



Norwegian University of
Science and Technology

Isogeometric Structural Analysis of Wind Turbine Blades

David Vågnes

Marine Technology

Submission date: June 2018

Supervisor: Josef Kiendl, IMT

Norwegian University of Science and Technology
Department of Marine Technology

Preface

Isogeometric analysis is an alternative approach to the established finite element method for structural analysis of complex structures. Moreover, the general concept can be applied to all fields of engineering mechanics, including computational fluid dynamics and fluid-structure interactions. Since the concept was introduced, several papers has been published on the topic, which implies that researchers finds the topic very interesting. This Master's thesis is written during the spring semester of 2018 as a part of the study program marine technology at NTNU. This thesis was proposed by Assoc Prof Josef Kiendl, and provides the reader an introduction to isogeometric structural analysis and shell structures in general. A wind turbine model designed for offshore sitings is under consideration, and will be supervised by Assoc Prof Erin Bachynski.

For this thesis it is assumed that the reader have some background in marine technology and basic structural engineering. It is also assumed that the reader has knowledge of mathematical formulations.

Trondheim, June 11, 2018

A handwritten signature in black ink, reading "David Vågnes". The signature is written in a cursive style with a large, stylized 'D' and a long, sweeping underline.

David Vågnes

Abstract

Isogeometric analysis, IGA, seeks to bridge the gap between engineering design and computational analysis by using spline functions as finite element bases in the analysis framework. NURBS has become the standard technique for representing geometrical shapes like spheres and cones, as well as free-form curves and surfaces. In this thesis, theory on the subject is presented and a NURBS-based Kirchhoff-Love shell formulation is utilized for structural analysis of the DTU 10MW RWT blade. Rhino is used as a tool for modelling of the NURBS model and an IGA research code in Matlab performs IGA. Two different methods for multi-patch coupling is investigated, where the first is coupling by penalty method and the second is bending strip method. A procedure for defining the regions of the blade is added to the code, and classical laminate theory is used to model the composite laminate material of the structure under consideration. Finally, a corresponding FEM analysis in Abaqus with two different load cases is compared to the IGA model in terms of accuracy and efficiency.

Acknowledgment

I would like to take this opportunity to thank my supervisor Assoc Prof Josef Kiendl for providing guidance and sharing his knowledge throughout the semester. His expertise on the subject never fails to amaze me, and has been of huge inspiration to me. A special thanks to Ph.D Candidate Davide Proserpio for guidance regarding the IGA code and the subject in general. Also, discussions in Ph.D group meetings working on the subject has been helpful, both with respect to clarifying topics and motivation. I would also like to acknowledge Assoc Prof Erin Bachynski for sharing her expertise on offshore wind turbines, as well as providing the data repository for the wind turbine under consideration.

I am very grateful for the help, guidance, and encouragement of fellow students, friends, and family during my studies. I have been lucky to share office with my mates - Erik Hveding, Jens Berg Ildstad, Kjetil Blindheim Hole, Sebastian Eriksson, Stian Rørvik Johansen and Vegard Arnetveit Liland - who have been a constant sources of insight and inspiration.

A special thank-you to my life long friends and team of proofreaders, Christer Dybvik Sunde, Marius Salen, Nikolai Vadseth, Sondre Bjørkedal Vågnes, Thomas Olav Dybvik and Øyvind Humlen. They have devoted several hours on spell checking during the finalization of this thesis. Finally, a thanks to my family: my parents, brother and two sisters. Their support has meant a lot to me during stressful times.

D.V.

Executive Summary

Isogeometric analysis, IGA, seeks to bridge the gap between engineering design and computational analysis by using spline functions as finite element bases in the analysis framework. The general concept can be applied to all fields of engineering mechanics, and has proven to perform with great accuracy and efficiency. In computer-aided design, CAD, NURBS has become the standard technique for representing exact geographical shapes like circles, spheres and cones, as well as free form curves and surfaces, which are fundamentals in geometry modelling. In the finite element method, FEM, the geometry model is typically represented by small finite element in a mesh, which is analyzed several sub-problems. The basis functions used in the analysis is typically based on simple Lagrange polynomials, whereas the design framework utilizes smooth spline functions. For awkward shaped structures, this conflict accounts for more than 80% of the overall design-analysis time. By utilizing the same NURBS basis functions in analysis, the meshing procedure is omitted and the exact geometry model created in CAD can be subject for analysis.

The use of wind turbines for electricity generation is increasingly common, and offshore wind turbines has become an attractive solution in the industry. Due to the large areas available and cost related to offshore installation, there has been a continuous upscaling of wind turbines. This has created a need for the DTU 10MW reference wind turbine, which has the scope of acting as benchmark for new wind turbine designs. In this thesis, one of the DTU 10MW RWT blades will be modelled subject for structural IGA.

The mathematical formulation of NURBS is presented, where important features to its development is pointed out. In finite element analysis, mesh refinement is used to converge and validate the solution by resolving the model with successivly finer and finer mesh. Theory behind a unique method for refining the NURBS model with respect to IGA is presented as a solution to FEM mesh refinement. Data is retrieved from the repository accompanying the reference wind turbine, and the commercial NURBS-based software Rhino is used as a tool for modelling of the NURBS model. A procedure is obtained to import a coarsen multipatch model to the IGA research code in Matlab.

In this thesis, a NURBS-based Kirchhoff-Love shell formulation is utilized for structural analysis of the blade model in IGA. Important features about the rotation-free shell formulation is pointed out, where two methods for multi patch coupling is used. Two different methods for multipatch coupling is investigated, where the first is coupling by penalty method for unmatching patches and the second is bending strip method. Furthermore, classical lamination plate theory is used to obtain the laminate stiffness matrix in the analysis.

The origin of the material parameters for the plies in the reference wind turbine blade is presented through the micromechanics equations, together with composite lamination theory.

Regional layup for the blade is presented, and a procedure for defining the regions is added to the code. Also, a stiffness function is adapted to compute the stiffness of the evaluated points in the analysis, where classical laminate theory is used to model the composite laminate material of the structure under consideration.

The stiffness function adapted in the IGA code is compared to a benchmark of a multistack and multidirectional laminate. Furthermore, the two coupling methods are compared for a simplified analysis, which in addition to controlling the unmatching patches challenge, also shows one of the advantages of IGA, where a modified geometry does not need a new meshing procedure. Finally, a corresponding FEM analysis in Abaqus with two different load cases is compared to the full IGA model in terms of accuracy and efficiency. The displacement pattern in general and tip deflection particularly for the two analyzes, shows that the IGA model performs with great accuracy and efficiency. With significantly fewer degrees of freedom, the tip displacement in IGA deviates with only 0.34% from the reference solution.

Sammendrag

Isogeometrisk analyse, IGA, prøver å kartlegge gapet mellom industrielt design og beregningsanalyse ved bruk av splinefunksjoner som formulering i elementmetode. Det generelle konseptet kan bli brukt på alle ingeniørfelt som involverer mekanikk, og har vist seg å være både effektivt og nøyaktig. I databasert design, CAD, har NURBS blitt den standardiserte teknikken for fremstilling av geometriske former som sirkler, sfærer og kjegler, samt frie formkurver og flater, som er grunnleggende geometrisk modellering. I elementmetoden er den geometriske modellen representert av flere mindre elementer i et *mesh*, som igjen blir analysert som flere mindre problemer. Basisfunksjonene som blir brukt i analysen er typisk basert på enkle Lagrange-polynomer, mens den geometriske modellen er basert på glatte splinefunksjoner. For kompliserte strukturer står denne geometriske konflikten for over 80% av hele analyseringsprosedyren.

For produksjon av elektrisk energi blir bruken av vindturbiner stadig mer populært, og da spesielt løsninger offshore vindturbin. På grunn av store tilgjengelige områder og kostnader relatert til installasjon, har det forekommet en trend mot stadig større vindturbiner. Dette har skapt et behov for DTU 10MW RWT, en referansevindturbin med mål om å være en retningslinje for nye vindturbindesign. I denne oppgaven, skal ett av bladene fra rotoren bli modellert og analysert med IGA.

Den matematiske formuleringen bak NURBS blir presentert, der viktige poeng knyttet teknikkens egenskaper og utvikling blir pekt ut. Det medfølgende datalageret til referanse vindturbinen er brukt for å importere informasjon om geometrien til bruk i modelleringsprogramvaren Rhino. Fra dette lages en prosedyre til modellering av vindturbin rotor med hensyn til IGA i en Matlab-basert forskningskode.

I denne oppgaven benyttes en NURBS-basert Kirchhoff-Love skallformulering til strukturanalyse av vindturbinbladet i IGA. I teorikapitlet presenteres to konvensjonelle metoder for kobling av flere NURBS overflater i den rotasjonsfrie skallformuleringen. Videre presenteres klassisk laminatteori for å kalkulere laminat stivhetsmatrisen i analysedelen.

Materialene brukt i komposittoppbygning av referanse vindturbinbladet blir presentert gjennom mikromekanikk-ligninger sammen med klassisk laminatteori. Oppbygningen av bladet presenteres, der en prosedyre for å definere bladets regioner blir supplert til koden. Videre blir en stivhetsfunksjon tilpasset beregning av stivheten for hvert enkelt punkt produsert, der klassisk laminatteori brukes til å modellere komposittlaminatmaterialet av strukturen.

Stivhetsfunksjonen som er tilpasset IGA-koden blir sammenlignet med et referansepunkt for et flerlags- og flerretningslaminat. Videre er de to koblingsmetodene sammenlignet i en forenklet analyse, der utfordringer knyttet til overflater uten felles kant blir kontrollert. Den samme testen viser frem en av de største fordelene knyttet til IGA, der en modifisert geometri ikke trenger et nytt mesh. Til slutt sammenlignes IGA av en fullt utrustet modell med to lastkondisjoner mot en referanseanalyse gjennomført med elementmetoden. Sammenligning av forskyvningsmønster mellom de to analysene viser at IGA er svært effektivt og presist. I tillegg er avviket for bøyning av rotorens frie ende bare 0.34% fra referansen, selv om mange færre frihetsgrader er benyttet.

Nomenclature

CAD	Computer-Aided Design
CC	Cap centre
CPU	Central Processing Unit
DTU	Technical University of Denmark
FEA	Finite element analysis
FEM	Finite element method
IGA	Isogeometric analysis
IPCC	Intergovernmental Panel on Climate Change
LC	Load case
MW	Mega Watt
NREAP	National Renewable Energy Action Plans
NURBS	Non-uniform rational B-splines
PA	Pitch axis
RWT	Reference Wind Turbine
USCS	United states customary units
α	Penalty parameter
δ_i^i	Kronecker delta
$\epsilon_{\alpha\beta}$	Membrane strains, middle surface shell
ϵ_{ij}	Strains in $i j$ direction
γ_{ij}	Shear strains in $i j$ direction
$\kappa_{\alpha\beta}$	Curvature, middle surface shell

ρ	Mass density
σ_{ij}	Cartesian Cauchy stresses in $i j$ direction
τ_{ij}	Cartesian Cauchy shear stresses in $i j$ direction
θ^i	Surface coordinates
ϵ	Strain
κ	Curvature
ν	Poisson's ratio
ξ/η	Knot
Ξ/H	Knot vector
$\bar{\mathbf{D}}_k$	Laminate stiffness matrix of ply k
\mathbf{A}	Extensional stiffness matrix
\mathbf{a}_α	Base vector on the middle surface p
\mathbf{B}	Coupling stiffness matrix
\mathbf{D}	Bending matrix
\mathbf{D}^{mat}	Material matrix
\mathbf{e}_i	Standard basis vectors in i direction
\mathbf{P}_i	control point coordinates i
\mathbf{R}	Reuter's matrix
\mathbf{S}	Compliance matrix
\mathbf{T}	Transformation matrix
\mathbf{u}	Deformation vector
\mathbf{X}	Position vector
$b_{\alpha\beta}$	Curvature tensor coefficient
$B_{i,p}$	Bernstein basis functions for i control points of degree p
c	Chord length
C^n	Parametric continuity of n degree

E_i	Young's modulus in i direction
$E_{\alpha\beta}$	Strain corefficients
$E_{\alpha\beta}$	Strains in the shell
E_s	Directional bending stiffness, bending strip
f_a	Magnitude area load
G	Transverse shear modulus
g	Gravity constant
g^i	Contravariant basis
$g^{\alpha\beta}$	Contravariant metric coefficient
g_i	Covariant basis
$g_{\alpha\beta}$	Covariant metric coefficient
ix	Counting index circumeferential direction
ix	Counting index side
iz	Counting index redial direction
$N_{i,p}$	Normalized B-spline basis functions for i control points of degree p
$R_{i,p}$	Normalized rational basis functions for i control points of degree p
S	Second Piola-Kirchhoff stresses
V	Volume
W	Work
z_k	Distance from ply k to neutral axis
a	Metric coefficient
b	Curvature coefficient
F^{ext}	External load vector
$K^{int,lin}$	Linearized internal stiffness matrix
p/q	Polynomial Degree
C	Curve representation

m	Bending moments
n	Normal forces
S	Surface representation
x	Position Vector
N	Basis functions
n	Number of Control Points
R	Rational basis functions
r	Radial position
t	Thickness

