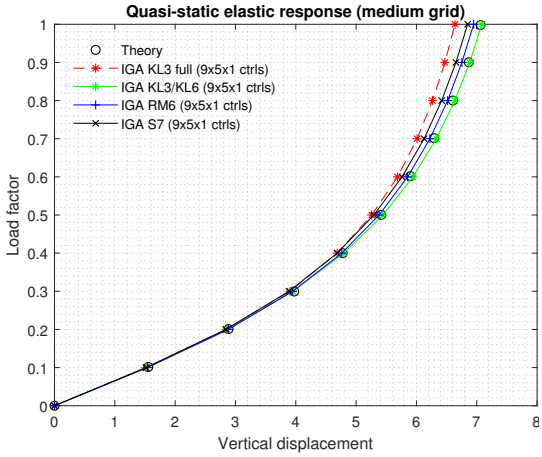
In contrast to the previous small deformation problems, a significant difference is observed between the shear corrected KL3 and full KL3 formulations. This is likely to be associated with curvature of the Bézier elements. The full KL3 formulation suffers from severe transversal shear locking and produces the worst results of all formulations. On the other hand, the shear corrected KL3 and KL6 formulations produce the best results, the difference between these two formulations



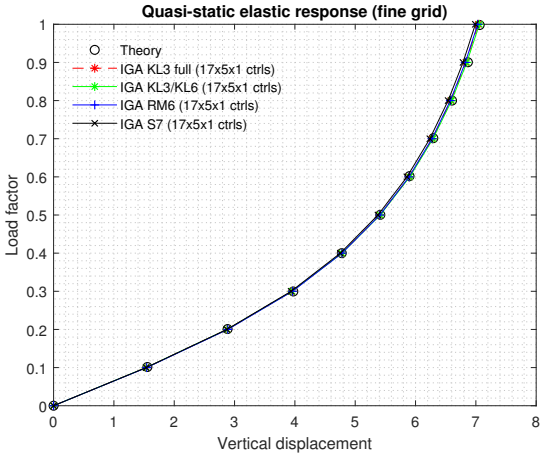
(a) The problem setup.



(b) Load factor versus the vertical tip displacement (coarse grid).



(c) Load factor versus the vertical tip displacement (medium grid).



(d) Load factor versus the vertical tip displacement (fine grid).

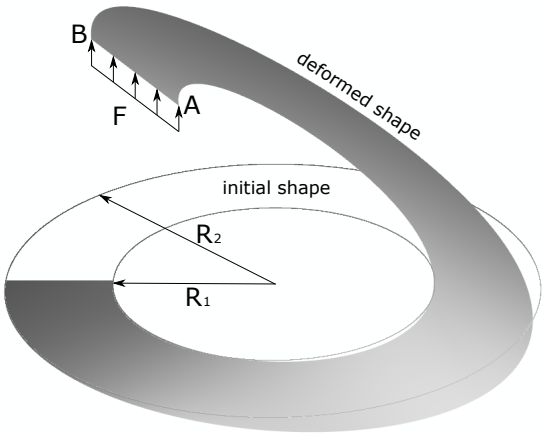
Figure 4.3: Cantilever plate strip under a tip line load.

being practically indistinguishable. The accuracy of the RM6 and S7 formulations is quite close to each other, although the RM6 formulation is slightly more accurate. All element types converge towards the theoretical solution when the mesh is refined.

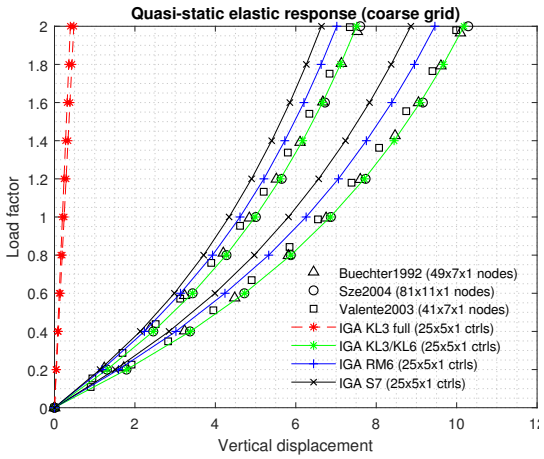
4.4 Cantilever ring plate

The thin cantilever ring plate shown in Figure 4.4a subjected to a vertical line load is considered. According to Cardoso and Yoon (2005), this example is one of the most sensitive tests for finite rotations and element warping. The geometric data is characterized by the inner radius $R_1 = 6.0$, the outer radius $R_2 = 10.0$, and the thickness $h = 0.03$. Young's modulus and Poisson's ratio are given by $E = 2.1 \times 10^{10}$

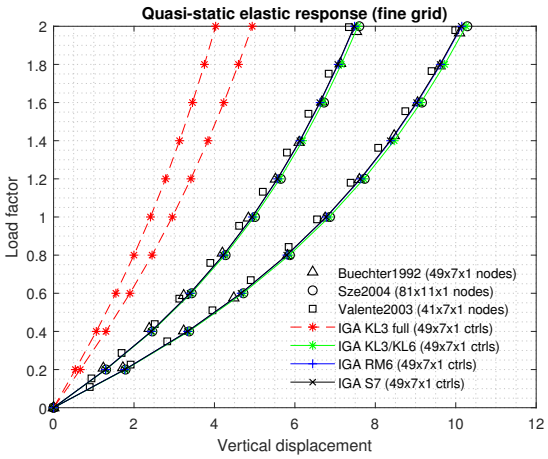
and $\nu = 0.0$, respectively. The applied line load is given as $F = 100\lambda$, where the load factor λ ranges from 0.0 to 2.0.



(a) The problem setup.



(b) The load factor versus the displacements of points A and B (coarse grid).



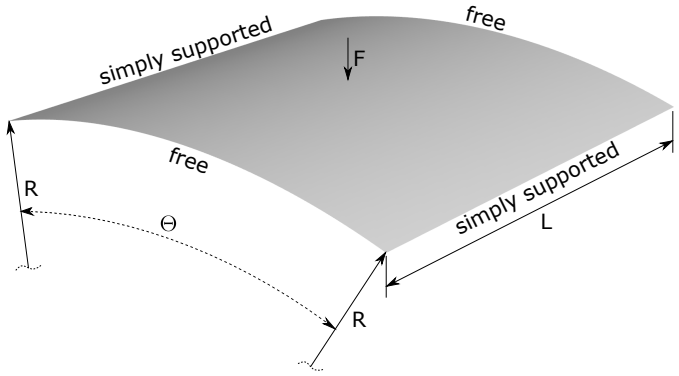
(c) The load factor versus the displacements of points A and B (fine grid).

Figure 4.4: Cantilever ring plate under a tip line load.

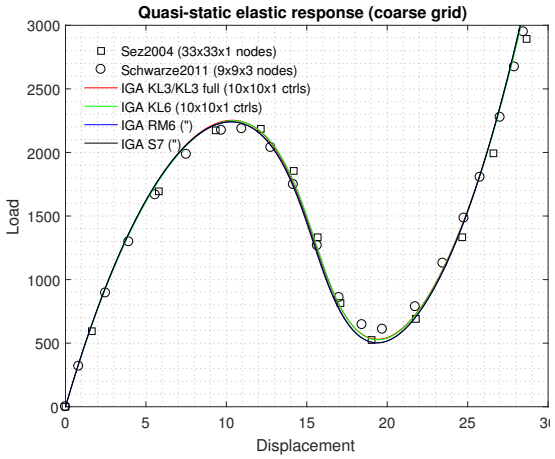
The vertical displacements of points A and B are compared with the reference results given by Buechter and Ramm (1992), Sze et al. (2004), and Valente et al. (2003) in Figures 4.4b and 4.4c for coarse and fine meshes using $25 \times 5 \times 1$ and $49 \times 7 \times 1$ control points, respectively. The full KL3 formulation suffers from extreme transversal shear locking and produces incorrect results for both mesh sizes. However, the shear corrected KL3 and KL6 formulations do not suffer from any noticeable locking and produce very good results even for the coarse mesh size. The RM6 and S7 formulations suffer from slight shear locking when the coarse mesh is used, the RM6 formulation being slightly ahead.

4.5 Snap-through of a thick roof

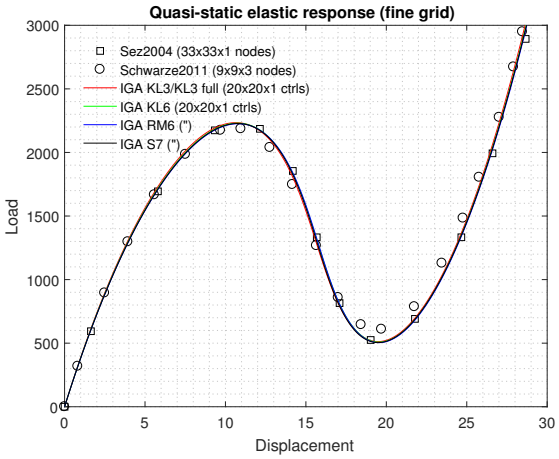
In this example, a snap-through behavior of a thick cylindrical roof structure is investigated. The geometric data is given by the edge length $L = 508$, the angle $\Theta = 0.2$, the radius $R = 2540$, and the thickness $h = 12.7$. Young's modulus and Poisson's ratio are set to $E = 3102.75$ and $\nu = 0.3$, respectively.



(a) The problem setup.



(b) The load factor versus the displacement of the center point (coarse grid).



(c) The load factor versus the displacement of the center point (fine grid).

Figure 4.5: A cylindrical roof structure under a point load.

The two opposite edges of the cylindrical shell are simply supported and the two other opposite edges are free. A concentrated vertical load F is applied in the mid-point of the roof as shown in Figure 4.5a. The analysis is carried out by prescribing the displacement in the explicit solver and moving slowly (i.e., keeping the inertial forces small). No external damping is applied. The full model is analyzed and the symmetry of the problem is not exploited.

The load-displacement curves for two mesh configurations are shown in Figures 4.5b

and 4.5c, where the given number of control points or nodes in the legend has been scaled for the full geometry mesh, even if only a quarter of the problem was originally analyzed.

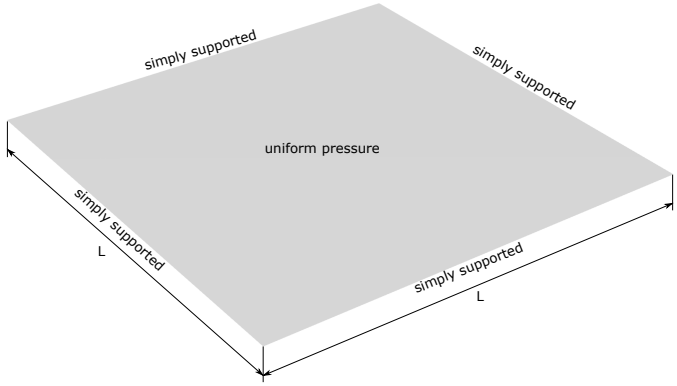
All element types are in a good agreement with the reference solutions given by Sze et al. (2004) as well as Schwarze and Reese (2011), but particularly with Sze et al. (2004) who used a very fine mesh. Furthermore, no difference is observed between the shear corrected and full KL3 formulations, i.e., shear locking does not seem to cause problems with the corresponding thick problem.

4.6 Dynamic elasto-plastic response of a square plate

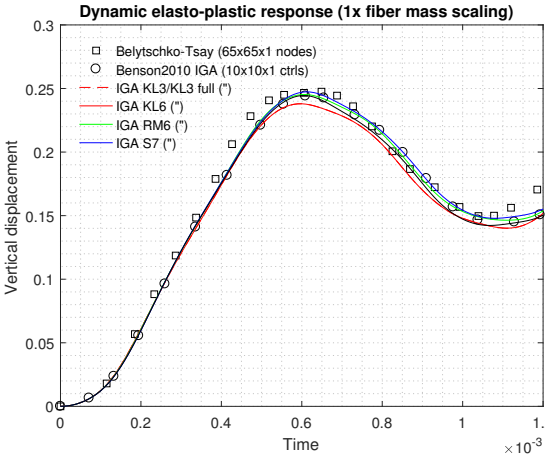
The last benchmark problem found from the literature evaluates the undamped dynamic elasto-plastic behavior of a square plate under an impulsively applied pressure load. The plate dimensions are given by the edge length $L = 10$ and the thickness $h = 0.5$. The properties of the elastic-perfectly-plastic material are given by the density $\rho = 2.588 \times 10^{-4}$, Young's modulus $E = 10^7$, Poisson's ratio $\nu = 0.3$, and the yield stress $\sigma_y = 3 \times 10^4$. The plate is simply supported and a uniform pressure $p = 300$ is applied to the reference surface (Figure 4.6a).

The dynamic response of all elements match well with the biquadratic isogeometric Reissner-Mindlin shell element proposed by Benson et al. (2010) as shown in Figure 4.6b. The response of the Belytschko-Tsay element (Belytschko et al., 1984) employing a significantly finer mesh in comparison to the other element types is given as a reference as well. The KL3 element deviates slightly from the shear deformable element formulations. Furthermore, no practical difference is observed between the shear corrected and full KL3 formulations.

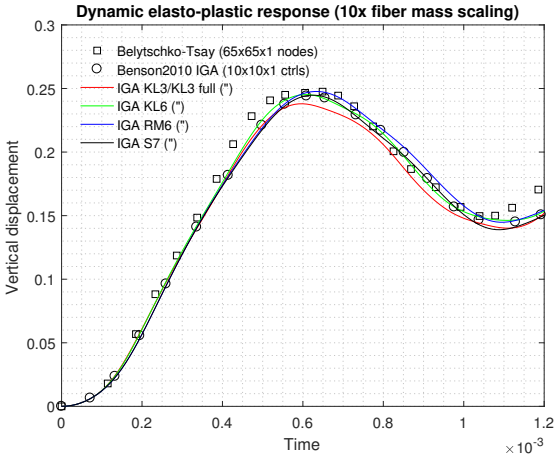
The degrees of freedom associated with the transverse deformations often needs be selectively scaled to increase the stable time step size. The influence of scaling these masses is evaluated in Figures 4.6c, 4.6d, and 4.6e for mass scaling factors of 10, 100, and 1000, respectively. Scaling these masses by 10 does not significantly alter the results for any element type. However, scaling of 100 already causes notable deviation in the dynamic response for the RM6 and S7 formulations. Increasing the corresponding mass scaling factor up to 1000 has a huge impact on the results provided by the RM6 and S7 formulations.



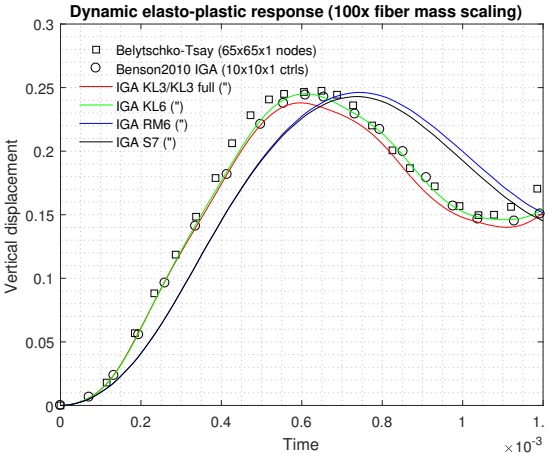
(a) The problem setup.



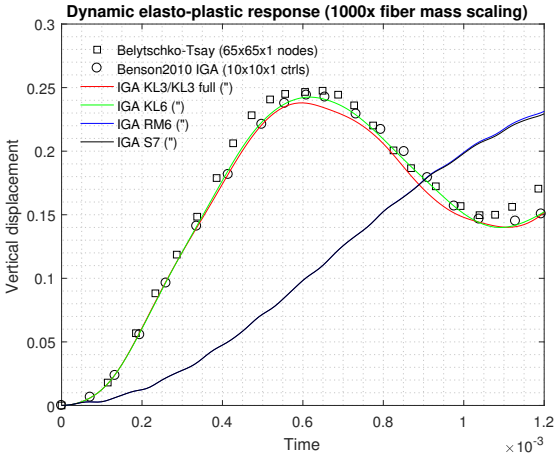
(b) The vertical displacement of the center point versus time (1x fiber mass scaling).



(c) The vertical displacement of the center point versus time (10x fiber mass scaling).



(d) The vertical displacement of the center point versus time (100x fiber mass scaling).



(e) The vertical displacement of the center point versus time (1000x fiber mass scaling).

Figure 4.6: Dynamic elasto-plastic response of a square sheet under a uniform pressure and the influence of fiber mass scaling.

The KL3 formulation possesses only 3 translational degrees of freedom associated with the mid-surface geometry and therefore is not affected by the applied mass scaling. However, the shear deformable KL6 formulation remains very insensitive to scaling the masses associated with the angular shear velocity. This is a very important feature of the proposed shear deformable formulation. In fact, if the selective mass scaling factor approaches infinity, the solution of the KL6 formulation just approaches the solution of the KL3 formulation as suggested by Figures 4.6c, 4.6d, and 4.6e.

The selective mass scaling is typically related to the element slenderness ratio such that when the element becomes thinner, the mass scaling factor must be increased. However, when the thickness approaches zero, the solution approaches the Kirchhoff-Love solution anyway as the transversal shear vanishes. Therefore, the proposed KL6 formulation which does not show any signs of shear locking is a very appealing choice for a wide variety of applications ranging from very thin problems to moderately thick problems.

4.7 Single point incremental forming example

In this section, the shell elements are applied to a single point incremental forming (SPIF) problem introduced by Hokkanen and Pedroso (2019b). A classical BWC shell element (Belytschko et al., 1992) is included as well. The formed part from the experiment shown in Figure 4.7 is 3D scanned to obtain a measure of the true thinning of the part. The resulting normalized thickness distribution is illustrated by Figure 4.8.

The simulation uses the stylus contact algorithm discussed in Section 3.6.2, which was also applied to ISF problems by Hokkanen et al. (2018) and Hokkanen and Pedroso (2019b). A uniform control point spacing is considered with three different space discretizations: 6 mm (coarse), 4 mm (medium), and 2 mm (fine). The resulting initial element dimensions along the direction of the edges of the sheet are therefore approximately 30%, 20%, and 10% of the stylus diameter, respectively. A detailed description of the geometry and numerical model is given by Hokkanen and Pedroso (2019b) (see Appendix B).

The deformed shape is plotted in Figure 4.9 for all space discretizations by considering a slice along the dotted line shown in Figure 4.8. The yellow region in the plots

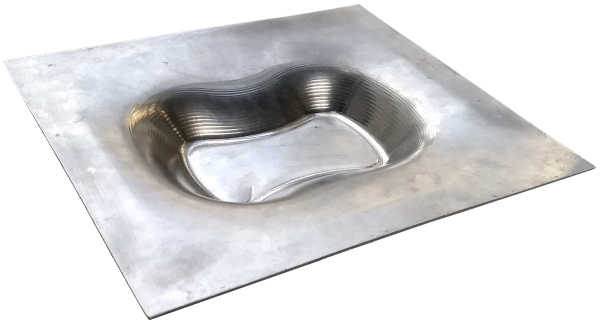


Figure 4.7: A photograph of the formed part (SPIF).

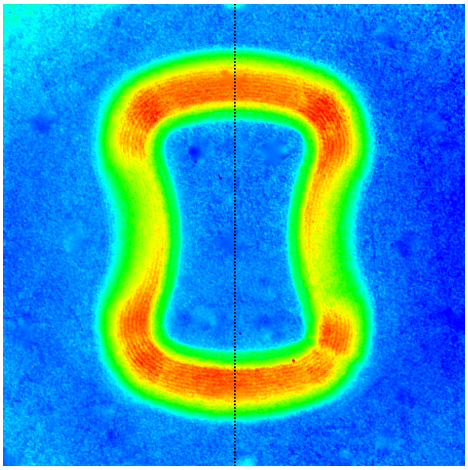
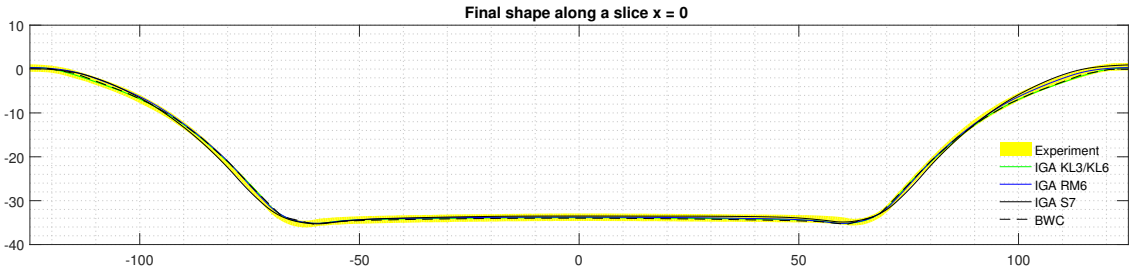
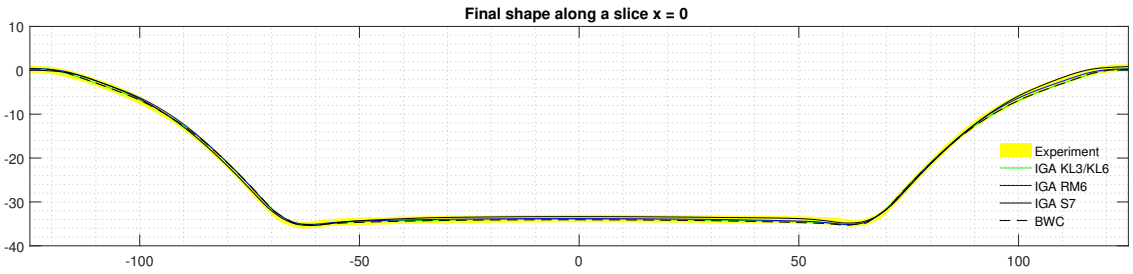


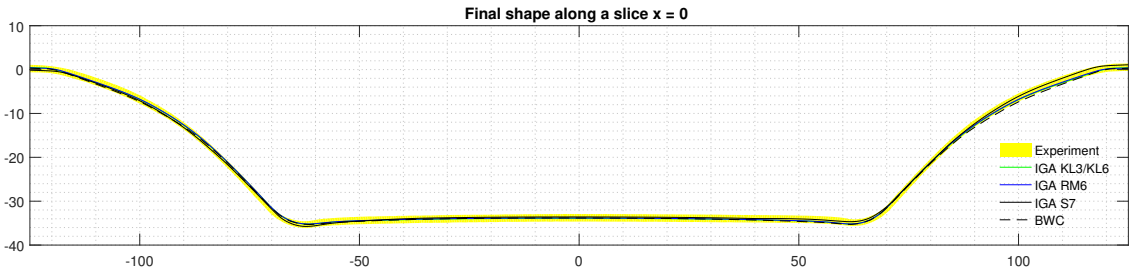
Figure 4.8: The experimental normalized thickness from the 3D scanning (SPIF). The contours vary linearly from 0.62 (red) to 1.0 (blue).



(a) 6mm grid.

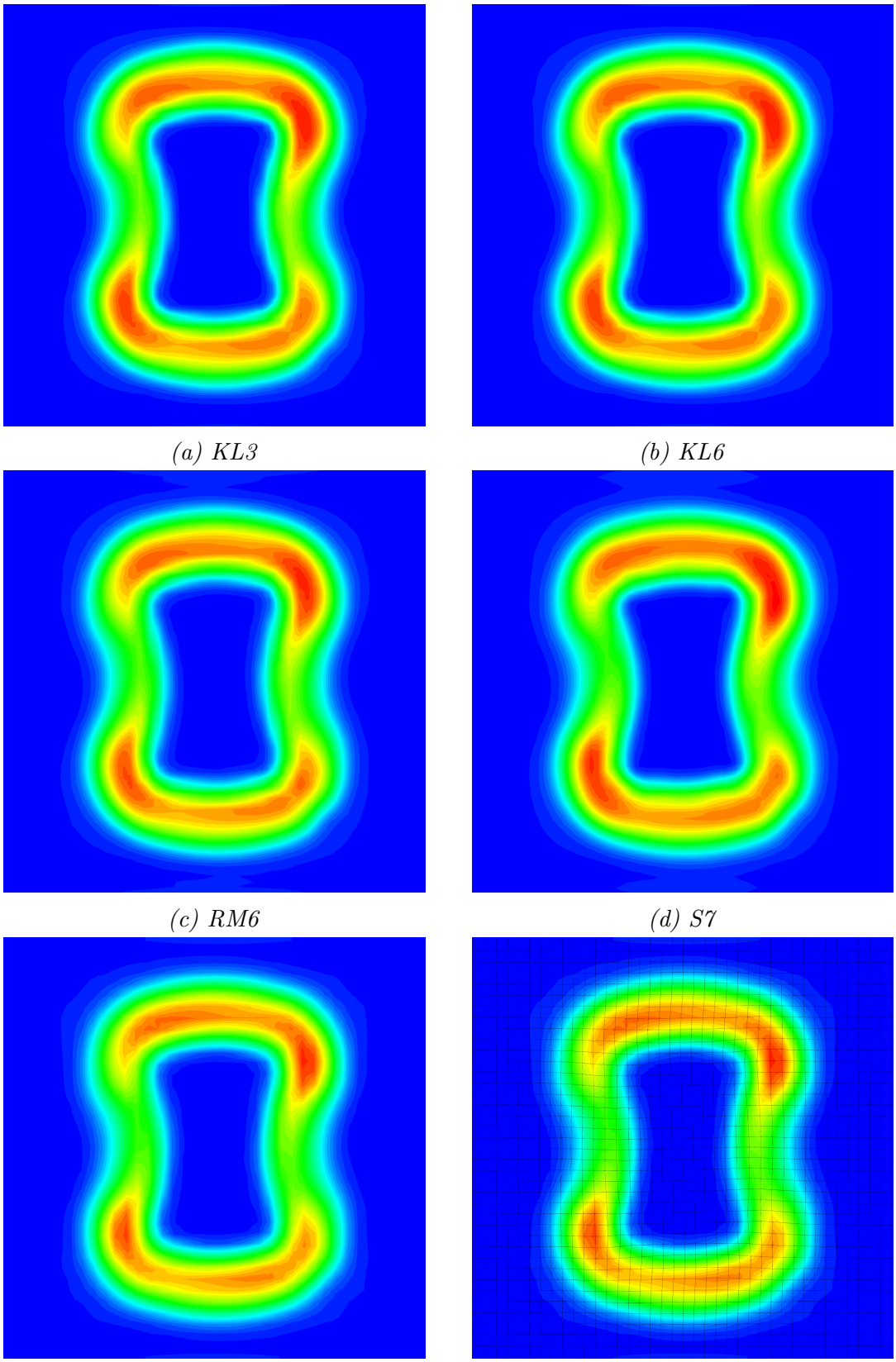


(b) 4mm grid.



(c) 2mm grid.

Figure 4.9: The final shape of the formed part along a slice $x = 0$ for the SPIF example.

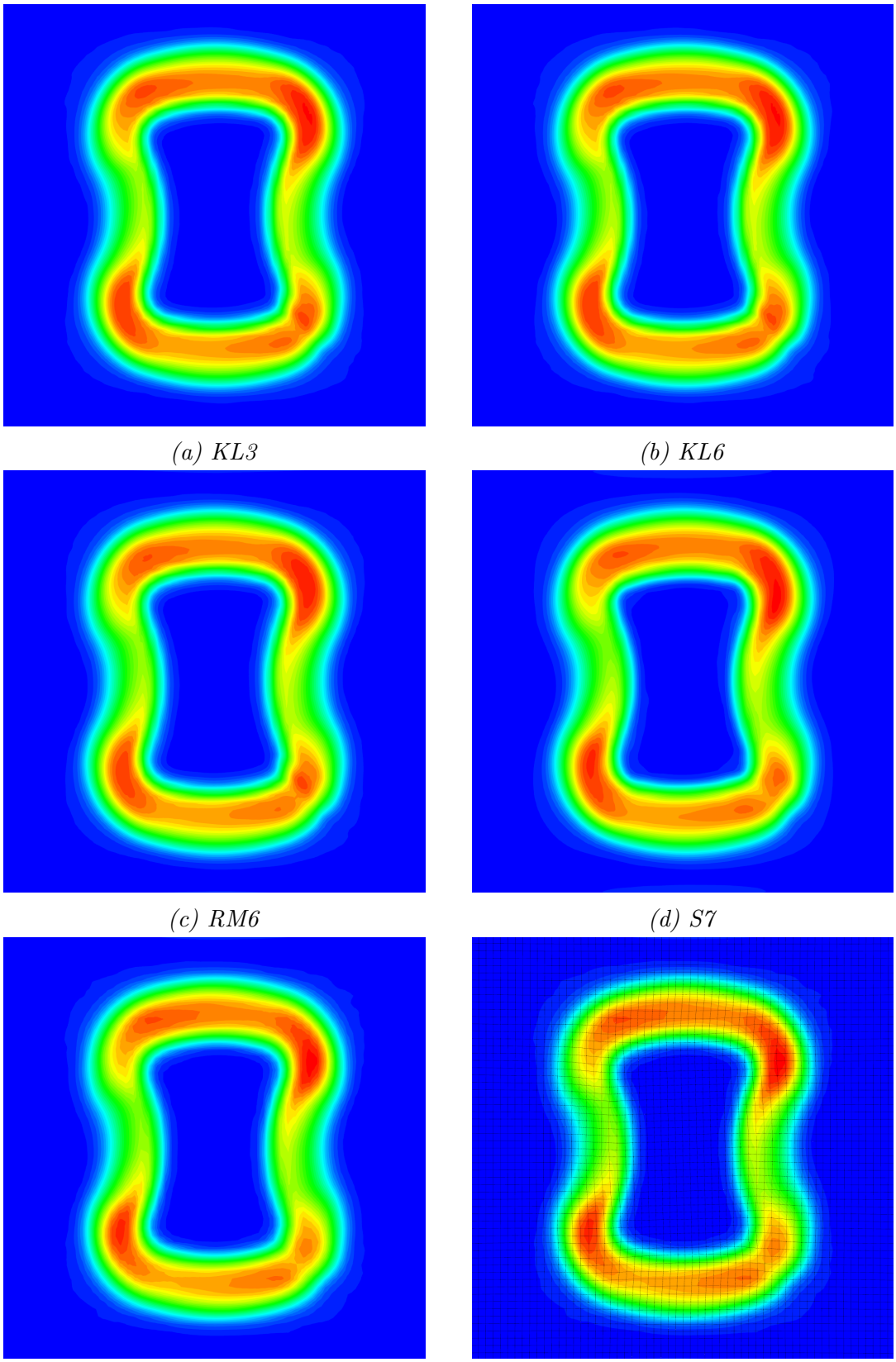


(a) KL3 *(b) KL6*

(c) RM6 *(d) S7*

(e) BWC w/o grid *(f) BWC w/ grid*

Figure 4.10: The normalized thickness plot of the SPIF example for each element type (6mm grid). The contours vary linearly from 0.62 (red) to 1.0 (blue).

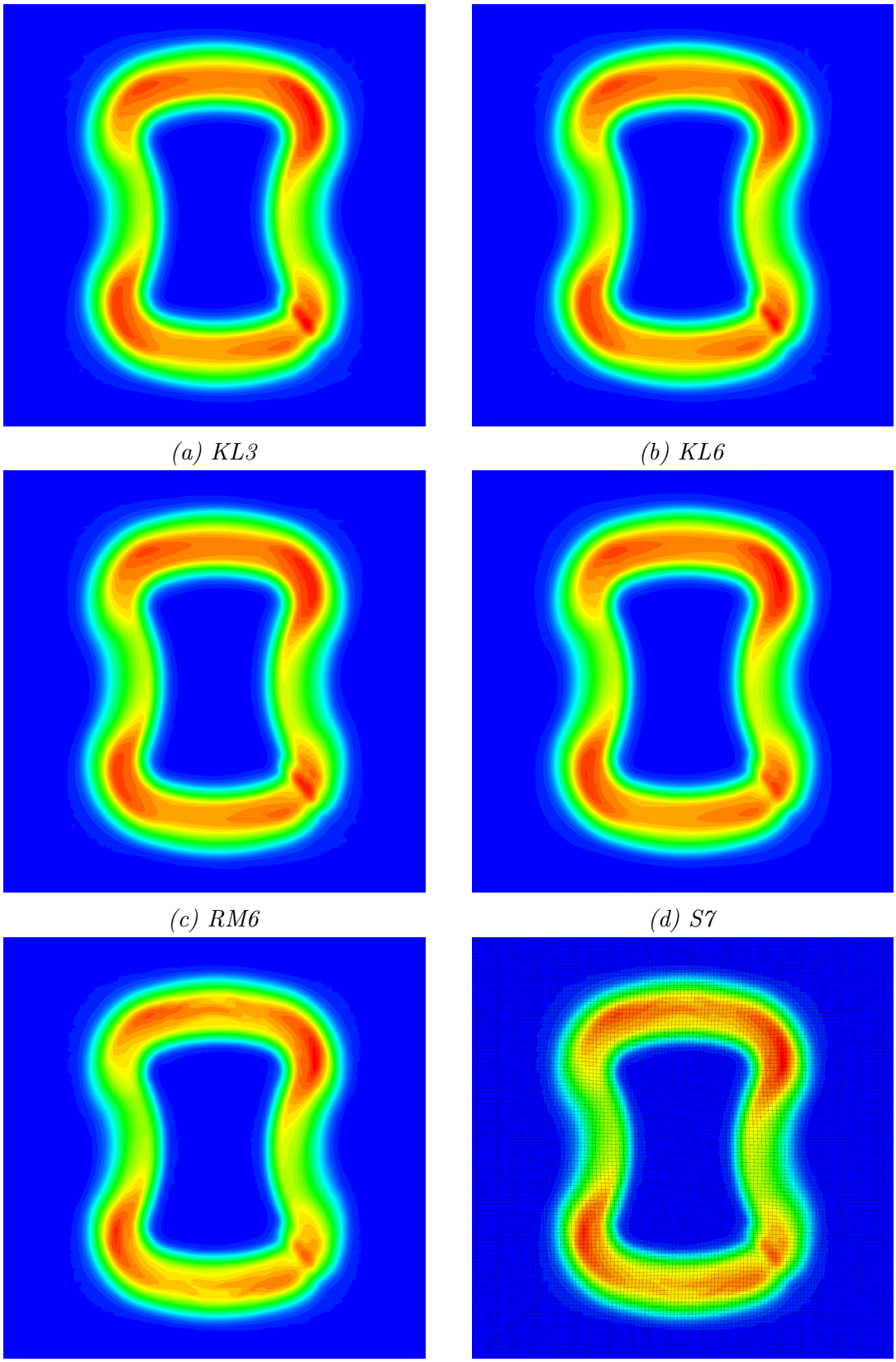


(a) KL3 *(b) KL6*

(c) RM6 *(d) S7*

(e) BWC w/o grid lines *(f) BWC w/ grid lines*

Figure 4.11: The normalized thickness plot of the SPIF example for each element type (4mm grid). The contours vary linearly from 0.62 (red) to 1.0 (blue).