Contents

Preface	i
Acknowledgment	iii
Executive Summary	iv
Nomenclature	ix
1 Introduction	3
1.1 Background	6
2 Geometrical Description	9
2.1 Curves and Surfaces	9
2.1.1 Bézier Curves	9
2.1.2 Bézier Surfaces	10
2.1.3 B-Spline Curves	11
2.1.4 B-Spline Surfaces	13
2.1.5 Knot Vector	14
2.1.6 Non-Uniform Rational B-Spline Curves	14
2.1.7 Non-Uniform Rational B-Spline Surfaces	16
2.2 NURBS Refinement	18
3 Isogeometric Analysis	21
3.1 NURBS-Based Isogeometric Analysis	21
3.2 Differential Geometry of Surfaces	23
3.3 NURBS-Based Kirchhoff-Love Shell Element	26
3.4 Multipatch Coupling	28
4 Orthotropic Materials	31
4.1 Micromechanics Theory	32
4.2 Classical Lamination Plate Theory	33
5 Modelling of Offshore Wind Turbine Blades	39
5.1 Blade Geometry Description	40
5.2 Rhinoceros As a Tool for Modelling	42
5.2.1 Importing Points	43
5.2.2 Creating NURBS Control Polygon	44

5.2.3	Model Coarsening	44
5.2.4	Internal Structure in Rhinoceros	45
5.3	Material Parameters	47
5.4	Regions Layup	49
6	Results and Discussion	53
6.1	Stiffness Function	53
6.2	Load Cases	55
6.3	IGA Multipatch Coupling Methods	55
6.4	Finite Element Method Reference Model	56
6.5	Structural Analysis of Wind Turbine Blade with IGA	58
7	Conclusions	63
	References	65
A	Modelling	69
B	Composite Layups	71
C	Matlab	77

List of Figures

1.1	Offshore wind turbine	4
2.1	Example B-spline basis functions	12
2.2	Example B-spline curve	13
2.3	Example B-spline surfaces	15
2.4	Example NURBS curve	16
2.5	Example free-form NURBS surface	18
2.6	NURBS refinement	19
2.7	Example k-refinement	20
3.1	NURBS element formulation	22
3.2	NURBS refinement conflict	23
3.3	Multipatch coupling with bending strip method and penalty method.	30
4.1	Transformation fibre directions	34
4.2	Deformations associated with Kirchhoff plate theory	35
5.1	Airfoil profile	39
5.2	Relative thickness	40
5.3	Fully assambled RWT	41
5.4	Reference wind turbine blade structure definitions	42
5.5	Regional definitions	43
5.6	Curve interpolation and control points	44
5.7	Definitions of internal structure.	46
5.8	Wind turbine blade in Rhino environment	46
5.9	Cross section w/ internal structure definitions	47
5.10	Layup plot and ply stack region	50
5.11	Mathematical definition of regions	51
6.1	Deformation with two different methods of multipatch coupling	56
6.2	Manipulated control points model	56
6.3	FEM setup	57
6.4	Deformed configuration of the reference FEM analysis	58
6.5	IGA setup	59

6.6	Accumulated thickness and effective longitudinal modulus	60
6.7	Deformed configuration of IGA	60
6.8	Displacement plot along the span for the two analyzes	61

List of Tables

3.1	Isogeometric analysis compared to finite element analysis	23
5.1	Combinations for model coarsening.	45
5.2	Mechanical properties of the epoxy matrix (Hinton et al., 2004).	48
5.3	Mechanical properties of the E-glass fibres (Hinton et al., 2004).	48
5.4	Calculated mechanical properties of the two different laminae types.	48
5.5	Calculated engineering mechanical properties of the two different laminae types.	49
5.6	Mechanical properties of the core material.	49
6.1	Multidirectional and multistack composite laminate stiffness benchmark input.	54
6.2	Input parameters for multipatch coupling method analysis.	55
6.3	Tip deflection of the two IGA models.	56
6.4	Tip deflection of the two load cases for the FEM analysis.	58
6.5	Initial guess for point search algorithm	59
6.6	Tip deflection of the two load cases for the FEM analysis.. . . .	60
A.1	Manipulated cross sectional deviation for section 1-12.	69
A.2	Manipulated cross sectional deviation for section 13-24.	69
A.3	Manipulated cross sectional deviation for section 25-36.	69
A.4	Manipulated cross sectional deviation for section 37-48.	70
A.5	Manipulated cross sectional deviation for section 49-60.	70

Chapter 1

Introduction

Shell structures are curved surfaces that carries loads "through its shape" and is thus very efficient in saving of material and weight. The term shell is often associated with structures which process huge strength despite of being thin. Its appearance can often be seen in nature through eggs, nuts, the human skull and the shell of a tortoise. The shell element differentiates from the plate element due to its curvature, and are typically applied at aircrafts, ship hulls, wind turbine blades and large roof structures at stadiums. A thin shell is defined as a shell which has thickness relatively small compared to its other dimensions. Moan (2003) states that thin shell theory can be applied for shells which has other dimensions at least ten times its thickness. August E.H. Love presented a shell theory developed from the "classical plate theory" of Gustav R. Kirchhoff. The theory is known as the Kirchhoff-Love shell theory, and its applicability is related to thin shells.

With global climate change already having an observable effects on the environment, and CO₂ emission as the main contributor to the change, renewable energy sources have become a major focus area. IPCC (2014) lists wind power as one of the key low-carbon options for energy supply in their latest report on climate change. Modern wind turbines are developed to convert wind energy into electricity, through kinetic energy in the wind to mechanical energy in the shaft, which subsequently converts into electrical energy in a generator. Even though most wind turbines are located onshore, The European Wind Energy Association (2017) points to a trend of rapidly increasing installed offshore wind capacity. This is rooted in the fact that European member states are completing their National Renewable Energy Action Plans, NREAPs, as an answer to the Renewable Energy directive of the European Union (2009). With offshore wind, the structure adds low noise and visual impact, large areas are available at a low price and the wind is considered to be less turbulent, i.e. greater efficiency and power production. However, offshore wind is in a wet and corrosive environment with difficult access for installation and maintenance. With this in mind, structural integrity of offshore wind turbines is an important topic. In order to meet the huge demand for renewable energy, it has been found more cost efficient to increase the wind turbine rotor radius and thereby increase power production per wind turbine. The upscaling of wind

turbines are challenging because the mass of the turbine increases with the cube of the rotor radius. To address this challenge, wind turbine blades are optimized to increase stiffness and overall performance.



Figure 1.1: Offshore wind turbine (The European Wind Energy Association, 2017)

The standard technology that is used by computer-aided design, CAD, to represent complex geometries is the non-uniform rational B-Splines, NURBS. In a traditional engineering design process the geometry is model in a CAD system. This model represents the *exact* geometry of the structure, and is the foundation for the analysis model. With the finite element method, FEM, the geometry model is represented by several finite elements in a *mesh*, and then the model is analyzed as several sub-problems. For simple geometries, automatic meshing procedures are very efficient. However, most structures has awkward geometries like curvature in multiple directions, bad aspect ratios and are considered to be complex. Here the meshing procedure can only be done semi-automatically, since the mesh has to be repaired manually by the analyst to yield acceptable results. Lovadina et al. (2014) states that mesh generation accounts for more than 80% of overall analysis time, meaning that this process is looked upon as a bottleneck for the industry. The reason for this situation, is that the mathematical description used in design is different from the ones used in analysis.

Hughes et al. (2005) introduced the idea of isogeometric analysis, IGA, which is to use the smooth NURBS basis functions as the basis for analysis. As a result, the exact geometry is introduced into the analysis without any further modifications. The general concept of IGA can be applied to all fields of engineering, including structural analysis, computation fluid

dynamics and fluid-structure interactions. Thanks to the high-regularity properties of its basis functions, IGA has shown greater accuracy per degree of freedom and an enhanced robustness with respect to standard finite element analysis. In this thesis, the reference wind turbine blade of Bak et al. (2013) is the subject for structural IGA, due to its shell characteristics, complex shape and topology. The thesis is outlined as follows:

- *Chapter 2:* The geometrical basics needed with respect to geometric modelling of curves and surfaces with *NURBS* in this thesis is presented. Also, parametrization of the curve and surface and *NURBS* refinement is reviewed.
- *Chapter 3:* *NURBS*-based IGA is explained and key differences to finite element analysis is pointed out. Equations related to differential geometry of surfaces used for the shell formulation are reviewed. Furthermore, the *NURBS*-based Kirchhoff-Love shell theory with a focus on linear analysis and orthotropic materials is presented. Finally, two methods for coupling the rotations between *NURBS* patches is discussed.
- *Chapter 4:* Theory related to fiber-reinforced laminated plates are outlined, leading to a computational scheme where classical lamination plate theory is applied to find the composite laminate stiffness matrix.
- *Chapter 5:* Terminology related to modelling wind turbine rotors is presented. The geometry and structural properties of the wind turbine blade under consideration is reviewed. *Rhino* is presented as a tool for modelling of the *NURBS* wind turbine blade for IGA. The mechanical properties of the plies used in the structural analysis is derived, and a method for assigning regions to the blade is obtained.
- *Chapter 6:* An adapted laminate stiffness function is tested on a benchmark example for a multistack, multidirectional laminate. The results of the finite element method reference model is presented and discussed. Two methods for patch coupling of the internal structure for the wind turbine blade is investigated. All the presented methods are applied to the structural analysis of the reference wind turbine blade, compared and discussed.
- *Chapter 7:* Main findings are concluded and further work based on the thesis is proposed.

1.1 Background

Problem formulation from supervisor:

This thesis is based on a project work done during the autumn semester of 2017. The main focus of the project thesis was to familiarize with NURBS modelling, NURBS-based Kirchhoff-Love shell theory and modelling of wind turbine blades. This was mainly a literature study where theoretical bases were covered. Judging by the number of publications, IGA has been of increasing interest in among computational mechanics researchers since it first was introduced by Hughes et al. (2005).

Problem Formulation

Within this thesis, isogeometric analysis shall be used for the modelling and structural analysis of a single blade of the DTU 10MW Reference Wind Turbine. A precise multi-patch NURBS model of the blade, taking into account different materials and cross sections shall be created. An IGA Matlab research code for performing the analyses is provided. Shell elements will be used for analysis and classical laminate theory shall be used to model composite laminate materials. The analyses to be performed shall include those described in the DTU report and results shall be compared to those reported. Additionally, corresponding FEM analyses in Abaqus (or similar) shall be performed for comparing IGA and FEM in terms of accuracy depending on the mesh size.

The work is to be carried out as follows:

1. Create a detailed structural model (multi-patch, multi-material) of a single blade of the DTU 10 MW reference wind turbine.
2. Apply the load cases, perform structural analyses, and compare to the results as described in the DTU report.
3. Perform corresponding FEM analyses (in Abaqus or similar) with mesh refinement studies and compare the accuracy of IGA and FEM.

The format of the thesis is carried out to be:

- The thesis should be organized in a rational manner to give a clear exposition of results, assessments, and conclusions. The text should be brief and to the point, with a clear language. Telegraphic language should be avoided.
- The thesis shall contain the following elements: A text defining the scope, preface, list of contents, summary, main body of thesis, conclusions with recommendations for further work, and references. All figures, tables and equations shall be numerated.

- The supervisors may require that the candidate, in an early stage of the work, presents a written plan for the completion of the work.
- The original contribution of the candidate and material taken from other sources shall be clearly defined. Work from other sources shall be properly referenced using an acknowledged referencing system.

Related work

The doctoral thesis of Kiendl (2011) on shape optimal shell structures with the NURBS based Kirchhoff-Love shell element has been used as a basis in this thesis. The well known textbooks Piegl and Tiller (1995) and Rogers (2001) has created a great foundation on modelling with NURBS, as well as offering deeper insight on the subject. Also, the report on the DTU 10MW reference wind turbine by Bak et al. (2013) has been a huge recourse in this thesis.

Chapter 2

Geometrical Description

Rogers (2001) states that curves and surfaces can be expressed explicitly, implicitly and parametrically. For an explicit representation of a curve a x-value can only represent one y-value, whereas an implicit representation can have a x-value describing several y-values, e.g. a circle. Parametric curves and surfaces is defined by parametric equations of one and two variables, respectively. Moreover, the parametric representation is the most common way of representing curves and surfaces in *Computer Aided Design*, or CAD, thus all methods described in the next sections are based on the parametric representation. Flexible properties like axis independence, easy representation of multiple values for the same variable and additional degrees of freedom makes the parametric representation perfect for defining free from curves and surfaces. Furthermore, the standard mathematical description method used in CAD are Non-Uniform Rational B-Splines, or NURBS. For working professionals and students, the mathematics behind NURBS are looked upon as highly advanced and very theoretical. However, on a working-level, it is not necessary to understand the deep mathematics. In this chapter, the historical development from Bézier to NURBS, via B-spline curves and their basis will be presented.

2.1 Curves and Surfaces

Rogers (2001) presents NURBS curves and surfaces with a historical perspective, where the antecedents, Bézier and B-Spline are presented in order to understand NURBS. In this thesis, the final mathematical description and distinguishing features of the three representation will be the focus point.

2.1.1 Bézier Curves

In contrast to a interpolating polynomial, where the curve is approximated by passing through control points, a Bézier curve is defined inside a *control polygon*, where only the first and the last point is interpolated. The order of the curve is defined as $p = n - 1$, where p is the order

and n is the number of control points. Furthermore, the curve is defined by the normalized parametric variable ξ as a function of the polygon control point coordinates and the Bernstein basis functions, which is given by Rogers (2001):

$$\mathbf{C}(\xi) = \sum_{i=1}^n B_{i,p}(\xi) \mathbf{P}_i \quad 0 \leq \xi \leq 1 \quad (2.1)$$

where P_i is the control point coordinates and $B_{i,p}$ is the Bernstein basis function defined as:

$$B_{i,p}(\xi) = \frac{n!}{i!(n-i)!} \xi^i (1-\xi)^{n-i} \quad (2.2)$$

An important property for Bézier curves with respect to design is that the starting point of the curve has the same coordinates as the first control point $\mathbf{C}(0) = \mathbf{P}_0$. Similarly, the ending point of the curve has the same coordinates as the last control point $\mathbf{C}(1) = \mathbf{P}_n$. This obviously gives more control of the curve for the designer, unlike to polynomial interpolated curves, where one does not have the same control of start and ending points. Piegl and Tiller (1995) states the shortcomings of Bézier curves; with respect to passing the curve through n control points, one needs a $(n-1)$ -degree polynomial curve. High order curves are inefficient, as well as the algorithm gets numerically unstable. Furthermore, modification of a control point has influence on the entire curve, i.e. no local changes to the curve can be done.