(a) KL3 *(b) KL6*

(c) RM6 *(d) S7*

(e) BWC w/o grid lines *(f) BWC w/ grid lines*

Figure 4.12: The normalized thickness plot of the SPIF example for each element type (2mm grid). The contours vary linearly from 0.62 (red) to 1.0 (blue).

represent the experimental result, i.e., area between the top and the bottom surfaces of the sheet. This is obtained from the 3D scan data. The curves denoting the simulation results represent only the mid-surface of the sheet and do not contain any information about the simulated thinning of the sheet.

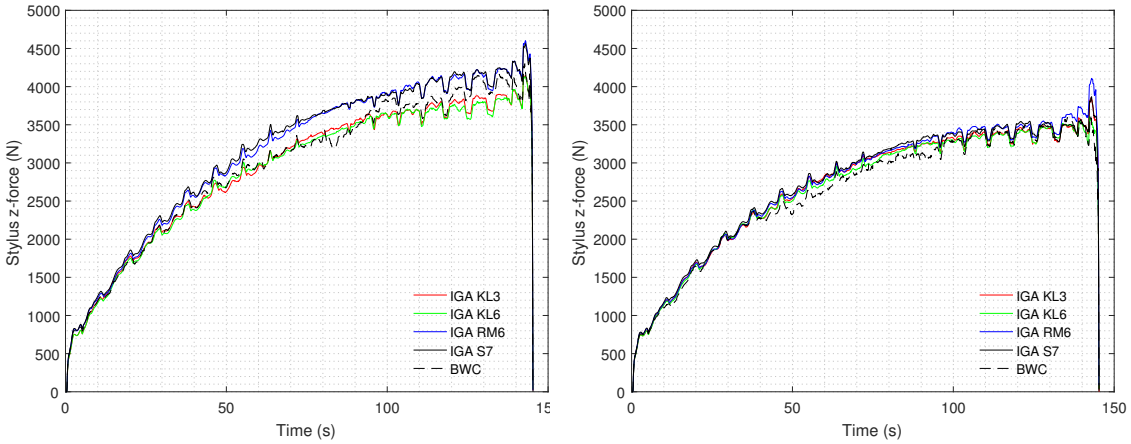
Overall, the simulated shapes along the slice agree very well with the experimental data. The KL3 and KL6 results are represented by a single curve as they produce nearly identical results. The thickness stretchable S7 formulation converges towards a slightly different solution as it does not make the plane stress assumption. A closer inspection reveals that the difference between the coarsest and the finest solutions is largest for the S7 formulation, and smallest for the KL3 and KL6 formulations. This suggests the best performance for the KL3 and KL6 element types.

The simulated thinning is shown in Figures 4.10, 4.11, and 4.12. The results match very well with the experimental thinning shown in Figure 4.8 when a fine grid is employed. A notable difference is observed between the coarsest and the finest discretizations. However, the differences between the formulations are quite small. Closer inspection of Figure 4.10 reveals that the BWC element underestimates the thinning most when compared to the finer grids. It is also noted that the BWC result for the finest discretization is slightly noisy in comparison with the isogeometric elements.

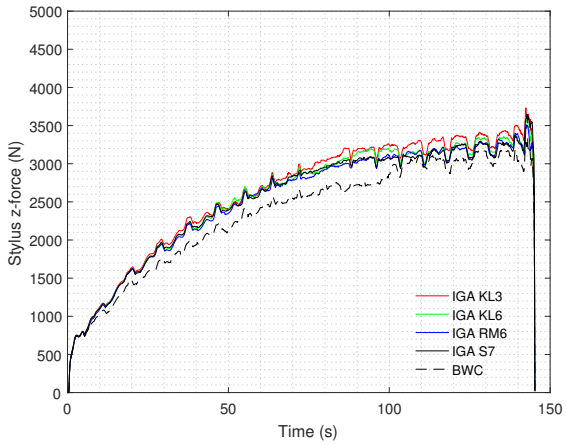
The simulated forming forces experienced by the stylus in the global z-direction (vertical direction) are shown in Figure 4.13. The forces are filtered as the raw data is somewhat noisy. Refining the grid consistently lowers the z-forces. This is expected behavior as the smaller elements can better capture the local material response, i.e., the local deformations do not influence an overly large area. The forces are still reduced over 5% by refining the grid size from 4 mm to 2 mm.

For the coarsest discretization, the RM6 and S7 formulations predict notably higher forces in comparison with KL3 and KL6 formulations. This is likely to be due to the KL3 and KL6 formulations being less prone to locking issues as demonstrated by the previous sections. However, the gap between the aforementioned elements narrows down when the grid is refined.

The forces predicted by the BWC formulation are slightly inconsistent in comparison with the isogeometric elements. The most notable deviation occurs for the finest discretization which may be related to the noisy thickness plot. Somewhat similar



(a) Coarse grid (6mm) *(b) Medium grid (4mm)*



(c) Fine grid (2mm).

Figure 4.13: Forming forces in the global z-direction (vertical direction) for the SPIF example.

behavior was observed by Hokkanen and Pedroso (2019b) with isogeometric elements when integration schemes that are prone to hourglassing were used. Therefore, the observation may be associated with the rank deficiency of the BWC element, even though an hourglass control is used.

Table 4.1: SPIF simulation runtime for each element type using Nvidia Titan V GPU.

Formulation	Order	Cost per el.	Time step	Runtime: 6/4/2 mm (s)
KL3	2	= 4	1.5x	542/1356/6893
KL6	2	= 4	1.5x	615/1483/7534
RM6	2	= 4	1.5x	585/1424/7651
S7	2	= 4	1.5x	489/1241/7145
BWC	1	= 1	0.9x	580/1157/4779

The simulation runtimes for different element types are shown in Table 4.1. The

isogeometric S7 formulation is fastest for the coarse grid whereas the KL6 formulation is slowest requiring ∼25% more time. For the medium grid, BWC formulation is fastest and the KL6 formulation, which is the slowest, takes ∼30% longer. For the finest mesh, the BWC is significantly faster than the isogeometric formulations, the slowest formulation requiring ∼60% more time than the BWC formulation.

In comparison with Lagrange finite elements, the isogeometric elements benefit from larger stable time step sizes. For the ISF problem of this section, the maximum stable time step size for the isogeometric elements is over 60% larger in comparison to the BWC shell element, and therefore, a significantly smaller number of time increments are required. On the other hand, the isogeometric elements use somewhat costlier spline interpolation and roughly four times as many stress integration points.

4.8 Two-point incremental forming example

The last section considers a two-point incremental forming (TPIF) problem introduced by Hokkanen et al. (2018). The formed part from the experiment shown in Figure 4.14 is 3D scanned to obtain a measure of the true thinning of the part. The resulting normalized thickness distribution is illustrated by Figure 4.15.

In addition to the stylus contact algorithm of Section 3.6.2, the die contact algorithm discussed in Section 3.6.3 is used. Similarly to Section 4.7, a uniform control point spacing is considered with three different space discretizations: 6 mm (coarse), 4 mm (medium), and 2 mm (fine). A detailed description of the geometry and numerical model is given by Hokkanen et al. (2018) (see Appendix C). However, the isotropic material model used in Section 4.7 is employed instead of the original anisotropic material model.

The deformed shape is plotted in Figure 4.16 for all space discretizations similarly to Section 4.7. The dotted line along which the slice is taken is shown in Figure 4.15. The simulated shapes along the slice are in a good agreement with the experimental result although the converged simulation results deviate slightly more from the experimental results in comparison with the single point forming problem of Section 4.7. All element formulations converge towards almost exactly the same solution.

The edges of the sheet bend upwards after the sheet is unclamped from the frame due to the springback effect. However, when the coarsest grid is employed, the RM6, S7, and BWC element formulations do not capture the springback effect correctly

Figure 4.14: A photograph of the formed part (TPIF).

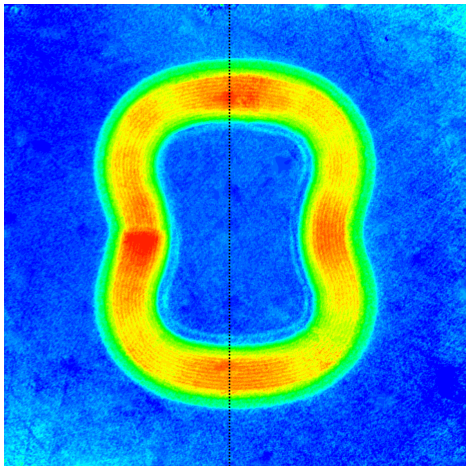
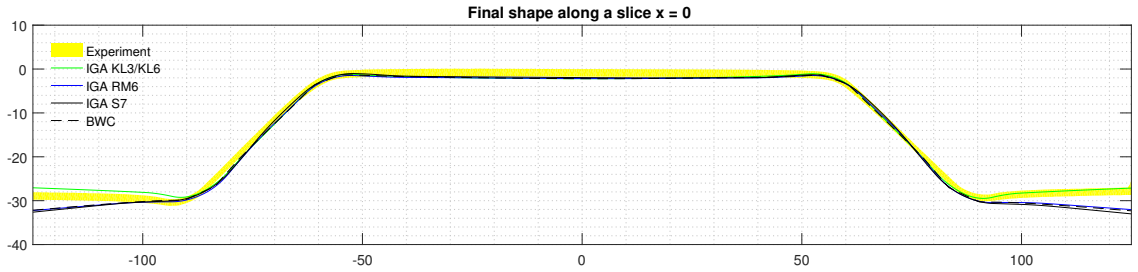
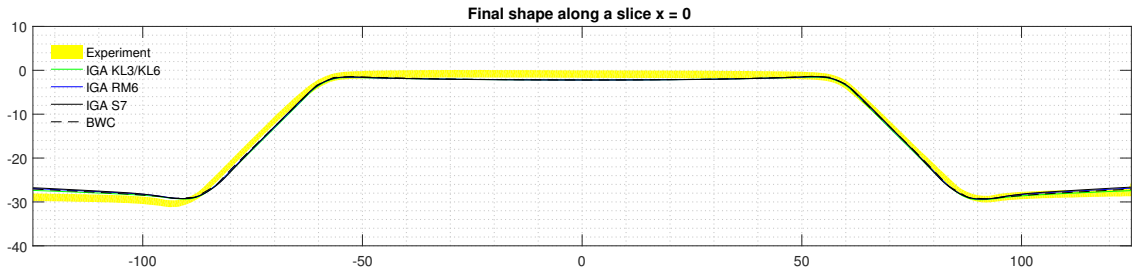


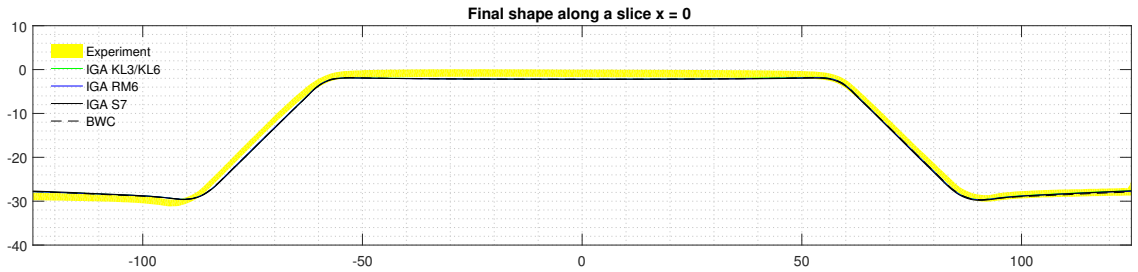
Figure 4.15: The experimental normalized thickness from the 3D scanning (TPIF). The contours vary linearly from 0.66 (red) to 1.0 (blue).



(a) 6mm grid.

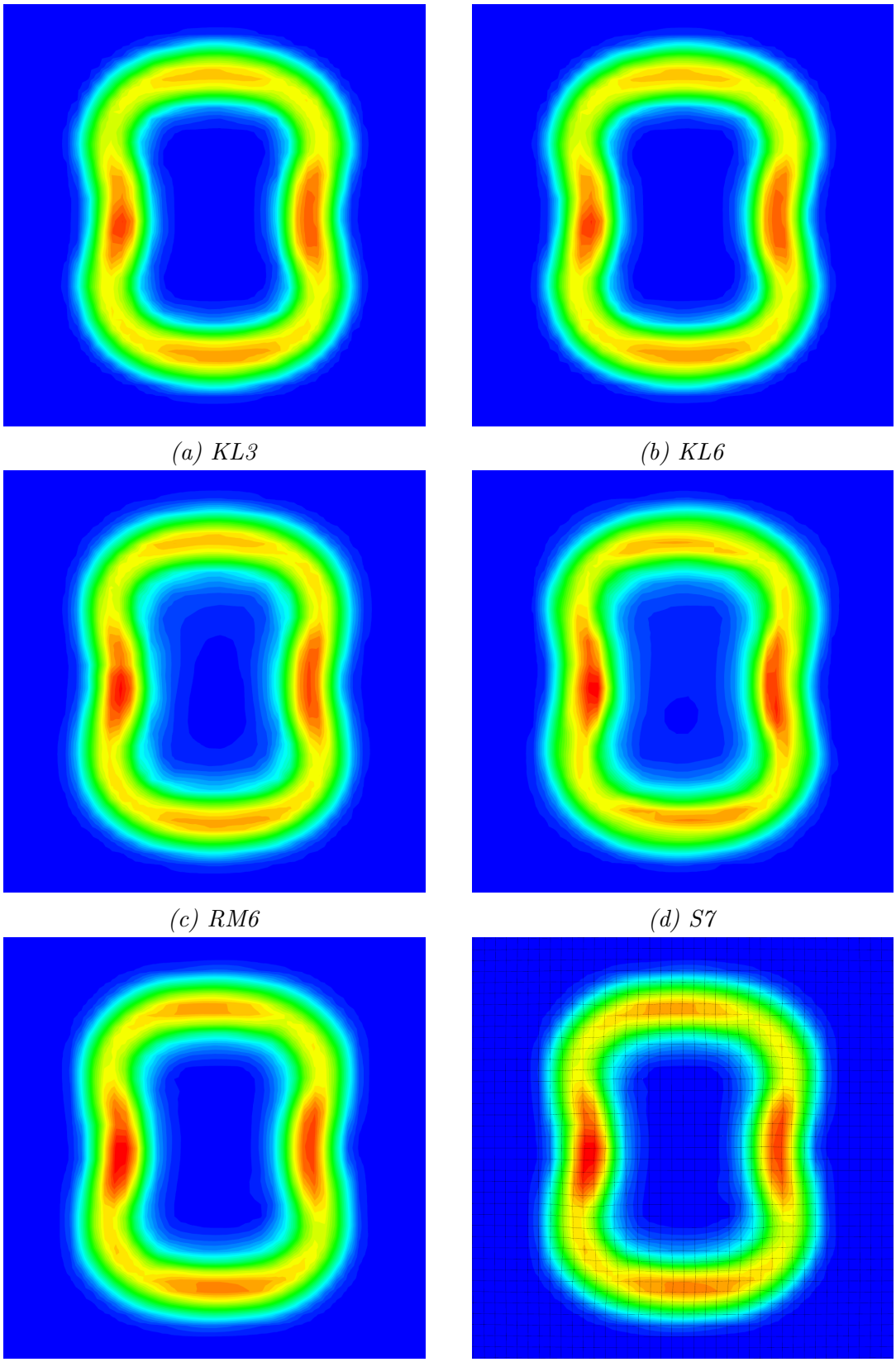


(b) 4mm grid.



(c) 2mm grid.

Figure 4.16: The final shape of the formed part along a slice $x = 0$ for the TPIF example.

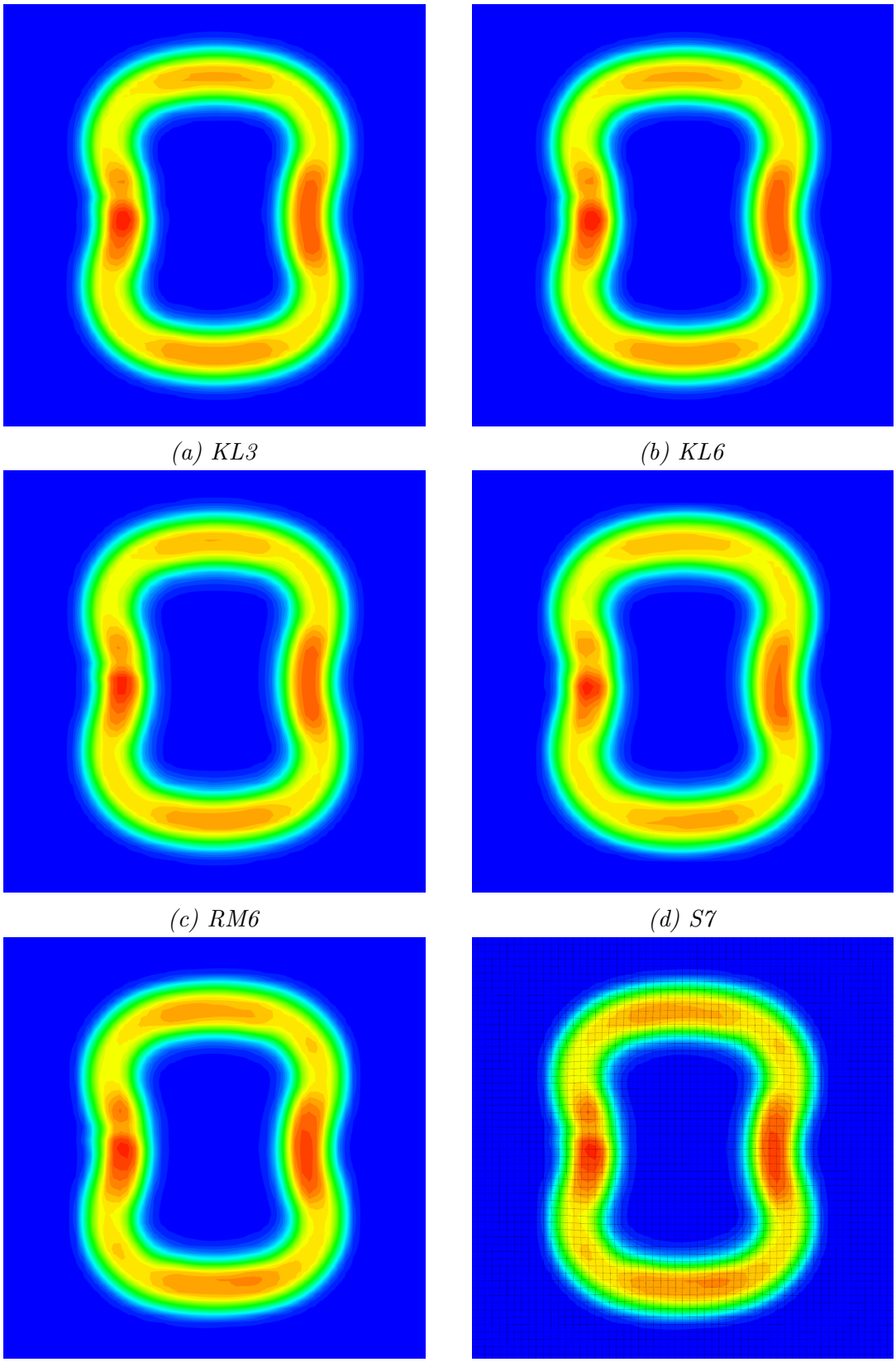


(a) KL3 *(b) KL6*

(c) RM6 *(d) S7*

(e) BWC w/o grid *(f) BWC w/ grid*

Figure 4.17: The normalized thickness plot of the TPIF example for each element type (6mm grid). The contours vary linearly from 0.66 (red) to 1.0 (blue).

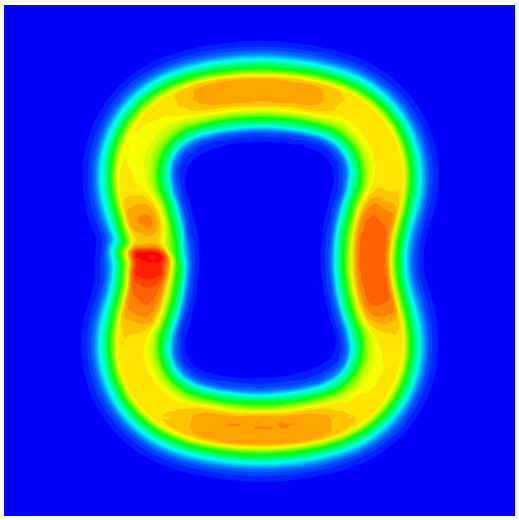


(a) KL3 *(b) KL6*

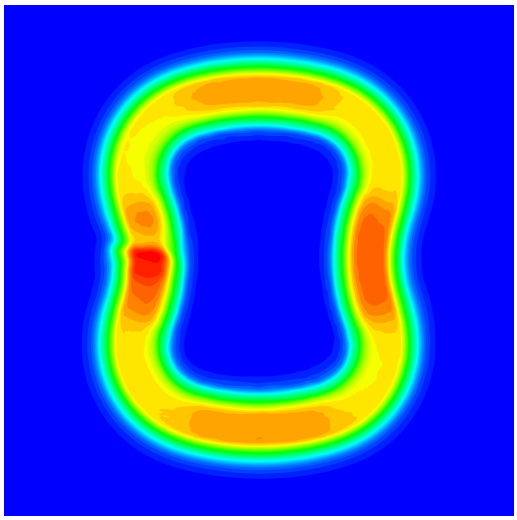
(c) RM6 *(d) S7*

(e) BWC w/o grid lines *(f) BWC w/ grid lines*

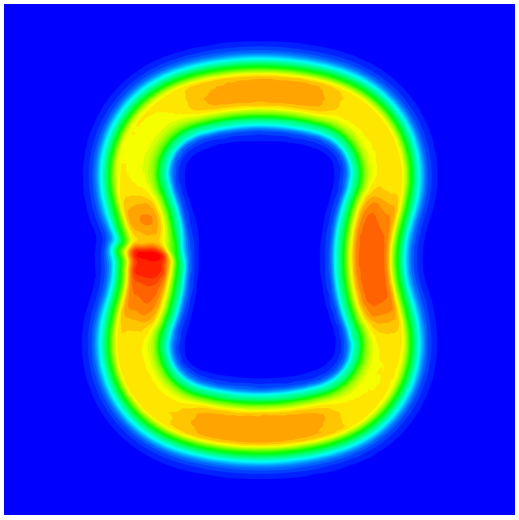
Figure 4.18: The normalized thickness plot of the TPIF example for each element type (4mm grid). The contours vary linearly from 0.66 (red) to 1.0 (blue).



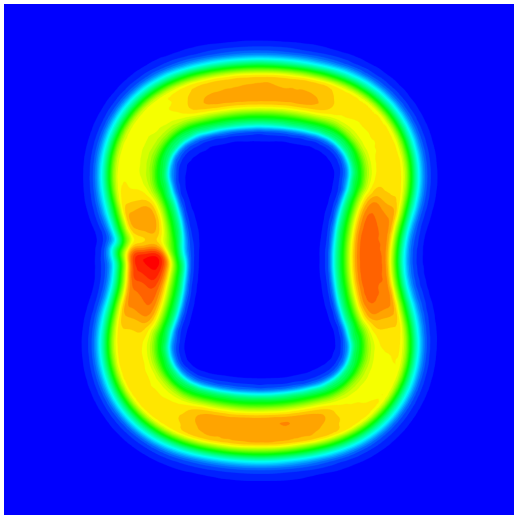
(a) KL3



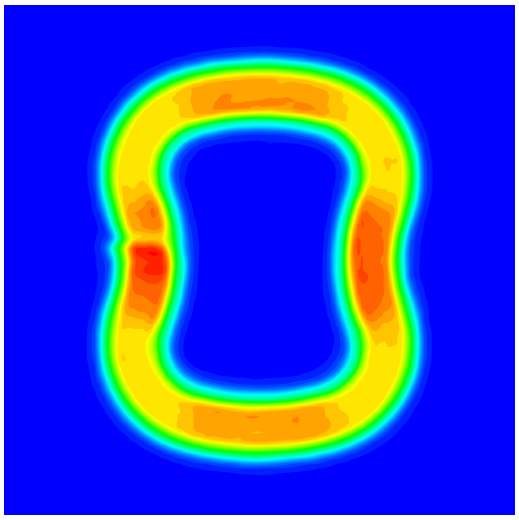
(b) KL6



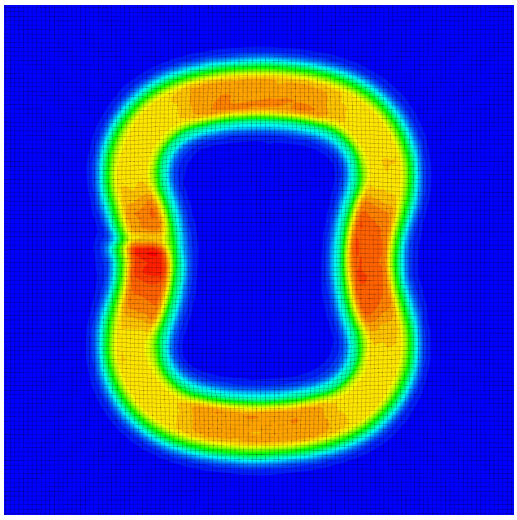
(c) RM6



(d) S7



(e) BWC w/o grid lines

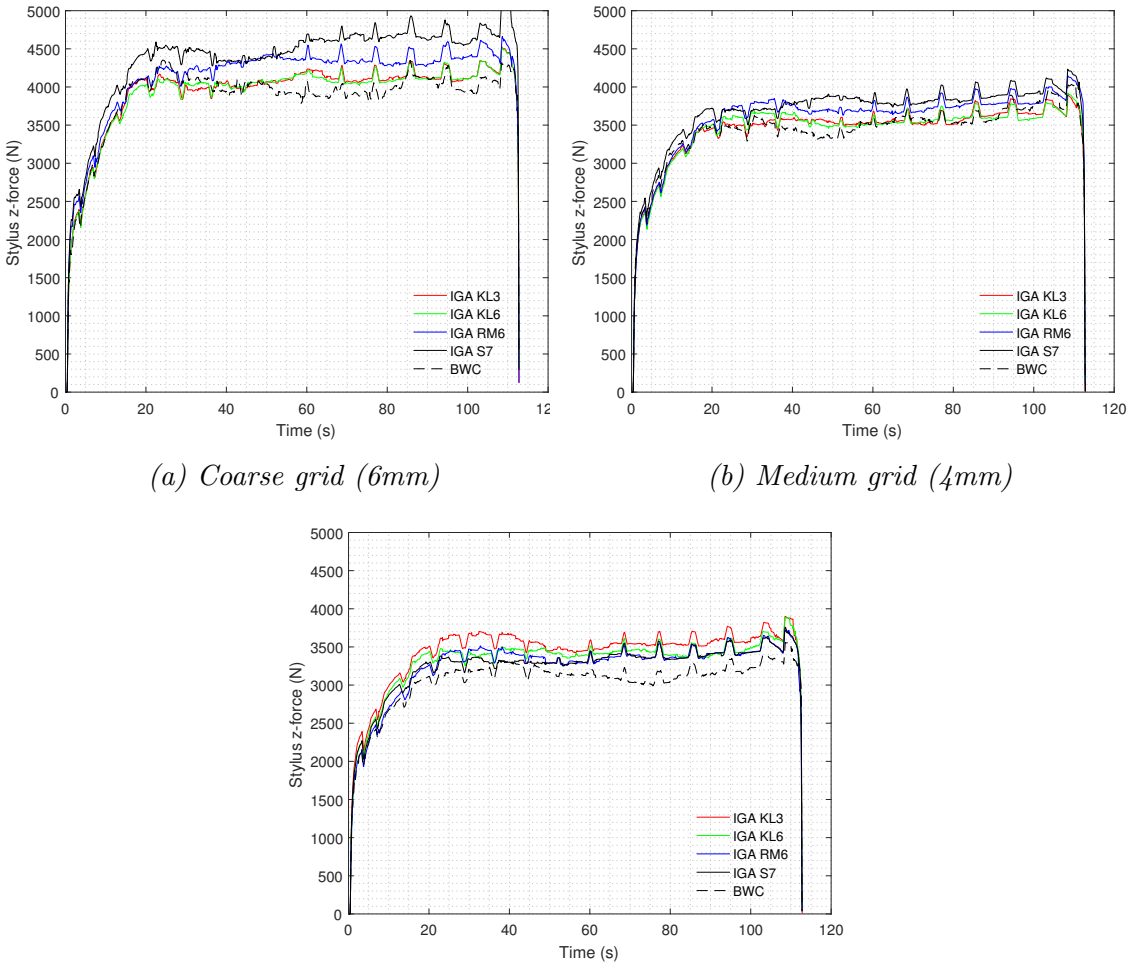


(f) BWC w/ grid lines

Figure 4.19: The normalized thickness plot of the TPIF example for each element type (2mm grid). The contours vary linearly from 0.66 (red) to 1.0 (blue).

whereas the KL3 and KL6 formulations do. This again suggests best accuracy for the KL3 and KL6 formulations.

The simulated thinning is shown in Figures 4.17, 4.18, and 4.19. Again, a very good match is obtained by using the finest discretization in comparison with the experimental thinning shown in Figure 4.15. However, now the difference between the coarsest and the finest discretizations is even larger in comparison with the single point forming problem, but the differences between the formulations are still small. The BWC element produces slightly thinner results than the isogeometric elements for the finest discretization.



(a) Coarse grid (6mm)

(b) Medium grid (4mm)

(c) Fine grid (2mm).

Figure 4.20: Forming forces in the global z-direction (vertical direction) for the TPIF example.

The simulated forming forces in the global z-direction are shown in Figure 4.20. Similarly to Section 4.7, refining the grid size consistently lowers the z-forces. Again, the RM6 and S7 formulations predict notably higher forces in comparison with the

KL3 and KL6 formulations for the coarsest discretization, although the gap between the former two element types is now much larger. The BWC element predicts consistently slightly lower forces than the isogeometric elements.

Table 4.2: TPIF simulation runtime for each element type using Nvidia Titan V GPU.

Formulation	Order	Cost per el.	Time step	Runtime: 6/4/2 mm (s)
KL3	2	= 4	1.5x	665/1409/7464
KL6	2	= 4	1.5x	647/1494/7926
RM6	2	= 4	1.5x	672/1467/7949
S7	2	= 4	1.5x	523/1367/7631
BWC	1	= 1	0.9x	671/1255/5593

The simulation runtimes for different element types are shown in Table 4.2. The results are very similar to Section 4.7 which implies that the die contact algorithm does not have significant influence to the relative performance between the elements. The isogeometric elements still suffer relatively more from the grid refinement in comparison to the BWC element. This is explained by the four times larger number of integration points per element and the limited parallel computing resources of the GPU. The computations start becoming increasingly serialized after a certain mesh size when the mesh is refined—the penalty due to this serialization is relatively larger for the isogeometric elements. This drawback can be avoided by employing a one-point in-plane quadrature for the isogeometric elements, but one should expect reduced accuracy and possible hourglassing problems (Hokkanen and Pedroso, 2019b).

Chapter 5

Conclusions

The main objective of this work was to investigate the applicability and beneficiality of isogeometric shell technology to incremental sheet forming simulations. However, most findings and contributions to the field are beneficial for a vast variety of applications ranging from modeling of bioprosthetic heart valves to automobile crash simulations. The most significant contributions include advancement of isogeometric shell technology, demonstration of usage of unclamped knot vectors in the isogeometric analysis, and evaluation of some recently proposed numerical integration rules using a complex real-world application. All important research objectives are fulfilled and ISF simulations are found to benefit from the isogeometric analysis in many ways with few disadvantages in comparison to classical finite elements.

An isogeometric thickness stretchable shell formulation was proposed by Hokkanen and Pedroso (2019a) and is given as Appendix A. This formulation supports full three-dimensional constitutive models and does not require the plane stress assumption. To improve the efficiency of the formulation, the stress resultant approach was adopted and the higher-order terms in the velocity gradient were neglected after performing a Taylor expansion about the reference surface. Furthermore, the concept of fiber mass scaling was successfully extended to the degree of freedom representing a quadratic thickness deformation to benefit from the larger stable time step size provided by the spline interpolation. The sensitivity of this degree of freedom to mass scaling was found to be small.

An explicit time integration scheme was used for numerical testing which makes the maximum stable time step size extremely small. Clamped and unclamped

knot vectors were compared, and unclamped knot vectors were found to provide a larger stable time step size for the boundary elements as suggested by Adam et al. (2015b). Moreover, an efficient technique compatible with diagonal mass matrices was proposed for enforcing Dirichlet boundary conditions at arbitrary locations within the patch. In the work of Hokkanen et al. (2018), the thickness stretchable shell was applied to an ISF simulation requiring an anisotropic material model and double-sided contact enforcement, see Appendix C.

In Section 3.3, three different shell formulations were developed based on Kirchhoff-Love and Reissner-Mindlin theories. The Kirchhoff-Love formulation which is comprised of three degrees of freedom per control point follows the developments of Benson et al. (2011), but the original formulation is modified in five ways: 1) the velocity gradient and the stress resultants are evaluated in a local coordinate system such that no computationally expensive transformations need be performed for any tensors of second order; 2) only terms up to linear along the thickness direction are considered for the velocity gradient in order to speed up the formulation; 3) the in-plane Jacobian matrix is decoupled from the through-thickness Jacobian such that only a 2×2 matrix inversion is calculated instead of a 3×3 inversion; 4) the through-thickness Jacobian is made independent of the control point normals and depends only on the local shell thickness; 5) the terms contributing to the transversal shear which were shown to vanish at the continuum limit are dropped in order to cure the shear locking problem. All five modifications reduce the number of floating point operations and therefore speed up the formulation. The last modification cures the observed shear locking issue, and as a consequence, improves the accuracy of the element significantly for thin problems.

Furthermore, a shear enhanced Kirchhoff-Love formulation was developed following the concept of hierarchic shells proposed by Echter et al. (2013). The resulting element does not show any signs of shear locking, and its only difference to the basic Kirchhoff-Love element is the additional degrees of freedom which represent angular shear velocity. This shear deformable formulation was shown to converge to the exact analytical Mindlin plate solution and it does not suffer from the increasing fiber mass scaling factor when the shell thickness decreases.

The third shell formulation resembles the Reissner-Mindlin shell proposed by Benson et al. (2013), but differs from the original formulation due to the aforementioned modifications 1–4. Typically to the Reissner-Mindlin shells, the accuracy of this

shear deformable shell element suffers from the increasing fiber mass scaling factor when approaching the thin limit. However, one advantage of this element is its ability to handle sharp corners (i.e., C^0 continuous geometry can be used), although this is rarely important in the context of ISF simulations.

A study on several quadrature rules recently proposed in the literature was conducted by Hokkanen and Pedroso (2019b) and is given as Appendix B. In contrast to most other works, a complex incremental sheet forming problem was used as one of the benchmark cases which led to some important new findings. The performance of the thickness stretchable shell element was found to benefit from using bicubic interpolation functions combined with a recently proposed patchwise quadrature rule. In comparison to the interpolation functions and the quadrature employed in the original paper (Hokkanen and Pedroso, 2019a), the results showed an improved convergence rate and less locking without introducing spurious modes or increasing the total computational cost of the ISF simulation. Additionally, an irregular patchwise quadrature recently proposed in the literature was found to result in a wavy pattern leading to a poor solution. Also, some simple stabilization strategies for quadratures suffering from spurious modes were found inadequate for a complex real-world simulation.

An automatically adjusting application-specific contact pressure smoothing algorithm was proposed in Section 3.6.2 which is applicable to the stylus contact modeling in ISF simulations. Furthermore, efficient contact tracking algorithms for the stylus and the die contact evaluation were discussed in Sections 3.6.2 and 3.6.3, respectively.

General-purpose computing on graphics processing units (GPGPU) was found to be a very cost-efficient way of speeding up the simulations, and is therefore highly recommended for future developments of IGA or FEM codes. The implemented GPU-based code was shown to be between one and two orders of magnitude faster for a representative incremental forming problem in comparison with traditional CPU-based implementations (e.g., Abaqus) on a modern desktop PC (Elford et al., 2018). The performance difference between the investigated isogeometric and classical elements is, however, much smaller. The developed isogeometric elements benefit from a larger stable time step size, but, on the other hand, require approximately four times as many integration points per element in comparison to the fastest classical shell elements. A reduction in the total number of time steps is however much more beneficial than a reduction in the number of integration points because

the time steps cannot be evaluated in parallel. The availability of parallel computing resources is likely to keep improving in the future, and therefore, the minimization of the number of integration points (which can be evaluated parallel) becomes less important. However, irrespective of the development of computing technology, the simulation time is expected to remain approximately proportional to the total number of time steps.

Although the implementation was moderately optimized, further optimizations have the potential to improve the performance significantly. Optimizing the GPU memory management using pitched memory allocations and optimal coalesced memory access patterns was shown to result in a very significant performance improvement for a trilinear solid-shell element with 24 degrees of freedom per element (Stephan et al., 2018). On the other hand, each isogeometric thickness stretchable shell element is associated with $7(p+1)^2$ degrees of freedom (where p denotes the order of the spline interpolation functions), and may therefore benefit proportionally more from optimizing the GPU memory management in comparison to the aforementioned solid-shell element, or any low-order classical shell element.

The following publication has been incorporated as Appendix A.

Hokkanen, J. and Pedroso, D. M. (2019). Isogeometric thickness stretchable shell: Efficient formulation for nonlinear dynamic problems. *International Journal for Numerical Methods in Engineering*, 119(2):105-127.

Contributor	Statement of contribution	%
Jaro Hokkanen	writing of text	100
	proof-reading	70
	theoretical derivations	80
	computer implementation	90
	numerical calculations	100
	preparation of figures	100
Dorival M. Pedroso	proof-reading	10
	supervision, guidance	80
	theoretical derivations	10
Michael C. Elford	proof-reading	10
	supervision, guidance	20
	computer implementation	10
Andrew J. E. Stephan	proof-reading	10
	theoretical derivations	10

Appendix A

Isogeometric thickness stretchable shell: efficient formulation for nonlinear dynamic problems

An isogeometric shell element with through-thickness stretch is introduced and applied to quasi-static and dynamic problems. The shell element supports full three-dimensional constitutive laws, i.e., the plane stress assumption is not required. An updated Lagrangian rate formulation is adopted, and biquadratic spline-based interpolation functions are used for in-plane interpolation. The concept of fiber mass scaling is proposed to lower the highest eigenfrequencies to improve the performance of the formulation. Clamped and unclamped knot vectors are compared, and the advantages of using unclamped knot vectors are demonstrated. The shell element is validated using several benchmark tests which indicate good performance of the proposed formulation.

A.1 Introduction

Isogeometric analysis (IGA) first proposed by Hughes et al. (2005) has attracted growing interest among the computational mechanics community. This concept adopts spline-based interpolation functions resulting in several potential advantages over conventional finite elements, such as increased accuracy per degree of freedom (Cottrell et al., 2006, 2007; Evans et al., 2009; Großmann et al., 2012), increased stable time step size (Adam et al., 2015b), and continuous stress and strain fields,

among others (Cottrell et al., 2009).

Many of the advantages result directly from the higher-order continuity of the interpolation functions, which also enables straightforward implementation of Kirchhoff-Love shell models (Kiendl et al., 2009; Benson et al., 2011, 2013). A Kirchhoff-Love shell formulation for general hyperelastic materials was proposed by Kiendl et al. (2015b), which was later applied to fluid-structure interaction simulations of bioprosthetic heart valves (Hsu et al., 2015). An application of a Kirchhoff-Love shell model to elasto-plastic materials was discussed by Ambati et al. (2018). A hierarchic family of isogeometric shell elements was proposed by Echter et al. (2013), and further developed by Oesterle et al. (2016, 2017), where a Kirchhoff-Love type shell model is used as a basis, and additional degrees of freedom are added to represent shear and thickness deformations.

Isogeometric Reissner-Mindlin shell elements were discussed by Benson et al. (2010), Dornisch et al. (2013), and Adam et al. (2015a), among others. Benson et al. (2010) considered explicit dynamics and scaled the rotational masses to increase the stable time step size.

Recently, the focus in the shell finite element technology has shifted towards more advanced formulations such as thickness stretchable shell elements and solid-shells. Two isogeometric solid-shell elements have been proposed by Hosseini et al. (2013, 2014); the latter formulation was extended to model propagation of delamination in composite materials (Hosseini et al., 2015). Other contributors to isogeometric solid-shell technology include Bouclier et al. (2013, 2015a,b), Cardoso and Cesar de Sa (2014), and Caseiro et al. (2015).

Most of the isogeometric solid-shell element formulations proposed in the literature do not consider dynamic problems nor the implications of using explicit time integration. However, numerous recently published contributions show a great interest in using solid-shell elements in explicit dynamic simulations, e.g., many formulations have been proposed in the context of classical finite elements (Tan and Vu-Quoc, 2005; Li et al., 2011b; Pagani et al., 2014; Mattern et al., 2015). Furthermore, explicit metal forming simulations with solid-shell elements were demonstrated by Li et al. (2011a) and Xu et al. (2012). Traditionally, bilinearly interpolated shell elements have dominated explicit analyses because using higher-order shape functions (which are Lagrange polynomials) reduces the stable time step size. This drawback of using higher-order interpolation functions does not necessarily show up in IGA. In fact,

the stable time step size increases with increasing smoothness across the elements (Adam et al., 2015b); this can be achieved by raising the order of the interpolation functions.

The purpose of this paper is to introduce an isogeometric solid-like shell formulation for dynamic simulations which is applicable to thick shell problems including double-sided contact. Therefore, no plane stress assumption is made. The proposed formulation strives for efficiency and robustness which is important for demanding highly nonlinear dynamic simulations such as automobile crash dynamics or sheet metal forming problems. The formulation is based on the degenerated solid approach with only translational degrees of freedom.

Reduced integration is used to alleviate several locking problems (Adam et al., 2015c) and to reduce computational cost. The chosen integration scheme preserves full rank of the stiffness matrix, and therefore no ad hoc hourglass control procedures are required. The concept of fiber mass scaling is adopted to prevent the eigenfrequencies associated with the transverse deformations from lowering the larger stable time step size allowed by the spline-based in-plane interpolation. Moreover, unclamped knot vectors are investigated due to their potential advantages over conventionally employed clamped knot vectors (Adam et al., 2015b).

The paper is structured as follows. Section A.2 briefly presents the calculation of b-spline and NURBS interpolation functions. Section A.3 explains thoroughly the internal, external, and inertial force calculations as well as the Dirichlet boundary conditions, the fiber mass scaling, and the stable time step estimate. In Section A.4, the proposed shell formulation is validated using biquadratic b-spline interpolation functions. Two linear elastic, and five nonlinear elastic or elasto-plastic benchmark cases are considered. Finally, conclusions are given in Section A.5.

The following notation is used throughout the text. The order of a tensor variable is indicated by the number of dots added under the corresponding symbol. For instance, $\underset{\cdot}{\boldsymbol{a}}$ and $\underset{\cdot\cdot}{\boldsymbol{a}}$ are different tensors of first and second orders, respectively. A similar dot on top of a variable indicates a time derivative. Furthermore, a partial derivative is denoted by a comma in the subscript. Lastly, although the concept of Bézier extraction is not employed, the subdomains bounded by the knots are still referred to as Bézier elements, similarly to the works of Schillinger et al. (2014), Adam et al. (2015c), and Hiemstra et al. (2017).

A.2 Isogeometric analysis fundamentals

The reference surface of the shell formulation is interpolated using spline-based interpolation functions. A general spline geometry is given by

$$\underset{\cdot}{\boldsymbol{r}}(\underset{\cdot}{\boldsymbol{\xi}}) = \sum_A N_A(\underset{\cdot}{\boldsymbol{\xi}})\underset{\cdot}{\boldsymbol{r}}_A \tag{A.1}$$

where $\underset{\cdot}{\boldsymbol{\xi}}$ represents all parametric coordinates and $\underset{\cdot}{\boldsymbol{r}}_A$ are the control points. The multivariate b-spline interpolation functions $N_A(\underset{\cdot}{\boldsymbol{\xi}})$ (later referred to as shape functions) are given as

$$N_A(\underset{\cdot}{\boldsymbol{\xi}}) = \prod_l N_A^l(\xi^l) \tag{A.2}$$

where $N_A^l(\xi^l)$ is the univariate b-spline basis function of the Ath control point. Denoting the order of the univariate b-spline basis function by p, one may calculate the basis function of the ith control point in the lth parametric direction using the Cox-de Boor recursion formula (Piegl and Tiller, 1996) as

$$\begin{aligned} N_{i0}^l(\xi^l) &= \begin{cases} 1 & \text{if } \xi_i^l \leq \xi^l \leq \xi_{i+1}^l \\ 0 & \text{otherwise} \end{cases} \\ N_{ip}^l(\xi^l) &= \frac{\xi^l - \xi_i^l}{\xi_{i+p}^l - \xi_i^l} N_{i(p-1)}^l(\xi^l) + \frac{\xi_{i+p+1}^l - \xi^l}{\xi_{i+p+1}^l - \xi_{i+1}^l} N_{(i+1)(p-1)}^l(\xi^l) \quad \forall\, p \in \mathbb{Z}^+ \end{aligned} \tag{A.3}$$

where $0/0 \equiv 0$ by convention. The knots ξ_i^l are stored in nondecreasing sets of real numbers known as knot vectors for which the following notation is used:

$$\Xi^l = \{\xi_1^l, \xi_2^l, \ldots, \xi_{n^l+p^l+1}^l\} \tag{A.4}$$

In (A.4) n^l and p^l represent the number of control points and the order associated with the lth parametric direction, respectively. If the values of the first $p^l + 1$ knots of a knot vector are equal, and also the last $p^l + 1$ knots are equal, the knot vector is called clamped. If neither end of the knot vector has $p^l + 1$ repeated knots, the knot vector is called unclamped. A simple algorithm to convert a geometry definition from clamped to unclamped knot vectors without changing the number of control points is given by Piegl and Tiller (1996). The exact geometry is preserved and all important mathematical properties required by the isogeometric analysis still hold throughout the geometric domain such as the partition of unity condition for the basis functions.

A generalized nonuniform rational b-spline (NURBS) geometry can be obtained by introducing a weight w associated with each control point, and replacing (A.2) by the rational multivariate interpolation functions

$$N_A(\dot{\boldsymbol{\xi}}) = \frac{w_A \prod_l N_A^l(\xi^l)}{\sum_B w_B \prod_l N_B^l(\xi^l)} \tag{A.5}$$

where $N_A^l(\xi^l)$ is still given by (A.3). It should be noted that the weights w_A do not change during the analysis, but come from the initial geometry definition.

A.3 Updated Lagrangian rate formulation

A.3.1 Shell kinematics

The kinematics of the proposed shell formulation are derived from the degenerated solid element approach where the shell geometry is described by the reference surface coordinates and the shell fiber vector which remains straight during the deformation. The shell geometry is hence written as

$$\dot{\boldsymbol{x}}(\xi,\eta,\zeta,t) = \underbrace{\tfrac{1}{2}\big(\dot{\boldsymbol{x}}_{top}(\xi,\eta,t) + \dot{\boldsymbol{x}}_{bot}(\xi,\eta,t)\big)}_{\dot{\boldsymbol{r}}(\xi,\eta,t)} + \tfrac{1}{2}\zeta \underbrace{\big(\dot{\boldsymbol{x}}_{top}(\xi,\eta,t) - \dot{\boldsymbol{x}}_{bot}(\xi,\eta,t)\big)}_{\dot{\boldsymbol{d}}(\xi,\eta,t)} \tag{A.6}$$

where $\dot{\boldsymbol{x}}_{top}(\xi,\eta,t)$ represents the top surface and $\dot{\boldsymbol{x}}_{bot}(\xi,\eta,t)$ the bottom surface. Furthermore, $\dot{\boldsymbol{r}}(\xi,\eta,t)$ and $\dot{\boldsymbol{d}}(\xi,\eta,t)$ represent the reference surface and the shell fiber vector, respectively. The extensible fiber shell formulations are capable of directly representing thickness changes and do support complete three-dimensional constitutive laws. Therefore, the plane stress enforcement is not needed. These type of formulations usually adopt additional degrees of freedom to avoid Poisson locking, which is caused by the lack of a linear transversal normal strain distribution, and which does not vanish even if the mesh is refined. The two common procedures to avoid the aforementioned problem are to either introduce internal degrees of freedom by following the enhanced assumed strain (EAS) procedure proposed by Simo and Rifai (1990), or to introduce a quadratic term in ζ contributing to the displacements. The latter approach was first proposed by Parisch (1995) who introduced the following shell geometry definition

$$\dot{\boldsymbol{x}} = \dot{\boldsymbol{r}} + \frac{\zeta}{2}\dot{\boldsymbol{d}} + (1-\zeta^2)q\dot{\boldsymbol{d}} \tag{A.7}$$

where the new term involving q vanishes when $\|\zeta\| = 1$, and therefore does not contribute to the top and bottom surface locations. The shell geometry definition

adopted herein was proposed by Seifert (1996) and results in a simpler formulation given by

$$\dot{\boldsymbol{x}} = \dot{\boldsymbol{r}} + \frac{\zeta}{2}\dot{\boldsymbol{d}} + \zeta^2 q \dot{\boldsymbol{d}} \tag{A.8}$$

where the quadratic term now vanishes at the reference surface. Regardless of the definition, the term involving q results in a linear transversal normal strain distribution (see Section A.3.4).

A.3.2 The principle of virtual power

The proposed shell element formulation is based on the Galerkin method. The expression for the virtual power including inertial, internal, and external virtual power contributions is given by (Belytschko et al., 2013)

$$\underbrace{\int_\Omega \rho \dot{\boldsymbol{a}} \cdot \delta \dot{\boldsymbol{v}} \, \mathrm{d}\Omega}_{\delta P_{iner}} + \underbrace{\int_\Omega \dot{\boldsymbol{\sigma}} : \delta \dot{\boldsymbol{D}} \, \mathrm{d}\Omega}_{\delta P_{int}} = \underbrace{\int_\Omega \dot{\boldsymbol{b}} \cdot \delta \dot{\boldsymbol{v}} \, \mathrm{d}\Omega + \int_\Gamma \dot{\boldsymbol{t}} \cdot \delta \dot{\boldsymbol{v}} \, \mathrm{d}\Gamma}_{\delta P_{ext}} \tag{A.9}$$

where ρ is the density, $\dot{\boldsymbol{a}}$ is the acceleration, $\delta\dot{\boldsymbol{v}}$ is the virtual velocity, $\dot{\boldsymbol{\sigma}}$ is the Cauchy stress, $\dot{\boldsymbol{b}}$ is the body force, and $\dot{\boldsymbol{t}}$ is the traction. The domain and its boundary are represented by Ω and Γ, respectively. The virtual rate-of-deformation, $\delta\dot{\boldsymbol{D}}$, is defined in terms of the spatial virtual velocity gradient $\delta\dot{\boldsymbol{v}}_{,\dot{\boldsymbol{x}}}$ by

$$\delta \dot{\boldsymbol{D}} = \frac{1}{2}(\delta \dot{\boldsymbol{v}}_{,\dot{\boldsymbol{x}}} + \delta \dot{\boldsymbol{v}}_{,\dot{\boldsymbol{x}}}^T) \tag{A.10}$$

A.3.3 Local coordinate system

A local orthogonal coordinate system independent of ζ is defined at each reference surface integration point. The covariant basis vectors $\dot{\boldsymbol{g}}_1$, $\dot{\boldsymbol{g}}_2$, and $\dot{\boldsymbol{g}}_3$ are defined as

$$\dot{\boldsymbol{g}}_1 = \dot{\boldsymbol{x}}_{,\xi}, \quad \dot{\boldsymbol{g}}_2 = \dot{\boldsymbol{x}}_{,\eta}, \quad \dot{\boldsymbol{g}}_3 = \dot{\boldsymbol{x}}_{,\zeta} \tag{A.11}$$

where $\dot{\boldsymbol{x}}$ represents the global coordinates of the current configuration. The basis vector $\dot{\boldsymbol{e}}_3$ for the local coordinate system is aligned along $\dot{\boldsymbol{g}}_3$:

$$\dot{\boldsymbol{e}}_3 = \frac{\dot{\boldsymbol{g}}_3}{\|\dot{\boldsymbol{g}}_3\|} \tag{A.12}$$

The remaining basis vectors $\dot{\boldsymbol{e}}_2$ and $\dot{\boldsymbol{e}}_1$ defining the local coordinate system are given as

$$\dot{\boldsymbol{e}}_2 = \frac{\dot{\boldsymbol{e}}_3 \times \dot{\boldsymbol{g}}_1}{\|\dot{\boldsymbol{e}}_3 \times \dot{\boldsymbol{g}}_1\|} \tag{A.13}$$

and

$$\dot{\boldsymbol{e}}_1 = \dot{\boldsymbol{e}}_2 \times \dot{\boldsymbol{e}}_3 \tag{A.14}$$

The transformation from the global to the local coordinate system is given by

$$\dot{\hat{\boldsymbol{x}}} = \ddot{\boldsymbol{R}}^T \cdot \dot{\boldsymbol{x}} \tag{A.15}$$

where the "hat" ($\hat{\bullet}$) is used to denote variables expressed in the local coordinate system. The rotation tensor $\ddot{\boldsymbol{R}}$ is constructed from the basis vectors $\dot{\boldsymbol{e}}_1$, $\dot{\boldsymbol{e}}_2$, and $\dot{\boldsymbol{e}}_3$ by

$$\boldsymbol{R} = [\boldsymbol{e}_1 \ \boldsymbol{e}_2 \ \boldsymbol{e}_3] \tag{A.16}$$

where the underdots are omitted due to the employed matrix notation.

Another option is to use a local orthogonal coordinate system where $\dot{\boldsymbol{e}}_3$ is always aligned along the surface normal, and the other two basis vectors are aligned with $\dot{\boldsymbol{g}}_1$ and $\dot{\boldsymbol{g}}_2$ as closely as possible (Hughes, 2012). This would result in many additional small terms to be calculated as the fiber is generally not aligned along the surface normal due to shear deformations. However, many Reissner-Mindlin shell elements assume that the fiber vector of the current configuration is normal to the reference surface in order to simplify the formulation and increase its robustness—without preventing transverse shear stresses from accumulating (Benson et al., 2013; Belytschko et al., 2013). This approach is suggested as the second option as it does not affect the computational efficiency and requires only a minimum change to the implementation (modification of the local coordinate system definition).

A.3.4 Internal force calculation

Taking the time derivative of (A.8), one ends up with the following expression for the velocity:

$$\dot{\boldsymbol{v}} = \dot{\boldsymbol{r}} + \frac{\zeta}{2}\dot{\boldsymbol{d}} + \zeta^2(\dot{q}\boldsymbol{d} + q\dot{\boldsymbol{d}}) \tag{A.17}$$

The spatial velocity gradient $\dot{\boldsymbol{v}}_{,\boldsymbol{x}}$ is calculated in the local coordinate system where $\hat{\boldsymbol{x}} = \hat{x}\hat{\boldsymbol{e}}_1 + \hat{y}\hat{\boldsymbol{e}}_2 + \hat{z}\hat{\boldsymbol{e}}_3$. Noting that $\boldsymbol{r}$, $\boldsymbol{d}$, and q are independent of $\hat{z}$, and since ζ is not a function of $\hat{x}$ or $\hat{y}$, the components of the spatial velocity gradient in the local coordinate system are given as

$$\begin{aligned}
\hat{\boldsymbol{v}}_{,\chi} &= \dot{\hat{\boldsymbol{r}}}_{,\chi} + \frac{\zeta}{2}\dot{\hat{\boldsymbol{d}}}_{,\chi} + \zeta^2(\dot{q}\hat{\boldsymbol{d}} + q\dot{\hat{\boldsymbol{d}}})_{,\chi} \\
\hat{\boldsymbol{v}}_{,\hat{z}} &= \frac{\zeta_{,\hat{z}}}{2}\dot{\hat{\boldsymbol{d}}} + 2\zeta\zeta_{,\hat{z}}(\dot{q}\hat{\boldsymbol{d}} + q\dot{\hat{\boldsymbol{d}}})
\end{aligned} \tag{A.18}$$

where χ represents either $\hat{x}$ or $\hat{y}$. Similarly,

$$\begin{aligned}\delta\hat{\dot{\boldsymbol{v}}}_{,\chi} &= \delta\dot{\hat{\boldsymbol{r}}}_{,\chi} + \frac{\zeta}{2}\delta\dot{\hat{\boldsymbol{d}}}_{,\chi} + \zeta^2(\delta\dot{q}\hat{\dot{\boldsymbol{d}}} + q\delta\dot{\hat{\boldsymbol{d}}})_{,\chi} \\ \delta\hat{\dot{\boldsymbol{v}}}_{,\hat{z}} &= \frac{\zeta_{,\hat{z}}}{2}\delta\dot{\hat{\boldsymbol{d}}} + 2\zeta\zeta_{,\hat{z}}(\delta\dot{q}\hat{\dot{\boldsymbol{d}}} + q\delta\dot{\hat{\boldsymbol{d}}})\end{aligned} \tag{A.19}$$

The derivative $\zeta_{,\hat{z}}$ is obtained from (A.8) by first calculating the component along $\hat{\dot{\boldsymbol{e}}}_3$ in the local coordinate system as

$$\hat{z} = \left(\hat{\dot{\boldsymbol{r}}} + \frac{\zeta}{2}\hat{\dot{\boldsymbol{d}}} + \zeta^2 q\hat{\dot{\boldsymbol{d}}}\right)\cdot\hat{\dot{\boldsymbol{e}}}_3 \tag{A.20}$$

Differentiating $\hat{z}$ with respect to ζ yields

$$\hat{z}_{,\zeta} = \left(\frac{1}{2}\hat{\dot{\boldsymbol{d}}} + 2\zeta q\hat{\dot{\boldsymbol{d}}}\right)\cdot\hat{\dot{\boldsymbol{e}}}_3 \tag{A.21}$$

hence,

$$\zeta_{,\hat{z}} = \frac{2}{(\hat{\dot{\boldsymbol{d}}} + 4\zeta q\hat{\dot{\boldsymbol{d}}})\cdot\hat{\dot{\boldsymbol{e}}}_3} \tag{A.22}$$

Substituting (A.22) into (A.18$_2$) and performing a Taylor expansion about the reference surface, one may represent the spatial velocity gradient in the following form:

$$\begin{aligned}\hat{\dot{\boldsymbol{v}}}_{,\chi} &= \dot{\hat{\boldsymbol{r}}}_{,\chi} + \frac{\zeta}{2}\dot{\hat{\boldsymbol{d}}}_{,\chi} + \zeta^2(\dot{q}\hat{\dot{\boldsymbol{d}}} + q\dot{\hat{\boldsymbol{d}}})_{,\chi} \\ \hat{\dot{\boldsymbol{v}}}_{,\hat{z}} &= \frac{1}{\hat{\dot{\boldsymbol{d}}}\cdot\hat{\dot{\boldsymbol{e}}}_3}\left(\dot{\hat{\boldsymbol{d}}} + 4\zeta\dot{q}\hat{\dot{\boldsymbol{d}}} - 16q\zeta^2\dot{q}\hat{\dot{\boldsymbol{d}}} + 64q^2\zeta^3\dot{q}\hat{\dot{\boldsymbol{d}}} - \dots\right)\end{aligned} \tag{A.23}$$

Parisch (1995), who first introduced a quadratic term in ζ for shell models, justified the exclusion of higher-order strain components in ζ by q being small in comparison with the other corresponding quantities. Considering only the terms constant and linear in ζ, and noting from (A.8) and (A.12) that

$$\hat{\dot{\boldsymbol{e}}}_3 = \frac{\hat{\dot{\boldsymbol{d}}}}{\hat{\dot{\boldsymbol{d}}}\cdot\hat{\dot{\boldsymbol{e}}}_3} \tag{A.24}$$

the final expressions for the spatial velocity gradient and the spatial virtual velocity gradient interpolated using spline-based shape functions become

$$\begin{aligned}\hat{\dot{\boldsymbol{v}}}_{,\chi} &= \sum_A N^0_{A,\chi}\dot{\hat{\boldsymbol{r}}}_A + \frac{\zeta}{2}\sum_A\left(N^1_{A,\chi}\dot{\hat{\boldsymbol{r}}}_A + N^0_{A,\chi}\dot{\hat{\boldsymbol{d}}}_A\right) \\ \hat{\dot{\boldsymbol{v}}}_{,\hat{z}} &= \frac{1}{\sum_B N_B\hat{\dot{\boldsymbol{d}}}_B\cdot\hat{\dot{\boldsymbol{e}}}_3}\sum_A N_A\dot{\hat{\boldsymbol{d}}}_A + 4\zeta\sum_A N_A\dot{q}_A\hat{\dot{\boldsymbol{e}}}_3\end{aligned} \tag{A.25}$$

and

$$\begin{aligned}\delta\hat{\dot{\boldsymbol{v}}}_{,\chi} &= \sum_A N^0_{A,\chi}\delta\hat{\dot{\boldsymbol{r}}}_A + \frac{\zeta}{2}\sum_A\left(N^1_{A,\chi}\delta\hat{\dot{\boldsymbol{r}}}_A + N^0_{A,\chi}\delta\hat{\dot{\boldsymbol{d}}}_A\right)\\ \delta\hat{\dot{\boldsymbol{v}}}_{,\hat{z}} &= \frac{1}{\sum_B N_B\hat{\dot{\boldsymbol{d}}}_B\cdot\hat{\dot{\boldsymbol{e}}}_3}\sum_A N_A\delta\hat{\dot{\boldsymbol{d}}}_A + 4\zeta\sum_A N_A\delta\dot{q}_A\hat{\dot{\boldsymbol{e}}}_3\end{aligned} \tag{A.26}$$

respectively. Due to definition (A.25), the quadratic term of (A.8) now only contributes to the linear component of the transversal normal strain, i.e., the component that is required to overcome Poisson locking. The proposed solid-like shell model is comprised of seven degrees of freedom associated with each A, which correspond to $\dot{\boldsymbol{r}}_A$, $\dot{\boldsymbol{d}}_A$, and $\dot{q}_A$.

In (A.25) and (A.26), the derivatives of the shape functions $N_{A,\chi}$ are approximated by the constant and linear terms in ζ, denoted by $N^0_{A,\chi}$ and $N^1_{A,\chi}$, respectively. A similar approach is suggested by Belytschko et al. (1989), who showed that the higher-order terms in ζ are small in the context of classical Reissner-Mindlin elements. As the shell formulation does not consider quadratic variation in ζ for the velocity strains, $N^1_{A,\chi}$ is not considered for $\dot{\boldsymbol{d}}_{,\chi}$. The relations for the derivatives of the shape functions with respect to the local coordinates are derived from

$$\begin{bmatrix}N_{A,\xi}\\ N_{A,\eta}\\ N_{A,\zeta}\end{bmatrix} = \boldsymbol{J}\begin{bmatrix}N_{A,\hat{x}}\\ N_{A,\hat{y}}\\ N_{A,\hat{z}}\end{bmatrix} = \begin{bmatrix}\hat{x}_{,\xi} & \hat{y}_{,\xi} & \hat{z}_{,\xi}\\ \hat{x}_{,\eta} & \hat{y}_{,\eta} & \hat{z}_{,\eta}\\ \hat{x}_{,\zeta} & \hat{y}_{,\zeta} & \hat{z}_{,\zeta}\end{bmatrix}\begin{bmatrix}N_{A,\hat{x}}\\ N_{A,\hat{y}}\\ N_{A,\hat{z}}\end{bmatrix} \tag{A.27}$$

where $\boldsymbol{J}$ is the Jacobian matrix of the transformation from the Euclidean to the parametric space. The expression for the shape function derivatives $N_{A,\hat{x}}$ and $N_{A,\hat{y}}$ can be written as

$$\begin{bmatrix}N_{A,\hat{x}}\\ N_{A,\hat{y}}\end{bmatrix} = \frac{1}{J}\begin{bmatrix}\hat{y}_{,\eta}\hat{z}_{,\zeta} - \hat{y}_{,\zeta}\hat{z}_{,\eta} & -\hat{y}_{,\xi}\hat{z}_{,\zeta} + \hat{y}_{,\zeta}\hat{z}_{,\xi}\\ -\hat{x}_{,\eta}\hat{z}_{,\zeta} + \hat{x}_{,\zeta}\hat{z}_{,\eta} & \hat{x}_{,\xi}\hat{z}_{,\zeta} - \hat{x}_{,\zeta}\hat{z}_{,\xi}\end{bmatrix}\begin{bmatrix}N_{A,\xi}\\ N_{A,\eta}\end{bmatrix} \tag{A.28}$$

where

$$J = \hat{x}_{,\zeta}\begin{vmatrix}\hat{y}_{,\xi} & \hat{z}_{,\xi}\\ \hat{y}_{,\eta} & \hat{z}_{,\eta}\end{vmatrix} - \hat{y}_{,\zeta}\begin{vmatrix}\hat{x}_{,\xi} & \hat{z}_{,\xi}\\ \hat{x}_{,\eta} & \hat{z}_{,\eta}\end{vmatrix} + \hat{z}_{,\zeta}\begin{vmatrix}\hat{x}_{,\xi} & \hat{y}_{,\xi}\\ \hat{x}_{,\eta} & \hat{y}_{,\eta}\end{vmatrix} \tag{A.29}$$

is the Jacobian determinant.