2.1.2 Bézier Surfaces

There exists several schemes for representing a surface, but the most common method is the tensor product scheme. This method can shortly be explained as a bidirectional curve scheme. The surface consists of basis functions and geometric coefficients. The basis functions are multi variable functions of ξ and η , constructed as products of single variable basis functions. A net constrains the arranged geometric coefficients in two directions. Piegl and Tiller (1995) obtains the Bézier surface with a net of control points and the Bernstein basis functions in two parametric directions:

$$\mathbf{S}(\xi, \eta) = \sum_{i=1}^n \sum_{j=1}^m B_{i,p}(\xi) B_{j,p}(\eta) \mathbf{P}_{i,j} \quad (2.3)$$

Where $\mathbf{P}_{i,j}$ form the bidirectional control net and $B_{j,p}(\eta)$ is the Bernstein basis functions in η -direction, expressed nearly identically as in Equation 2.2:

$$B_{j,p}(\eta) = \frac{m!}{j!(m-j)!} \eta^j (1-\eta)^{m-j} \quad (2.4)$$

2.1.3 B-Spline Curves

Schoenberg (1946) was the first to suggest theory on B-splines. The B-Spline curve originates from the Bézier curve, and is built on the same basic principles, where the curve is a linear combination of control points and basis functions through a parametric space. However, the basis functions for B-spline curves are different and is called B-Spline basis. Rogers (2001) states that this basis is generally non-global and each control point is associated to its own unique basis function, i.e. the control points have local influence on the curve by giving contribution only where the parameters of the basis function is non-zero. Another feature of the B-spline basis, is that it allows changing the degree of the basis function, and hence the degree of the resulting curve, without changing the number of control points.

Letting $\mathbf{C}(\xi)$ be the parametric curve, with ξ as the normalized variable, the B-spline curve is given as:

$$\mathbf{C}(\xi) = \sum_{i=1}^n N_{i,p}(\xi) \mathbf{P}_i \quad 0 \leq \xi \leq 1 \quad (2.5)$$

where $N_{i,p}$ is the normalized B-spline function generally and specially defined by the Cox-deBoor recursion formula. The special case, where the degree p of the basis function is equal to zero is defined as:

$$N_{i,0} = \begin{cases} 1 & \text{if } \xi_i \leq \xi \leq \xi_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (2.6)$$

where ξ_i are elements of a knot vector, which will be described in more detail in Section 2.1.5. Then the general definition of B-spline basis function with $p \leq 1$ is given as:

$$N_{i,p}(\xi) = \frac{(\xi - \xi_i) N_{i,p-1}(\xi)}{\xi_{i+p} - \xi_i} + \frac{(\xi_{i+p+1} - \xi) N_{i+1,p-1}(\xi)}{\xi_{i+p+1} - \xi_{i+1}} \quad (2.7)$$

The resulting plot for cubic B-spline basis functions can be seen in Figure 2.1. Some important properties about the B-spline basis functions are deduced:

- For all basis function of higher order than $p=1$, each basis function has exact one vertex.
- The values for the basis function is either zero or positive, i.e. $N_{i,p}(\xi) \geq 0$.

- de Boor (1972) claims that the sum of B-spline basis functions for any parameter ξ is one, expressed mathematically as:

$$\sum_{i=1}^n N_{i,p}(\xi) \equiv 1 \quad (2.8)$$

- Piegl and Tiller presents the linear independence of each respective basis function, i.e. $\sum_{i=1}^n \alpha_i N_{i,p}(\xi) = 0$

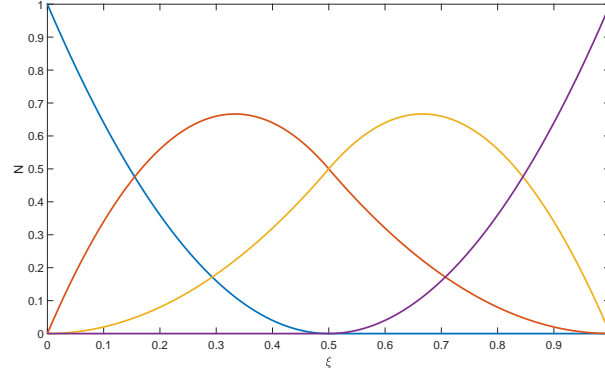


Figure 2.1: Quadratic B-spline basis functions with the open knot vector $\Xi = [0, 0, 0, 0.5, 1, 1, 1]$

Continuity, C , is defined as the "smoothness" inside or in the join between curves and surfaces. The orders of continuity, C^0 , C^1 and C^2 , refers to *join*, *equal tangent* and *equal curvature* in a point on a curve, respectively. An example of a B-spline curve, with illustrative control points are shown in Figure 2.5. The blue crosses, marks the knots on the curve from the parametric space. Examining the the knots reveals that the curve is tangent to the control polygon in these points. The reason for the tangency is that the curve drops down to C^1 -continuous at the knot, which makes the knot a tangent point. In addition, the C^0 -continuity can be seen from the kink at the point $(4, -1)$. It is also worth noting that the curve is interpolated at both the first and last control point.

Hughes et al. (2005) summarizes the important properties of B-spline curves with respect to IGA:

- The curve has continuous derivatives of order $p-1$, except where it has repeated knots.
- The curve is C^{p-1} -continuous at a single knot and C^∞ -continuous inside a knot span.
- By applying an affine transformation to the control points, one obtains the transformation of the curve, i.e translations, rotations, scaling and shear.

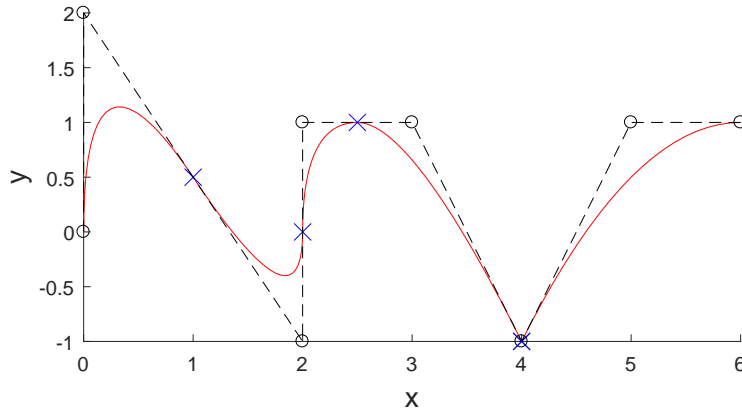


Figure 2.2: B-spline curve inside a control polygon with $p=2$ and $\Xi = [0, 0, 0, 0.2, 0.4, 0.6, 0.8, 0.8, 1, 1, 1]$, where the blue crosses marks the knot spans of the curve.

2.1.4 B-Spline Surfaces

The tensor product of B-spline basis functions in two parametric dimensions ξ and η will compute a B-spline surface, and is expressed mathematically by Rogers (2001) as:

$$\mathbf{S}(\xi, \eta) = \sum_{i=1}^n \sum_{j=1}^m N_{i,p}(\xi) M_{j,q}(\eta) \mathbf{P}_{i,j} \quad (2.9)$$

where $N_{i,p}(\xi)$ and $M_{j,q}(\eta)$ are the corresponding basis functions to the definition space as a net of control points, the basis functions knot vectors Ξ and H , and the two polynomial degrees p and q , respectively. $N_{i,p}(\xi)$ is defined as before in Equation 2.6-2.7 and $M_{j,q}(\eta)$ is defined similarly as:

$$M_{j,q}(\eta) = \frac{(\eta - \eta_j) M_{j,q-1}(\eta)}{\eta_{j+q} - \eta_j} + \frac{(\eta_{j+q+1} - \eta) M_{j+1,q-1}(\eta)}{\eta_{j+q+1} - \eta_{j+1}} \quad (2.10)$$

2.1.5 Knot Vector

Piegl and Tiller (1995) defined the knot vector as $\Xi = [\xi_1, \xi_2, \dots, \xi_{n+p+1}]$, where ξ_i are knots in the parametric space. The knots are a non-decreasing sequence of real numbers, i.e. $\xi_i \leq \xi_{n+p+1}$. A B-spline representation of a curve with the knot vector:

$$\Xi = [\underbrace{0, \dots, 0}_{p+1}, \underbrace{1, \dots, 1}_{p+1}]$$

is a generalization of the Bézier representation of a curve.

The term multiplicity refers to the number the same knot is repeated. There are two types of knot vectors. The open knot vector, as its commonly called, has multiplicity of $p + 1$ at the first and the last knot. This knot vector may also referred to as non-periodic or clamped in other textbook. According to Piegl and Tiller (1995), the open knot vector is the standard in CAD, due to its ability to decide the exact starting and ending point of a curve. The knot vector type is periodic otherwise. The periodic knot vectors basis functions is described by Rogers (2001) to be a translate of each other. However, only open knot vectors will be used in this thesis, meaning periodic knot vectors will only be mentioned once here. The knot vector types are divided into two flavours; a knot vector is uniform if the interior knots are equally spaced, which means there exists a real number $d = \xi_{i+1} - \xi_i$ for all knots, with $p \leq i \leq n - p - 1$, and non-uniform otherwise.

Figure 2.3a shows a free form B-spline surface where the knot vectors are equal and open in both parametric directions. Rogers (2001) states that the shape and character of a B-spline surface, as with a B-spline curve, is highly dependant on the knot vectors Ξ and H , which not necessarily need to be equal. He mentions the cylinder as an example, where the knot vectors are unequal, where H has multiplicity $k = 2$ at $\xi = 0.25$, $\xi = 0.5$ and $\xi = 0.75$, which creates a *kink* in the respective points. The knot vector Ξ is open with no internal knots. This is the way of expressing a perfect cylinder with a B-spline surface, as can be seen in Figure 2.3b. As a result, one will have C^0 -continuity of the surface at the multiple knots. However, since these knots also are the tangential points between the surface and control polygon, the geometry is still smooth due to the unique position of its control polygon.

2.1.6 Non-Uniform Rational B-Spline Curves

Versprille (1975) was the first to present work on Non-Uniform Rational B-Splines, or *NURBS*. The acronym refers to the knot vector, which is not uniform, and the basis functions, which are piecewise rational polynomials. Technically, NURBS is the special case of rational B-splines, with an open knot vector. The control points that defines the control polygon has in addition to its coordinates an individual weight, w_i . This yields the control point $P_i = (x_i, y_i, z_i, w_i)$, which also can be given in projective coordinates $P_i^w = (w_i x_i, w_i y_i, w_i z_i, w_i)$

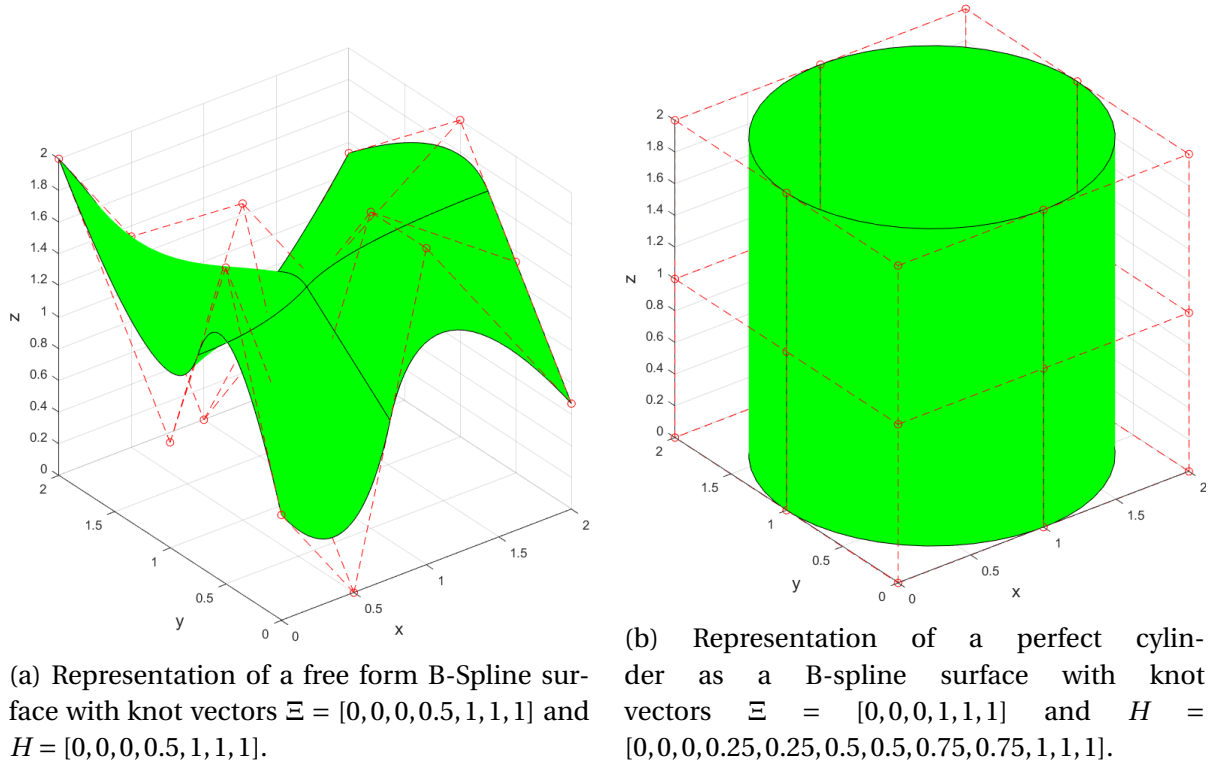


Figure 2.3: B-Spline surfaces with equal and unequal knot vectors in both directions.

in a four-dimensional space. The NURBS curve becomes the projection of a B-spline curve in the four-dimensional space with projective coordinates, onto the three-dimensional space, which yields the "rational" part of the NURBS acronym. Piegl and Tiller (1995) derives the NURBS representation of a curve as:

$$\mathbf{C}(\xi) = \frac{\sum_{i=1}^n N_{i,p}(\xi) w_i \mathbf{P}_i}{\sum_{i=1}^n N_{i,p}(\xi) w_i} \quad 0 \leq \xi \leq 1 \quad (2.11)$$

where w_i is assumed positive for all i . The rational basis functions in Equation 2.11 are then given as:

$$R_{i,p}(\xi) = \frac{N_{i,p}(\xi) w_i}{\sum_{i=1}^n N_{i,p}(\xi) w_i} \quad (2.12)$$

The rational basis functions are piecewise rational functions on $\xi \in [0, 1]$, and rewrites the NURBS curve formulation in Equation 2.11 on the common form as a sum of the respective rational basis functions times the control point coordinates:

$$\mathbf{C}(\xi) = \sum_{i=1}^n R_{i,p}(\xi) \mathbf{P}_i \quad (2.13)$$

The combination of deciding where the curve should both start and end, position of knots and the individual weight of control points, makes the NURBS curve highly adaptable for design purpose. These properties are illustrated in Figure 2.4.