As $\hat{z}$ is aligned along ζ, $\hat{x}_{,\zeta} = \hat{y}_{,\zeta} = 0$. Additionally, as the shape functions are independent of ζ, they are independent of $\hat{z}$ as well, and therefore $N_{A,\hat{z}} = N_{A,\zeta} = 0$. Hence, (A.28) is simplified to

$$\begin{bmatrix}N_{A,\hat{x}}\\ N_{A,\hat{y}}\end{bmatrix} = \frac{1}{J}\begin{bmatrix}\hat{y}_{,\eta}\hat{z}_{,\zeta} & -\hat{y}_{,\xi}\hat{z}_{,\zeta}\\ -\hat{x}_{,\eta}\hat{z}_{,\zeta} & \hat{x}_{,\xi}\hat{z}_{,\zeta}\end{bmatrix}\begin{bmatrix}N_{A,\xi}\\ N_{A,\eta}\end{bmatrix} \tag{A.30}$$

where

$$J = \hat{z}_{,\zeta} \begin{vmatrix} \hat{x}_{,\xi} & \hat{y}_{,\xi} \\ \hat{x}_{,\eta} & \hat{y}_{,\eta} \end{vmatrix} \tag{A.31}$$

After carrying out a Taylor expansion of (A.30) about the reference surface, and retaining only the constant and linear terms in ζ, one obtains

$$\begin{bmatrix} N_{A,\hat{x}} \\ N_{A,\hat{y}} \end{bmatrix} \approx \begin{bmatrix} N^0_{A,\hat{x}} \\ N^0_{A,\hat{y}} \end{bmatrix} + \frac{\zeta}{2} \begin{bmatrix} N^1_{A,\hat{x}} \\ N^1_{A,\hat{y}} \end{bmatrix} \tag{A.32}$$

The constant terms in ζ are given as

$$\begin{bmatrix} N^0_{A,\hat{x}} \\ N^0_{A,\hat{y}} \end{bmatrix} = \frac{1}{J_0} \begin{bmatrix} r_{\hat{y},\eta} & -r_{\hat{y},\xi} \\ -r_{\hat{x},\eta} & r_{\hat{x},\xi} \end{bmatrix} \begin{bmatrix} N_{A,\xi} \\ N_{A,\eta} \end{bmatrix} \tag{A.33}$$

and the linear terms by

$$\begin{bmatrix} N^1_{A,\hat{x}} \\ N^1_{A,\hat{y}} \end{bmatrix} = \frac{1}{J_0} \begin{bmatrix} d_{\hat{y},\eta} - \frac{J_1}{J_0} r_{\hat{y},\eta} & -d_{\hat{y},\xi} + \frac{J_1}{J_0} r_{\hat{y},\xi} \\ -d_{\hat{x},\eta} + \frac{J_1}{J_0} r_{\hat{x},\eta} & d_{\hat{x},\xi} - \frac{J_1}{J_0} r_{\hat{x},\xi} \end{bmatrix} \begin{bmatrix} N_{A,\xi} \\ N_{A,\eta} \end{bmatrix} \tag{A.34}$$

where

$$\begin{aligned} J_0 &= r_{\hat{x},\xi} r_{\hat{y},\eta} - r_{\hat{x},\eta} r_{\hat{y},\xi} \\ J_1 &= r_{\hat{x},\xi} d_{\hat{y},\eta} + r_{\hat{y},\eta} d_{\hat{x},\xi} - r_{\hat{x},\eta} d_{\hat{y},\xi} - r_{\hat{y},\xi} d_{\hat{x},\eta} \end{aligned} \tag{A.35}$$

Instead of directly interpolating the fiber vectors, only the displacements of the fiber vectors are interpolated by the control points. In the initial undeformed configuration, the fibers remain normal to the surface and can be obtained from the surface normal. For quadratic and higher order interpolation, the surface normal is fully defined within the patch if no repeated knots are introduced. The fiber vector $\dot{\boldsymbol{d}}$ is modeled by summing the exact initial fiber and the interpolated fiber displacement as

$$\dot{\boldsymbol{d}} = h^0 \dot{\boldsymbol{n}}^0 + \sum_A N_A \dot{\boldsymbol{w}}_A \tag{A.36}$$

where h^0 is the initial thickness, $\dot{\boldsymbol{w}}_A$ is the fiber displacement of the Ath control point, and the initial normal vector $\dot{\boldsymbol{n}}^0$ is defined by the basis vectors $\dot{\boldsymbol{g}}^0_1$ and $\dot{\boldsymbol{g}}^0_2$ of (A.11) in the undeformed configuration as

$$\dot{\boldsymbol{n}}^0 = \frac{\dot{\boldsymbol{g}}^0_1 \times \dot{\boldsymbol{g}}^0_2}{\|\dot{\boldsymbol{g}}^0_1 \times \dot{\boldsymbol{g}}^0_2\|} \tag{A.37}$$

The derivatives $\dot{\boldsymbol{d}}_{,\xi}$ and $\dot{\boldsymbol{d}}_{,\eta}$ needed in the calculation of the Jacobian matrix are given by

$$\dot{\boldsymbol{d}}_{,\xi} = h^0 \dot{\boldsymbol{n}}^0_{,\xi} + \sum_A N_{A,\xi} \dot{\boldsymbol{w}}_A, \quad \dot{\boldsymbol{d}}_{,\eta} = h^0 \dot{\boldsymbol{n}}^0_{,\eta} + \sum_A N_{A,\eta} \dot{\boldsymbol{w}}_A \tag{A.38}$$

where the initial thickness h^0 is assumed to be constant across the domain, and the derivatives of the normal vector in the initial local coordinate system are given by

$$\begin{bmatrix} n^0_{\hat{x},\xi} \\ n^0_{\hat{y},\xi} \\ n^0_{\hat{z},\xi} \end{bmatrix} = \sum_A \begin{bmatrix} -N_{A,\hat{x}\xi} r^0_{\hat{z}A} \\ -N_{A,\hat{y}\xi} r^0_{\hat{z}A} \\ 0 \end{bmatrix}, \quad \begin{bmatrix} n^0_{\hat{x},\eta} \\ n^0_{\hat{y},\eta} \\ n^0_{\hat{z},\eta} \end{bmatrix} = \sum_A \begin{bmatrix} -N_{A,\hat{x}\eta} r^0_{\hat{z}A} \\ -N_{A,\hat{y}\eta} r^0_{\hat{z}A} \\ 0 \end{bmatrix} \tag{A.39}$$

The second derivatives of the shape functions $N_{A,\hat{x}\xi}$, $N_{A,\hat{x}\eta}$, $N_{A,\hat{y}\xi}$, and $N_{A,\hat{y}\eta}$ appear in the calculation of $\dot{\boldsymbol{n}}^0_{,\xi}$ and $\dot{\boldsymbol{n}}^0_{,\eta}$. These initial normal derivatives are constant throughout the analysis and can be calculated for each reference surface integration point in advance.

To evaluate the Cauchy stress correctly, one must employ an objective stress rate. The commonly used Jaumann rate of Kirchhoff stress (Bazant and Cedolin, 2010) is given by

$$\overset{\triangledown}{\dot{\boldsymbol{\tau}}} = \dot{\boldsymbol{\tau}} + \boldsymbol{\tau} \cdot \boldsymbol{\omega} - \boldsymbol{\omega} \cdot \boldsymbol{\tau} \tag{A.40}$$

where $\boldsymbol{\omega} = \frac{1}{2}(\boldsymbol{v}_{,\boldsymbol{x}} - \boldsymbol{v}^T_{,\boldsymbol{x}})$ is the skew-symmetric spin tensor and $\boldsymbol{\tau}$ is the symmetric Kirchhoff stress. The constitutive relation between $\overset{\triangledown}{\dot{\boldsymbol{\tau}}}$ and the rate-of-deformation tensor $\boldsymbol{D} = \frac{1}{2}(\boldsymbol{v}_{,\boldsymbol{x}} + \boldsymbol{v}^T_{,\boldsymbol{x}})$ is

$$\overset{\triangledown}{\dot{\boldsymbol{\tau}}} = \boldsymbol{C} : \boldsymbol{D} \tag{A.41}$$

where $\boldsymbol{C}$ is the elasto-plastic tangent modulus for the Jaumann rate of Kirchhoff stress. The employed relation for $\overset{\triangledown}{\dot{\boldsymbol{\tau}}}$ and $\dot{\boldsymbol{\sigma}}$ which preserves second-order accuracy in the internal work of infinitesimal stress increments is given as follows (Bazant and Cedolin, 2010):

$$\dot{\boldsymbol{\sigma}} = \overset{\triangledown}{\dot{\boldsymbol{\tau}}} + \boldsymbol{\omega} \cdot \boldsymbol{\sigma} - \boldsymbol{\sigma} \cdot \boldsymbol{\omega} - \mathrm{tr}(\boldsymbol{D})\boldsymbol{\sigma} \tag{A.42}$$

The stress rotation between the global and the local coordinate systems is given by

$$\hat{\boldsymbol{\sigma}} = \boldsymbol{R}^T \cdot \boldsymbol{\sigma} \cdot \boldsymbol{R} \tag{A.43}$$

However, the formulation is not restricted to the Jaumann stress rate, and one is free to choose another objective stress rate. For example, the rate of corotational Cauchy

stress may be an appealing choice with the second local coordinate system suggested in Section A.3.3, particularly if an anisotropic material model is used (Belytschko et al., 2013).

The internal forces for each control point are given in the global coordinate system by

$$\begin{aligned}
\underset{\cdot}{\boldsymbol{f}}^r_{intA} &= \int_S \underset{\cdot\cdot}{\boldsymbol{R}} \cdot \left(N^0_{A,\hat{\underset{\cdot}{\boldsymbol{x}}}} \cdot \hat{\underset{\cdot\cdot}{\boldsymbol{f}}}^0 + N^1_{A,\hat{\underset{\cdot}{\boldsymbol{x}}}} \cdot \hat{\underset{\cdot\cdot}{\boldsymbol{f}}}^1 \right) \mathrm{d}\xi \mathrm{d}\eta \\
\underset{\cdot}{\boldsymbol{f}}^d_{intA} &= \int_S \underset{\cdot\cdot}{\boldsymbol{R}} \cdot \left(N^0_{A,\hat{\underset{\cdot}{\boldsymbol{x}}}} \cdot \hat{\underset{\cdot\cdot}{\boldsymbol{f}}}^1 + \frac{N_A \hat{\underset{\cdot}{\boldsymbol{e}}}_3}{\sum_B N_B \hat{\underset{\cdot}{\boldsymbol{d}}}_B \cdot \hat{\underset{\cdot}{\boldsymbol{e}}}_3} \cdot \hat{\underset{\cdot\cdot}{\boldsymbol{f}}}^0 \right) \mathrm{d}\xi \mathrm{d}\eta \\
f^q_{intA} &= \int_S 8 N_A \hat{\underset{\cdot\cdot}{\boldsymbol{f}}}^1 : (\hat{\underset{\cdot}{\boldsymbol{e}}}_3 \otimes \hat{\underset{\cdot}{\boldsymbol{e}}}_3) \mathrm{d}\xi \mathrm{d}\eta
\end{aligned} \tag{A.44}$$

where S represents the reference surface. The resultants $\hat{\underset{\cdot\cdot}{\boldsymbol{f}}}^0$ and $\hat{\underset{\cdot\cdot}{\boldsymbol{f}}}^1$ are obtained by integrating along the fiber as

$$\hat{\underset{\cdot\cdot}{\boldsymbol{f}}}^0 = \int_{-1}^{1} \hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}} J \mathrm{d}\zeta, \qquad \hat{\underset{\cdot\cdot}{\boldsymbol{f}}}^1 = \int_{-1}^{1} \frac{\zeta}{2} \hat{\underset{\cdot\cdot}{\boldsymbol{\sigma}}} J \mathrm{d}\zeta \tag{A.45}$$

where the small terms in the Jacobian determinant J involving q or its derivatives may be neglected. It should be further emphasized that all "hatted" quantities ($\hat{\bullet}$) are meant to be evaluated in the local coordinate system, e.g., $\hat{\underset{\cdot}{\boldsymbol{e}}}_3 = [0\ 0\ 1]^T$.

A.3.5 External force calculation

The discretized external force contributions to the virtual power can be derived knowing that the virtual velocity $\delta \underset{\cdot}{\boldsymbol{v}}$ is given as

$$\delta \underset{\cdot}{\boldsymbol{v}} = \delta \dot{\underset{\cdot}{\boldsymbol{r}}} + \frac{\zeta}{2} \delta \dot{\underset{\cdot}{\boldsymbol{d}}} + \zeta^2 (\delta \dot{q} \underset{\cdot}{\boldsymbol{d}} + q \delta \dot{\underset{\cdot}{\boldsymbol{d}}}) \tag{A.46}$$

Now, the external body and traction force contributions to the virtual power are given by

$$\delta P^b_{ext} = \int_\Omega \underset{\cdot}{\boldsymbol{b}} \cdot \delta \underset{\cdot}{\boldsymbol{v}} \, \mathrm{d}\Omega, \qquad \delta P^t_{ext} = \int_\Gamma \underset{\cdot}{\boldsymbol{t}} \cdot \delta \underset{\cdot}{\boldsymbol{v}} \, \mathrm{d}\Gamma \tag{A.47}$$

After discretizing and substituting the virtual velocity $\delta \underset{\cdot}{\boldsymbol{v}}$ from (A.46) into (A.47), one obtains

$$\delta P^b_{ext} = \sum_A \Big(\delta \dot{\underset{\cdot}{\boldsymbol{r}}}_A \cdot \underbrace{\int_\Omega N_A \underset{\cdot}{\boldsymbol{b}} \, \mathrm{d}\Omega}_{\underset{\cdot}{\boldsymbol{f}}^r_{extA}} + \delta \dot{\underset{\cdot}{\boldsymbol{d}}}_A \cdot \underbrace{\int_\Omega (\frac{\zeta}{2} + \zeta^2 q_A) N_A \underset{\cdot}{\boldsymbol{b}} \, \mathrm{d}\Omega}_{\underset{\cdot}{\boldsymbol{f}}^d_{extA}} + \delta \dot{q}_A \underbrace{\int_\Omega \zeta^2 N_A \underset{\cdot}{\boldsymbol{d}}_A \cdot \underset{\cdot}{\boldsymbol{b}} \, \mathrm{d}\Omega}_{f^q_{extA}} \Big) \tag{A.48}$$

for the body force. The external traction force contribution is given similarly, but the contribution is usually applied only to the top, $\zeta = 1$, or bottom surface, $\zeta = -1$. Furthermore, the term involving q_A in (A.48) is small in most cases and may be neglected.

Additionally, point and line loads are considered as external traction loads applied over an infinitesimal area. For example, the external traction force contribution to the weak form (A.47) now takes the following form for a point load applied to $\xi = \xi^{pl}$, $\eta = \eta^{pl}$, $\zeta = \zeta^{pl} = 0$:

$$\delta P^t_{ext} = \sum_i \sum_j \delta \dot{\boldsymbol{r}}_{ij} \cdot \underbrace{\lim_{\substack{\Delta\xi\to 0\\ \Delta\eta\to 0}} \int_{\eta^{pl}-\Delta\eta}^{\eta^{pl}+\Delta\eta} \int_{\xi^{pl}-\Delta\xi}^{\xi^{pl}+\Delta\xi} N_i^{\xi}(\xi) N_j^{\eta}(\eta) \dot{\boldsymbol{t}}^{pl} J_S \mathrm{d}\xi \mathrm{d}\eta}_{\dot{\boldsymbol{f}}^{pl}_{ij}} \tag{A.49}$$

In (A.49), the multivariate shape function has been decoupled into its univariate spline basis functions (see (A.3)), i and j are control point indices associated with the parametric directions ξ and η, respectively, and J_S is the Jacobian determinant. As the integral is only evaluated over an infinitesimal area, the basis functions are considered to be constant within the integrated area and can thus be brought outside the integral expression. The control point forces due to the point load can be now written as

$$\dot{\boldsymbol{f}}^{pl}_{ij} = N_i^{\xi}(\xi^{pl}) N_j^{\eta}(\eta^{pl}) \underbrace{\lim_{\substack{\Delta\xi\to 0\\ \Delta\eta\to 0}} \int_{\eta^{pl}-\Delta\eta}^{\eta^{pl}+\Delta\eta} \int_{\xi^{pl}-\Delta\xi}^{\xi^{pl}+\Delta\xi} \dot{\boldsymbol{t}}^{pl} J_S \mathrm{d}\xi \mathrm{d}\eta}_{\dot{\boldsymbol{f}}^{pl}(\xi^{pl},\eta^{pl},\zeta^{pl})} \tag{A.50}$$

where the integral expression is equal to the known point load $\dot{\boldsymbol{f}}^{pl}(\xi^{pl}, \eta^{pl}, \zeta^{pl})$.

Line loads are treated in a similar way. In the case of a line load along $\xi = \xi^{ll}$, $\zeta = \zeta^{ll} = 0$

$$\dot{\boldsymbol{f}}^{ll}_{ij} = N_i^{\xi}(\xi^{ll}) \int_{\eta} N_j^{\eta}(\eta) \underbrace{\lim_{\Delta\xi\to 0} \int_{\xi^{ll}-\Delta\xi}^{\xi^{ll}+\Delta\xi} \dot{\boldsymbol{t}}^{ll} J_S \mathrm{d}\xi}_{\bar{\dot{\boldsymbol{f}}}^{ll}(\xi^{ll},\eta,\ \zeta^{ll})} \mathrm{d}\eta \tag{A.51}$$

where $\bar{\dot{\boldsymbol{f}}}^{ll}(\xi^{ll}, \eta,\ \zeta^{ll})$ represents the line load in the parametric space and can be obtained from the line load $\dot{\boldsymbol{f}}^{ll}(\xi^{ll}, \eta,\ \zeta^{ll})$ given in the Euclidean space by

$$\bar{\dot{\boldsymbol{f}}}^{ll}(\xi^{ll}, \eta,\ \zeta^{ll}) = \sqrt{x_{,\eta}^2 + y_{,\eta}^2 + z_{,\eta}^2}\ \Big|_{\substack{\xi=\xi^{ll}\\ \zeta=\zeta^{ll}}} \dot{\boldsymbol{f}}^{ll}(\xi^{ll}, \eta,\ \zeta^{ll}) \tag{A.52}$$

A.3.6 Inertial force calculation

Differentiating the velocity given by (A.17) with respect to time, one ends up with the following expression for the acceleration:

$$\underset{\cdot}{\boldsymbol{a}} = \ddot{\underset{\cdot}{\boldsymbol{r}}} + \frac{\zeta}{2}\ddot{\underset{\cdot}{\boldsymbol{d}}} + \zeta^2(\ddot{q}\underset{\cdot}{\boldsymbol{d}} + 2\dot{q}\dot{\underset{\cdot}{\boldsymbol{d}}} + q\ddot{\underset{\cdot}{\boldsymbol{d}}}) \tag{A.53}$$

The inertial force contribution to the virtual power is given by

$$\delta P_{iner} = \int_\Omega \rho \underset{\cdot}{\boldsymbol{a}} \cdot \delta \underset{\cdot}{\boldsymbol{v}} \, \mathrm{d}\Omega \tag{A.54}$$

and the discretized form as

$$\delta P_{iner} = \sum_{A,B} \int_\Omega \rho N_A \delta \underset{\cdot}{\boldsymbol{v}}_A \cdot N_B \underset{\cdot}{\boldsymbol{a}}_B \, \mathrm{d}\Omega \tag{A.55}$$

In the context of explicit analyses, a lumped mass matrix is preferred for the sake of computational efficiency. The mass lumping is accomplished by replacing the shape function associated with the acceleration by the Kronecker delta. This corresponds to the row sum method in the context of solid elements with three degrees of freedom per control point. Now, the expression can be written as

$$\delta P_{iner} \approx \sum_{A,B} \int_\Omega \rho N_A \delta \underset{\cdot}{\boldsymbol{v}}_A \cdot \delta_{AB} \underset{\cdot}{\boldsymbol{a}}_B \, \mathrm{d}\Omega \tag{A.56}$$

Substituting the acceleration $\underset{\cdot}{\boldsymbol{a}}_B$ from the discretized (A.53) into (A.56):

$$\delta P_{iner} \approx \sum_{A} \int_\Omega \rho N_A \delta \underset{\cdot}{\boldsymbol{v}}_A \cdot \left(\ddot{\underset{\cdot}{\boldsymbol{r}}}_A + (\frac{\zeta}{2} + \zeta^2 q_A)\ddot{\underset{\cdot}{\boldsymbol{d}}}_A + \zeta^2 \underset{\cdot}{\boldsymbol{d}}_A \ddot{q}_A + 2\zeta^2 \dot{q}_A \dot{\underset{\cdot}{\boldsymbol{d}}}_A \right) \mathrm{d}\Omega \tag{A.57}$$

Substituting the virtual velocity $\delta \underset{\cdot}{\boldsymbol{v}}_A$ from the discretized (A.46) into (A.57), the total expression for the virtual power including internal and external forces as well as optional damping is given by

$$\begin{aligned}
\sum_A \delta \dot{\underset{\cdot}{\boldsymbol{r}}}_A \cdot \left(M_A^{rr} \ddot{\underset{\cdot}{\boldsymbol{r}}}_A + M_A^{rd} \ddot{\underset{\cdot}{\boldsymbol{d}}}_A + \underset{\cdot}{\boldsymbol{M}}_A^{rq} \ddot{q}_A \right) &= \sum_A \delta \dot{\underset{\cdot}{\boldsymbol{r}}}_A \cdot \left(\underset{\cdot}{\boldsymbol{f}}_{extA}^r - \underset{\cdot}{\boldsymbol{f}}_{intA}^r - \underset{\cdot}{\boldsymbol{f}}_{dampA}^r - \underset{\cdot}{\boldsymbol{f}}_{vA}^r \right) \\
\sum_A \delta \dot{\underset{\cdot}{\boldsymbol{d}}}_A \cdot \left(M_A^{rd} \ddot{\underset{\cdot}{\boldsymbol{r}}}_A + M_A^{dd} \ddot{\underset{\cdot}{\boldsymbol{d}}}_A + \underset{\cdot}{\boldsymbol{M}}_A^{dq} \ddot{q}_A \right) &= \sum_A \delta \dot{\underset{\cdot}{\boldsymbol{d}}}_A \cdot \left(\underset{\cdot}{\boldsymbol{f}}_{extA}^d - \underset{\cdot}{\boldsymbol{f}}_{intA}^d - \underset{\cdot}{\boldsymbol{f}}_{dampA}^d - \underset{\cdot}{\boldsymbol{f}}_{vA}^d \right) \\
\sum_A \delta \dot{q}_A \left(\underset{\cdot}{\boldsymbol{M}}_A^{rq} \cdot \ddot{\underset{\cdot}{\boldsymbol{r}}}_A + \underset{\cdot}{\boldsymbol{M}}_A^{dq} \cdot \ddot{\underset{\cdot}{\boldsymbol{d}}}_A + M_A^{qq} \ddot{q}_A \right) &= \sum_A \delta \dot{q}_A \left(f_{extA}^q - f_{intA}^q - f_{dampA}^q - f_{vA}^q \right)
\end{aligned} \tag{A.58}$$

where the mass terms and the forces denoted by v in the subscript are given in Appendix A.A. Equation (A.58) forms a symmetric 7×7 system of equations in the

following form independently for each control point:

$$\begin{bmatrix} M^{rr} & 0 & 0 & M^{rd} & 0 & 0 & M_x^{rq} \\ 0 & M^{rr} & 0 & 0 & M^{rd} & 0 & M_y^{rq} \\ 0 & 0 & M^{rr} & 0 & 0 & M^{rd} & M_z^{rq} \\ M^{rd} & 0 & 0 & M^{dd} & 0 & 0 & M_x^{dq} \\ 0 & M^{rd} & 0 & 0 & M^{dd} & 0 & M_y^{dq} \\ 0 & 0 & M^{rd} & 0 & 0 & M^{dd} & M_z^{dq} \\ M_x^{rq} & M_y^{rq} & M_z^{rq} & M_x^{dq} & M_y^{dq} & M_z^{dq} & M^{qq} \end{bmatrix} \begin{bmatrix} \ddot{r}_x \\ \ddot{r}_y \\ \ddot{r}_z \\ \ddot{d}_x \\ \ddot{d}_y \\ \ddot{d}_z \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} F_{total}^{r_x} \\ F_{total}^{r_y} \\ F_{total}^{r_z} \\ F_{total}^{d_x} \\ F_{total}^{d_y} \\ F_{total}^{d_z} \\ F_{total}^{q} \end{bmatrix} \tag{A.59}$$

It is further noted that although a mass lumping procedure is applied in (A.56), a nondiagonal submatrix appears due to substitution steps (A.57) and (A.58) for each control point A. This submatrix couples the different degrees of freedom internally for each control point but does not introduce a coupling between adjacent control points.

A.3.7 Dirichlet boundary conditions

In the context of classical finite elements, Dirichlet boundary conditions are usually applied by directly specifying the nodal values. In IGA, the control points are generally not interpolatory and applying a boundary condition directly to a noninterpolatory control point is rarely physically justifiable. Many authors circumvent the problem by introducing repeated knots such that the boundary conditions can be directly applied to the control points that are now interpolatory. However, this approach lowers the patch continuity to C^0 which is not always desirable.

An efficient technique compatible with diagonal mass matrices is proposed for enforcing Dirichlet boundary conditions at arbitrary locations within a patch. This technique is directly applicable to the proposed shell formulation if the off-diagonal terms in each submatrix (A.59) are neglected. Alternatively, the Lagrange multiplier method is employed to solve the full inertia coupling. A boundary condition of a prescribed acceleration along $\xi = \xi^{bc}$, $\zeta = 0$ requires that

$$\sum_i \sum_j N_i^{\xi}(\xi^{bc}) N_j^{\eta}(\eta) \ddot{\mathbf{r}}_{ij} = \ddot{\mathbf{r}}^{prescribed}(\xi^{bc}, \eta) \tag{A.60}$$

where the multivariate shape function has been decoupled into its univariate spline basis functions similarly to (A.49). However, the boundary condition can be satisfied exactly only if the prescribed acceleration $\ddot{\mathbf{r}}^{prescribed}(\xi^{bc}, \eta)$ is representable for all η

by the underlying spline interpolation, i.e.,

$$\sum_j N_j^\eta(\eta)\ddot{\mathbf{r}}_j^{prescribed} = \ddot{\mathbf{r}}^{prescribed}(\xi^{bc},\eta) \tag{A.61}$$

where $\ddot{\mathbf{r}}_j^{prescribed}$ are the control values for the prescribed acceleration $\ddot{\mathbf{r}}^{prescribed}(\xi^{bc},\eta)$. The acceleration $\ddot{\mathbf{r}}_{ij}$ of (A.60) is decoupled into an uncorrected acceleration $\ddot{\mathbf{r}}_{ij}^{uc}$, i.e., the acceleration that is obtained without applying the boundary condition, and a change in the acceleration due to the boundary condition $\ddot{\mathbf{r}}_{ij}^{bc}$:

$$\sum_i \sum_j N_i^\xi(\xi^{bc}) N_j^\eta(\eta)(\ddot{\mathbf{r}}_{ij}^{uc} + \ddot{\mathbf{r}}_{ij}^{bc}) = \sum_k N_k^\eta(\eta)\ddot{\mathbf{r}}_k^{prescribed} \tag{A.62}$$

Now, it is sufficient to guarantee that

$$\sum_i N_i^\xi(\xi^{bc})(\ddot{\mathbf{r}}_{ij}^{uc} + \ddot{\mathbf{r}}_{ij}^{bc}) = \ddot{\mathbf{r}}_j^{prescribed} \tag{A.63}$$

for all j.

However, (A.63) is often underdetermined as there are generally multiple unknowns $\ddot{\mathbf{r}}_{ij}^{bc}$. Therefore, additional equations are required. Considering (A.51), the control point forces due to a line load that exactly satisfy the boundary condition are given by

$$\mathbf{f}_{ij}^{bc} = N_i^\xi(\xi^{bc}) \underbrace{\int_\eta N_j^\eta(\eta) \lim_{\Delta\xi\to 0} \int_{\xi^{bc}-\Delta\xi}^{\xi^{bc}+\Delta\xi} \mathbf{t}^{bc} J_S \mathrm{d}\xi \mathrm{d}\eta}_{\boldsymbol{\lambda}_j} \tag{A.64}$$

where $\mathbf{t}^{bc}$ is the unknown traction exerted by the boundary condition. From (A.64), the following relation is obtained:

$$\mathbf{f}_{ij}^{bc} = N_i^\xi(\xi^{bc}) \frac{\mathbf{f}_{kj}^{bc}}{N_k^\xi(\xi^{bc})} \tag{A.65}$$

Considering Newton's second law $\mathbf{f}_{ij}^{bc} = M_{ij}^{rr}\ddot{\mathbf{r}}_{ij}^{bc}$, i.e.,

$$M_{ij}^{rr}\ddot{\mathbf{r}}_{ij}^{bc} = N_i^\xi(\xi^{bc}) \frac{M_{kj}^{rr}\ddot{\mathbf{r}}_{kj}^{bc}}{N_k^\xi(\xi^{bc})} \tag{A.66}$$

and thus

$$\ddot{\mathbf{r}}_{ij}^{bc} = N_i^\xi(\xi^{bc}) \frac{M_{kj}^{rr}\ddot{\mathbf{r}}_{kj}^{bc}}{N_k^\xi(\xi^{bc}) M_{ij}^{rr}} \tag{A.67}$$

where M_{ij}^{rr} represents the mass of the control point ij which is obtained from a diagonal mass matrix. Substituting (A.67) into (A.63) gives the change in the acceleration due to the boundary condition:

$$\ddot{\boldsymbol{r}}_{kj}^{bc} = \frac{N_k^{\xi}(\xi^{bc})\left(\ddot{\boldsymbol{r}}_j^{prescribed} - \sum_l N_l^{\xi}(\xi^{bc})\ddot{\boldsymbol{r}}_{lj}^{uc}\right)}{\sum_i \frac{N_i^{\xi}(\xi^{bc})^2}{M_{ij}^{rr}} M_{kj}^{rr}} \tag{A.68}$$

The control point accelerations that exactly satisfy the boundary condition are now given by

$$\ddot{\boldsymbol{r}}_{ij} = \ddot{\boldsymbol{r}}_{ij}^{uc} + \ddot{\boldsymbol{r}}_{ij}^{bc} \tag{A.69}$$

The unknown control point forces $\boldsymbol{f}_{ij}^{bc}$ are only used to form the relation (A.67) between the accelerations $\ddot{\boldsymbol{r}}_{ij}^{bc}$ of the neighboring control points, and need not be explicitly calculated for the purposes of evaluating the boundary condition. Furthermore, the unknowns $\boldsymbol{\lambda}_j$ can be understood as Lagrange multipliers that exactly satisfy the boundary condition.

It is further noted that if the boundary condition is enforced along a parametric coordinate that is simultaneously a knot location, the summations in (A.68) are evaluated only over p control points, where p is the order of the shape function. Therefore, for a typical edge constraint with unclamped knot vectors, only two control points need be considered for each j assuming biquadratic interpolation. However, if two edge boundary conditions intersect at a patch corner, then a coupled problem of four control points must be solved at that location.

A.3.8 Fiber mass scaling and time step estimate

For the underlying mechanical problem, the maximum stable time step size for the explicit central difference time integration scheme is given by (Adam et al., 2015b)

$$\Delta t_{max} = \frac{2}{\omega_{max}} \tag{A.70}$$

where ω_{max} is the maximum natural frequency of the system.

To minimize the computational cost, the maximum stable time step calculation is based on the element time step estimate. This commonly used estimate assumes that the maximum natural frequency of the system is bounded from above by the maximum natural frequency among individual Bézier elements, i.e.,

$$\omega_{max} \leq \max\{\omega_{max}^e\}_{e=1:n} \tag{A.71}$$

where n is the number of Bézier elements. Now, the estimate based on the element characteristic length is given as

$$\Delta t_{max} = \frac{1}{c} \min\{\alpha_e l_e\}_{e=1:n} \tag{A.72}$$

where α_e and l_e are the time step scaling coefficient (Adam et al., 2015b) and the characteristic length of the eth Bézier element, respectively. The dilatational wave speed in the material is calculated as

$$c = \sqrt{\frac{E(1-\nu)}{(1+\nu)(1-2\nu)\rho}} \tag{A.73}$$

In contrast to the conventional Lagrange elements, raising the order of the spline-based interpolation functions generally increases the stable time step size as the order of continuity between the elements is elevated. Therefore, repeated knots typically reduce the stable time step, e.g., if clamped knot vectors are used (which consist of repeated knots in both ends), the boundary elements penalize the time step. The stable time step estimates in isogeometric analysis are discussed in more detail by Adam et al. (2015b) as well as Hartmann and Benson (2014).

Solid and solid-shell elements are known to suffer from ill-conditioned systems of equations in the thin limit (Gee et al., 2005; Klöppel et al., 2011). This is a result of an increasing gap between the highest and lowest eigenfrequencies of the system when the thickness decreases. The highest eigenfrequencies typically relate to the transverse shear and thickness deformations whereas the lowest eigenfrequencies are associated with the bending and in-plane deformations. Unfortunately, as pointed out by Simo et al. (1990), the shell formulations adopting only displacement-based degrees of freedom and coupling the thickness with the director vector are prone to the same ill-conditioning problem. One possible way to circumvent the problem is to decouple the thickness from the shell fiber and introduce rotational degrees of freedom. This approach was adopted by El-Abbasi and Meguid (2000) as well as Cardoso and Yoon (2005), but the increased complexity would increase the computational cost of the proposed formulation.

In this paper, the highest eigenfrequencies are lowered by scaling the fiber masses M^{dd} and M^{qq}. This concept of fiber mass scaling is similar to the rotational mass scaling often adopted by many explicit Reissner-Mindlin shell formulations as it prevents the transversal deformations from reducing the maximum stable time step size. An intriguing way of determining the adequate fiber mass scaling is to scale

the fiber masses such that the highest eigenfrequency is associated with the in-plane deformations and therefore the larger stable time step size provided by the spline-based in-plane interpolation can be exploited. The influence of fiber mass scaling is investigated in Section A.4.

It is further noted that quasi-static solvers where the inertia term is neglected, the ill-conditioning of the stiffness matrix can be efficiently circumvented by the scaled director conditioning (SDC) or the scaled thickness conditioning (STC) procedures as proposed by Gee et al. (2005) and Klöppel et al. (2011), respectively.

A.4 Numerical examples

In this section, the performance of the proposed shell formulation is evaluated using biquadratic b-spline interpolation functions and a hypoelastic or hypoelastic-plastic material assumption. The benchmark problems consist of two linear elastic, and five nonlinear elastic or elasto-plastic examples: Sections A.4.1 and A.4.2 consider static linear elasticity, Sections A.4.3–A.4.5 quasi-static geometrically nonlinear elasticity, and Sections A.4.6 and A.4.7 explicit dynamic elasto-plasticity.

Each Bézier element is integrated with 2×2 reduced Gauss-Legendre quadrature rule in-plane and 9-point Simpson's rule through-thickness. In comparison with full integration, the in-plane 2×2 reduced Gauss-Legendre quadrature lowers the computational overhead and alleviates several locking phenomena while still preserving full rank of the stiffness matrix and providing good accuracy (Schillinger et al., 2014). The formulation is, however, not restricted to the chosen interpolation functions and quadrature rules, and higher order spline interpolation and the selection of the in-plane quadrature are currently under investigation (Hokkanen and Pedroso, 2019b). The central difference scheme is used for the time integration.

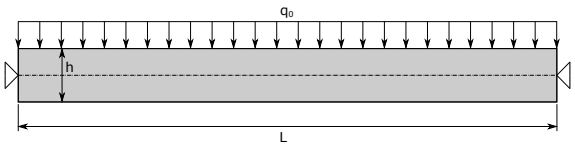
Benchmark problems considered in Sections A.4.1–A.4.3 are evaluated using clamped and unclamped uniform knot vectors. Rough estimates for the maximum stable time step scaling coefficients are presented based on numerical experimentation where $\alpha_e = \alpha$ for each e. Three different techniques to form the space discretization are investigated: clamped knot vectors with uniform control grid (Type C1), clamped knot vectors with uniform Bézier element size (Type C2), and unclamped knot vectors with uniform control grid (Type U). The Type C1 discretization leads to larger Bézier element sizes at the boundaries, whereas the Type C2 case leads to smaller

control point spacings at the boundaries. The Type U discretization employing unclamped knot vectors considers a uniform control grid which simultaneously leads to a uniform Bézier element size.

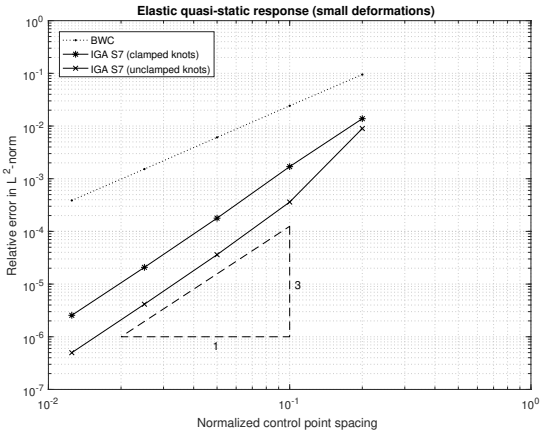
Different types of boundary conditions are treated as follows. A typical simply supported boundary condition is employed by constraining the degrees of freedom associated with the velocity of the reference surface. However, the analytical solution in Section A.4.2 assumes no shear deformations along the simply supported edges (also known as hard support). Therefore, the degrees of freedom representing the velocity component of the fiber vector in the direction parallel to the simply supported edges are also constrained in this case. The fixed boundary conditions encountered in Sections A.4.2–A.4.4 and A.4.7 are enforced by constraining all degrees of freedom for the corresponding edges.