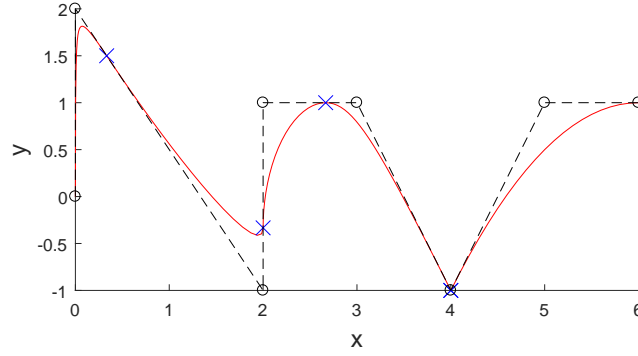


Figure 2.4: NURBS curve with start and end point at (0,0) and (6,1) respectively, kink at (4,-1) and weighted control points at (0,2)

2.1.7 Non-Uniform Rational B-Spline Surfaces

As mentioned earlier, dealing with NURBS curves yields excellent local and global control due to the knots. The same applies for NURBS surfaces. Also, the NURBS surface has open knot vectors in the two parametric directions which defines the surface. The NURBS surface is expressed by Piegl and Tiller (1995) as

$$\mathbf{S}(\xi, \eta) = \sum_{i=1}^n \sum_{j=1}^m R_{i,j}^{p,q}(\xi, \eta) \mathbf{P}_{i,j} \quad 0 \leq \xi, \eta \leq 1 \quad (2.14)$$

where $R_{i,j}^{p,q}(\xi, \eta)$ is the rational basis functions of a surface:

$$R_{i,j}^{p,q}(\xi, \eta) = \frac{N_{i,p}(\xi) M_{j,q}(\eta) w_{i,j}}{\sum_{i=1}^n \sum_{j=1}^m N_{i,p}(\xi) M_{j,q}(\eta) w_{i,j}} \quad (2.15)$$

With knot vectors:

$$\Xi = [\underbrace{0, \dots, 0}_{p+1}, u_{p+1}, \dots, u_{r-p-1}, \underbrace{1, \dots, 1}_{p+1}]$$

$$H = [\underbrace{0, \dots, 0}_{q+1}, u_{q+1}, \dots, u_{s-q-1}, \underbrace{1, \dots, 1}_{q+1}]$$

The important properties of $R_{i,j}^{p,q}(\xi, \eta)$ are roughly the same as $N_{i,j}^{p,q}(\xi)$, mentioned in Section 2.1.3. Piegl and Tiller (1995) summarizes these properties for the piecewise rational basis functions:

- Non-negativity for all i, j, ξ and η , i.e. $R_{i,j}(\xi, \eta) \geq 0$
- Partition of unity for all $(\xi, \eta) \in [0, 1] \times [0, 1]$, i.e. $\sum_{i=1}^n \sum_{j=1}^m R_{i,j} = 1$
- Local support if (ξ, η) outside the rectangle given by $[\xi_i, \xi_{i+p+1}) \times [\eta_j, \eta_{j+q+1})$, i.e. $R_{i,j}(\xi, \eta) = 0$ outside this rectangle.
- If $p > 0$ and $q > 0$, only one maximum value exist for each basis function $R_{i,j}(\xi, \eta)$
- The First and last basis function starts and ends at the value 1, i.e. $R_{0,0}(0, 0) = R_{n,0}(1, 0) = R_{0,m}(0, 1) = R_{n,m}(1, 1) = 1$
- Inside the rectangles formed by the ξ and η knots, all partial derivatives of $R_{i,j}(\xi, \eta)$ exist. At a ξ and η knot, the function is $p - k$ and $q - k$ differentiable in the ξ and η direction, respectively.

The properties of piecewise rational basis functions yields some important geometric properties of NURBS surfaces:

- The corner points of the surface are interpolated, i.e. $\mathbf{S}(0, 0) = \mathbf{P}_{1,1}$, $\mathbf{S}(1, 0) = \mathbf{P}_{n,1}$, $\mathbf{S}(0, 1) = \mathbf{P}_{0,m}$ and $\mathbf{S}(1, 1) = \mathbf{P}_{n,m}$
- As long as $w_{i,j} \geq 0$ for all i, j and $(\xi, \eta) \in [\xi_i, \xi_{i+p+1}) \times [\eta_j, \eta_{j+q+1})$, then $\mathbf{S}(\xi, \eta)$ is inside the convex hull of the control points $\mathbf{P}_{i,j}$, $i_1 - p \leq i \leq i_1$ and $j_1 - q \leq j \leq j_1$
- Local influence to the curve, if the coordinates of $\mathbf{P}_{i,j}$ or weight of $w_{i,j}$ is changed, the shape of the surface only inside the rectangle $[\xi_i, \xi_{i+p+1}) \times [\eta_j, \eta_{j+q+1})$ is changed.
- Bézier and B-spline surfaces are special cases of NURBS surfaces.

The control net that forms the surface in Figure 2.3a is shown in Figure 2.5 as a NURBS surface with increased weight of control point $\mathbf{P}_{2,1}$.

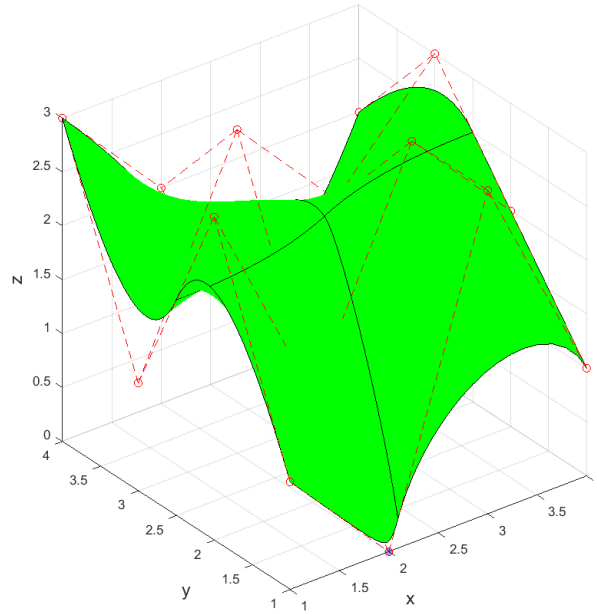


Figure 2.5: NURBS surface with open knot vectors $\Xi = [0, 0, 0, 0.5, 1, 1, 1]$ and $H = [0, 0, 0, 0.5, 1, 1, 1]$, highlighted control point in blue $\mathbf{P}_{2,1}$ and individual $w_{2,1}=10$.

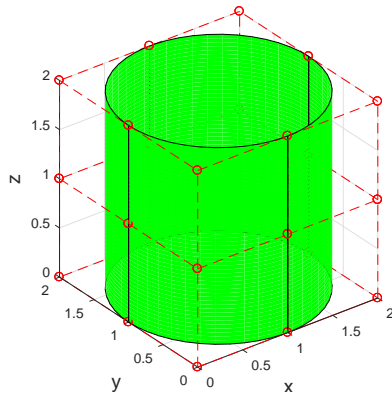
2.2 NURBS Refinement

Rogers (2001) presents two methods for increasing flexibility, or refinement to a NURBS surface. The two common types of refinement are:

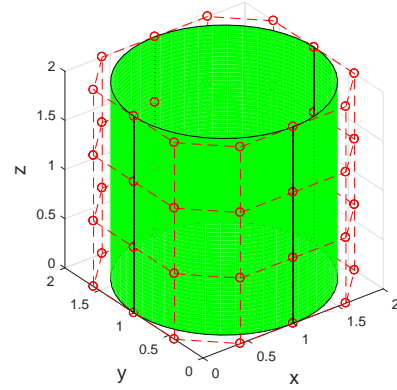
- Knot insertion, corresponding to FEM h-refinement
- Order elevation, corresponding to FEM p-refinement

Common for the two methods is that they both add control points to the grid which defines the surface, without changing the geometry. Knot insertion adds a knot to the curve or surface, where each knot adds one control point. When adding control points to the surface description, other control points are recomputed as described in the algorithm by Boehm and Prautzsch (1985). Like before, adding a knot to the curve reduces continuity by one where the knot is added. Moreover, order elevation increases polynomial degree of the basis function, while keeping the knot interval constant before and after the refinement. This allows the curve to be of same differentiable degree. Figure 2.6 shows an example of a cylinder, where the algorithm from Piegl and Tiller (1995) is used to refine the surface. The example illustrates a surface where the degree of the basis functions in both directions are elevated with one degree. In addition to elevate the order, all knots in the knot vector increases its multiplicity to maintain the same continuity and thereby geometric shape of the surface. As can be seen, more control points are added along the edge of the surface in both parametric directions.

Hughes et al. (2005) introduces k-refinement, where one combines order elevation and knot insertion for the refinement. Remembering that order elevation preserves continuity at all



(a) Surface of a cylinder with polynomial degrees $p=2$ and $q=2$. Knot vectors $\Xi = [0, 0, 0, 1, 1, 1]$ and $H = [0, 0, 0, 0.25, 0.25, 0.5, 0.5, 0.75, 0.75, 1, 1, 1]$



(b) Refined surface: Unchanged geometry, different coordinates for control points, and new polynomial degrees $p=3$ and $q=3$. More control points and all knots in both directions has increased their multiplicity with one.

Figure 2.6: NURBS surface of a cylinder with displayed control points, before and after refinement algorithm.

knots, while knot insertion reduces continuity at the point where it is inserted, it is important to note that the two refinement sequences are not interchangeable. An example in Figure 2.7 shows that combining the two methods of refinement will give two different outcomes for the basis functions. The example shows the linear basis functions, $p = 1$, elevated to quadratic basis functions, $p = 2$, together with knot insertion of two knots.

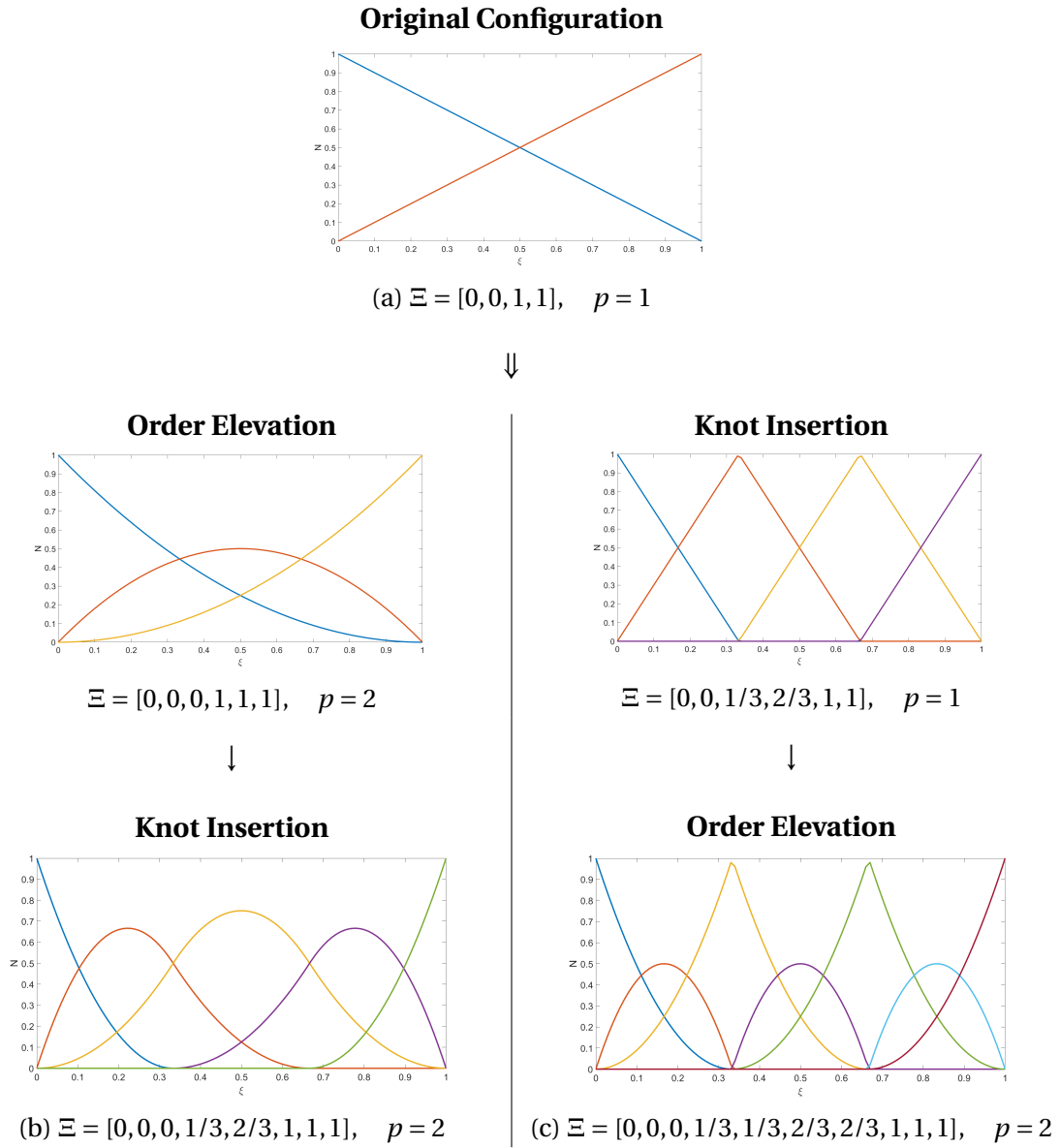


Figure 2.7: Two cases of order elevation and knot insertion. (a) shows the original configuration for the basis functions that is to be refined. (b) is the sequence where the basis functions is order elevated first, then the knot is inserted. This is what one refers to as k-refinement. (c) is the sequence when the basis functions has a knot insertion first, then order elevated.

Chapter 3

Isogeometric Analysis

Hughes et al. (2005) defined the term "isogeometric analysis" as a model who utilizes the same mathematical description in both the analysis and geometry. In traditional finite element analysis, FEA, one typically has basis functions of lower order, mostly linear, Lagrange polynomials, whereas techniques involving higher order, spline functions, are applied in CAD. This creates a barrier between the modelling in the two platforms, which is solved in the meshing process, where the geometry model is divided into a mesh of finite elements. Now the finite element model only becomes an approximation of the true model, which may cause problems if the geometry has small imperfections that governs structural behaviour like buckling of thin shells. Another aspect is the time consumption of the meshing, Smith (2012) states that automatic meshing techniques is limiting with respect to element types, especially for complex geometries. Also, the process needs to be redone every time the model is modified. With an analysis method who uses the same basis functions as the geometry, the meshing process is omitted and the design and analysis is merged into the same model.

3.1 NURBS-Based Isogeometric Analysis

In order to be applicable as basis functions in isogeometric analysis, conditions like partition of unity and linear independence needs to be fulfilled (Hughes et al., 2005). NURBS basis functions, who also is the standard in CAD, acts according to these conditions and is thereby adoptable in analysis. Similarly to the FEA formulation, one utilizes elements in isogeometric analysis. However, the definition of the boundary between elements can be interoperated in two ways. The first is to define the full NURBS patch as a single element. This thesis will use the same definition of Kiendl (2011), which is between knot spans in the parametric space of each patch. This division of elements can be seen in Figure 3.1. The reason for this definition is that inside a knot span, the B-spline basis functions polynomials, implying that gauss quadrature can be used on element level. NURBS basis functions are rational polynomials, meaning gauss quadrature is only an approximation for NURBS basis functions. However, in a conventional FEM formulation with mapped elements one always end up with integrating rational polynomials, due to the inverse of the Jacobian. This im-

plies that using gauss quadrature for NURBS elements are reliable as well, which is stated by Hughes et al. (2010). Similarly to finite element, NURBS are defined by a set of nodes and corresponding basis functions. However, for NURBS the control points acts as nodes. This means that translations in x-, y- and z-direction, as well as boundary conditions are carried in the control points.

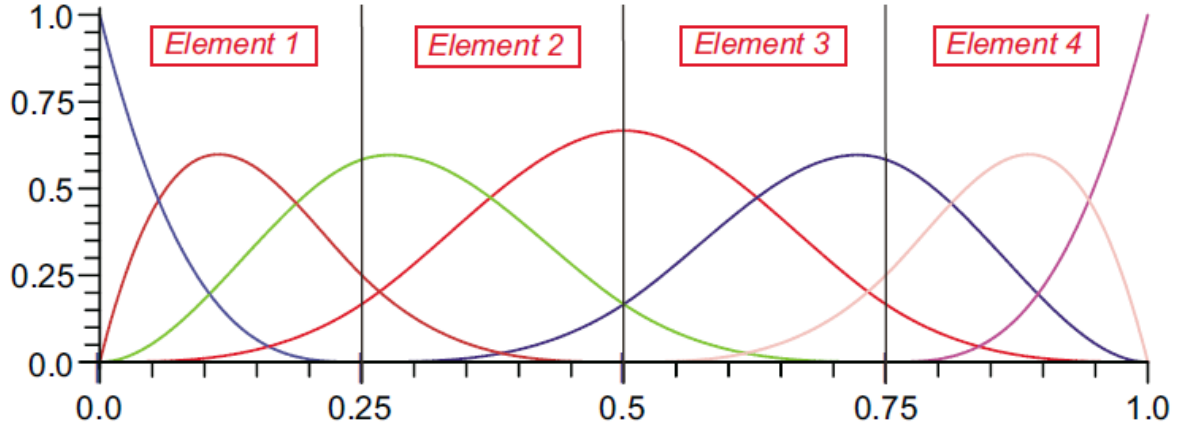


Figure 3.1: Division of elements in NURBS basis functions (Kiendl, 2011).

Furthermore, it is important to note that the basis functions are not confined to one element, but extends over a series of elements. As stated in Equation 2.12, the basis functions inside a knot span is defined by the Cox-deBoor recursion formula and depended on neighbouring knot spans. This implies that it is not possible to define a single element of a NURBS curve without knowing the complete patch. The stiffness matrix is shown as an example to state why an element formulation is important. In classical FEM, the stiffness matrix for each element is defined and then assembled to a global stiffness matrix. The same method can be applied on a NURBS patch, where the stiffness matrix can be evaluated on element level and then assembled to a global level. This enables the NURBS-elements to be treated the same way as in a finite element code. The only difference is the use of basis functions, who in NURBS-elements has higher-order continuities between elements. Also, locking effects from low-order basis functions can be precluded ab initio.

With respect to refinement, NURBS-based isogeometric analysis uses the method of knot insertion and order elevation. In classic finite element analysis, the two refinement methods corresponds to h-refinement and p-refinement, respectively. Unless low continuities are desired, k-refinement referred to in Section 2.2 is applied in NURBS based isogeometric analysis. In contrast to FEA, the geometry is not changed when refinement of NURBS geometries are involved. This means that further refinement does not need reference in the original geometry. Moreover, Hughes et al. (2005) states that in contrast to finite element analysis, NURBS-refinement follow patches, i.e. a knot insertion in Ξ - or H-direction at one edge, would imply knot insertion at the opposing edge, as can be seen in Figure 3.2. This

illustrates the challenge of local NURBS refinement.

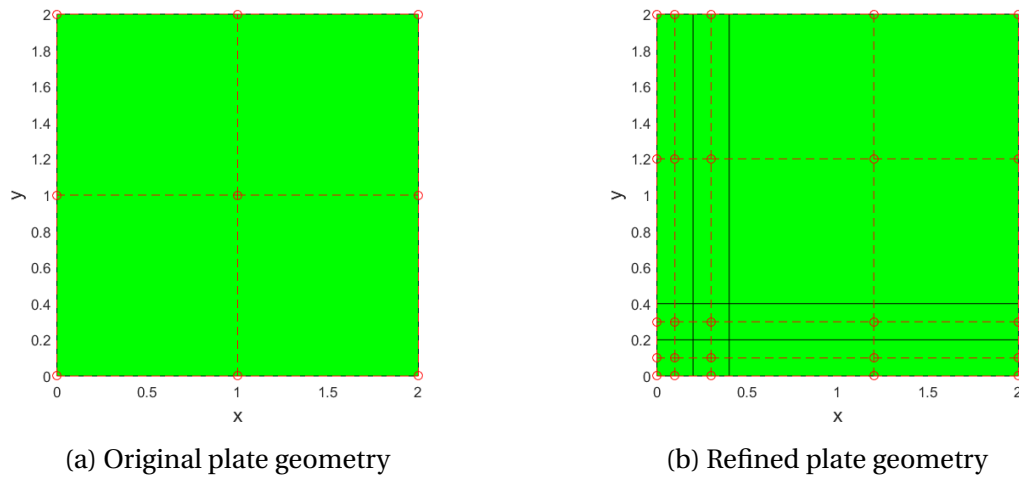


Figure 3.2: Refinement of plate in a single direction

Hughes et al. (2005) summarizes the comparison of finite element analysis and isogeometric analysis based on NURBS, which is presented in Table 3.1

Finite element analysis	Isogeometric analysis
Nodal points	Control points
Nodal variable	Control variables
Mesh	Knots
h-refinement	Knot insertion
p-refinement	Order elevation
Basis interpolates nodal points and variables	Basis does <i>not</i> interpolate control points and variables
Approximate geometry	Exact geometry
Polynomial basis	NURBS basis
Sub domains	Patches

Table 3.1: Comparison of finite element analysis and isogeometric analysis

3.2 Differential Geometry of Surfaces

Differential geometry of surfaces is the subfield geometry that deals with determining metric properties like lengths, curvature, angle and areas of curved surfaces. In this section, the fundamental forms of a surface utilized in Section 3.3 will be presented and explained. The equations used for deriving the expressions of the first and second fundamental forms are presented by Klingenberg (1978).

Arfken and Weber (1999) describes an arbitrary vector as a linear combination of the basis vectors \mathbf{e} and their coordinate coefficients x as:

$$\mathbf{x} = x^1 \mathbf{e}_1 + x^2 \mathbf{e}_2 + x^3 \mathbf{e}_3 = x^i \mathbf{e}_i \quad (3.1)$$

When summing vectors of the same upper and lower index, the vector can be written on a more compact form with the Einstein summation convention. In Equation 3.1, the same index is seen in both the coordinate and base vectors, i.e. Einstein summing convention is applicable. Furthermore, in this and the following section, indices in Latin letters (i and j) takes the values $\{1,2,3\}$, whereas Greek letters (α and β) takes the values $\{1,2\}$. Moreover, italic symbols indicate scalars and bold symbols denotes a tensor or a vector.

In order to describe a point on a free-form geometry like a surface, it is advantageous to use curvilinear coordinates and local bases. The two important bases are the *covariant* \mathbf{g}_i and the *contravariant* \mathbf{g}^i basis. Subscript indices denotes covariant and superscript indices denotes contravariant. Now an arbitrary vector is expressed as the covariant and contravariant base vectors with their corresponding surface coordinates θ^i and θ_i , as:

$$\mathbf{x} = \theta^i \mathbf{g}_i = \theta_i \mathbf{g}^i \quad (3.2)$$

where θ^α are surface coordinates and θ^3 are thickness coordinates. Furthermore, the covariant base vector is defined as the partial derivative of the position vector with respect to corresponding contravariant surface coordinates:

$$\mathbf{g}_i = \frac{\partial \mathbf{x}}{\partial \theta^i} \quad (3.3)$$

Worth noting about the two bases is that they are related by the Kronecker delta δ_i^j function:

$$\mathbf{g}_i \cdot \mathbf{g}^j = \delta_i^j = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases} \quad (3.4)$$

This relation reveals that the covariant and contravariant basis are orthogonal if $i \neq j$. Furthermore, the third covariant base vector \mathbf{g}_3 is defined as the normalized vector orthogonal to \mathbf{g}_1 and \mathbf{g}_2 :

$$\mathbf{g}_3 = \frac{\mathbf{g}_1 \times \mathbf{g}_2}{|\mathbf{g}_1 \times \mathbf{g}_2|} \quad (3.5)$$

As shown in Equation 3.5, the contravariant base vector \mathbf{g}^α lie in the tangential plane spanned by the covariant base vectors. Therefore, the third contravariant base vector \mathbf{g}^3 must be equal to \mathbf{g}_3 :

$$\mathbf{g}^3 = \mathbf{g}_3 \quad (3.6)$$

The covariant base vectors is utilized to define a local Cartesian basis. The first base vector \mathbf{e}_1 is parallel to \mathbf{g}_1 , expressed mathematically as:

$$\mathbf{e}_1 = \frac{\mathbf{g}_1}{\|\mathbf{g}_1\|} \quad (3.7)$$

where the vector inside double absolute signs indicates the norm. Furthermore, \mathbf{e}_2 is orthogonal to \mathbf{e}_1 and in the plane of the covariant base vectors.

$$\mathbf{e}_2 = \frac{\mathbf{g}_2 - (\mathbf{g}_2 \cdot \mathbf{e}_1)\mathbf{e}_1}{\|\mathbf{g}_2 - (\mathbf{g}_2 \cdot \mathbf{e}_1)\mathbf{e}_1\|} \quad (3.8)$$

The third base vector is equal to \mathbf{g}_3 :

$$\mathbf{e}_3 = \mathbf{g}_3 \quad (3.9)$$

Furthermore, the metric tensor \mathbf{g} is defined as an expression in the covariant and contravariant basis:

$$\mathbf{g} = g^{\alpha\beta} \mathbf{g}_\alpha \otimes \mathbf{g}_\beta = g_{\alpha\beta} \mathbf{g}^\alpha \otimes \mathbf{g}^\beta \quad (3.10)$$

where $g_{\alpha\beta}$ is the covariant metric coefficient computed by the scalar product of covariant base vectors:

$$g_{\alpha\beta} = \mathbf{g}_\alpha \cdot \mathbf{g}_\beta \quad (3.11)$$

The relation in Equation 3.11 is commonly known as the *first fundamental form of surfaces*. According to Weisstein, the first fundamental form is extremely important and useful in determining the metric properties, such as length and area of a surface. The contravariant metric coefficients $g^{\alpha\beta}$ is obtained by the inverse of the covariant coefficient matrix:

$$[g^{\alpha\beta}] = [g_{\alpha\beta}]^{-1} \quad (3.12)$$

The contravariant base vector is then computed from the contravariant metric coefficient matrix:

$$\mathbf{g}^\alpha = g^{\alpha\beta} \mathbf{g}_\beta \quad (3.13)$$

and vice versa:

$$\mathbf{g}_\alpha = g_{\alpha\beta} \mathbf{g}^\beta \quad (3.14)$$

Finally, the *second fundamental form of surfaces* is obtained to describe the curvature of a surface. The curvature tensor coefficients $b_{\alpha\beta}$ is retrieved from Klingenberg (1978) and defined as:

$$b_{\alpha\beta} = -\mathbf{g}_\alpha \cdot \mathbf{g}_{3,\beta} = -\mathbf{g}_\beta \cdot \mathbf{g}_{3,\alpha} = \mathbf{g}_{\alpha,\beta} \cdot \mathbf{g}_3 \quad (3.15)$$

The metric and curvature metric coefficients will be utilized in Section 3.3.