A.4.1 Timoshenko beam (SS)

In this example, the linear elastic response of a simply supported Timoshenko beam under a uniformly distributed load of magnitude q_0 is assessed (see Figure A.1a). The analytical expression for the nondimensionalized deflection $\bar{w}$ is given in Appendix A.B. The beam of slenderness ratio $L/h = 10$ and Poisson's ratio $\nu = 0$ is considered, where L and h represent the length and the thickness of the beam, respectively.



(a) The problem setup.



(b) The relative error in the L^2-norm of the nondimensionalized deflection.

Figure A.1: Timoshenko beam under a uniformly distributed load.

The relative error in the L^2-norm of the nondimensionalized deflection $\bar{w}$ given by

$$\|e_{\bar{w}}\|_{L_2} = \sqrt{\frac{\int_\Omega (\bar{w}_{exact} - \bar{w}_{iga})^2 \, \mathrm{d}\Omega}{\int_\Omega \bar{w}^2_{exact} \, \mathrm{d}\Omega}} \tag{A.74}$$

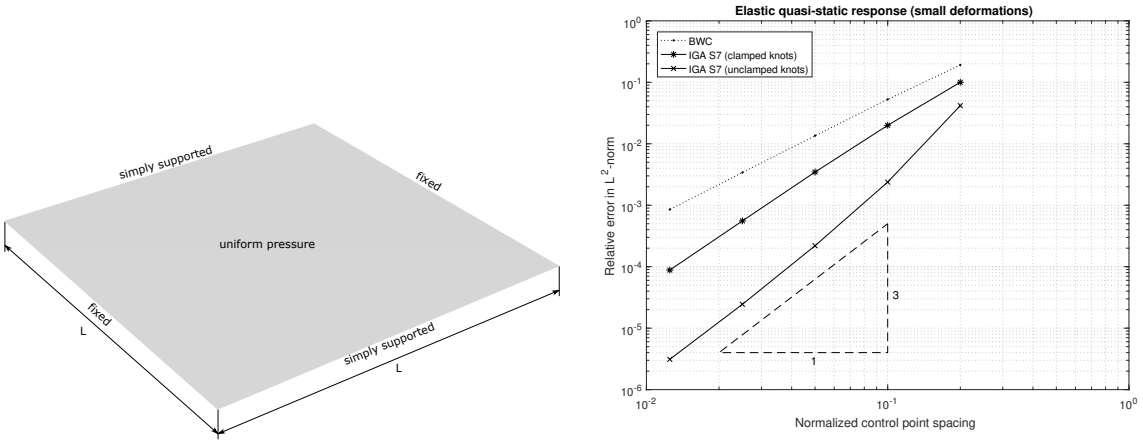
is plotted against the averaged normalized control point spacing in Figure A.1b for Type C1, Type C2, and Type U discretizations. Significantly worse accuracy is observed for the Type C1 discretization in comparison with the Type C2 or Type U discretizations for which identical results are obtained. This is due to the larger boundary elements resulting from the clamped knot vectors and uniform control grid being less accurate than the interior elements. In the Type C2 and Type U cases, the boundary element size does not differ from the interior elements and better accuracy is preserved. The rate of convergence, however, approaches a similar value when the mesh is refined. The convergence of the Belytschko-Wong-Chiang element (Belytschko et al., 1992) employing bilinear interpolation functions is given as a reference.

However, the maximum stable time step size for the Type C2 discretization is significantly smaller in comparison with the Type C1 or Type U discretizations. The Type C2 case remains stable up to the time step scaling coefficient of $\alpha^{C2} \approx 0.8$, whereas the Type C1 and Type U cases remain stable up to $\alpha^{C1} \approx \alpha^{U} \approx 1.3$. Therefore, the discretizations adopting clamped knot vectors suffer from either lower accuracy or lower stable time step size associated with the boundary elements in comparison with the Type U discretization adopting unclamped knot vectors. The maximum stable time step scaling coefficient for the classical Belytschko-Wong-Chiang element is $\alpha \approx 1$.

A.4.2 Mindlin plate (SCSC)

This example considers the linear elastic response of a Mindlin plate under a uniformly distributed load. The boundary conditions are imposed such that two opposite edges are simply supported (hard support) and two opposite edges are fixed as shown in Figure A.2a. The analytical expression for the nondimensionalized deflection $\bar{w}$ is given in Appendix A.C. The square plate of slenderness ratio $L/h = 10$ and Poisson's ratio $\nu = 0.3$ is considered, where L and h represent the edge length and the thickness of the plate, respectively.

The relative error in the L^2-norm of the nondimensionalized deflection $\bar{w}$ is plotted against the averaged normalized control point spacing in Figure A.2b. Again, Type



(a) The problem setup.

(b) The relative error in the L^2-norm of the nondimensionalized deflection.

Figure A.2: Mindlin plate under a uniformly distributed load.

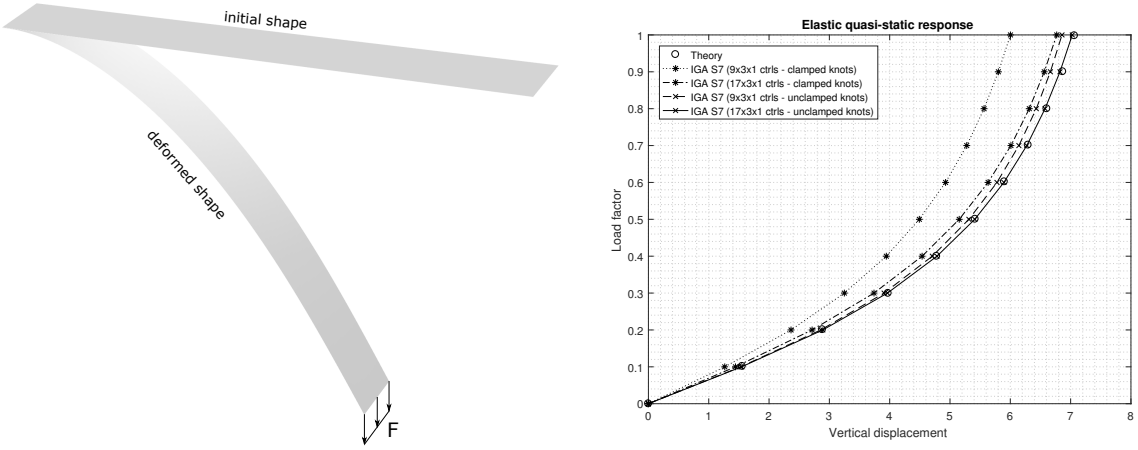
C1, Type C2, and Type U discretizations are considered. The latter two still produce identical results by computer precision. However, now an even bigger difference in results is observed between the Type C1 and the other two discretizations in comparison to the beam problem (Figure A.1b). The rate of convergence still approaches a similar value when the mesh is refined, but slower.

The maximum stable time step size for the Type C2 discretization is again significantly smaller in comparison with the Type C1 or Type U discretizations. However, the Type C2 case now remains stable up to $\alpha^{C2} \approx 1.3$, whereas the Type C1 and Type U cases remain stable up to $\alpha^{C1} \approx \alpha^{U} \approx 1.6$. The larger time step scaling coefficients in comparison to the beam problem of Section A.4.1 are explained by more strictly constrained boundaries. Due to sufficient fiber mass scaling, the degrees of freedom that determine the stable time step size represent the velocity of the reference surface. In the current example, these degrees of freedom are constrained throughout the whole boundary. This prevents instabilities originating from the boundary control points which are associated with smaller masses than the interior control points. The connection between the boundary conditions and the maximum stable time step size was also observed by Hokkanen et al. (2018) in a metal forming simulation.

A.4.3 Straight cantilever plate strip

In this example, the elastic straight cantilever plate strip shown in Figure A.3a is considered. The plate strip has the length $L = 10$, the width $w = 1.0$ and the thickness $h = 0.1$. Young's modulus and Poisson's ratio are $E = 10^7$ and $\nu = 0.3$,

respectively. The applied line load is given as $F = 40\lambda$, where λ is the load factor ranging from 0.0 to 1.0.



(a) The problem setup.

(b) Load factor versus the vertical tip displacement.

Figure A.3: Cantilever plate strip under a tip line load.

The maximum vertical tip displacement is plotted against the load factor λ in Figure A.3b. The performance of the element is evaluated for Type C1, Type C2, and Type U discretizations using 9 and 17 control points along the length of the plate strip. The results obtained from the Type C1 case and the coarser mesh ($9 \times 3 \times 1$ control points) underestimate the maximum deflection by almost 15% due to locking. However, the Type C2 and Type U cases show significantly better results in comparison with the theoretical solution (Frisch-Fay, 1962)—even with the coarse mesh. The difference between the solid line and the fully converged solution is less than 0.5% for any given load factor.

In the case of the coarse mesh, the maximum stable time step scaling coefficients for Type C1, Type C2, and Type U discretizations are now $\alpha^{C1} \approx 0.7$, $\alpha^{C2} \approx 0.7$, and $\alpha^{U} \approx 1.1$, respectively. For the fine mesh, the respective coefficients are $\alpha^{C1} \approx 1.1$, $\alpha^{C2} \approx 0.7$, and $\alpha^{U} \approx 1.2$.

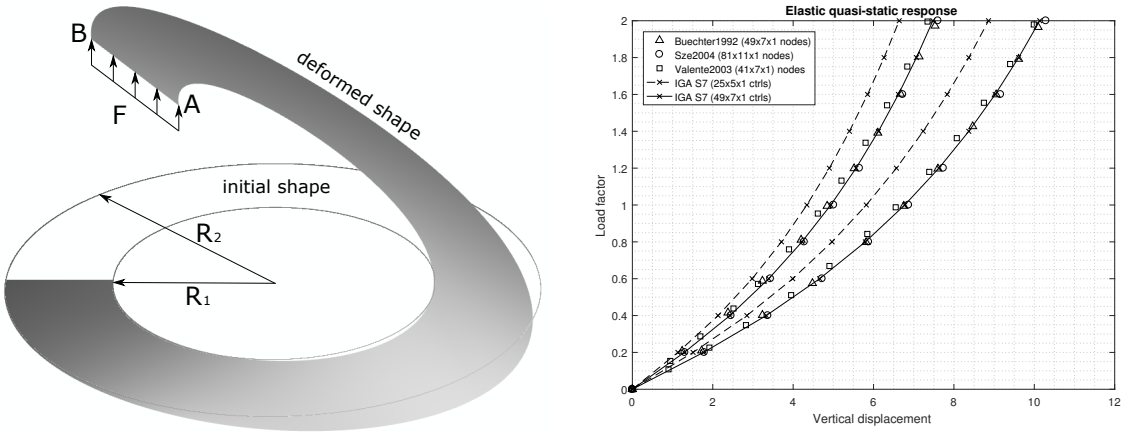
The different time step scaling coefficient values for different mesh sizes are explained by the fact that only one element along the width of the cantilever plate strip is used. For the coarse mesh, the dimension along the width of the plate constrains the time step. The discretization along this dimension is identical for the Type C1 and Type C2 cases which explains their identical time step scaling coefficients. For the fine mesh, it is the dimension associated with the length of the plate that constrains

the time step.

These observations further demonstrate the importance of using unclamped knot vectors to take full advantage of isogeometric analysis, at least, in the context of explicit analyses.

A.4.4 Cantilever ring plate

The thin cantilever ring plate shown in Figure A.4a subjected to a vertical line load is considered. According to Cardoso and Yoon (2005), this example is one of the most sensitive tests for finite rotations and element warping. The geometric data is characterized by the inner radius $R_1 = 6.0$, the outer radius $R_2 = 10.0$, and the thickness $h = 0.03$. Young's modulus and Poisson's ratio are given by $E = 2.1 \times 10^{10}$ and $\nu = 0.0$, respectively. The applied line load is given as $F = 100\lambda$, where the load factor λ ranges from 0.0 to 2.0.



(a) The problem setup.

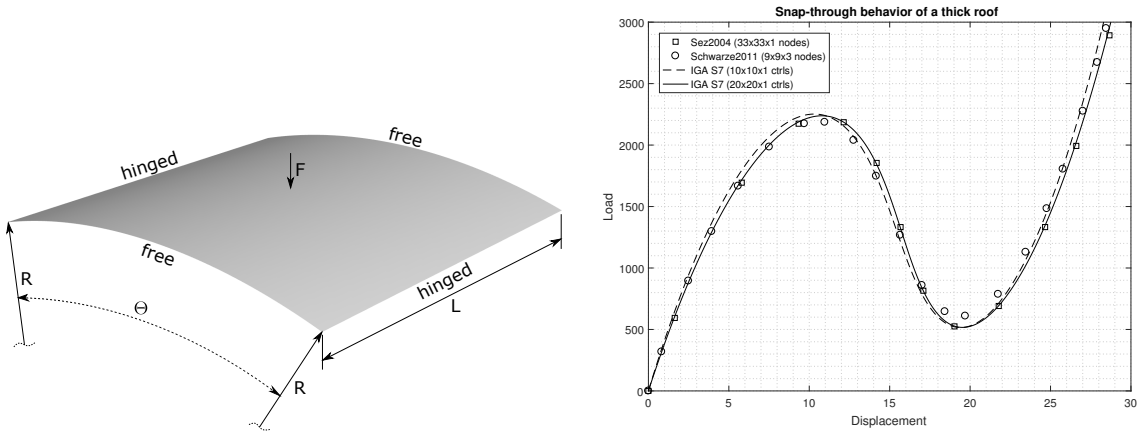
(b) The load factor versus the displacements of points A and B.

Figure A.4: Cantilever ring plate under a tip line load.

Two mesh configurations are considered with unclamped knot vectors (Type U discretization) only; the coarser mesh with $25 \times 5 \times 1$ control points and the finer mesh with $49 \times 7 \times 1$ control points. The vertical displacements of points A and B are compared with the reference results in Figure A.4b. The coarser mesh case clearly suffers from locking as this example represents a moderately thin problem. However, the results obtained with the finer mesh are in good accordance with the reference solutions (Buechter and Ramm, 1992; Sze et al., 2004; Valente et al., 2003).

A.4.5 Snap-through of a thick roof

In this example, a snap-through behavior of a thick cylindrical roof structure is investigated. The geometric data is given by the edge length $L = 508$, the angle $\Theta = 0.2$, the radius $R = 2540$, and the thickness $h = 12.7$. Young's modulus and Poisson's ratio are set to $E = 3102.75$ and $\nu = 0.3$, respectively. The two opposite straight edges of the cylindrical shell are simply supported and the two other opposite edges are free. A concentrated vertical load F is applied in the mid-point of the roof as shown in Figure A.5a.



(a) The problem setup.

(b) The load factor versus the displacement of the center point.

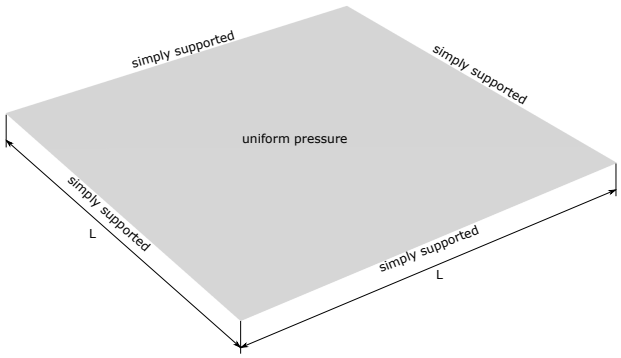
Figure A.5: Snap-through behavior of a cylindrical roof structure under a point load.

The analysis is carried out by prescribing the displacement in the explicit solver and moving slowly (i.e., keeping the inertial forces small). No external damping is applied. The full model is analyzed and the symmetry of the problem is not exploited. The load-displacement plot is shown in Figure A.5b, where the given number of control points or nodes has been scaled for the full geometry mesh, even if only a quarter of the problem was originally analyzed. The two mesh configurations run with the Type U discretization are in very good agreement with the reference solutions (Sze et al., 2004; Schwarze and Reese, 2011), particularly with that of Sze et al. (2004), who used a very fine mesh.

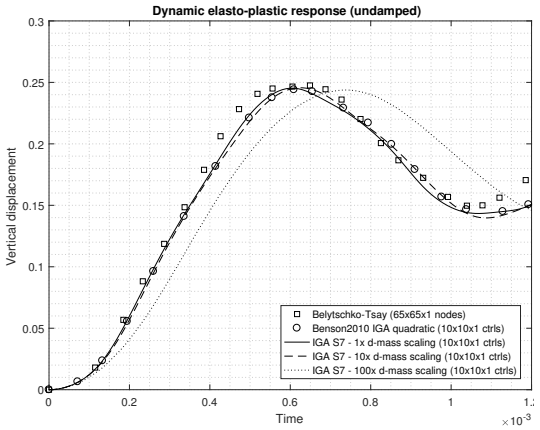
A.4.6 Dynamic elasto-plastic response of a square plate

This example tests the undamped dynamic elasto-plastic behavior of a square plate under an impulsively applied pressure load. The plate dimensions are given by the

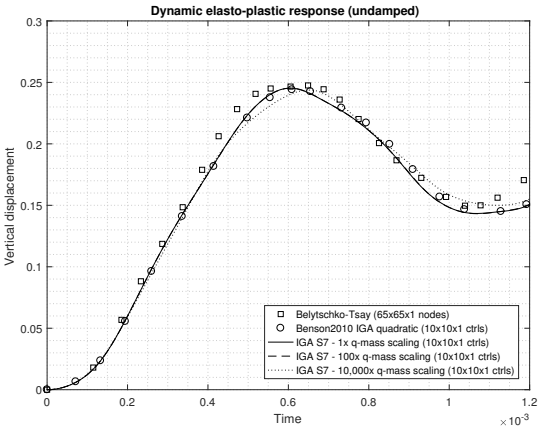
edge length $L = 10$ and the thickness $h = 0.5$. The properties of the elastic-perfectly-plastic material are given by the density $\rho = 2.588 \times 10^{-4}$, Young's modulus $E = 10^7$, Poisson's ratio $\nu = 0.3$, and the yield stress $\sigma_y = 3 \times 10^4$. The plate is simply supported and a uniform pressure $p = 300$ is applied to the reference surface (Figure A.6a).



(a) The problem setup.



(b) The vertical displacement of the center point versus time and the influence of d-mass scaling.



(c) The vertical displacement of the center point versus time and the influence of q-mass scaling.

Figure A.6: Dynamic elasto-plastic response of a square plate under a uniform pressure.

The dynamic response obtained by the Type U discretization matches well with the biquadratic isogeometric Reissner-Mindlin shell element proposed by Benson et al. (2010) as shown in Figures A.6b and A.6c. The response of the Belytschko-Tsay element (Belytschko et al., 1984) employing a very fine mesh is given as a reference as well. To prevent the degrees of freedom associated with the fiber vector from reducing the stable time step size, one may need to scale the corresponding masses. The influence of scaling the mass M^{dd} is evaluated for the scaling factors of 10 and 100 as shown in Figure A.6b. Scaling the mass M^{dd} by 10 does not significantly alter

the response of the plate. However, scaling this mass by 100 already causes notable deviation in the dynamic response.

The quadratic term contributing to the geometry poses the strictest requirement for the time step size if no mass scaling is applied. Nevertheless, this requirement can be circumvented by scaling the mass M^{qq}. The influence of M^{qq} scaling is given in Figure A.6c for the scaling factors of 100 and 10,000. In contrast to M^{dd} scaling, applying the scaling factor of 100 to M^{qq} makes practically no difference to the results, and, in fact, the scaling factor can be increased up to 10,000 while still leading to a good result. This implies that M^{qq} can easily be scaled such that q does not penalize the time step. However, it should be noted that further increasing M^{qq} slowly worsens the results, which demonstrates the importance of the corresponding degree of freedom.

A.4.7 Dynamic elasto-plastic response of a straight cantilever plate strip

The last example evaluates the damped dynamic elasto-plastic behavior of a cantilever plate strip. The geometric dimensions of the plate strip are given by the length $L = 6000$, the width $w = 200$, and the thickness $h = 100$. The properties of the elastic-perfectly-plastic material are given by the density $\rho = 7.5 \times 10^{-9}$, Young's modulus $E = 2 \times 10^5$, Poisson's ratio $\nu = 0.3$, and the yield stress $\sigma_y = 250$. The setup is similar to Figure A.3a; the other end of the plate strip is fixed and the line load $F = 80$ is applied to the reference surface of the other end. The diagonal damping matrix is obtained by scaling the mass matrix by the damping coefficient $\tilde{\gamma} = 10$, e.g., the damping term D_A^{dd} is given as $D_A^{dd} = \tilde{\gamma} M_A^{dd}$.

Again, the Type U discretization is employed. The damped dynamic response agrees well with the solid-shell reference element proposed by Pagani et al. (2014) and is given in Figure A.7. Additionally, the figure shows the effect of scaling the mass M^{dd} associated with the fiber vector by 100 and 10,000. In contrast to the previous example (cf. Figure A.6b), scaling M^{dd} by 100 produces an almost indistinguishable curve with respect to the nonscaled reference case. However, a significantly different response is obtained by applying the mass scaling factor of 10,000 for which a large error even in the quasi-static final state is obtained. Smaller sensitivity to the M^{dd} mass scaling compared to Figure A.6b is explained by the slower dynamic response of the damped problem.

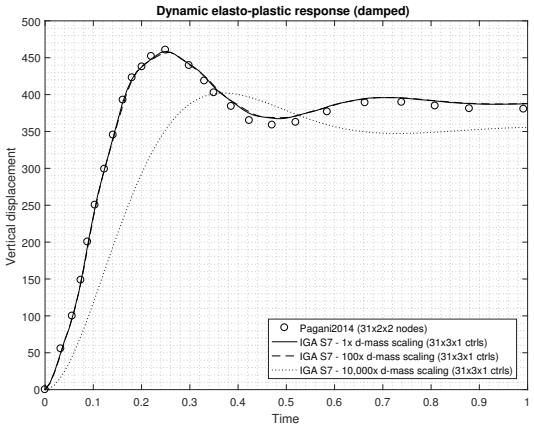


Figure A.7: Dynamic elasto-plastic response of a straight cantilever plate strip under a tip line load.

A.5 Conclusions

An isogeometric thick shell formulation incorporating thickness stretch was developed. The shell element is derived from the degenerated solid approach and results in seven degrees of freedom per control point. Various benchmark tests including geometric and material nonlinearities, as well as dynamic response, were evaluated to demonstrate the performance of the proposed formulation. It is also noted that the proposed formulation has been verified to produce a constant transversal normal stress state of unit magnitude when the bottom and the top surfaces of the shell are subjected to a positive uniform pressure of unit magnitude, although a separate benchmark problem was not constructed for this trivial verification. The formulation has also been applied to a two-point incremental forming problem requiring treatment of double-sided contact and anisotropic material by Hokkanen et al. (2018), wherein both the thinning and final geometry of the formed part were found to match well with the experimental results.

Several locking phenomena are alleviated by biquadratic spline-based interpolation, reduced integration and quadratic through thickness displacement field. However, the formulation is not fully locking-free, and a small amount of locking is still present, although this diminishes when the mesh is refined. It is further observed that unclamped knot vectors offer advantages over clamped knot vectors in that they provide either better accuracy or a larger stable time step size for the boundary elements. The accuracy of the boundary elements usually plays a big role especially in the academic benchmark cases where coarse meshes are often used to evaluate the element performance. However, this may not always be the case in the industrial

applications, particularly, if a fine mesh is used or the boundary elements do not significantly contribute to the overall solution. Unclamped uniform knot vectors offer also implementational benefits, e.g., b-spline shape functions need not be stored separately for each Bézier element as the boundary elements do not differ from the interior elements, at least, if each Bézier element uses the same quadrature. On the other hand, increased complexity is encountered in enforcing the boundary conditions.

The main contributions of IGA to the speed are increased accuracy per degree of freedom (fewer elements) and increased stable time step size relative to the element size. Furthermore, it is observed that running only the diagonal terms of the 7×7 inertia coupling system makes practically no difference in any of the benchmark cases. Therefore, the off-diagonal terms may be neglected to reduce the computational overhead. Another possibility to speed up the formulation, particularly if the shear strains are small, is to treat the stress stored in the local coordinate system as the corotational Cauchy stress, i.e., the rotation tensor defining the local coordinate system is assumed to follow the rotation given by the polar decomposition of the deformation gradient. As a consequence, the Cauchy stress does not need to be rotated between the time steps, and no Jaumann type update is required. However, the second local coordinate system suggested in Section A.3.3 is often more robust with this assumption.

The formulation was implemented using CUDA parallel computing platform, and most computations were performed parallel on a modern consumer level GPU. Some more complex simulations carried out by the authors (Hokkanen et al., 2018) indicate that the formulation is very competitive even against traditional low-order reduced integration shell elements in speed. In many applications, such as metal forming simulations, often over 50% larger time step size can be used in comparison to classical shell elements, either due to suitable boundary conditions or a larger boundary element size. This increase in the time step alone has been observed to make the proposed isogeometric element faster in comparison to a similarly implemented classical shell element (Belytschko et al., 1992)—even for the same mesh size (provided that sufficient computing resources are available for parallelization).

A.A Mass terms

Due to the conservation of mass $\int_{\Omega_0} \rho_0 \mathrm{d}\Omega_0 = \int_{\Omega} \rho \, \mathrm{d}\Omega$. This implies that $\rho_0 J_0 = \rho J = const$ throughout the analysis, where ρ_0 and J_0 represent the density and the Jacobian determinant in the initial configuration, respectively (note that J_0 in this appendix differs from the definition (A.35) in Section A.3). Now, all the masses can be evaluated only once during the analysis and no recurring mass assemblies are required in contrast to many Reissner-Mindlin shell elements adopting rotational degrees of freedom. The physical mass contribution for each control point is calculated as

$$M_A^{rr} = \int_{\Omega_0} \rho_0 N_A \mathrm{d}\Omega_0 = \rho_0 J_0 \int_\xi \int_\eta N_A \int_{-1}^{1} 1 \mathrm{d}\zeta \mathrm{d}\eta \mathrm{d}\xi \tag{A.75}$$

where the constant $\rho_0 J_0$ has been moved outside the integral. The other terms are derived similarly as

$$\begin{aligned}
M_A^{rd} &= \rho_0 J_0 \int_\xi \int_\eta N_A \int_{-1}^{1} \Big(\frac{\zeta}{2} + \zeta^2 q_A\Big) \mathrm{d}\zeta \mathrm{d}\eta \mathrm{d}\xi = \frac{q_A}{3} M_A^{rr} \\
\boldsymbol{M}_A^{rq} &= \rho_0 J_0 \boldsymbol{d}_A \int_\xi \int_\eta N_A \int_{-1}^{1} \zeta^2 \mathrm{d}\zeta \mathrm{d}\eta \mathrm{d}\xi = \frac{\boldsymbol{d}_A}{3} M_A^{rr} \\
\boldsymbol{f}_{vA}^{r} &= 2\rho_0 J_0 \dot{q}_A \dot{\boldsymbol{d}}_A \int_\xi \int_\eta N_A \int_{-1}^{1} \zeta^2 \mathrm{d}\zeta \mathrm{d}\eta \mathrm{d}\xi = \frac{2\dot{q}_A \dot{\boldsymbol{d}}_A}{3} M_A^{rr} \\
M_A^{dd} &= \rho_0 J_0 \int_\xi \int_\eta N_A \int_{-1}^{1} \Big(\frac{\zeta^2}{4} + \zeta^3 q_A + \zeta^4 q_A^2\Big) \mathrm{d}\zeta \mathrm{d}\eta \mathrm{d}\xi = \Big(\frac{q_A^2}{5} + \frac{1}{12}\Big) M_A^{rr} \\
\boldsymbol{M}_A^{dq} &= \rho_0 J_0 \boldsymbol{d}_A \int_\xi \int_\eta N_A \int_{-1}^{1} \Big(\frac{\zeta^3}{2} + \zeta^4 q_A\Big) \mathrm{d}\zeta \mathrm{d}\eta \mathrm{d}\xi = \frac{q_A \boldsymbol{d}_A}{5} M_A^{rr} \\
\boldsymbol{f}_{vA}^{d} &= 2\rho_0 J_0 \dot{q}_A \dot{\boldsymbol{d}}_A \int_\xi \int_\eta N_A \int_{-1}^{1} \Big(\frac{\zeta^3}{2} + \zeta^4 q_A\Big) \mathrm{d}\zeta \mathrm{d}\eta \mathrm{d}\xi = \frac{2 q_A \dot{q}_A \dot{\boldsymbol{d}}_A}{5} M_A^{rr} \\
M_A^{qq} &= \rho_0 J_0 \boldsymbol{d}_A \cdot \boldsymbol{d}_A \int_\xi \int_\eta N_A \int_{-1}^{1} \zeta^4 \mathrm{d}\zeta \mathrm{d}\eta \mathrm{d}\xi = \frac{\boldsymbol{d}_A \cdot \boldsymbol{d}_A}{5} M_A^{rr} \\
f_{vA}^{q} &= 2\rho_0 J_0 \dot{q}_A \dot{\boldsymbol{d}}_A \cdot \boldsymbol{d}_A \int_\xi \int_\eta N_A \int_{-1}^{1} \zeta^4 \mathrm{d}\zeta \mathrm{d}\eta \mathrm{d}\xi = \frac{2\dot{q}_A \dot{\boldsymbol{d}}_A \cdot \boldsymbol{d}_A}{5} M_A^{rr}
\end{aligned} \tag{A.76}$$

where the integrals along ζ can be evaluated analytically in advance (as q_A can always be moved outside the integral and the shape functions N_A are only functions of ξ and η).

A.B Analytical solution for Timoshenko beam (SS)

The analytical solution for the nondimensionalized deflection $\bar{w}(x)$ of a simply supported Timoshenko beam under a uniformly distributed load is given as follows:

$$\bar{w}(x) = \frac{w(x) E I}{q_0 L^4} \tag{A.77}$$

The absolute deflection $w(x)$ is given by

$$w(x) = \frac{q_0 L^4}{24EI}\Big(\frac{x}{L} - 2\frac{x^3}{L^3} + \frac{x^4}{L^4}\Big) + \frac{q_0 L^2}{2\kappa GA}\Big(\frac{x}{L} - \frac{x^2}{L^2}\Big) \tag{A.78}$$

where q_0 is the uniformly distributed load, L is the beam length, E is Young's modulus, I is the second moment of area, κ is the shear correction factor, G is the shear modulus, A is the cross-sectional area, and x is the coordinate along the length of the beam proceeding from 0 to L.

A.C Analytical solution for Mindlin plate (SCSC)

The analytical solution for the nondimensionalized deflection $\bar{w}(x, y)$ of a Mindlin plate under a uniformly distributed load that has two opposite edges simply supported (hard support) and two opposite edges fixed is given as follows:

$$\bar{w}(x, y) = \frac{w(x, y)D}{q_0 a^4} \tag{A.79}$$

The absolute deflection $w(x, y)$ is given by (Wang et al., 2000)

$$w(x, y) = w_K(x, y) + \sum_{m=1}^{\infty} \frac{q_m}{\kappa Gh}\Big(\frac{a}{m\pi}\Big)^2 \Big(\beta_{1m}(y) + \beta_{2m}(y)\Big) \sin\frac{m\pi x}{a} \tag{A.80}$$

where

$$\begin{aligned}
\beta_{1m}(y) &= 1 - 2B_m \cosh\frac{m\pi y}{a} \\
\beta_{2m}(y) &= \Big(1 - 2B_m \cosh\psi_m\Big)\Big(\bar{\xi}_m \cosh\frac{m\pi y}{a} - \xi_m \frac{ay}{2m\pi}\sinh\frac{m\pi y}{a}\Big) \\
B_m &= \frac{1}{\cosh\psi_m + \psi_m \operatorname{csch}\psi_m}, \quad q_m = \frac{2q_0}{m\pi}(1 - (-1)^m) \\
\bar{\xi}_m &= \xi_m \frac{ab}{4m\pi}\tanh\psi_m - \operatorname{sech}\psi_m \\
\xi_m &= \frac{1}{B_{1m} - B_{2m} - B_{3m}}\Big(\tanh\psi_m \cosh\frac{\lambda_m b}{2} - \frac{m\pi}{a\lambda_m}\sinh\frac{\lambda_m b}{2}\Big)
\end{aligned} \tag{A.81}$$

and

$$\begin{aligned}
B_{1m} &= A_{1m}\cosh\psi_m \sinh\frac{\lambda_m b}{2} \\
B_{2m} &= A_{2m}\sinh\psi_m \cosh\frac{\lambda_m b}{2} \\
B_{3m} &= A_{3m}\operatorname{sech}\psi_m \cosh\frac{\lambda_m b}{2} \\
A_{1m} &= \frac{m\pi}{a\lambda_m}\frac{D}{\kappa Gh}, \quad A_{2m} = \frac{D}{\kappa Gh} + \frac{1}{2}\Big(\frac{a}{m\pi}\Big)^2, \quad A_{3m} = \frac{ab}{4m\pi} \\
\lambda_m &= \frac{m\pi}{a} + \sqrt{\frac{2\kappa Gh}{D(1-\nu)}}, \quad \psi_m = \frac{m\pi b}{2a}
\end{aligned} \tag{A.82}$$

The Kirchhoff plate solution w_K, i.e., the solution neglecting shear deformations, is given as follows:

$$\begin{aligned} w_K(x,y) &= \frac{a^4}{D\pi^4}\sum_{m=1}^{\infty}\frac{q_m}{m^4}\Big(1 + A_m\cosh\frac{m\pi y}{a} + B_m\frac{m\pi y}{a}\sinh\frac{m\pi y}{a}\Big)\sin\frac{m\pi x}{a} \\ A_m &= -\frac{1+\psi_m\coth\psi_m}{\cosh\psi_m + \psi_m\operatorname{csch}\psi_m} \end{aligned} \tag{A.83}$$

In the above equations, a represents the lengths of the fixed edges at $y = -\frac{b}{2}$ and $y = \frac{b}{2}$ whereas b represents the lengths of the simply supported edges at $x = 0$ and $x = a$. For a square plate $a = b$. Furthermore, q_0 is the uniformly distributed load, κ is the shear correction factor, and G is the shear modulus. The flexural rigidity is given as $D = \frac{Eh^3}{12(1-\nu^2)}$, where E is Young's modulus, h is the thickness, and ν is Poisson's ratio.

The following submitted manuscript has been incorporated as Appendix B.

Hokkanen, J. and Pedroso, D. M. (2019). Quadrature rules for isogeometric shell formulations: study using a real-world application about metal forming. Submitted to *Computer Methods in Applied Mechanics and Engineering* on 25th March 2019.

Contributor	Statement of contribution	%
Jaro Hokkanen	writing of text	90
	proof-reading	70
	theoretical derivations	80
	computer implementation	90
	numerical calculations	100
	experimental results	20
	preparation of figures	80
Dorival M. Pedroso	writing of text	10
	proof-reading	30
	theoretical derivations	10
	supervision, guidance	80
Michael C. Elford	computer implementation	10
	experimental results	40
	preparation of figures	10
	supervision, guidance	20
Andrew J. E. Stephan	theoretical derivations	10
	experimental results	40
	preparation of figures	10

Appendix B

Quadrature rules for isogeometric shell formulations: study using a real-world application about metal forming

This paper studies quadrature rules for simulating large deformations of shells using isogeometric analysis. Several recently proposed rules and their effects on a real-world application known as incremental sheet forming are investigated. It is observed that, when tackling real-world applications, unexpected problems arise and, therefore, theoretical studies only with manufactured solutions are not enough for a complete verification of a method. The chosen application reveals problems with certain quadratures and that some simple stabilization strategies cannot completely suppress hourglass modes. Additionally, the effects of quadrature rules on the total computational costs are demonstrated and the influence of the maximum stable time step is assessed using a highly demanding simulation.

B.1 Introduction

In isogeometric analysis (IGA), two notions of elements can be identified which have features in common with the classical perception of a finite element: Bézier elements and patches. Early papers on isogeometric analysis follow the standard practice from the classical finite element analysis and use Gauss-Legendre quadratures for each

Bézier element for numerical integration. However, in IGA, the integrand function is often continuous across the Bézier elements, which makes it possible for a single integration point to contribute to multiple Bézier elements. In 2010, Hughes et al. initiated a study on optimal quadrature rules for isogeometric analysis by considering the whole patch as the basic entity for determining the quadrature instead of a single Bézier element. This subject has attracted significant interest among the computational mechanics community leading to several published contributions (Auricchio et al., 2012b; Schillinger et al., 2014; Adam et al., 2014, 2015a,c; Hillman et al., 2015; Ait-Haddou et al., 2015; Barto and Calo, 2016; Barto et al., 2017; Johannessen, 2017; Hiemstra et al., 2017), most of which concentrate on methods for determining the optimal patchwise quadratures.

If the corresponding integrand is polynomial in the parametric coordinates and is exactly integrated using a numerical method, the quadrature rule is referred to as a full integration rule. On the other hand, a rule resulting in an approximate integration is referred to as a reduced rule. However, spline-based interpolation functions may also be rational, and a quadrature resulting in an approximate integration of the corresponding rational function is still considered a full integration rule, provided that it exactly integrates the polynomial numerator of the respective rational function. The main motivations to use reduced integration are to reduce the computational cost and to alleviate numerical locking.

Many quadrature rules are proposed for isogeometric analysis in the literature. In most cases, the testing applications are limited to small-scale elastic and geometrically linear problems without considering the locking aspect. This paper, however, compares the capability of each quadrature rule to alleviate locking, and also extends benchmarking to a large-scale metal forming application, which leads to a discovery of certain new problems associated with the previously proposed patchwise quadratures. In contrast to most previous works, the comparisons are made by using unclamped knot vectors to maintain good comparability between different orders of interpolation functions and to improve the performance of the boundary elements (Hokkanen and Pedroso, 2019a). Moreover, based on numerical experimentation, the maximum stable time step estimates for the explicit central difference time integration scheme are listed, and the computational cost is evaluated for a large-scale problem using a modern parallel GPU-based implementation.

The paper is structured as follows. Section B.2 describes the investigated quadrature

rules. Section B.3 assesses the performance of the quadrature rules using two simple examples with manufactured solutions: a Mindlin plate and a straight cantilever plate strip. Section B.4 presents the application to the real-world metal forming problem—a manufacturing process known as incremental sheet forming. Finally, some conclusions are given in Section B.5.

B.2 Quadrature rules

The investigation is limited to quadrature rules which have been proposed for polynomial interpolation functions of orders $p = 2$ and $p = 3$ with regularity $q = p - 1$. Assuming affine transformation from the parametric domain to the physical domain, the required target space to exactly integrate the corresponding mass and stiffness integrals has the minimum order of $2p$ and the maximum regularity of $q - 1$; this target space is denoted as follows:

$$\text{target space} \equiv S^{2p}_{q-1} \quad \text{with p:order and q:regularity} \tag{B.1}$$

In this paper, a quadrature rule that integrates a certain space exactly using the minimum number of integration points is referred to as optimal rule. Optimal rules for spaces other than the corresponding target space are used as reduced integration rules. Five of the quadrature rules considered for biquadratic interpolation functions are reduced rules, whereas the sixth rule results in full integration. Furthermore, two reduced rules are tested for bicubic interpolation.

The following six quadrature rules have been applied to biquadratic Bézier elements by Benson et al. (2010), Schillinger et al. (2014), Adam et al. (2015a,c), and Hiemstra et al. (2017). The first rule is referred to as the Center scheme and is exact for up to $S^1_{-1} \otimes S^1_{-1}$ space. It was proposed (in the context of IGA) by Adam et al. (2015a) who suggested over-integrating the boundary elements to avoid hourglass modes (Figure B.1a). The optimal rule for $S^2_0 \otimes S^2_0$ space (Figure B.1b) was proposed by Adam et al. (2015c), who also recommended over-integrating the boundary elements to suppress the hourglass modes. In this paper, the corner Bézier elements are integrated using the Gauss 2×2 scheme (Figure B.1e) to obtain the over-integrated Center and $S^2_0 \otimes S^2_0$ schemes. The Center-edge and the Gauss 2×2 integration schemes discussed by Schillinger et al. (2014) and illustrated in Figures B.1c and B.1e, respectively, were reported to result in the same accuracy as the full Gauss rule in numerical testing. The Center-edge scheme is classified as a monomial rule (as opposed to the

tensor-product rules) and integrates monomials exactly up to a total degree of 3. The Gauss 2×2 scheme is exact for up to $S^3_{-1} \otimes S^3_{-1}$ space. Hiemstra et al. (2017) suggested using the optimal rule for $S^3_0 \otimes S^3_0$ space (Figure B.1d). The sixth scheme considered in this paper is the optimal rule for the corresponding $S^4_0 \otimes S^4_0$ target space (Figure B.1f).

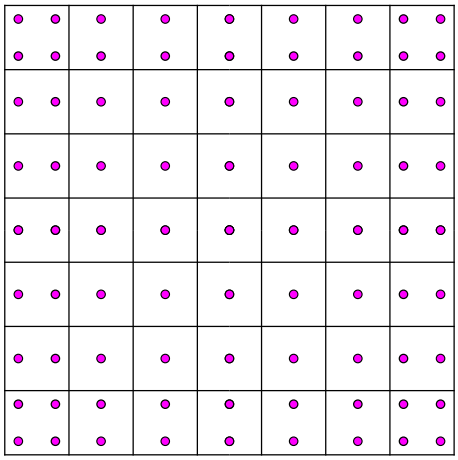
Moreover, the following two reduced rules proposed for bicubic Bézier elements are investigated. Adam et al. (2015c) suggested using the optimal rule for $S^3_1 \otimes S^3_1$ space with over-integrated boundary elements. In this paper, the over-integrated $S^3_1 \otimes S^3_1$ scheme (Figure B.2a) is obtained by taking the corner Bézier element integration point locations and weights from the optimal scheme for $S^4_1 \otimes S^4_1$ space (see Adam et al. (2015c)) to minimize locking. The second rule is the optimal rule for $S^5_1 \otimes S^5_1$ space (Figure B.2b) and was proposed by Hiemstra et al. (2017).

The performance of the aforementioned quadrature rules is evaluated using the shell formulation of Hokkanen and Pedroso (2019a), which uses spline-based interpolation along the in-plane dimensions of the shell. Three benchmark cases are examined: typical geometrically linear and nonlinear small-scale elastic problems, and a large-scale elasto-plastic incremental sheet forming problem with contact. All examples use unclamped uniform knot vectors and uniform control point spacing. The through-thickness integration uses 9-point Simpson's rule. The optimal rule for $S^2_0 \otimes S^2_0$ space is evaluated with and without over-integrated boundary Bézier elements. The Center scheme as well as the optimal scheme for $S^3_1 \otimes S^3_1$ space do not produce meaningful results for all benchmark cases without any type of hourglass stabilization and are hence only considered with over-integrated boundary elements. The central difference scheme is used for the time integration.

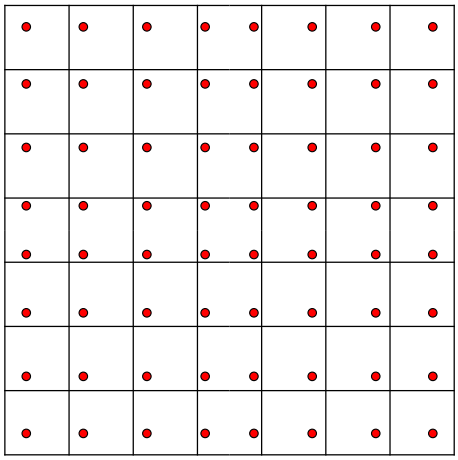
B.3 Verifications using manufactured solutions

B.3.1 Mindlin plate (SCSC)

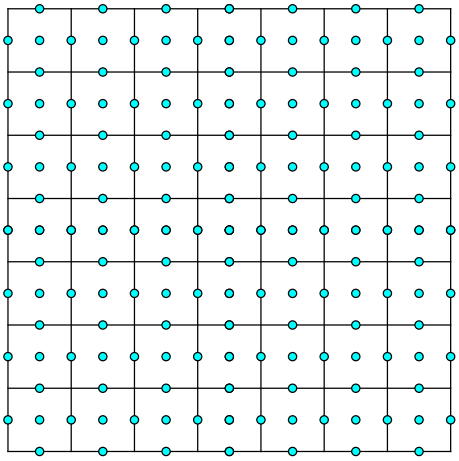
The first example considers the linear elastic response of a Mindlin plate under a uniformly distributed load. The boundary conditions are imposed such that two opposite edges are simply supported and two opposite edges are clamped as shown in Figure B.3a. The analytical expression for the nondimensionalized deflection $\bar{w}$ is given by Hokkanen and Pedroso (2019a). The square plate of slenderness ratio $L/h = 100$ and Poisson's ratio $\nu = 0.3$ is considered, where L and h represent the



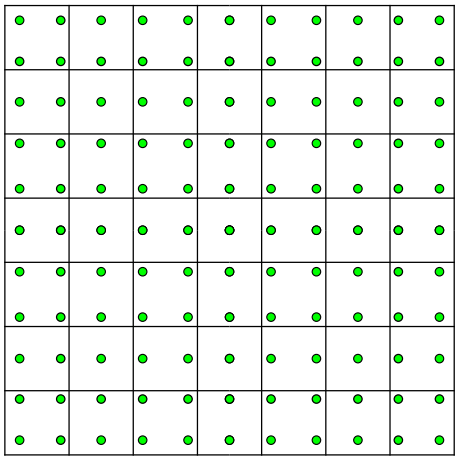
(a) Over-integrated Center scheme (cost $\to 1$).



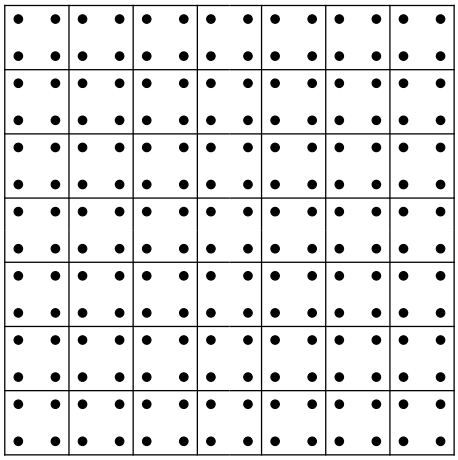
(b) $S_0^2 \otimes S_0^2$ scheme (cost $\to 1$).



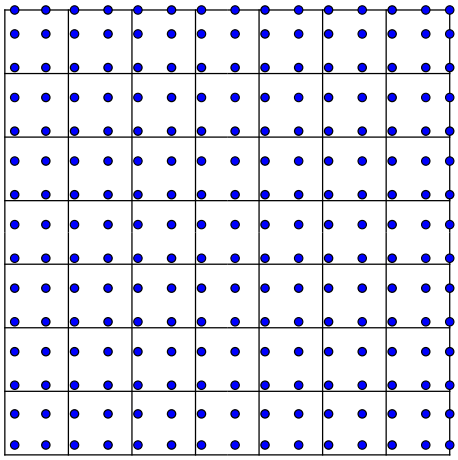
(c) Center-edge scheme (cost $\to 3$).



(d) $S_0^3 \otimes S_0^3$ scheme (cost $\to 2.25$).

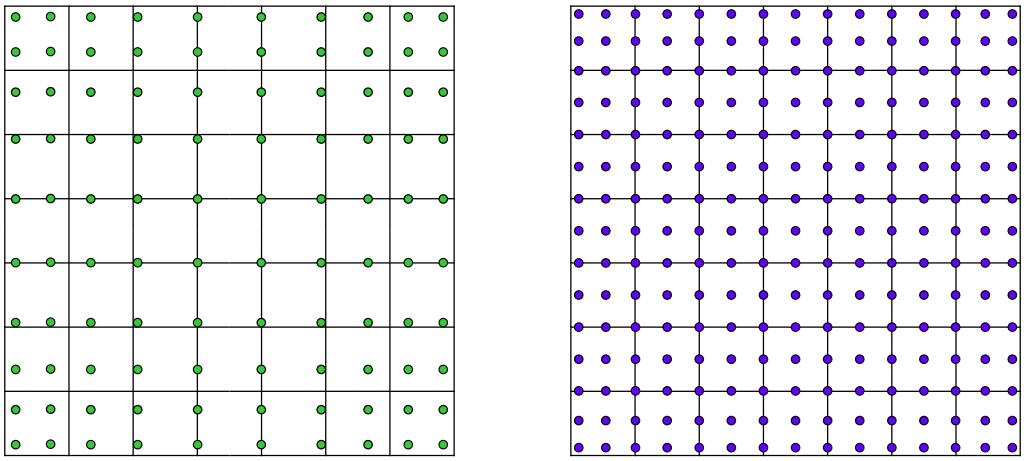


(e) Gauss 2×2 scheme (cost $= 4$).



(f) $S_0^4 \otimes S_0^4$ scheme (cost $\to 4$).

Figure B.1: In-plane quadrature point locations proposed for biquadratic splines.



(a) Over-integrated $S_1^3 \otimes S_1^3$ scheme (cost $\to 1$).

(b) $S_1^5 \otimes S_1^5$ scheme (cost $\to 4$).

Figure B.2: In-plane quadrature point locations proposed for bicubic splines.

edge length and the thickness of the plate, respectively.

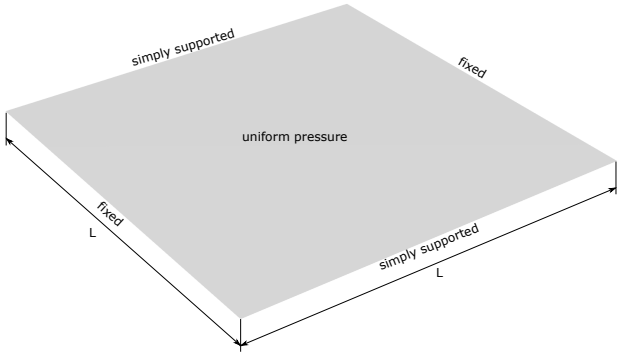
The relative error in the L^2-norm of the nondimensionalized deflection $\bar{w}$ is plotted against the normalized control point spacing in Figures B.3b and B.3c, for biquadratic and bicubic splines, respectively. The results from the coarsest discretizations suggest the largest errors for the Center-edge and $S_0^4 \otimes S_0^4$ schemes. This is a result of numerical locking and does not indicate inaccuracy in calculating the corresponding integrals.

The observed numerical locking vanishes by refining the mesh, which explains the higher than optimal convergence rates for the Center-edge and $S_0^4 \otimes S_0^4$ rules. The convergence rates are expected to approach the optimal rate as the mesh is refined. The Gauss 2×2 scheme is a good example of this behavior as it is less prone to locking and reaches the optimal convergence rate for the finest discretizations.

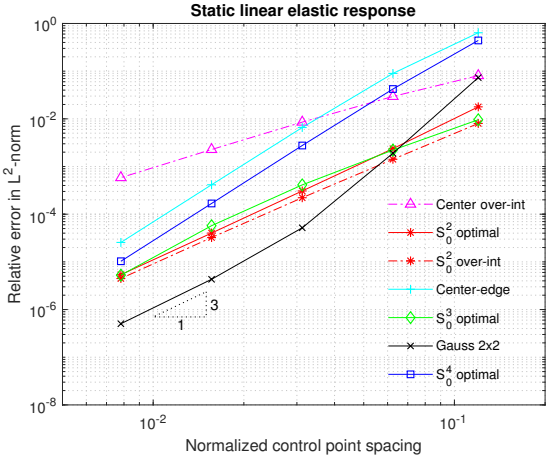
The over-integrated Center scheme suffers from severe hourglassing and converges towards the correct solution much slower than any other scheme. The $S_0^2 \otimes S_0^2$ and $S_0^3 \otimes S_0^3$ schemes show no significant locking tendencies in this case and perform much better for the coarse discretizations than any other scheme used with biquadratic interpolation. Furthermore, the accuracy of the over-integrated $S_0^2 \otimes S_0^2$ scheme is close to its non-over-integrated counterpart. For biquadratic splines, the best accuracy for the fine discretizations is obtained by the Gauss 2×2 scheme.

In contrast to the work of Adam et al. (2015c), the over-integrated $S_1^3 \otimes S_1^3$ scheme is observed to suffer from hourglassing and does not achieve the optimal convergence rate for bicubic splines. The $S_1^5 \otimes S_1^5$ scheme shows slight locking tendency for the

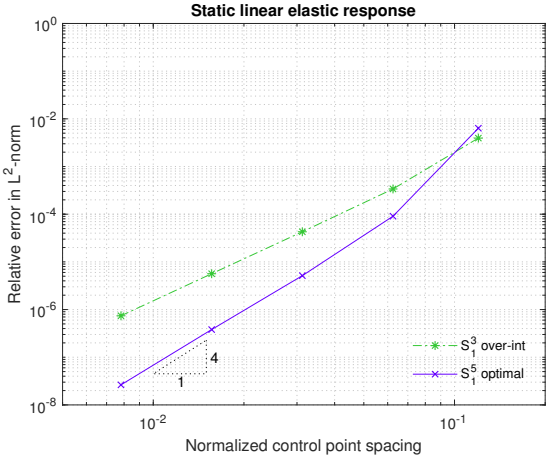
coarsest discretization but quickly achieves the optimal convergence rate.



(a) The problem setup.



(b) The relative error in the L^2-norm of the nondimensionalized deflection (biquadratic splines).



(c) The relative error in the L^2-norm of the nondimensionalized deflection (bicubic splines).

Figure B.3: Mindlin plate under a uniformly distributed load.

B.3.2 Straight cantilever plate strip

In this example, the elastic straight cantilever plate strip shown in Figure B.4 is considered. The plate strip has length $L = 10$, width $w = 1.0$ and thickness $h = 0.1$. Young's modulus and Poisson's ratio are $E = 10^7$ and $\nu = 0.3$, respectively. The applied line load is given as $F = 40\lambda$, where λ is the load factor ranging from 0.0 to 1.0.

The maximum vertical tip displacement is plotted against the load factor λ in Figure B.5. The theoretical solution is given by Frisch-Fay (1962). The performance of the quadrature rules is evaluated using three different discretizations—5, 9, and 17 control points along the length of the plate strip. For biquadratic splines, the $S_0^2 \otimes S_0^2$ scheme performs the best and shows no locking tendencies. In contrast to

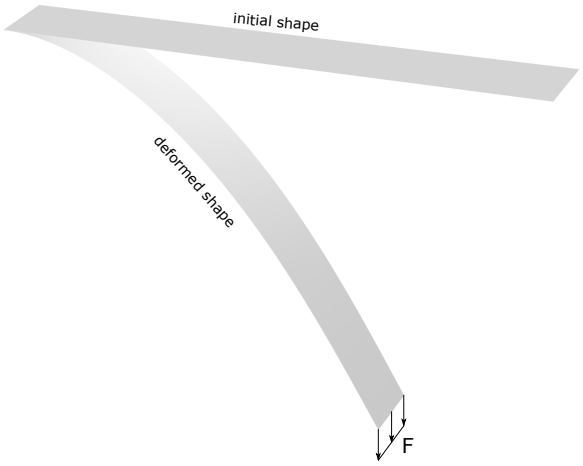


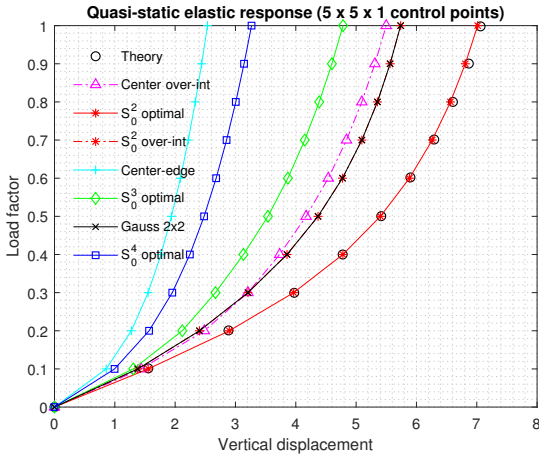
Figure B.4: Cantilever plate strip under a tip line load—the problem setup.

the Mindlin plate problem, the performance of the over-integrated Center scheme is not significantly hindered by hourglassing as only small numbers of control points are used, and the over-integrated boundary elements are able to provide sufficient resistance against the hourglass modes. The over-integrated $S_0^2 \otimes S_0^2$ scheme is identical to the Gauss 2×2 scheme for the coarsest mesh due to a small number of Bézier elements and performs only slightly better after mesh refinement against the latter. The $S_0^3 \otimes S_0^3$ scheme shows a clear locking tendency in this problem and performs significantly worse than the Gauss 2×2 scheme. Furthermore, the Center-edge and $S_0^4 \otimes S_0^4$ schemes suffer from severe locking, the Center-edge scheme being clearly the worst.

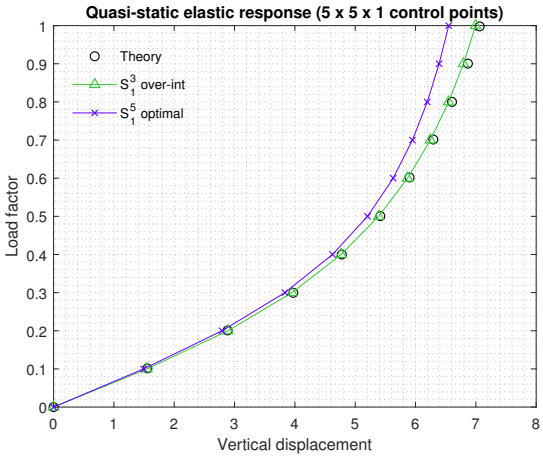
The over-integrated $S_1^3 \otimes S_1^3$ scheme performs very well for bicubic splines and shows no locking tendencies. Similarly to the Center scheme, hourglassing does not deteriorate the results due to small numbers of control points. The $S_1^5 \otimes S_1^5$ rule performs reasonably well and experiences slight locking only for the coarsest mesh.

B.4 Real-world application: incremental sheet forming

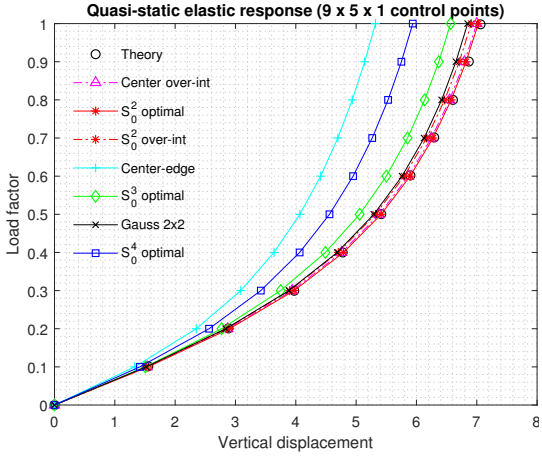
The real-world application considers a manufacturing process known as incremental sheet forming (ISF). The ISF process is a challenging large-scale problem well suited for a study about efficient explicit shell formulations. The study proposed in this paper also includes a detailed qualitative evaluation of the deformed geometry and



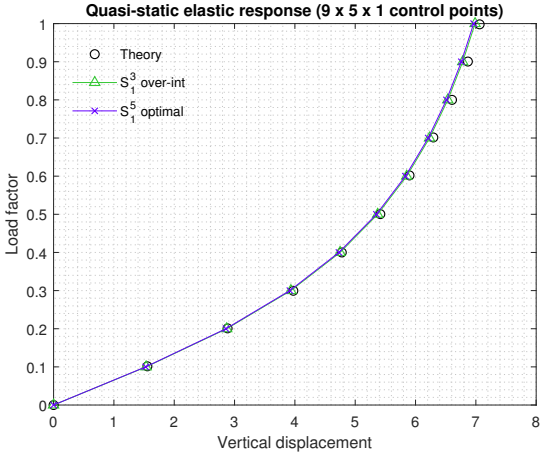
(a) $5 \times 5 \times 1$ *control points (biquadratic splines).*



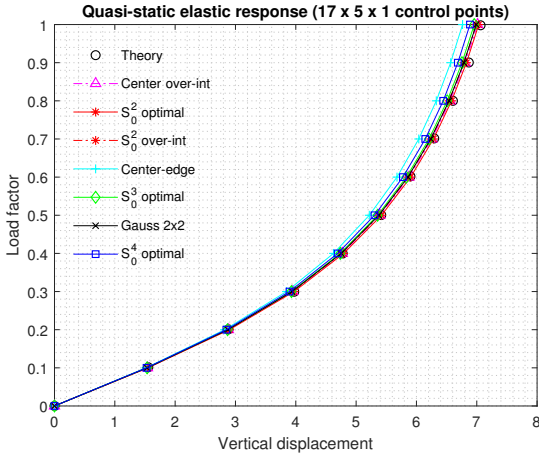
(b) $5 \times 5 \times 1$ *control points (bicubic splines).*



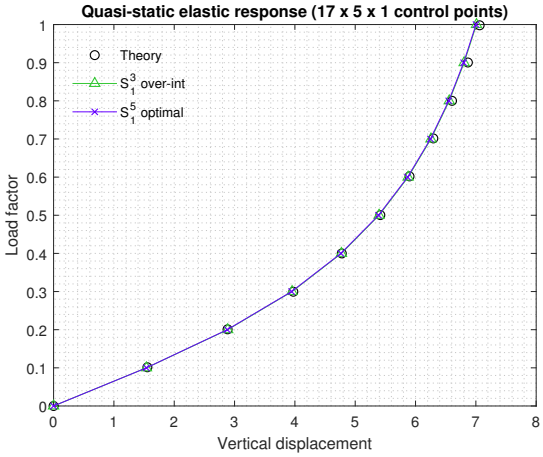
(c) $9 \times 5 \times 1$ *control points (biquadratic splines).*



(d) $9 \times 5 \times 1$ *control points (bicubic splines).*



(e) $17 \times 5 \times 1$ *control points (biquadratic splines).*



(f) $17 \times 5 \times 1$ *control points (bicubic splines).*

Figure B.5: Cantilever plate strip under a tip line load—load factor versus the vertical tip displacement.

thinning based on the results from an innovative experimental programme carried out at The University of Queensland in Australia. The numerical results obtained with the different quadrature rules are processed to visually demonstrate the implications resulting from the choice of the quadrature rule. Furthermore, the simulations are used to compare the computational overhead between each quadrature rule and to give a rough estimate of the maximum stable time step size for the interior Bézier elements.

B.4.1 Problem definition and numerical model

The target geometry dimensions for the ISF simulation are given in Figure B.6. The path followed by the indenting tool (stylus) is illustrated in Figure B.7. The tool has a hemispherical head, 20 mm in diameter, which makes sliding contact with the metal sheet. The square sheet has the thickness of 1.6002 mm (0.063 inches) and the edge length of 290 mm.

The isogeometric formulation uses a nonmortar Gauss-point-to-surface (GPTS) penalty contact algorithm (Temizer et al., 2011; De Lorenzis et al., 2014) for the stylus contact similarly to the work of Hokkanen et al. (2018). The stylus tool proceeds in the counterclockwise direction and steps down 2 mm after each full revolution. The assumed tool speed is 4000 mm/min.

Nondimensional mass scaling of 4000 is applied to increase the stable time step size and to speed up the simulation. A relatively coarse control point spacing (4 mm) is used to better demonstrate the differences between the quadrature rules. The six outer rows of control points are fixed for each edge to account for sheet clamping. After the stylus tool has finished the whole toolpath, the constraints are removed and the sheet finds its unclamped equilibrium state.

The simulation assumes hypoelastic-plastic material behavior. The material model is based on the isotropic von Mises yield criterion. The required parameters describing the constitutive relations for aluminum 7075-O are the density $\rho = 2.81 \times 10^{-9}\,\frac{\mathrm{Ns}^2}{\mathrm{mm}^4}$, Young's modulus $E = 71.7 \times 10^3\,\mathrm{MPa}$, Poisson's ratio $\nu = 0.33$, and the initial yield stress $\sigma_{y0} = 261\,\mathrm{MPa}$. The hardening coefficient and the hardening exponent required by the Hollomon type hardening law $\sigma_y = C\varepsilon_p^n$ are given as $C = 334.59\,\mathrm{MPa}$ and $n = 0.157$, respectively.

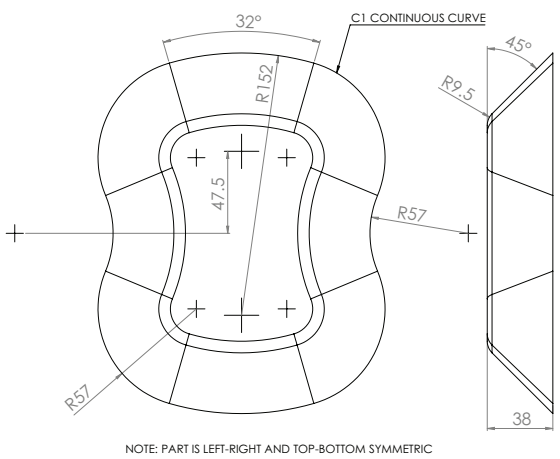


Figure B.6: The target geometry dimensions.

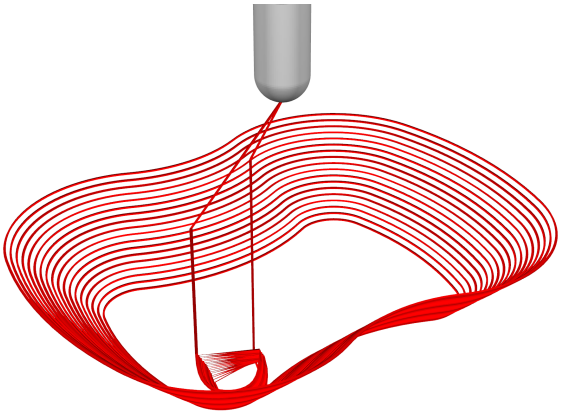


Figure B.7: The path followed by the indenting stylus (toolpath).

B.4.2 Numerical results

The formed parts from the laboratory experiments are brought to a 3D scanner to measure the change in thickness due to the indentation process. The measured thickness can then be compared with the numerical results. Figure B.8 shows the formed part of the experimental test and Figure B.9 shows the measured thickness distribution over the part.

The simulation results for biquadratic splines are given by Figures B.10 (geometry) and B.11 (thickness) for each quadrature rule. It is observed that the Center rule suffers from severe hourglassing as shown in Figure B.10a. The hourglass problem also appears in the thickness plot (Figure B.11a), but it is less obvious. The thickness plot is somewhat rough but still well in line with the experimental result.

The geometry plot (Figure B.10b) resulting from the $S_0^2 \otimes S_0^2$ scheme shows no obvious signs of hourglassing. However, small "imperfections" in the formed surface can be observed under closer inspection compared to the Center-edge, Gauss 2×2, and $S_0^4 \otimes S_0^4$ schemes. The thickness plot (Figure B.11b) is also clearly rougher in comparison with the three aforementioned schemes.

The $S_0^3 \otimes S_0^3$ rule produces a wavy pattern associated with the deformed areas only. According to Hiemstra et al. (2017), this quadrature rule does not possess any hourglass modes. The wavy pattern is suspected to originate from the irregular nature of the scheme. It is worth noting that the integration point locations with respect to the Bézier elements differ from one element to another. The elements with more integration points are typically more susceptible to numerical locking which

may explain the observed behavior. The same wavy pattern can be observed in the thickness plot (Figure B.11d).

The Center-edge, Gauss 2×2 and $S_0^4 \otimes S_0^4$ schemes all result in a very clean geometry and thickness plots (Figures B.10c, B.10e, B.10f and B.11c, B.11e, B.11f). However, it is noticed that the interruption in the thickness contours in the bottom right region of each plot (i.e., the stylus tool step down location) is less visible the more the scheme suffers from locking.

The simulation results for the bicubic splines are given in Figures B.12 (geometry) and B.13 (thickness). The $S_1^3 \otimes S_1^3$ scheme suffers from hourglassing producing a similar out-of-plane hourglass mode as the Center scheme for biquadratic splines. The $S_1^5 \otimes S_1^5$ scheme produces clean geometry and thickness plots which match well with the experimental results.

For each quadrature rule, the vertical forces experienced by the stylus tool are shown in Figures B.14a and B.14b. It is observed that the schemes that suffer from hourglassing such as the over-integrated Center and $S_1^3 \otimes S_1^3$ schemes experience the smallest forming forces and deviate most from the average. The differences in the forming forces between the other quadrature rules are relatively small. The over-integrated $S_0^2 \otimes S_0^2$ scheme predicts the smallest forces among the schemes that do not suffer from obvious hourglassing. It is noted that these results correspond to the previous benchmark problems in Section B.3.