3.3 NURBS-Based Kirchhoff-Love Shell Element

Kiendl (2011) developed a shell element based on the Kirchhoff-Love shell theory with NURBS basis functions. In Kirchhoff-Love theory, curvature is expressed as the second derivative of the geometry description. This means that C^1 -continuity is required between elements in order to fulfill the compatibility conditions. With Lagrange polynomials as basis functions, this is generally not possible, unless higher order elements of rectangular shape is applied. In finite element, the Reissner-Mindlin shell formulation is more common, since transverse shear strain is accounted for and C^0 -continuity is sufficient between elements. With NURBS basis functions, the Kirchhoff-Love theory can be implemented without additional modification, due to the high continuity between the element.

The NURBS-based Kirchhoff-Love shell needs neither rotational degrees of freedom nor nodal directors. However, in clamped boundary condition and coupling between NURBS patches, the rotations needs to be described at the shells boundary. For the clamped boundary condition, this is solved by fixing the first two rows of control points along the fixed edge. It is important to note that by fixing the second control point does not fix the displacements inside the plate, but only the rotation at the boundary. For coupling of NURBS patches in this thesis, two methods will be reviewed in Section 3.4.

For the detailed formulation of the NURBS-based Kirchhoff-Love shell formulation, the reader is referred to Kiendl (2011). Only an outline of the formulation will be highlighted here. One distinguishes between undeformed and deformed configuration by referring to the two configurations with upper and lower case letters, respectively. Other notations and definitions are described in Section 3.2. In Kirchhoff-Love shell theory transverse shear deformation is neglected, i.e. plane surfaces remain plane. Due to this assumption, every point in the shell continuum can be described by the middle surface and its normal vector. The base vectors \mathbf{a}_i on the middle surface ($\theta^3 = 0$) are obtained by:

$$\mathbf{a}_\alpha = \mathbf{x}_{,\alpha} \quad (3.16)$$

$$\mathbf{a}_3 = \frac{\mathbf{a}_1 \times \mathbf{a}_2}{|\mathbf{a}_1 \times \mathbf{a}_2|} \quad (3.17)$$

With an expression for the base vectors, Equations 3.11 and 3.15 defines the metric and curvature of the shell:

$$a_{\alpha\beta} = \mathbf{a}_\alpha \cdot \mathbf{a}_\beta \quad (3.18)$$

$$b_{\alpha\beta} = -\mathbf{b}_\alpha \cdot \mathbf{b}_{3,\beta} = -\mathbf{b}_\beta \cdot \mathbf{b}_{3,\alpha} = \mathbf{b}_{\alpha,\beta} \cdot \mathbf{b}_3 \quad (3.19)$$

For a thin and moderately thick shell, strains in the shell represented by the metric coefficient $a_{\alpha\beta}$ and curvature coefficient $b_{\alpha\beta}$ in the middle surface are given by:

$$E_{\alpha\beta} = \frac{1}{2}(a_{\alpha\beta} - A_{\alpha\beta}) + \theta^3(B_{\alpha\beta} - b_{\alpha\beta}) \quad (3.20)$$

The strains consists of a constant and a linear part, where the constant membrane strains $\epsilon_{\alpha\beta}$ describes the the strains in the middle surface and are defined as:

$$\epsilon_{\alpha\beta} = \frac{1}{2}(a_{\alpha\beta} - A_{\alpha\beta}) \quad (3.21)$$

The linear part represents the change in curvature $\kappa_{\alpha\beta}$ as an effect of bending and is defined as:

$$\kappa_{\alpha\beta} = B_{\alpha\beta} - b_{\alpha\beta} \quad (3.22)$$

Now the strains in equation 3.20 can be rewritten as:

$$E_{\alpha\beta} = \epsilon_{\alpha\beta} + \theta^3 \kappa_{\alpha\beta} \quad (3.23)$$

Note that equations 3.22-3.23 are non-linear measures and that for linearized analysis, the membrane strains and curvature is expressed in terms of the displacement and geometric variables of the undeformed configuration as in Kiendl et al. (2016):

$$\epsilon_{\alpha\beta}^{m,lin} = \frac{1}{2}(\mathbf{A}_\beta \cdot \mathbf{u}_{,\alpha} + \mathbf{A}_\alpha \cdot \mathbf{u}_{,\beta}) \quad (3.24)$$

$$\kappa_{\alpha\beta}^{lin} = -\mathbf{A}_3 \cdot \mathbf{u}_{,\alpha\beta} + \frac{1}{|A_1 \times A_2|} \left((\mathbf{X}_1 \times \mathbf{X}_{\alpha\beta}) \cdot \mathbf{u}_{,2} + \mathbf{X}_{,\alpha\beta} \cdot \mathbf{A}_3 \left((\mathbf{A}_2 \times \mathbf{A}_3) \cdot \mathbf{u}_{,1} + (\mathbf{A}_3 \times \mathbf{A}_1) \cdot \mathbf{u}_{,2} \right) \right) \quad (3.25)$$

The linearized membrane strains and curvature will be applied in this thesis. Furthermore, the strain and the stress tensor are connected by the material matrix \mathbf{D}^{mat} . Since the material matrix has physical material parameters referring to the local Cartesian coordinate system, the stresses and strains are transformed to coefficients with a local Cartesian basis with a transformation rule. These coefficients are fitted with an upper bar, which gives the second Piola-Kirchhoff stresses:

$$\begin{bmatrix} \bar{S}^{11} \\ \bar{S}^{22} \\ \bar{S}^{12} \end{bmatrix} = \mathbf{D}^{mat} \cdot \begin{bmatrix} \bar{E}^{11} \\ \bar{E}^{22} \\ 2\bar{E}^{12} \end{bmatrix} \quad (3.26)$$

In this thesis, an orthotropic material matrix is applied and will be described in detail in section 4.2. Here the extensional stiffness \mathbf{A} , the coupling stiffness \mathbf{B} and the bending stiffness \mathbf{D} is derived and shown to make up the laminate stiffness matrix. The matrices expresses the normal forces and bending moments as:

$$\begin{bmatrix} \bar{n}^{11} \\ \bar{n}^{22} \\ \bar{n}^{12} \end{bmatrix} = \mathbf{A} \cdot \begin{bmatrix} \bar{\epsilon}_{11} \\ \bar{\epsilon}_{22} \\ 2\bar{\epsilon}_{12} \end{bmatrix} + \mathbf{B} \cdot \begin{bmatrix} \bar{\kappa}_{11} \\ \bar{\kappa}_{22} \\ 2\bar{\kappa}_{12} \end{bmatrix} \quad (3.27)$$

$$\begin{bmatrix} \bar{m}^{11} \\ \bar{m}^{22} \\ \bar{m}^{12} \end{bmatrix} = \mathbf{B} \cdot \begin{bmatrix} \bar{\epsilon}_{11} \\ \bar{\epsilon}_{22} \\ 2\bar{\epsilon}_{12} \end{bmatrix} + \mathbf{D} \cdot \begin{bmatrix} \bar{\kappa}_{11} \\ \bar{\kappa}_{22} \\ 2\bar{\kappa}_{12} \end{bmatrix} \quad (3.28)$$

A virtual displacement is a change in the geometric configuration of the body compatible with all kinematic constraints, made while keeping all forces and stresses on the body frozen. With reference to Kiendl (2011), the kinematics of the NURBS-based Kirchhoff-Love shell element is derived using the principle of virtual work at the equilibrium condition ($\delta W = 0$), forcing it to be fulfilled for any arbitrary variation of the displacement. The linearised internal stiffness $K_{rs}^{int,lin}$ is then given as:

$$K_{rs}^{int,lin} = \int_A \left(\frac{\partial \mathbf{n}}{\partial u_s} : \frac{\partial \boldsymbol{\epsilon}}{\partial u_r} + \frac{\partial \mathbf{m}}{\partial u_s} : \frac{\partial \boldsymbol{\kappa}}{\partial u_r} \right) dA, \quad (3.29)$$

which leads to the equation system for a geometrically linear problem:

$$\mathbf{K}^{lin} \mathbf{u} = \mathbf{F}^{ext}, \quad (3.30)$$

where \mathbf{F}^{ext} is the external load vector, defined from the applied load.

3.4 Multipatch Coupling

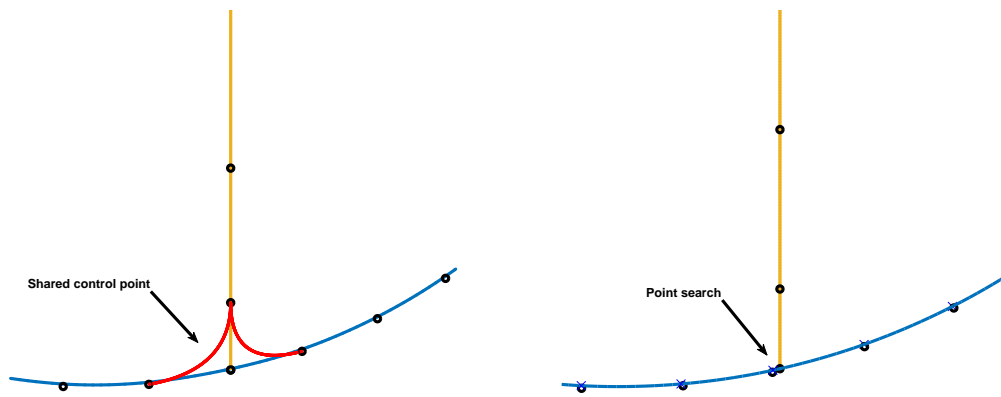
For complex geometric structural models, multiple patches have to be considered, because a single NURBS patch cannot represent geometries of arbitrary complexity. For multipatch NURBS surfaces, rotation-free IGA elements require special treatment as patch boundaries where the basis functions are C^0 -continuous. Apostolatos et al. (2014) presents three *domain decomposition methods* (DDMs) for solving the boundary value problem in multiple domains. The interest reader is referred to Toselli and Widlund (2006) for more theory on DDMs. For coupling and managing rotations along two NURBS patches, the popular penalty method is presented. The method is widely used because of its simple implementation, where the size of the equation system remains unaltered. The main drawback is that the equation system rapidly becomes ill-conditioned, and is highly dependant on the penalty parameter. The involved parameter needs to be set constant, and is in this thesis chosen as $\alpha = 10^3$.

In this thesis, a point search algorithm is applied to find the penalty coupling point. The point search algorithm utilize orthogonal projection, which is explained by Ko and Sakkalis (2014) to be a process of finding a point on a curve or a surface such that the vector connecting the point in space and the point on the curve or the surface becomes perpendicular to the curve or the surface. The approach applied for this thesis is the one proposed by Piegl and Tiller (1995). As input, the algorithm requires a tolerance for the angle, tolerance for the distance and an initial guess. The tolerance of the angle decides how much the orthogonal tangent may deviate, and the tolerance distance tells how far the projected point is allowed to be. Initial guess tells where in the parametric space of the opposite surface or curve the algorithm shall start its iterations. This has been an important subject in this thesis, as there might be two or more points on a surface that fulfils the orthogonality condition. Figure 3.3b shows the coupling condition by penalty method and point search.

Another coupling between patches is with the so called *bending strip* method. A bending strip is an additional patch which constrains the angle between two patches. Bending strips does not represent structural parts but only constraints on the angle between patches. This is due to the special material properties of this additional patch, which is no membrane stiffness and bending stiffness only in the direction transverse to the interface, i.e the mending strip stiffness is given as:

$$\begin{bmatrix} \bar{S}^{11} \\ \bar{S}^{22} \\ \bar{S}^{12} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{\epsilon}^{11} \\ \bar{\epsilon}^{22} \\ \bar{\epsilon}^{12} \end{bmatrix} + \theta^3 \begin{bmatrix} E_s & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{\kappa}^{11} \\ \bar{\kappa}^{22} \\ 2\bar{\kappa}^{12} \end{bmatrix} \quad (3.31)$$

The bending strip method is tested in Kiendl et al. (2009) and proven to be reliable, and it can handle both smooth geometries and patches forming a kink. Due to the regular topology of NURBS, the bending strips can be created automatically. They do not necessarily need to be created in CAD, but can be added during analysis. In the analysis, a bending strip patch is created from a common interface or shared control point between the patches and the two next control points of the respective patches. Figure 3.3a shows how the bending strips are implemented in analysis. The yellow and blue curves are separate patches, and the red curves are bending strips. The shared control point is highlighted.



(a) Coupling with bending strip method and (b) Coupling with penalty method and point shared control point.

Figure 3.3: 2D view of multipatch coupling with bending strip method and non-matching penalty method.

Chapter 4

Structural Mechanics of Orthotropic Materials

Several textbooks (González-Viñas and Mancini (2015); see also Hearn (1977); Timoshenko and Woinowsky-Krieger (1959)) explain isotropic materials as materials with same elastic properties in all three directions, i.e. the deformation in response to a load is invariant with respect to direction. If an object has different material properties varying with the direction they are measured, the material is said to be anisotropic. Furthermore, an orthotropic material is a subset of anisotropic materials, but they differ along three mutually-orthogonal twofold axes of rotational symmetry. In the majority of anisotropic structural materials, the anisotropy is orthogonal, i.e. the properties are uniform and differ only in three or two orthogonal directions. Gopalakrishnan (2016) states that laminated composites are understood to act orthotropic on a lamina level, and anisotropic on a laminate level. The anisotropic behaviour results in a bending-axial-shear coupling in beams and plates, that makes analysis of laminated composites less comprehensible than isotropic structures. In this chapter, theory for performing a simple micro-mechanical analysis will be supplied. Also, classical lamination plate theory is presented as an option to determine normal forces and bending moments imposed on a composite laminate.

When evaluating an orthotropic material, subscripts are important to distinguish direction type of property, direction and plane. In the micro-mechanical equations, subscript f and m are used for fibre and matrix, respectively. In addition, properties along the fibre direction is denoted with subscript 1 and are conventionally called longitudinal or radial direction. Similarly, those transverse to the fibre direction have subscript 2, and are called transverse or circumferential direction. Furthermore, radial-tangential and vice versa are denoted 1-2 and 2-3, respectively. This notation also applies for the classical lamination plate theory.

4.1 Micromechanics Theory

In the case of a composite, the materials under consideration have different material properties. In micromechanics theory, the *rule of mixtures* is used to obtain material properties at the lamina level. From the individual mechanical properties of a matrix and fibre material, mechanical properties of a unidirectional lamina are computed using a set of micromechanical equations of simple form. Several textbooks present these equations, where Chamis (1983) is one of them. First the apparent Young's modulus is defined as:

$$E_1 = E_{f1} V_f + E_m V_m \quad (4.1)$$

where E_{f1} is the longitudinal Young's modulus of the fibres, E_m is the Young's modulus of the matrix and V_f and V_m is the volume fraction of fibres and the matrix, respectively. Furthermore, the transverse Young's modulus is computed as:

$$E_2 = \frac{E_m}{1 - \sqrt{V_f} \left(1 - \frac{E_m}{E_{f2}}\right)} \quad (4.2)$$

where E_{f2} is the transverse modulus of the fibres. Moreover, the in-plane and transverse shear modulus is computed as:

$$G_{12} = \frac{G_m}{1 - \sqrt{V_f} \left(1 - \frac{G_m}{G_{f12}}\right)} \quad (4.3)$$

and

$$G_{23} = \frac{G_m}{1 - \sqrt{V_f} \left(1 - \frac{G_m}{G_{f23}}\right)} \quad (4.4)$$

where G_{f12} and G_{f23} is the longitudinal and transverse shear modulus of the fibres, and G_m is the shear modulus of the matrix. The in-plane Poisson's ratio, ν_{12} , for the unidirectional lamina is computed as:

$$\nu_{12} = \nu_{f12} V_f + \nu_m V_m \quad (4.5)$$

where ν_{f12} is the major Poisson's ratio of the fibres, and ν_m is the Poisson's ratio of the matrix. By using the Young's modulus transverse to the fibres and out-of-plane shear modulus in Equation 4.2 and 4.4, the out-of-plane Poisson's ratio is computed as:

$$\nu_{23} = \frac{E_2}{2G_{23}} \quad (4.6)$$

Then the mass density of the unidirectional lamina is computed as:

$$\rho = \rho_f V_f + \rho_m V_m \quad (4.7)$$

where ρ_f and ρ_m is the mass density of the fibres and matrix, respectively. Finally, the remaining elastic properties are decided due to symmetry considerations:

$$E_3 = E_2 \quad (4.8)$$

$$G_{13} = G_{12} \quad (4.9)$$

$$\nu_{13} = \nu_{12} \quad (4.10)$$

4.2 Classical Lamination Plate Theory

Roylance (2000) outlines the mechanics of fiber-reinforced laminated plates, leading to a computational scheme that relates the in-plane strain and curvature of a laminate to the normal forces and bending moments imposed on it. This is done by retrieving the extensional-, coupling- and bending stiffness matrix from each individual layup in a laminate. In this section a short derivation and the important steps of how to retrieve the mentioned stiffness matrices through classical lamination plate theory are presented.

In a state of plane stress ($\sigma_{33} = \sigma_{13} = \sigma_{23} = 0$), one writes the strain-stress relation of an anisotropic material ($E_1 \neq E_2 \neq E_3$) as:

$$\begin{Bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ \gamma_{12} \end{Bmatrix} = \begin{bmatrix} 1/E_1 & \nu_{21}/E_2 & 0 \\ \nu_{12}/E_1 & 1/E_2 & 0 \\ 0 & 0 & 1/G_{12} \end{bmatrix} \begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \tau_{12} \end{Bmatrix} \quad (4.11)$$

where E_1, E_2 are the Young's moduli, ν_{12}, ν_{21} Poisson's ratios and the matrix in the brackets are known as the *compliance stiffness matrix*. Next, one has to transform the axes between the global x-y frame and the natural material frame with axes labelled 1-2, corresponding to the longitudinal and transverse fibre direction. The stresses in the two configurations are illustrated in Figure 4.1.

The transformation law for Cartesian Cauchy stress in two dimensions (plane stress) can be used to transform the stresses with the lamina principal axes to the global axes:

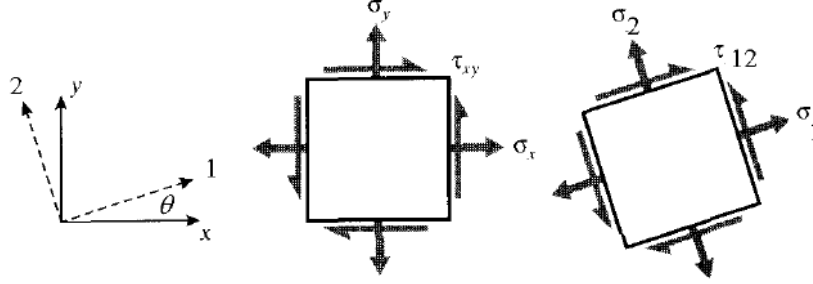


Figure 4.1: Transformation between global and fibre axis (Roylance, 2000).

$$\begin{Bmatrix} \sigma_{11} \\ \sigma_{22} \\ \tau_{12} \end{Bmatrix} = \begin{bmatrix} c^2 & s^2 & 2sc \\ s^2 & c^2 & -2sc \\ -sc & sc & c^2 - s^2 \end{bmatrix} \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{Bmatrix} \quad (4.12)$$

where $c = \cos\theta$, $s = \sin\theta$ and θ is the angle between the x- and the 1 axis. The transformation matrix \mathbf{T} can be used to transform infinitely small strain components. Next, the strains are related in a similar manner:

$$\begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \frac{1}{2}\gamma_{12} \end{Bmatrix} = \mathbf{T} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \frac{1}{2}\gamma_{xy} \end{Bmatrix} \quad (4.13)$$

Note that in order to use the same transformation matrix, the shear component is multiplied by a factor of $\frac{1}{2}$. Without going into much detail, this is related to the classical definition of shear strain, which is twice the tensorial shear strain. This introduces some awkwardness in the transformation, which is repaired with by introducing the *Reuter's matrix*, \mathbf{R} , defined as:

$$[\mathbf{R}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix} \quad \text{and} \quad [\mathbf{R}]^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{2} \end{bmatrix} \quad (4.14)$$

Now the relationship between the strains in an arbitrary x-y direction can be related to the strains in the 1-2 direction with the transformation matrix \mathbf{T} and the Reuter's matrix \mathbf{R} :

$$\begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \gamma_{12} \end{Bmatrix} = \mathbf{R} \begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \frac{1}{2}\gamma_{12} \end{Bmatrix} = \mathbf{RT} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \frac{1}{2}\gamma_{xy} \end{Bmatrix} = \mathbf{RTR}^{-1} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} \quad (4.15)$$

By means of the transformation and compliance stiffness matrix in the 1-2 configuration, the x-y strains are related to the x-y stresses:

$$\begin{aligned}
\begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} &= \mathbf{R} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \frac{1}{2}\gamma_{xy} \end{Bmatrix} = \mathbf{R}\mathbf{T}^{-1} \begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \frac{1}{2}\gamma_{12} \end{Bmatrix} = \mathbf{R}\mathbf{T}^{-1}\mathbf{R}^{-1} \begin{Bmatrix} \epsilon_1 \\ \epsilon_2 \\ \gamma_{12} \end{Bmatrix} \\
&= \mathbf{R}\mathbf{T}^{-1}\mathbf{R}^{-1}\mathbf{S} \begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{Bmatrix} = \mathbf{R}\mathbf{T}^{-1}\mathbf{R}^{-1}\mathbf{S}\mathbf{T} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} \equiv \bar{\mathbf{S}} \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix}
\end{aligned}$$

where $\bar{\mathbf{S}}$ is known as the *transformed compliance matrix*. Furthermore, the stiffness matrix $\bar{\mathbf{D}}$ is derived from the inverse of the transformed compliance matrix. This is summarized by the two equations:

$$\bar{\mathbf{S}} = \mathbf{R}\mathbf{T}^{-1}\mathbf{R}^{-1}\mathbf{S}\mathbf{T} \quad (4.16)$$

$$\bar{\mathbf{D}} = \bar{\mathbf{S}}^{-1} \quad (4.17)$$

Now that the transformed stiffness $\bar{\mathbf{D}}$ of a single ply is obtained, the *laminate stiffness matrix* can be derived from the classical lamination plate theory. We allow in-plane normal forces and bending moments, which gives the normal forces \mathbf{n} and bending moments \mathbf{m} at a position x, y :

$$\mathbf{n} = \begin{Bmatrix} n_x \\ n_y \\ n_{xy} \end{Bmatrix} \quad \text{and} \quad \mathbf{m} = \begin{Bmatrix} m_x \\ m_y \\ m_{xy} \end{Bmatrix} \quad (4.18)$$

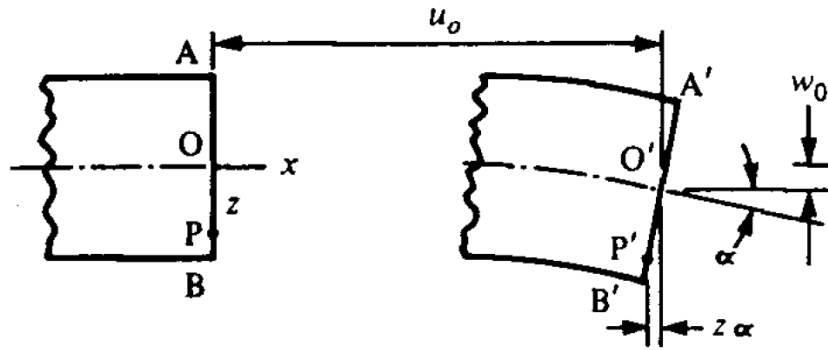


Figure 4.2: Deformations associated with Kirchhoff plate theory (Powell and Housz, 1998)

with the well known Kirchhoff assumption in Figure 4.2, the strains can be expressed as the gradient of the displacement as:

$$\begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} = \begin{Bmatrix} u_{,x} \\ v_{,y} \\ u_{,y} + v_{,x} \end{Bmatrix} = \begin{Bmatrix} u_{0,x} - zw_{0,xx} \\ v_{0,y} - zw_{0,yy} \\ (u_{0,y} + v_{0,x}) - 2zw_{0,xy} \end{Bmatrix} = \boldsymbol{\epsilon}^0 + z\boldsymbol{\kappa} \quad (4.19)$$

where $\boldsymbol{\epsilon}^0$ is the midplane strain and $\boldsymbol{\kappa}$ is curvature, expressed as the second derivative of the displacement and has components in x-, y- and xy-direction. Here κ_{xy} is known as the twisting curvature. Thus, stresses $\boldsymbol{\sigma}$ in the k th lamina at any vertical position is expressed with the transformed stiffness matrix and strains as:

$$\boldsymbol{\sigma}_k = \bar{\mathbf{D}}_k \boldsymbol{\epsilon}^0 + z \bar{\mathbf{D}}_k \boldsymbol{\kappa} \quad (4.20)$$

Furthermore, the normal forces are balanced by taking the integral of the stresses in the laminate over the height h . This is equivalent to adding up the stresses of each ply. The distance z is the vertical distance from the bottom of respective ply to the midpoint of the laminate and the index k marks the ply number:

$$\mathbf{n} = \int_{-h/2}^{+h/2} \boldsymbol{\sigma} dz \quad (4.21)$$

$$= \sum_{k=1}^N \int_{z_k}^{z_{k+1}} \boldsymbol{\sigma}_k dz \quad (4.22)$$

$$= \sum_{k=1}^N \left(\int_{z_k}^{z_{k+1}} \bar{\mathbf{D}}_k \boldsymbol{\epsilon}^0 dz + \int_{z_k}^{z_{k+1}} \bar{\mathbf{D}}_k \boldsymbol{\kappa} z dz \right) \quad (4.23)$$

$$= \sum_{k=1}^N \left(\bar{\mathbf{D}}_k \boldsymbol{\epsilon}^0 \int_{z_k}^{z_{k+1}} dz + \bar{\mathbf{D}}_k \boldsymbol{\kappa} \int_{z_k}^{z_{k+1}} z dz \right) \quad (4.24)$$

$$= \mathbf{A} \boldsymbol{\epsilon}^0 + \mathbf{B} \boldsymbol{\kappa} \quad (4.25)$$

where \mathbf{A} and \mathbf{B} is the extensional and coupling stiffness matrix, respectively, defined as:

$$\mathbf{A} = \sum_{k=1}^N \bar{\mathbf{D}}_k (z_{k+1} - z_k) \quad (4.26)$$

and

$$\mathbf{B} = \frac{1}{2} \sum_{k=1}^N \bar{\mathbf{D}}_k (z_{k+1}^2 - z_k^2) \quad (4.27)$$

Note that the z inside the integral of the curvature term is the contributes to a squared z term in the expression for the coupling stiffness matrix.