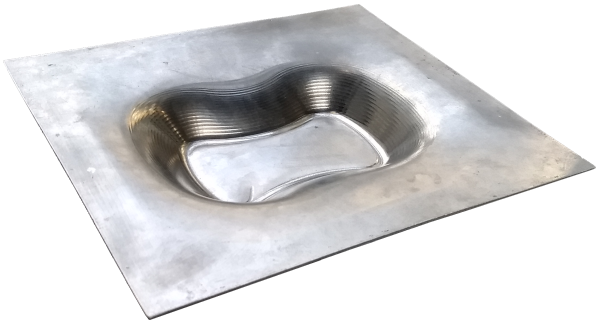


Figure B.8: A photograph of the formed part.

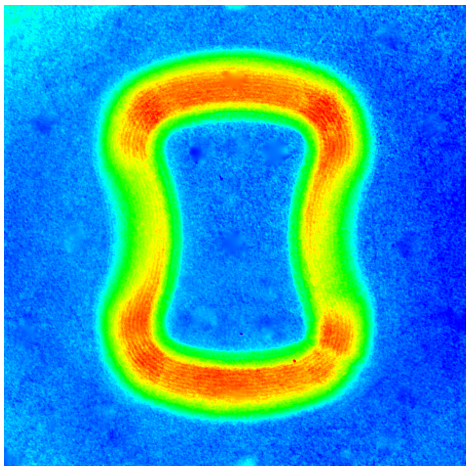
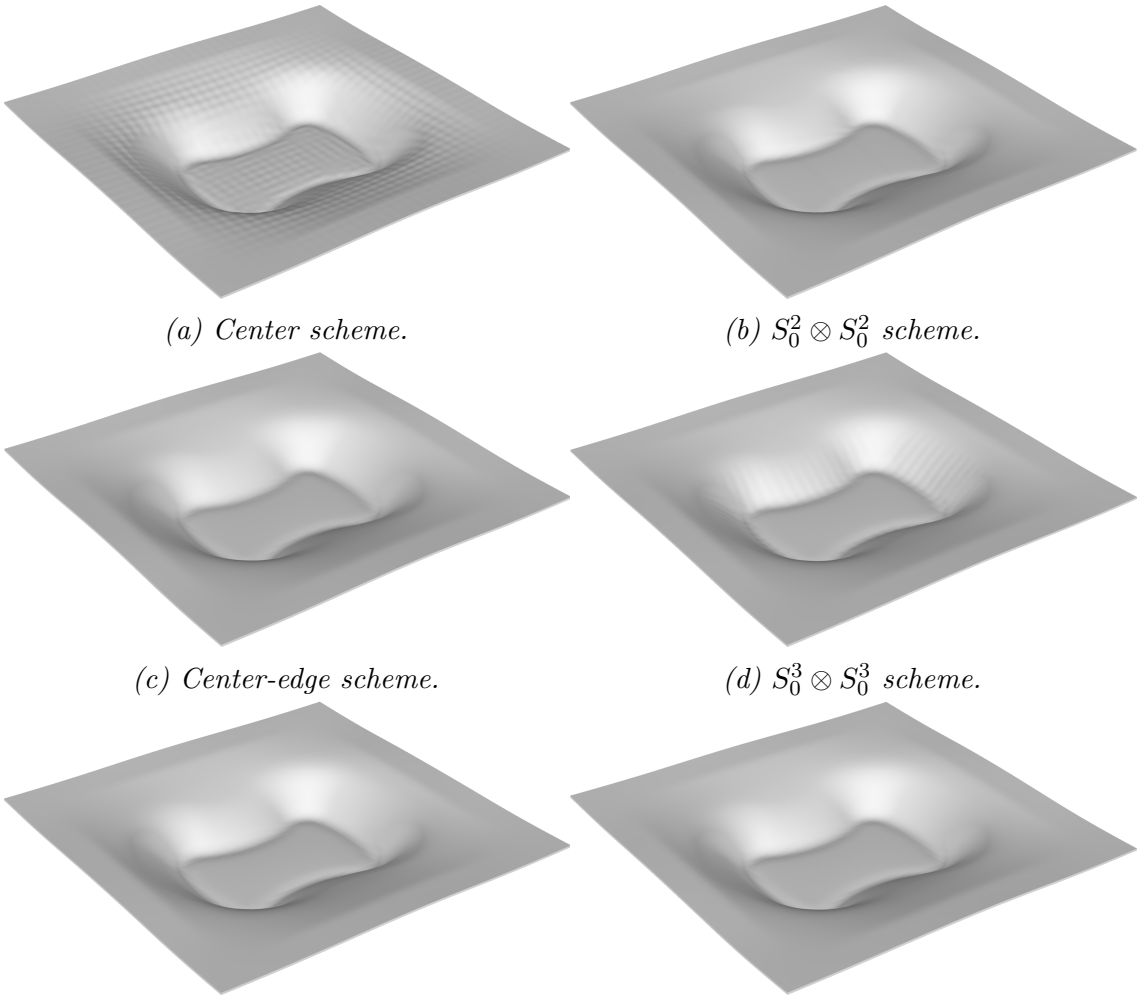


Figure B.9: The experimental normalized thickness from the 3D scanning. The contours vary linearly from 0.62 to 1.0.



(a) Center scheme. *(b) $S_0^2 \otimes S_0^2$ scheme.*

(c) Center-edge scheme. *(d) $S_0^3 \otimes S_0^3$ scheme.*

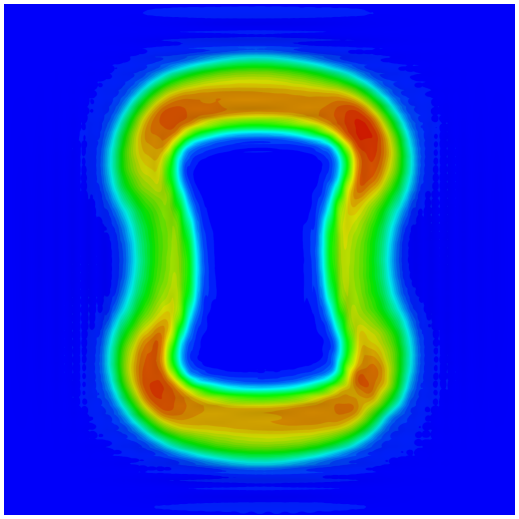
(e) Gauss 2×2 scheme. *(f) $S_0^4 \otimes S_0^4$ scheme (full integration).*

Figure B.10: The deformed geometry for each quadrature rule (biquadratic splines).

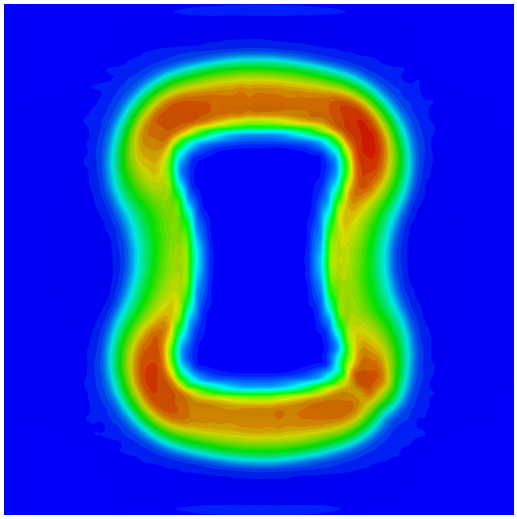
B.4.3 Time step and computational cost

An important advantage of the isogeometric formulation is the possibility of an increased stable time step size in explicit time integration. In IGA, the stable time step size increases when the order of spline-based interpolation functions is increased (if no repeated knots are introduced). It is worth noting that this behavior is opposite to simulations based on conventional Lagrange elements. From the experiments carried out in this paper, the maximum time step sizes relative to the element time step estimate (Adam et al., 2015b; Hokkanen and Pedroso, 2019a) that lead to a somewhat stable solution are given in Table B.1 along with the total computational time. The total computational times are obtained by using the maximum stable time step size for each quadrature rule.

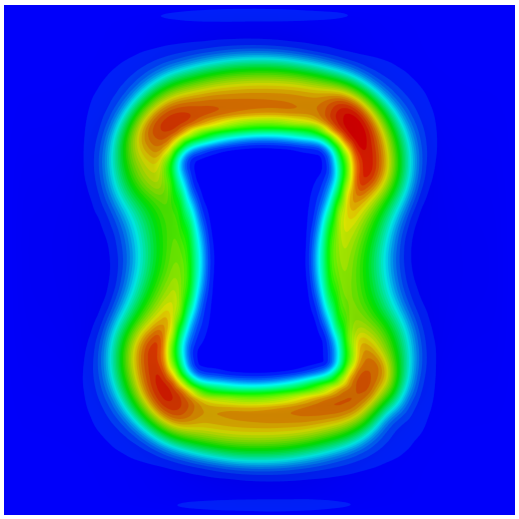
It is important to note that the placement of the integration points is not only crucial



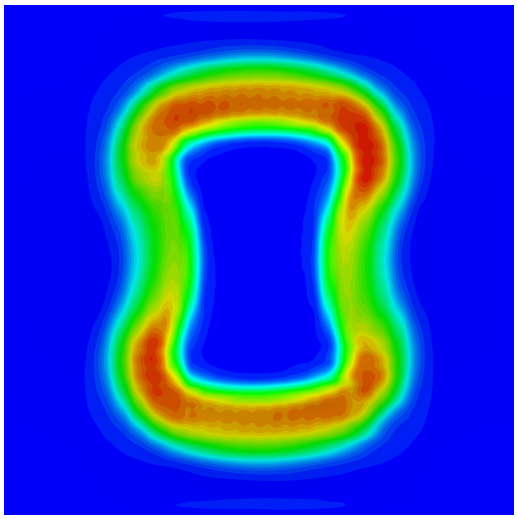
(a) Center scheme.



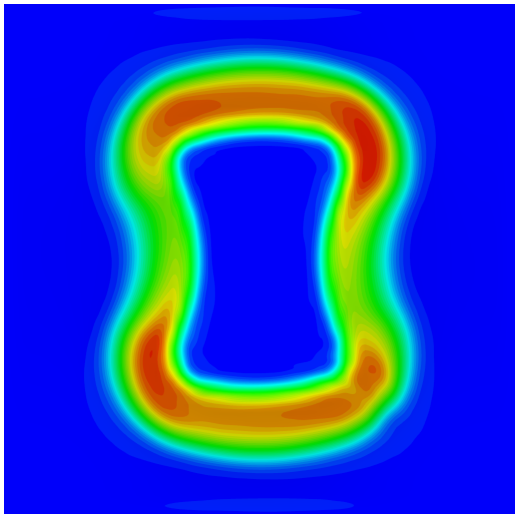
(b) $S_0^2 \otimes S_0^2$ scheme.



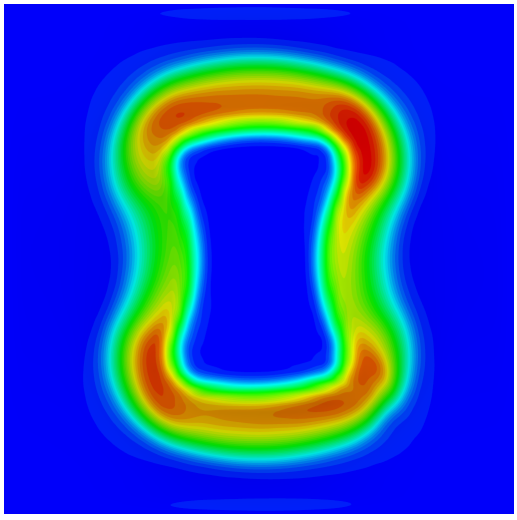
(c) Center-edge scheme.



(d) $S_0^3 \otimes S_0^3$ scheme.

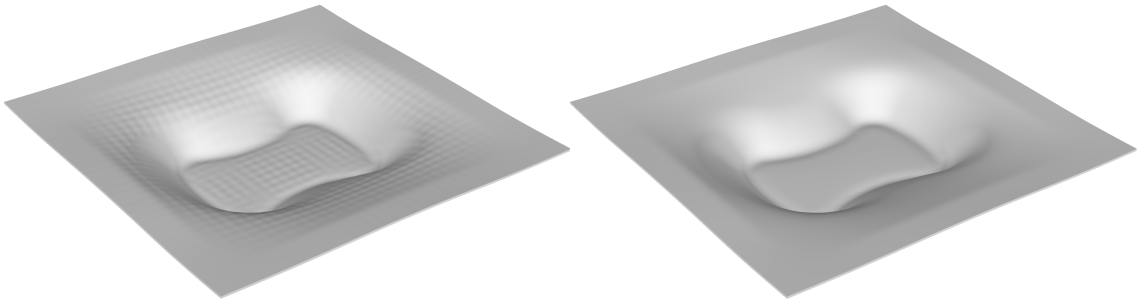


(e) Gauss 2×2 scheme.



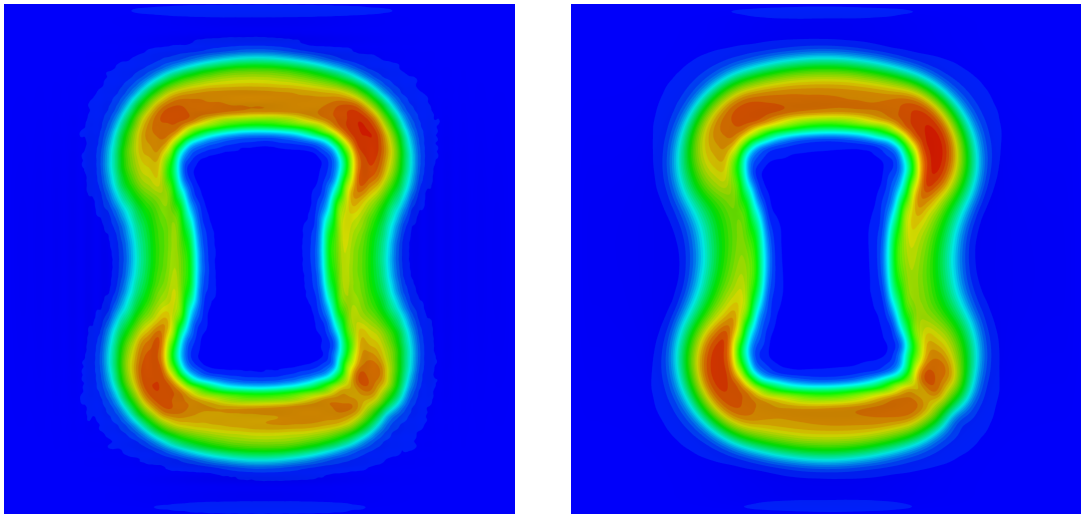
(f) $S_0^4 \otimes S_0^4$ scheme (full integration).

Figure B.11: The normalized thickness plot for each quadrature rule (biquadratic splines). The contours vary linearly from 0.62 to 1.0.



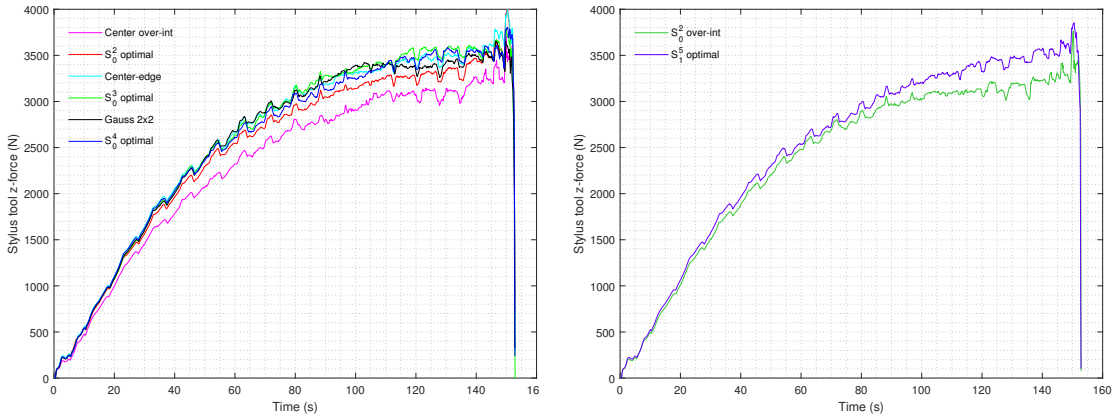
(a) $S_1^3 \otimes S_1^3$ *scheme.* *(b)* $S_1^5 \otimes S_1^5$ *scheme.*

Figure B.12: The deformed geometry for each quadrature rule (bicubic splines).



(a) $S_1^3 \otimes S_1^3$ *scheme.* *(b)* $S_1^5 \otimes S_1^5$ *scheme.*

Figure B.13: The normalized thickness plot for each quadrature rule (bicubic splines). The contours vary linearly from 0.62 to 1.0.



(a) Biquadratic splines. *(b) Bicubic splines.*

Figure B.14: Forming forces in the global z-direction (vertical direction).

for the accuracy of the solution, but it also affects the computational cost of the simulation. When using bicubic splines, each Bézier element is associated with 16 control points. However, only 9 of these control points have nonzero interpolation function or the first derivative of the interpolation function at a vertex location. This makes the cost of evaluating a vertex integration point for bicubic splines the same as the cost of evaluating an interior integration point for biquadratic splines. If an integration point is located at a Bézier element edge, but not a vertex, then 12 control points are required for interpolation (assuming bicubic interpolation functions).

The computational cost per degree of freedom increases by raising the order of interpolation functions, as the support of the basis functions becomes wider. For an explicit solver, this is usually a significant cost. However, the $S_1^3 \otimes S_1^3$ and $S_1^5 \otimes S_1^5$ schemes greatly benefit from having the majority of the integration points located at Bézier element edge or vertex locations. As shown by Table B.1, the cost of increasing the order of the interpolation functions from 2 to 3 is minimal, mostly due to the increased time step size, but also because the placement of the integration points is ideal. In comparison to biquadratic splines, bicubic splines offer higher convergence rate and better accuracy for the same number of degrees of freedom, without necessarily increasing the total computational cost even if an explicit solver is used.

According to Table B.1, the single most important factor affecting the computational cost is the number of integration points. The optimal $S_0^2 \otimes S_0^2$ and $S_1^3 \otimes S_1^3$ schemes offer significant savings in computational costs while being locking-free. Unfortunately, this comes at the cost of lower accuracy and possible hourglass problems. Both of these schemes have been shown to lead to rank-deficient stiffness matrices without over-integrated boundary elements (Adam et al., 2015c). However, severe hourglassing is observed even for the over-integrated $S_1^3 \otimes S_1^3$ scheme. The over-integrated $S_0^2 \otimes S_0^2$ scheme does not show obvious signs of hourglassing.

B.4.4 General observations

It is further noted that the hourglassing problems associated with the over-integrated Center and $S_1^3 \otimes S_1^3$ schemes appear well before the boundary conditions are removed at the end of the simulation. The same conclusion applies to the wavy pattern associated with the $S_0^3 \otimes S_0^3$ scheme. Moreover, it is observed that removing the boundary conditions from all edges reduces the maximum stable time step size even

Table B.1: Quadrature rule cost comparison using Nvidia Titan V GPU.

Quadrature rule	Order	Cost per el.	Time step	Total time (s)
Center over-int	2	$\rightarrow 1$	1.6x	726
$S_0^2 \otimes S_0^2$ over-int	2	$\rightarrow 1$	1.6x	742
Center-edge	2	$\rightarrow 3$	1.6x	1088
$S_0^3 \otimes S_0^3$ optimal	2	$\rightarrow 2.25$	1.6x	912
Gauss 2×2	2	$= 4$	1.6x	1279
$S_0^4 \otimes S_0^4$ optimal	2	$\rightarrow 4$	1.6x	1304
$S_1^3 \otimes S_1^3$ over-int	3	$\rightarrow 1$	1.8x	710
$S_1^5 \otimes S_1^5$ optimal	3	$\rightarrow 4$	1.8x	1288

though unclamped knot vectors are used. In this case, the interpolation functions between the boundary elements and the interior elements do not differ in any way. However, the control points located at the boundaries are associated with smaller masses than the interior control points.

Another important observation is that the instabilities occurring after removing the boundary conditions (if the time step size is not adjusted appropriately) originate from the corners of the patch where the control points with smallest masses are located. This time step penalty can be circumvented by selectively scaling the masses of the affected control points or by increasing the size of the boundary Bézier elements, although both methods reduce the accuracy of the solution.

B.5 Conclusions

In many applications, the motivation to use reduced integration is not only to reduce the computational cost, but also to alleviate numerical locking. However, reduced integration often leads to rank-deficient stiffness matrices and hourglassing. Some recent papers have suggested that the hourglass problem can be controlled by stabilizing only small part of Bézier elements within the patch (Adam et al., 2015a; Bouclier et al., 2015a). Although this approach guarantees full rank of the global stiffness matrix, the only resistance against the hourglass modes comes from the stabilized Bézier elements, and therefore, the other elements are still prone to hourglassing. In fact, this behavior is not any different from the classical finite elements where the global spurious zero-energy modes can also be suppressed by stabilizing only a small part of the elements, but that does not guarantee nonexistence of spurious finite energy modes. In the context of IGA, this problem was recognized

for the Center-vertex quadrature by Schillinger et al. (2014). Herein further evidence is provided that the patchwise quadratures are not an exception, and stabilizing only a small proportion of Bézier elements is generally not sufficient to suppress hourglass modes.

In the context of biquadratic splines, the $S_0^2 \otimes S_0^2$ scheme provides a locking-free low-cost option while maintaining the optimal convergence rate. However, the accuracy of this scheme is worse than the accuracy provided by the Gauss 2×2 scheme which was reported to provide accuracy comparable to the full integration in numerical tests by Schillinger et al. (2014). In the incremental sheet forming application studied in this paper, the $S_0^2 \otimes S_0^2$ scheme produces a somewhat rough solution, which is suspected to originate from the lower accuracy of the quadrature, but the possibility of hourglassing playing a role is not ruled out. The Gauss 2×2 scheme provides good accuracy while alleviating locking to some extent, but is not fully locking-free. Furthermore, the Gauss 2×2 scheme offers no significant reduction in the computational cost in comparison to full integration. It is further noted that the Gauss-Lobatto scheme investigated for biquadratic splines by Schillinger et al. (2014) produces almost identical results with the Center-edge scheme for the benchmark problems investigated in this paper.

A new problem associated with irregular quadrature rules such as the $S_0^3 \otimes S_0^3$ scheme proposed by Hiemstra et al. (2017) is discovered. This problem exhibits a wavy pattern in the incremental sheet forming example. The problem may also concern the schemes for which the integration points are positioned unsymmetrically with respect to each Bézier element (e.g., the $S_0^2 \otimes S_0^2$ and $S_0^4 \otimes S_0^4$ schemes), as these schemes require a *remainder row* of integration points for both parametric directions, i.e., at least one row of Bézier elements in both parametric directions must have more integration points in comparison with the other rows. The positioning of the remainder rows can be adjusted freely to minimize the potential problem. However, if the remainder rows are not positioned symmetrically, the result may become unsymmetric even if a symmetric problem is analyzed.

The over-integrated low-cost $S_1^3 \otimes S_1^3$ scheme proposed for bicubic splines by Adam et al. (2015c) suffers from severe hourglassing even with over-integrated boundary elements and is hence not recommended without any form of additional hourglass stabilization. The $S_1^5 \otimes S_1^5$ scheme proposed by Hiemstra et al. (2017) provides good accuracy, maintains the optimal convergence rate for bicubic splines, and

suffers from only slight locking. Due to the larger maximum stable time step size and favorable placement of the integration points, the bicubic splines do not require longer computational times for the investigated problems in comparison with biquadratic splines and are, therefore highly recommended.

The following publication has been incorporated as Appendix C.

Hokkanen, J., Pedroso, D. M., Elford, M. C., Stephan, A. J. E., and Zhang, Y. (2018). Efficient isogeometric shell element with through-thickness stretch: application to incremental sheet forming. *Journal of Physics: Conference Series*, 1063(1).

Contributor	Statement of contribution	%
Jaro Hokkanen	writing of text	100
	proof-reading	60
	theoretical derivations	60
	computer implementation	80
	numerical calculations	100
	experimental results	20
	preparation of figures	80
Dorival M. Pedroso	proof-reading	10
	theoretical derivations	10
	supervision, guidance	80
Michael C. Elford	proof-reading	10
	theoretical derivations	10
	computer implementation	10
	experimental results	40
	preparation of figures	10
	supervision, guidance	20
Andrew J. E. Stephan	proof-reading	10
	theoretical derivations	10
	experimental results	40
	preparation of figures	10
Yunpeng Zhang	proof-reading	10
	theoretical derivations	10
	computer implementation	10

Appendix C

Efficient isogeometric shell element with through-thickness stretch: application to incremental sheet forming

An isogeometric shell element with through-thickness stretch is applied to a two-point incremental forming problem. The shell element supports full three-dimensional constitutive laws and therefore does not make the plane stress assumption. An anisotropic material model is implemented to account for the sheet rolling direction. Automatically adjusting penalty stiffness is proposed for modeling the contact between the stylus tool and the sheet, whereas the die contact algorithm uses traditional constant penalty stiffness. A comparison is made between experimental results as well as results from a conventional shell formulation.

C.1 Introduction

Highly nonlinear problems such as automobile crash dynamics or sheet metal forming simulations often utilize the four-node, bilinearly interpolated Reissner-Mindlin shell elements combined with explicit time integration. These Reissner-Mindlin elements imposing the plane stress condition have dominated the finite element shell analysis for decades. The use of bilinear interpolation functions has resulted in efficient formulations and larger stable time step sizes compared to higher-order

Lagrange shell elements. However, the interest in more advanced formulations such as thickness stretchable shell elements and solid-shells has increased as the plane stress assumption is not always acceptable, particularly if thick plates or problems involving double-sided contact are considered. Many recent contributions show a great interest in using solid-shell elements in explicit dynamic simulations. However, the computational speed has yet to be improved for demanding problems such as incremental sheet forming.

Isogeometric analysis (IGA) first proposed by Hughes et al. (2005) has recently attracted growing interest among the computational mechanics community. This concept adopts spline-based interpolation functions resulting in several potential advantages over conventional finite elements, such as increased accuracy per degree of freedom, increased stable time step size, and continuous stress and strain fields, among others (Cottrell et al., 2006, 2007; Evans et al., 2009; Großmann et al., 2012; Adam et al., 2015b). Isogeometric solid-shell elements have been proposed by Hosseini et al. (2013), Bouclier et al. (2013, 2015a), and Caseiro et al. (2015). However, most of the isogeometric solid-shell element formulations proposed in the literature do not consider dynamic problems nor the implications of using explicit time integration.

The isogeometric shell element introduced by Hokkanen and Pedroso (2019a) supports full three-dimensional constitutive laws and is capable of simulating geometrically nonlinear dynamic problems. The purpose of this paper is to evaluate the suitability of the aforementioned shell element to incremental sheet forming (ISF) using explicit time integration.

C.2 ISF simulation

C.2.1 Shell formulation

The implemented formulation is based on the degenerated solid approach with only displacement degrees of freedom. Reduced integration is used to alleviate several locking problems and to reduce computational cost. The reduced Gauss 2×2 quadrature preserves full rank of the stiffness matrix and therefore no ad hoc hourglass control procedures are required. The fiber mass scaling prevents the eigenfrequencies related to the transverse deformations from lowering the larger stable time step size associated with the spline-based in-plane interpolation. The

implementation relies on CUDA and most computations are performed parallel on a modern consumer level GPU. A detailed description of the shell element is given by Hokkanen and Pedroso (2019a). The BWC shell element (Belytschko et al., 1992) based on the plane stress assumption is used as reference.

C.2.2 Material model

The simulation assumes hypoelastic-plastic material behavior. The material model is based on Yld2004-18p yield criterion proposed by Barlat et al. (2005) to consider the anisotropy introduced by the process of rolling the sheet. The required parameters describing the constitutive relations for aluminum 7075-O are the density $\rho = 2.81 \times 10^{-9}\,\frac{\mathrm{Ns}^2}{\mathrm{mm}^4}$, Young's modulus $E = 71.7 \times 10^3\,\mathrm{MPa}$, Poisson's ratio $\nu = 0.33$, and the initial yield stress $\sigma_y = 261\,\mathrm{MPa}$. The hardening coefficient and the hardening exponent required by the Swift type hardening law are given as $C_{sw} = 333\,\mathrm{MPa}$ and $n_{sw} = 0.16$, respectively. Moreover, the 18 parameters required by Yld2004-18p yield criterion are given by Esmaeilpour et al. (2017). The return mapping uses the explicit closest point projection method (Grilo et al., 2015).

C.2.3 Contact

The isogeometric formulation uses a nonmortar Gauss-point-to-surface (GPTS) penalty contact algorithm (Temizer et al., 2011; De Lorenzis et al., 2014) for the stylus contact. The linear pressure-overclosure relationship is given as

$$p = \varepsilon(g + s) \tag{C.1}$$

where ε is the penalty stiffness, g is the overclosure and s is the shift which causes the contact pressure to start increasing slightly before the actual contact occurs. The penalty stiffness ε (i.e., the slope of the pressure-overclosure relationship) adjusts automatically such that zero maximum penetration is maintained (i.e., $g \lesssim 0$). However, a minimum value for ε is specified. The described contact formulation is found robust for the stylus contact in many ISF simulations as the contact between the sheet and the spherical-headed tool comprises mostly of a single contact region. The die contact algorithm uses nonmortar Gauss-point-to-surface (GPTS) penalty contact, a linear pressure-overclosure relationship, and traditional constant penalty stiffness.

Moreover, the traditional BWC shell formulation uses a simple node-to-surface (NTS) contact formulation and traditional constant penalty stiffness for the stylus and the die.

C.2.4 Results

The die geometry and the toolpath for the ISF simulation are given in Figures C.1 and C.2, respectively. The square sheet has the thickness of 0.063 inches and the edge length of 290 mm. The tool has a hemispherical head, 20 mm in diameter, which makes sliding contact with the sheet. The stylus tool proceeds in the counterclockwise direction and steps down 2 mm after each full revolution. The assumed tool speed is 4000 mm/min. Mass scaling of 4000 is applied to increase the stable time step size and to speed up the simulation. The control point (or node) spacing is 2 mm.

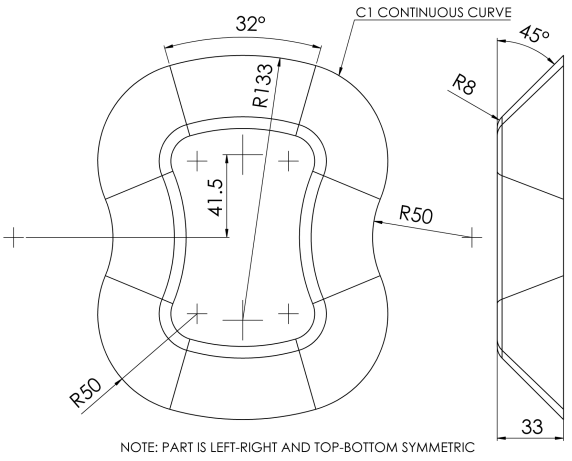


Figure C.1: The dimensions of the die.

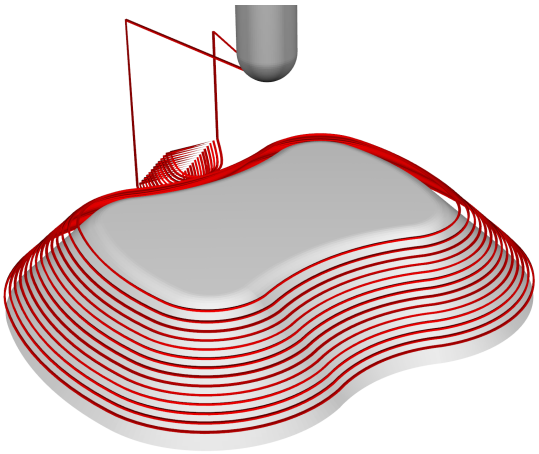


Figure C.2: The toolpath.

The formed part is shown in Figure C.3. The smooth results from the isogeometric solver are given in Figures C.5 and C.7. The simulation accounts for the unclamping step, i.e., the release of the clamps after the forming process has finished. After the unclamping step, the springback effect bends the edges of the sheet slightly upwards. The edges perpendicular to the longer sides of the die become more curved than the other edges. The springback predicted by the simulation matches very well with the experimental results. Furthermore, Figure C.4 shows the actual thickness determined by 3D scanning both surfaces and aligning them to produce the thickness contours. The simulated thinning shown in Figure C.7 is in a good agreement with the experimental results (same scales). The results from the bilinearly interpolated BWC shell element that is based on the plane stress assumption are given as a reference in Figures C.6 and C.8. The solution is rough in comparison with IGA as

the displacement field is now only C^0 continuous. The thickness plot is even rougher as the thickness is only available at the integration points. Nevertheless, the results are still good in comparison with the experimental results.

Unfortunately, both element types overestimate the averaged tool forces by ∼50%. However, this is a common problem which has been associated with many FEM formulations in the past (Elford et al., 2013). Furthermore, the isogeometric shell formulation overestimates these forces slightly more than the traditional BWC element. The overestimation is suspected to originate from the material model (mostly due to lack of a damage model), but also the contact formulation and element locking may play a role. Further investigation is required.

Figure C.3: A photograph of the formed part.

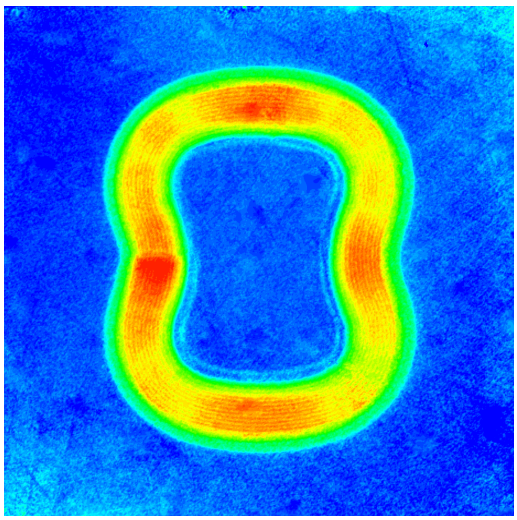


Figure C.4: The 3D scanned normalized thickness. The contours vary linearly from 0.66 to 1.0.

In the isogeometric analysis, the boundary Bézier elements pose stricter requirements for the stable time step size compared to the interior elements as shown by Adam et al. (2015b). One way to overcome this limitation is to increase the boundary element size, which, indeed, may be acceptable for many ISF simulations. However, it is noticed that preserving the uniform control point spacing, but simply clamping all the edges also effectively prevents instabilities originating from the boundaries. The implemented explicit isogeometric shell formulation is found visually stable with up to ∼50% larger maximum time step size compared to the traditional BWC shell. This increase in the time step size makes close to no difference to the final results. However, it is found that the large time step size cannot be preserved if all edges are not clamped (with uniform control point spacing). Therefore, the time step size must be reduced right before the unclamping step at

the end of the simulation.

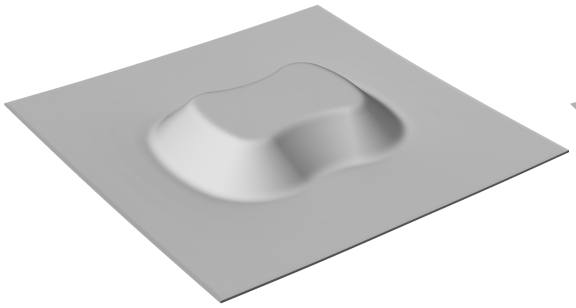


Figure C.5: The simulation result of the isogeometric shell formulation.

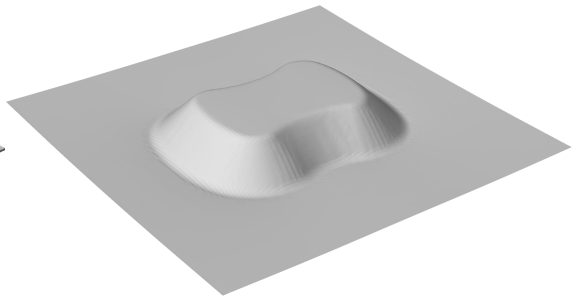


Figure C.6: The simulation result of BWC shell formulation.

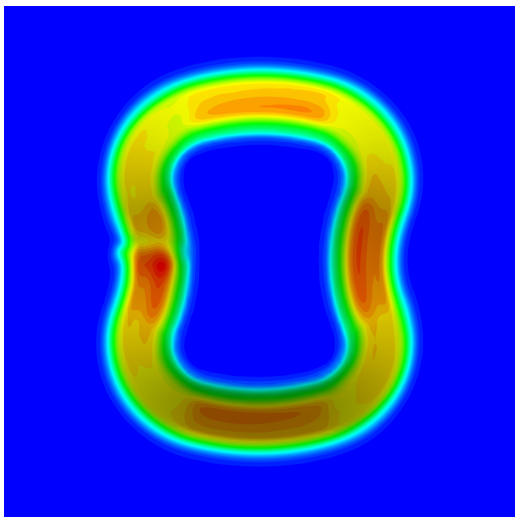


Figure C.7: The normalized thickness plot of the isogeometric shell formulation. The contours vary linearly from 0.66 to 1.0.

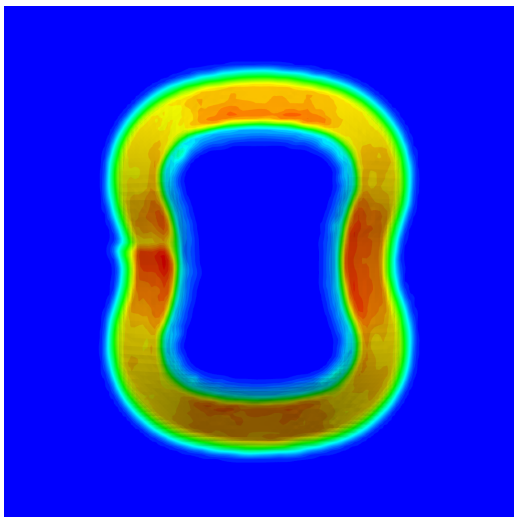


Figure C.8: The normalized thickness plot of BWC shell formulation. The contours vary linearly from 0.66 to 1.0.

C.3 Conclusions

The isogeometric shell formulation proposed by Hokkanen and Pedroso (2019a) is applied to a challenging ISF problem. A comprehensive anisotropic material model is implemented to take into account the sheet rolling direction. As the chosen problem requires a die, double-sided contact must be considered. The automatically adjusting penalty stiffness is proposed to model the contact between the stylus tool and the sheet, whereas the die contact algorithm uses traditional constant penalty stiffness.

Qualitatively good results are obtained by the isogeometric shell as well as the traditional BWC shell formulation. The displacements, springback effects, and

thinning agree very well with the experimental results. However, the forming forces are significantly overestimated by both shell types which requires further investigation. The isogeometric shell formulation is found very competitive even against low-order reduced integration shell elements in speed due to the increased accuracy per degree of freedom and larger stable time step size.

Bibliography

Adam, C., Bouabdallah, S., Zarroug, M., and Maitournam, H. (2014). Improved numerical integration for locking treatment in isogeometric structural elements, Part I: Beams. *Computer Methods in Applied Mechanics and Engineering*, 279:1–28.

Adam, C., Bouabdallah, S., Zarroug, M., and Maitournam, H. (2015a). Improved numerical integration for locking treatment in isogeometric structural elements. Part II: Plates and shells. *Computer Methods in Applied Mechanics and Engineering*, 284:106–137.

Adam, C., Bouabdallah, S., Zarroug, M., and Maitournam, H. (2015b). Stable time step estimates for NURBS-based explicit dynamics. *Computer Methods in Applied Mechanics and Engineering*, 295:581–605.

Adam, C., Hughes, T. J. R., Bouabdallah, S., Zarroug, M., and Maitournam, H. (2015c). Selective and reduced numerical integrations for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 284:732–761.

Ait-Haddou, R., Barto, M., and Calo, V. M. (2015). Explicit Gaussian quadrature rules for C1 cubic splines with symmetrically stretched knot sequences. *Journal of Computational and Applied Mathematics*, 290:543–552.

Altan, T. and Tekkaya, A. E. (2012). *Sheet Metal Forming: Processes and Applications*. ASM International.

Ambati, M., Kiendl, J., and De Lorenzis, L. (2018). Isogeometric Kirchhoff-Love shell formulation for elasto-plasticity. *Computer Methods in Applied Mechanics and Engineering*, 340:320–339.

Auricchio, F., Beirão da Veiga, L., Hughes, T. J. R., Reali, A., and Sangalli, G. (2010). Isogeometric collocation methods. *Mathematical Models and Methods in Applied Sciences*, 20(11):2075–2107.

Auricchio, F., Beirão da Veiga, L., Hughes, T. J. R., Reali, A., and Sangalli, G. (2012a). Isogeometric collocation for elastostatics and explicit dynamics. *Computer Methods in Applied Mechanics and Engineering*, 249-252:2–14.

Auricchio, F., Calabrò, F., Hughes, T., and Reali, A. (2012b). A simple algorithm for obtaining nearly optimal quadrature rules for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 249-252:15–27.

Bambach, M. (2008). *Process strategies and modelling approaches for asymmetric incremental sheet forming.* Umformtechnische Schriften. Shaker.

Barlat, F., Aretz, H., Yoon, J., Karabin, M., Brem, J., and Dick, R. (2005). Linear transfomation-based anisotropic yield functions. *International Journal of Plasticity*, 21(5):1009–1039.

Barto, M., Ait-Haddou, R., and Calo, V. M. (2017). Gaussian quadrature rules for C1 quintic splines with uniform knot vectors. *Journal of Computational and Applied Mathematics*, 322:57–70.

Barto, M. and Calo, V. M. (2016). Gaussian quadrature for splines via homotopy continuation: Rules for C2 cubic splines. *Journal of Computational and Applied Mathematics*, 296:709–723.

Bazilevs, Y., Calo, V., Cottrell, J., Evans, J., Hughes, T., Lipton, S., Scott, M., and Sederberg, T. (2010). Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5):229–263.

Bazant, Z. P. and Cedolin, L. (2010). *Stability of Structures: Elastic, Inelastic, Fracture, and Damage theories.* Dover Publications.

Belgacem, F. (2000). Numerical Simulation of Some Variational Inequalities Arisen from Unilateral Contact Problems by the Finite Element Methods. *SIAM Journal on Numerical Analysis*, 37(4):1198–1216.

Belgacem, F., Hild, P., and Laborde, P. (1998). The mortar finite element method for contact problems. *Mathematical and Computer Modelling*, 28(4-8):263–271.

Belytschko, T., Lin, J., and Tsay, C.-S. (1984). Explicit algorithms for the nonlinear dynamics of shells. *Computer Methods in Applied Mechanics and Engineering*, 42(2):225–251.

Belytschko, T., Liu, W. K., Moran, B., and Elkhodary, K. (2013). *Nonlinear finite elements for continua and structures*. John Wiley & Sons.

Belytschko, T., Wong, B., and Chiang, H. (1992). Advances in one-point quadrature shell elements. *Computer Methods in Applied Mechanics and Engineering*, 96(1):93–107.

Belytschko, T., Wong, B., and Stolarski, H. (1989). Assumed strain stabilization procedure for the 9-node Lagrange shell element. *International Journal for Numerical Methods in Engineering*, 28(2):385–414.

Benson, D., Bazilevs, Y., Hsu, M.-C., and Hughes, T. (2011). A large deformation, rotation-free, isogeometric shell. *Computer Methods in Applied Mechanics and Engineering*, 200(13-16):1367–1378.

Benson, D. J., Bazilevs, Y., Hsu, M.-C., and Hughes, T. J. R. (2010). Isogeometric shell analysis: the Reissner-Mindlin shell. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):276–289.

Benson, D. J., Hartmann, S., Bazilevs, Y., Hsu, M. C., and Hughes, T. J. R. (2013). Blended isogeometric shells. *Computer Methods in Applied Mechanics and Engineering*, 255:133–146.

Bernstein, S. (1912). Demonstration du th'eoreme de Weierstrass fondee sur le calcul des probabilites. *Communications de la Societe mathematique de Kharkow*, 13(1):1–2.

Bouclier, R., Elguedj, T., and Combescure, A. (2013). Efficient isogeometric NURBS-based solid-shell elements: Mixed formulation and B-method. *Computer Methods in Applied Mechanics and Engineering*, 267:86–110.

Bouclier, R., Elguedj, T., and Combescure, A. (2015a). An isogeometric locking-free NURBS-based solid-shell element for geometrically nonlinear analysis. *International Journal for Numerical Methods in Engineering*, 101(10):774–808.

Bouclier, R., Elguedj, T., and Combescure, A. (2015b). Development of a mixed displacement-stress formulation for the analysis of elastoplastic structures under

small strains: Application to a locking-free, NURBS-based solid-shell element. *Comput. Methods Appl. Mech. Engrg*, 295:543–561.

Bruhns, O., Xiao, H., and Meyers, A. (1999). Self-consistent Eulerian rate type elasto-plasticity models based upon the logarithmic stress rate. *International Journal of Plasticity*, 15(5):479–520.

Buechter, N. and Ramm, E. (1992). Shell theory versus degeneration—a comparison in large rotation finite element analysis. *International Journal for Numerical Methods in Engineering*, 34(1):39–59.

Buffa, A., Cho, D., and Sangalli, G. (2010). Linear independence of the T-spline blending functions associated with some particular T-meshes. *Computer Methods in Applied Mechanics and Engineering*, 199(23):1437–1445.

Cardoso, R. P. and Yoon, J. W. (2005). One point quadrature shell element with through-thickness stretch. *Computer Methods in Applied Mechanics and Engineering*, 194(9):1161–1199.

Cardoso, R. P. R. and Cesar de Sa, J. M. A. (2014). Blending moving least squares techniques with NURBS basis functions for nonlinear isogeometric analysis. *Computational Mechanics*, 53(6):1327–1340.

Caseiro, J., Valente, R., Reali, A., Kiendl, J., Auricchio, F., and Alves de Sousa, R. (2015). Assumed Natural Strain NURBS-based solid-shell element for the analysis of large deformation elasto-plastic thin-shell structures. *Computer Methods in Applied Mechanics and Engineering*, 284:861–880.

Catmull, E. and Clark, J. (1978). Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-Aided design*, 10(6):350–355.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2001). *Introduction to algorithms*. The MIT Press.

Cottrell, J., Hughes, T., and Reali, A. (2007). Studies of refinement and continuity in isogeometric structural analysis. *Computer Methods in Applied Mechanics and Engineering*, 196(41-44):4160–4183.

Cottrell, J., Reali, A., Bazilevs, Y., and Hughes, T. (2006). Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering*, 195(41-43):5257–5296.

Cottrell, J. A., Hughes, T. J. R., and Bazilevs, Y. (2009). *Isogeometric Analysis: Toward Integration of CAD and FEA*. John Wiley & Sons, Ltd, Chichester, UK.

Courant, R., Friedrichs, K., and Lewy, H. (1928). Über die partiellen Differenzengleichungen der mathematischen Physik. *Mathematische Annalen*, 100(1):32–74.

Cox, M. G. (1972). The numerical evaluation of B-splines. *IMA Journal of Applied Mathematics*, 10(2):134–149.

Curry, H. and Schoenberg, I. (1947). On spline distributions and their limits: the Polya distribution functions. *Bulletin of the American Mathematical Society*, 53:1114–1114.

de Boor, C. (1972). On Calculating with B-splines. *Journal of Approximation Theory*, 6(1):50–62.

de Boor, C. (1978). *A practical guide to splines*. Springer-Verlag New York.

de Borst, R., Crisfield, M. A., Remmers, J. J., and Verhoosel, C. V. (2012). *Nonlinear Finite Element Analysis of Solids and Structures*. John Wiley & Sons, Ltd, Chichester, UK.

De Lorenzis, L., Evans, J. A., Hughes, T. J. R., and Reali, A. (2015). Isogeometric collocation: Neumann boundary conditions and contact. *Computer Methods in Applied Mechanics and Engineering*, 284:21–54.

De Lorenzis, L., Wriggers, P., and Hughes, T. J. R. (2014). Isogeometric contact: A review. *GAMM Mitteilungen*, 37(1):85–123.

de Souza Neto, E. A., Peric, D., and Owen, D. R. J. (2011). *Computational Methods for Plasticity: Theory and Applications*. John Wiley & Sons.

Dhondt, G. D. C. (2004). *The finite element method for three-dimensional thermomechanical applications*. Wiley.

Dimitri, R., De Lorenzis, L., Scott, M., Wriggers, P., Taylor, R., and Zavarise, G. (2014). Isogeometric large deformation frictionless contact using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 269:394–414.

Dornisch, W., Klinkel, S., and Simeon, B. (2013). Isogeometric Reissner-Mindlin shell analysis with exactly calculated director vectors. *Computer Methods in Applied Mechanics and Engineering*, 253:491–504.

Echter, R., Oesterle, B., and Bischoff, M. (2013). A hierarchic family of isogeometric shell finite elements. *Computer Methods in Applied Mechanics and Engineering*, 254:170–180.

El-Abbasi, N. and Meguid, S. (2000). A new shell element accounting for through-thickness deformation. *Computer Methods in Applied Mechanics and Engineering*, 189(3):841–862.

Elford, M., Saha, P., Seong, D., Haque, M. Z., and Yoon, J. W. (2013). Benchmark 3 - Incremental sheet forming. In *AIP Conference Proceedings*, volume 1567, pages 227–261. American Institute of Physics.

Elford, M. C., Hokkanen, J., and Stephan, A. J. E. (2018). A Software Package for Toolpath Generation and Process Simulation of Incremental Sheet Forming. *Journal of Physics: Conference Series*, 1063(1).

Esmaeilpour, R., Kim, H., Park, T., Pourboghrat, F., and Mohammed, B. (2017). Comparison of 3D yield functions for finite element simulation of single point incremental forming (SPIF) of aluminum 7075. *International Journal of Mechanical Sciences*, 133:544–554.

Evans, J. A., Bazilevs, Y., Babuška, I., and Hughes, T. J. (2009). n-Widths, supinfs, and optimality ratios for the k-version of the isogeometric finite element method. *Computer Methods in Applied Mechanics and Engineering*, 198(21-26):1726–1741.

Farouki, R. T. (2012). The Bernstein polynomial basis: A centennial retrospective. *Computer Aided Geometric Design*, 29(6):379–419.

Fischer, K. A. and Wriggers, P. (2005). Frictionless 2D contact formulations for finite deformations based on the mortar method. *Computational Mechanics*, 36(3):226–244.

Fischer, K. A. and Wriggers, P. (2006). Mortar based frictional contact formulation for higher order interpolations using the moving friction cone. *Computer Methods in Applied Mechanics and Engineering*, 195(37-40):5020–5036.

Franzen, V., Kwiatkowski, L., Sebastiani, G., Shankar, R., Tekkaya, A. E., and Kleiner, M. (2008). Dyna-Die: Towards full kinematic incremental forming. *International Journal of Material Forming*, 1:1163–1166.

Freed, A. (2014). Hencky strain and logarithmic rates in Lagrangian analysis. *International Journal of Engineering Science*, 81:135–145.

Frisch-Fay, R. (1962). *Flexible bars*. Butterworths.

Gee, M., Ramm, E., and Wall, W. A. (2005). Parallel multilevel solution of nonlinear shell structures. *Computer Methods in Applied Mechanics and Engineering*, 194(21-24):2513–2533.

Grilo, T. J., Valente, R. A. F., and Alves de Sousa, R. J. (2015). Modelling non-quadratic anisotropic yield criteria and mixed isotropic-nonlinear kinematic hardening: analysis of forward- and backward-Euler formulations. *International Journal of Material Forming*, 8(4):533–547.

Großmann, D., Jüttler, B., Schlusnus, H., Barner, J., and Vuong, A.-V. (2012). Isogeometric simulation of turbine blades for aircraft engines. *Computer Aided Geometric Design*, 29(7):519–531.

Hallquist, J. O. (2006). LS-DYNA theory manual. *Livermore software Technology corporation*.

Hartmann, S. and Benson, D. (2014). Mass scaling and stable time step estimates for isogeometric analysis. *International Journal for Numerical Methods in Engineering*, 102(3-4):671–687.

Hetnarski, R. B. and Eslami, M. R. (2009). *Thermal Stresses Advanced Theory and Applications*, volume 158. Springer.

Hiemstra, R. R., Calabrò, F., Schillinger, D., and Hughes, T. J. (2017). Optimal and reduced quadrature rules for tensor product and hierarchically refined splines in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 316:966–1004.

Hild, P. (2000). Numerical implementation of two nonconforming finite element methods for unilateral contact. *Computer Methods in Applied Mechanics and Engineering*, 184(1):99–123.

Hillman, M., Chen, J., and Bazilevs, Y. (2015). Variationally consistent domain integration for isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 284:521–540.

Hirt, G., Ames, J., Bambach, M., Kopp, R., and Kopp, R. (2004). Forming strategies and Process Modelling for CNC Incremental Sheet Forming. *CIRP Annals - Manufacturing Technology*, 53(1):203–206.

Hokkanen, J. and Pedroso, D. M. (2019a). Isogeometric thickness stretchable shell: Efficient formulation for nonlinear dynamic problems. *International Journal for Numerical Methods in Engineering*, 119(2):105–127.

Hokkanen, J. and Pedroso, D. M. (2019b). Quadrature rules for isogeometric shell formulations: study using a real-world application about metal forming. *[Submitted to Computer Methods in Applied Mechanics and Engineering]*.

Hokkanen, J., Pedroso, D. M., Elford, M. C., Stephan, A. J. E., and Zhang, Y. (2018). Efficient isogeometric shell element with through-thickness stretch: application to incremental sheet forming. *Journal of Physics: Conference Series*, 1063(1).

Hosseini, S., Remmers, J. J., Verhoosel, C. V., and de Borst, R. (2013). An isogeometric solid-like shell element for nonlinear analysis. *International Journal for Numerical Methods in Engineering*, 95(3):238–256.

Hosseini, S., Remmers, J. J., Verhoosel, C. V., and De Borst, R. (2014). An isogeometric continuum shell element for non-linear analysis. *Comput. Methods Appl. Mech. Engrg.*, 271:1–22.

Hosseini, S., Remmers, J. J., Verhoosel, C. V., and de Borst, R. (2015). Propagation of delamination in composite materials with isogeometric continuum shell elements. *International Journal for Numerical Methods in Engineering*, 102(3-4):159–179.

Hsu, M.-C., Kamensky, D., Xu, F., Kiendl, J., Wang, C., Wu, M. C. H., Mineroff, J., Reali, A., Bazilevs, Y., and Sacks, M. S. (2015). Dynamic and fluid-structure interaction simulations of bioprosthetic heart valves using parametric design with T-splines and Fung-type material models. *Computational Mechanics*, 55(6):1211–1225.

Hughes, T. and Liu, W. (1981). Nonlinear finite element analysis of shells: Part I. Three-dimensional shells. *Computer Methods in Applied Mechanics and Engineering*, 26(3):331–362.

Hughes, T. J. R. (2012). *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Courier Corporation.

Hughes, T. J. R., Cottrell, J. A., and Bazilevs, Y. (2005). Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194(39-41):4135–4195.

Hughes, T. J. R., Reali, A., and Sangalli, G. (2010). Efficient quadrature for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):301–313.

Jeswiet, J., Micari, F., Hirt, G., Bramley, A., Duflou, J., and Allwood, J. (2005). Asymmetric Single Point Incremental Forming of Sheet Metal. *CIRP Annals - Manufacturing Technology*, 54(2):88–114.

Jiao, Y. and Fish, J. (2017). Is an additive decomposition of a rate of deformation and objective stress rates passé? *Computer Methods in Applied Mechanics and Engineering*, 327:196–225.

Jiao, Y. and Fish, J. (2018). On the equivalence between the multiplicative hyper-elasto-plasticity and the additive hypo-elasto-plasticity based on the modified kinetic logarithmic stress rate. *Computer Methods in Applied Mechanics and Engineering*, 340:824–863.

Johannessen, K. A. (2017). Optimal quadrature for univariate and tensor product splines. *Computer Methods in Applied Mechanics and Engineering*, 316:84–99.

Kiendl, J., Auricchio, F., Beirão da Veiga, L., Lovadina, C., and Reali, A. (2015a). Isogeometric collocation methods for the ReissnerMindlin plate problem. *Computer Methods in Applied Mechanics and Engineering*, 284:489–507.

Kiendl, J., Bletzinger, K.-U., Linhard, J., and Wüchner, R. (2009). Isogeometric shell analysis with Kirchhoff-Love elements. *Computer Methods in Applied Mechanics and Engineering*, 198(49-52):3902–3914.

Kiendl, J., Hsu, M.-C., Wu, M. C., and Reali, A. (2015b). Isogeometric Kirchhoff-Love shell formulations for general hyperelastic materials. *Computer Methods in Applied Mechanics and Engineering*, 291:280–303.

Klöppel, T., Gee, M., and Wall, W. A. (2011). A scaled thickness conditioning for solid- and solid-shell discretizations of thin-walled structures. *Computer Methods in Applied Mechanics and Engineering*, 200(9-12):1301–1310.

Koric, S., Hibbeler, L. C., and Thomas, B. G. (2009). Explicit coupled thermo-mechanical finite element model of steel solidification. *International Journal for Numerical Methods in Engineering*, 78(1):1–31.

Krieg, R. D. and Key, S. W. (1976). Implementation of a time independent plasticity theory into structural computer programs. *Constitutive equations in viscoplasticity: Computational and engineering aspects*, pages 125–137.

Kruse, R., Nguyen-Thanh, N., De Lorenzis, L., and Hughes, T. (2015). Isogeometric collocation for large deformation elasticity and frictional contact problems. *Computer Methods in Applied Mechanics and Engineering*, 296:73–112.

Lehmann, T., Guo, Z., and Liang, H. (1991). Conjugacy between Cauchy stress and logarithm of the left stretch tensor. *Eur. J. Mech. A/Solids*, 10:395–404.

Leszak, E. (1967). Apparatus and process for incremental dieless forming. U.S. Patent: 3,342,051.

Lewis, R., Nithiarasu, P., and Seetharamu, K. (2004). *Fundamentals of the Finite Element Method for Heat and Fluid flow*. Wiley.

Li, L., Li, D., and Peng, Y. (2011a). An Explicit Formulation of Solid-Shell Element and Its Application in Sheet Metal Forming Processes. *Advanced Science Letters*, 4(4):1761–1766.

Li, L., Peng, Y., and Li, D. (2011b). A stabilized underintegrated enhanced assumed strain solid-shell element for geometrically nonlinear plate/shell analysis. *Finite Elements in Analysis and Design*, 47(5):511–518.

Li, X., Zheng, J., Sederberg, T. W., Hughes, T. J., and Scott, M. A. (2012). On linear independence of T-spline blending functions. *Computer Aided Geometric Design*, 29(1):63–76.

Liu, L., Zhang, Y. J., and Wei, X. (2015a). Handling Extraordinary Nodes with Weighted T-spline Basis Functions. *Procedia Engineering*, 124:161–173.

Liu, L., Zhang, Y. J., and Wei, X. (2015b). Weighted T-splines with application in reparameterizing trimmed NURBS surfaces. *Computer Methods in Applied Mechanics and Engineering*, 295:108–126.

Lu, J. (2011). Isogeometric contact analysis: Geometric basis and formulation for frictionless contact. *Computer Methods in Applied Mechanics and Engineering*, 200(5-8):726–741.

Mattern, S., Schmied, C., and Schweizerhof, K. (2015). Highly efficient solid and solid-shell finite elements with mixed strain-displacement assumptions specifically

set up for explicit dynamic simulations using symbolic programming. *Computers & Structures*, 154:210–225.

Matzen, M., Cichosz, T., and Bischoff, M. (2013). A point to segment contact formulation for isogeometric, NURBS based finite elements. *Computer Methods in Applied Mechanics and Engineering*, 255:27–39.

McDevitt, T. W. and Laursen, T. A. (2000). A mortar-finite element formulation for frictional contact problems. *International Journal for Numerical Methods in Engineering*, 48(10):1525–1547.

Naumenko, K. and Altenbach, H. (2018). Angular Velocities, Twirls, Spins and Rotation Tensors in the Continuum Mechanics Revisited. In *Generalized Models and Non-classical Approaches in Complex Materials 1*, pages 621–632. Springer, Cham.

Newmark, N. M. (1959). A Method of Computation for Structural Dynamics. *Journal of the Engineering Mechanics Division*, 85(3):67–94.

Nicholson, D. W. (2008). *Finite element analysis: thermomechanics of solids*. CRC Press.

Oden, J. T. and Kikuchi, N. (1982). Finite element methods for constrained problems in elasticity. *International Journal for Numerical Methods in Engineering*, 18(5):701–725.

Oesterle, B., Ramm, E., and Bischoff, M. (2016). A shear deformable, rotation-free isogeometric shell formulation. *Computer Methods in Applied Mechanics and Engineering*, 307:235–255.

Oesterle, B., Sachse, R., Ramm, E., and Bischoff, M. (2017). Hierarchic isogeometric large rotation shell elements including linearized transverse shear parametrization. *Computer Methods in Applied Mechanics and Engineering*, 321:383–405.

Pagani, M., Reese, S., and Perego, U. (2014). Computationally efficient explicit nonlinear analyses using reduced integration-based solid-shell finite elements. *Computer Methods in Applied Mechanics and Engineering*, 268:141–159.

Parisch, H. (1995). A continuum-based shell theory for non-linear applications. *International Journal for Numerical Methods in Engineering*, 38(11):1855–1883.

Piegl, L. and Tiller, W. (1996). *The NURBS Book*. Springer.

Prager, W. (1961). An elementary discussion of definitions of stress rate.

Ramshaw, L. (1987). *Blossoming: A Connect-the-Dots Approach to Splines*. Digital Systems Research Center Palo Alto.

Reinhardt, W. and Dubey, R. (1995). Eulerian strain-rate as a rate of logarithmic strain. *Mechanics Research Communications*, 22(2):165–170.

Reinhardt, W. D. and Dubey, R. N. (1996). Coordinate-independent representation of spins in continuum mechanics. *Journal of Elasticity*, 42(2):133–144.

Riffnaller-Schiefer, A., Augsdörfer, U., and Fellner, D. (2016). Isogeometric shell analysis with NURBS compatible subdivision surfaces. *Applied Mathematics and Computation*, 272:139–147.

Rodrigues, O. (1840). Des lois géométriques qui régissent les déplacements d'un système solide dans l'espace: et de la variation des cordonnées provenant de ces déplacements considérés indépendamment des causes qui peuvent les produire. *J. Math. Pures Appl.*, 5:380–440.

Roux, P. (1960). Machines for shaping sheet metal. U.S. Patent: 2,945,528.

Sauer, R. A. and De Lorenzis, L. (2015). An unbiased computational contact formulation for 3D friction. *International Journal for Numerical Methods in Engineering*, 101(4):251–280.

Schillinger, D., Evans, J. A., Reali, A., Scott, M. A., and Hughes, T. J. (2013). Isogeometric collocation: Cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations. *Computer Methods in Applied Mechanics and Engineering*, 267:170–232.

Schillinger, D., Hossain, S. J., and Hughes, T. J. R. (2014). Reduced Bézier element quadrature rules for quadratic and cubic splines in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 277:1–45.

Schoenberg, I. J. (1946). Contributions to the problem of approximation of equidistant data by analytic functions. *Quarterly of Applied Mathematics*, 4(1-2):112–141.

Schwarze, M. and Reese, S. (2011). A reduced integration solid-shell finite element based on the EAS and the ANS concept—Large deformation problems. *International Journal for Numerical Methods in Engineering*, 85(3):289–329.

Scott, M., Li, X., Sederberg, T., and Hughes, T. (2012). Local refinement of analysis-suitable T-splines. *Computer Methods in Applied Mechanics and Engineering*, 213-216:206–222.

Sederberg, T. W., Zheng, J., Bakenov, A., and Nasri, A. (2003). T-splines and T-NURCCs. In *Proceedings of ACM SIGGRAPH 2003*, volume 22, pages 477–484, New York, USA. ACM Press.

Seifert, B. (1996). *Zur theorie und numerik finiter elastoplastischer deformationen von schalenstrukturen*. PhD thesis, Universität Hannover.

Shima, A., Yoshikawa, T., Nakamura, K., Sudo, Y., and Suzuki, S. (1997). Formation of successively expanding metallic plate and apparatus therefor. Japanese Patent: 09-085355.

Simo, J., Rifai, M., and Fox, D. (1990). On a stress resultant geometrically exact shell model. Part IV: Variable thickness shells with through-the-thickness stretching. *Computer Methods in Applied Mechanics and Engineering*, 81(1):91–126.

Simo, J. C. and Rifai, M. S. (1990). A class of mixed assumed strain methods and the method of incompatible modes. *International Journal for Numerical Methods in Engineering*, 29(8):1595–1638.

Simo, J. C., Wriggers, P., and Taylor, R. L. (1985). A perturbed Lagrangian formulation for the finite element solution of contact problems. *Computer Methods in Applied Mechanics and Engineering*, 50(2):163–180.

Stam, J. (1998). Exact evaluation of Catmull-Clark subdivision surfaces at arbitrary parameter values. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques - SIGGRAPH '98*, pages 395–404, New York. ACM Press.

Stephan, A. J., Daniel, W. J., and Elford, M. C. (2018). A GPU Based Explicit Solid-Shell Finite Element Solver. *Journal of Physics: Conference Series*, 1063(1).

Sze, K., Liu, X., and Lo, S. (2004). Popular benchmark problems for geometric nonlinear analysis of shells. *Finite Elements in Analysis and Design*, 40(11):1551–1569.

Tan, X. G. and Vu-Quoc, L. (2005). Efficient and accurate multilayer solid-shell element: non-linear materials at finite strain. *International Journal for Numerical Methods in Engineering*, 63(15):2124–2170.

Temizer, I., Wriggers, P., and Hughes, T. (2011). Contact treatment in isogeometric analysis with NURBS. *Computer Methods in Applied Mechanics and Engineering*, 200(9-12):1100–1112.

Valente, R. A. F., Jorge, R. M. N., Cardoso, R. P. R., Cesar de Sa, J. M. A., and Gracio, J. J. A. (2003). On the use of an enhanced transverse shear strain shell element for problems involving large rotations. *Computational Mechanics*, 30(4):286–296.

von Mises, R. (1913). Mechanik der festen Körper im plastisch-deformablen Zustand. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, pages 582–592.

Wang, A., Zhao, G., and Li, Y.-D. (2014). Linear independence of the blending functions of T-splines without multiple knots. *Expert Systems with Applications*, 41(8):3634–3639.

Wang, C. M., Reddy, J. N., and Lee, K. H. (2000). *Shear deformable beams and plates: relationships with classical solutions*. Elsevier.

Wang, Y. and Peng, W. (2015). A deformation analysis and experimental study for a novel full kinematic incremental forming. *Australian Journal of Mechanical Engineering*, 14(2):73–81.

Wei, X., Zhang, Y., Liu, L., and Hughes, T. J. (2016). Truncated T-splines: Fundamentals and methods. *Computer Methods in Applied Mechanics and Engineering*.

Wriggers, P. (2006). *Computational Contact Mechanics*. Springer Berlin Heidelberg, Berlin, Heidelberg.

Wu, S. R. and Gu, L. (2012). *Introduction to the explicit finite element method for nonlinear transient dynamics*. Wiley.

Xiao, H., Bruhns, O., and Meyers, A. (1998). Objective corotational rates and unified work-conjugacy relation between Eulerian and Lagrangean strain and stress measures. *Archives of Mechanics*, 50(6):1015 – 1045.

Xiao, H., Bruhns, O., and Meyers, A. (2006). Objective stress rates, cyclic deformation paths, and residual stress accumulation. *ZAMM*, 86(11):843–855.

Xiao, H., Bruhns, O. T., and Meyers, A. (1997). Logarithmic strain, logarithmic spin and logarithmic rate. *Acta Mechanica*, 124(1-4):89–105.

Xiao, H., Bruhns, O. T., and Meyers, A. (1999). Existence and uniqueness of the integrable-exactly hypoelastic equation (...) and its significance to finite inelasticity. *Acta Mechanica*, 138(1-2):31–50.

Xiao, H., Bruhns, O. T., and Meyers, A. (2000). The choice of objective rates in finite elastoplasticity: general results on the uniqueness of the logarithmic rate. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 456:1865–1882.

Xiao, H. and Chen, L. S. (2002). Hencky's elasticity model and linear stress-strain relations in isotropic finite hyperelasticity. *Acta Mechanica*, 157(1-4):51–60.

Xu, H., Liu, Y., and Zhong, W. (2012). Three-dimensional finite element simulation of medium thick plate metal forming and springback. *Finite Elements in Analysis and Design*, 51:49–58.

Young, D. and Jeswiet, J. (2004). Wall thickness variations in single-point incremental forming. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 218(11):1453–1459.

Zhou, X. and Tamma, K. (2003). On the applicability and stress update formulations for corotational stress rate hypoelasticity constitutive models. *Finite Elements in Analysis and Design*, 39(8):783–816.

Zienkiewicz, O., Taylor, R., and Zhu, J. (2013). *The Finite Element Method: its Basis and Fundamentals*. Elsevier.