Similarly to the normal forces, the bending moments are balanced by the internal moments caused by the internal stresses:

$$\mathbf{m} = \int_{-h/2}^{+h/2} \boldsymbol{\sigma} z \, dz \quad (4.28)$$

$$= \sum_{k=1}^N \int_{z_k}^{z_{k+1}} \boldsymbol{\sigma}_k z \, dz \quad (4.29)$$

$$= \sum_{k=1}^N \left(\int_{z_k}^{z_{k+1}} \bar{\mathbf{D}}_k \boldsymbol{\epsilon}^0 z \, dz + \int_{z_k}^{z_{k+1}} \bar{\mathbf{D}}_k \boldsymbol{\kappa} z^2 \, dz \right) \quad (4.30)$$

$$= \sum_{k=1}^N \left(\bar{\mathbf{D}}_k \boldsymbol{\epsilon}^0 \int_{z_k}^{z_{k+1}} z \, dz + \bar{\mathbf{D}}_k \boldsymbol{\kappa} \int_{z_k}^{z_{k+1}} z^2 \, dz \right) \quad (4.31)$$

$$= \mathbf{B} \boldsymbol{\epsilon}^0 + \mathbf{D} \boldsymbol{\kappa} \quad (4.32)$$

where \mathbf{D} is the bending stiffness matrix:

$$\mathbf{D} = \frac{1}{3} \sum_{k=1}^N \bar{\mathbf{D}}_k (z_{k+1}^3 - z_k^3) \quad (4.33)$$

Now the squared z term inside the curvature term contributes to a cubed z -contribution in the bending stiffness matrix. When an expression for the \mathbf{A} , \mathbf{B} and \mathbf{D} are established, forces and moments can be expressed as:

$$\begin{Bmatrix} \mathbf{n} \\ \mathbf{m} \end{Bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B} & \mathbf{D} \end{bmatrix} \begin{Bmatrix} \boldsymbol{\epsilon}^0 \\ \boldsymbol{\kappa} \end{Bmatrix} \quad (4.34)$$

The matrix inside the brackets is known as the *laminate stiffness matrix*, and can be used in the shell formulation without any further modifications.

The inverse of the laminate stiffness matrix is known as the *laminate compliance matrix*. By examining the (1,1) and (2,2) elements of the laminate compliance matrix, the *effective modulus* of the laminate can be determined with $E_x = 1/a_{11}$ and $E_y = 1/a_{22}$ where a_{11} and a_{22} are the (1,1) and (2,2) elements of the laminate compliance matrix.

Chapter 5

Modelling of Offshore Wind Turbine Blades

Bak et al. (2013) provides a data repository for the DTU 10MW RWT. This includes all necessary data for modelling both the external shape and internal structure of the rotor. A common terminology in order to describe the airfoil shape of the wind turbine blade is provided by Hau (2013). Figure 5.1 shows an air foil profile, or the cross section of the outer surface of a wind turbine blade. The dashed line located halfway between the top and bottom surface of the air foil is called the mean camber line. Furthermore, the forward end point of the mean camber line marks the leading edge, and the rear point marks the trailing edge. The connecting line drawn between the two edges is named the chord line. The chord line has a significant role when it comes to describe the design with respect to distances between regions and webs, which is described in detail in Section 5.1.

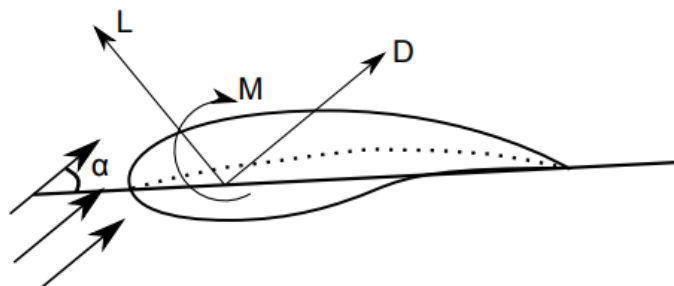
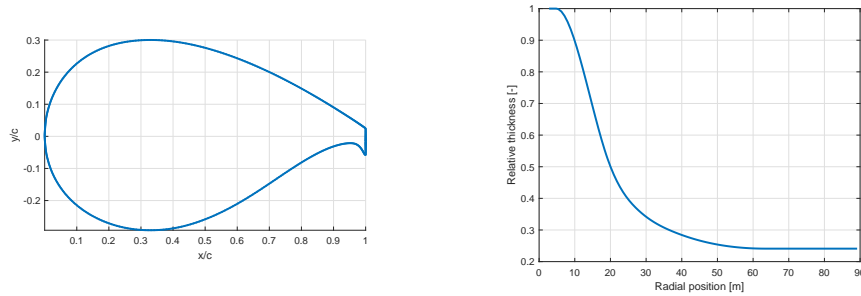


Figure 5.1: Basic concept of airfoil and plotted lines (Bachynski, 2017)

The thickness is defined as the distance between upper and lower surface, measured perpendicular to the chord line. The term relative thickness is introduced to describe the thickness of an airfoil, and is defined as the thickness divided by the chord length. This creates an expression of the thickness as a ratio of the chord length. Figure 5.2 presents two different ways of interpreting the term relative thickness. The term is understood to be applicable for both a local and global context. In Figure 5.2a, a single airfoil section is considered in 2D, and the term relative thickness is expressed in a local manner. Here the relative thickness is

plotted for all x- and y-coordinates, giving a detailed view of the airfoil at the respective blade position. Furthermore, Figure 5.2b shows the relative thickness across the span of the blade. Here the relative thickness is interpreted globally, where the maximum relative thickness for all sections is extracted and plotted along the span.



(a) 2D view of the relative thickness for a FFA-W3-600 airfoil with Gurney flap. (b) Relative thickness as a function across the blade span.

Figure 5.2: Two ways of interpreting the term relative thickness.

The angle of attack, α , referred to in Figure 5.1, is the angle between the relative wind direction and chord line. This angle contributes to the lift of the foil, which is not relevant for this thesis. With reference in Figure 5.2a, the upper and lower side of a wind turbine rotor is known as the suction and pressure side. When referring to the position along the span, one uses the radial position r in wind turbine terminology. Furthermore, terminology like flapwise direction refers out of plane loading, and edgewise direction refers in-plane loading of the blade.

5.1 Blade Geometry Description

The DTU 10MW Reference Wind Turbine is described in its completeness by Bak et al. (2013) in the report, and some key features will be highlighted here. Since this is a reference design for commercial use, the wind turbine has many common design features. Among these features are the number of rotor blades, which is three. The hub has a radius 2.8 m and the blades has a length of 86.466 m. In order to ensure tower clearance, the blades has a pre-bend which decreases the projected blade radius to 86.366 m. For the structural analysis the pre-bend is not included. The fully assembled wind turbine is presented in Figure 5.3

Different circumferential regional definitions are applied to distinguish the material properties of the blade. These regions are shown in Figure 5.4 defined from a straight line perpendicular to the chord line. The size of each region is decided from a percentage of the chord length, and is kept constant throughout the entire span of the blade.

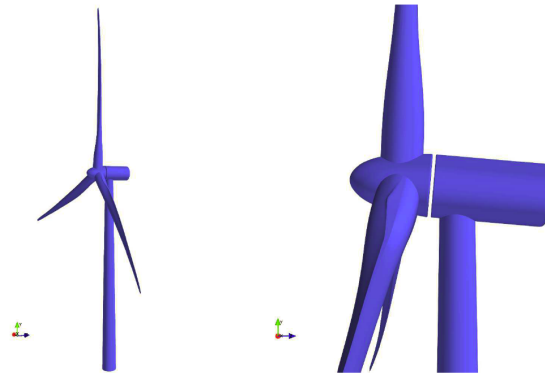


Figure 5.3: The fully assembled geometry model of the reference wind turbine, including tower, nacelle, spinner and rotor (Bak et al., 2013).

The blade is fitted with two shear webs with a cap in between, which makes up what is known as a box spar. Traditionally, the box spar and outer shell appear as constituent parts which are manufactured separately and then joined together in a bonding process. However, for the DTU 10 MW RWT blade, an alternative solution is utilized, where the webs connect the caps from each side of the blade, which also makes up the spars. Thomsen (2009) states that mainly the spars restrict most of the effects from flapwise bending, and that the webs carry the torsional and shear loading. A third web is fitted close to the tail with same purpose as the first two webs, in addition to take care of the buckling strength in the tail area. This shear web starts at the radial position $r = 21.8m$

Unlike the definition of the regions, which are perpendicular to the chord, the webs are perpendicular to the x-axis. In Figure 5.4, three key distances along the chord line for locating the webs are presented. The distance a represents the distance between the two webs. The cap centre is located along the chord line in the centre of the two webs. Furthermore, the distance b marks the distance between the cap centre and pitch axis. Finally, the distance c is the distance from the leading edge to the pitch axis. These three distances are defined as a function along the span of the blade and presented by Bak et al. (2013) as a spline in the data repository. Section 5.2.4 describes how these splines are utilized in order to create the internal structure of the isogeometric analysis model.

Thomsen (2009) claims that edgewise and flapwise bending are responsible for 97% of the fatigue damage in the blade. The edgewise bending is primarily resisted by the leading and trailing edges of the outer shell, which is strengthened due to this fact. The nose-region and the four different tail-regions viewed in Figure 5.4 is designed for this exact purpose.

In addition to the 11 circumferential regions, the structural design of the wind turbine blade is defined with 100 radial regions. These regions have varying material thickness and properties along the span. Figure 5.5 shows the complete build-up of the $11 \cdot 100 = 1100$ regions, where the external regions are extracted from the internal structure.

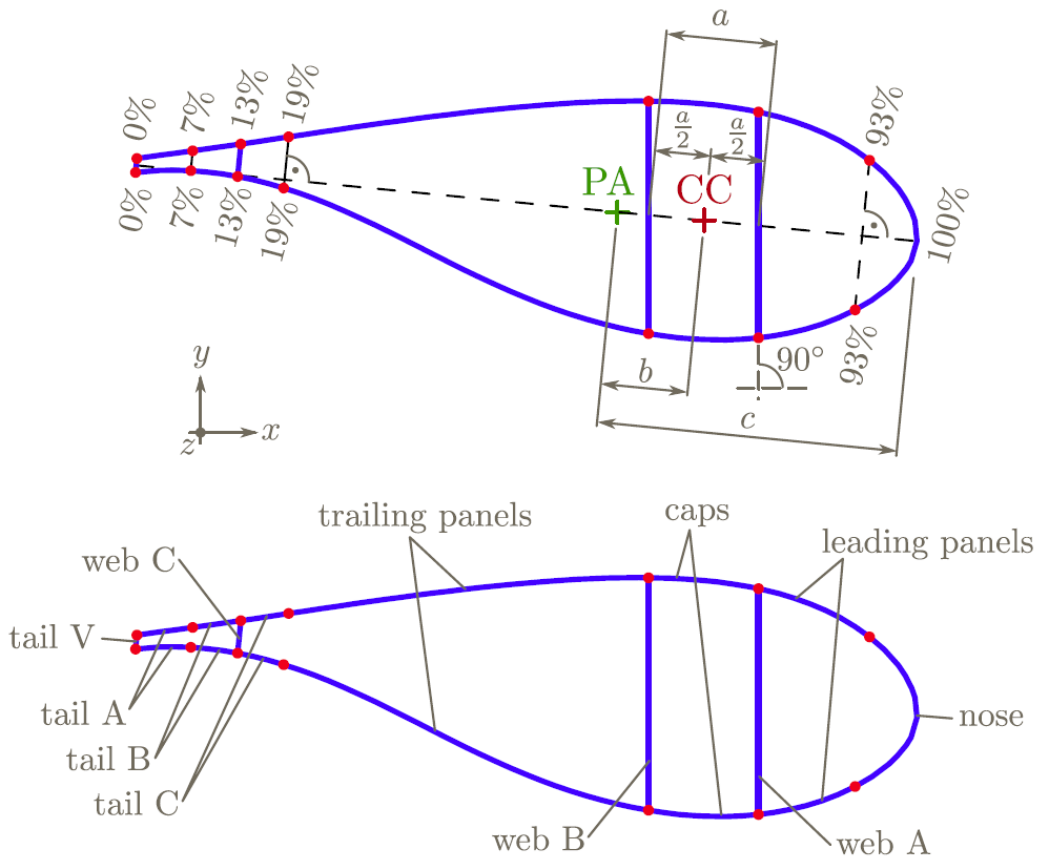


Figure 5.4: Definition of blade regions and internal structure (Bak et al., 2013), where PA is the pitch axis and CC is the cap centre.

5.2 Rhinoceros As a Tool for Modelling

In this thesis, Rhinoceros, or *Rhino*, by Robert McNeel & Associates is utilized as a tool to create the NURBS-model for the isogeometric analysis. The modelling is done in the way which is best suited for a multipatch analysis. The outer geometry is represented by two patches, where the lofted airfoils with unconnected edges is one patch, and the second patch is the trailing edge, tail V, of the blade. For the internal structure, three separate patches are applied to represent the three webs. In this section, the procedure for how the analysis model is obtained with data from the report and repository of the DTU 10MW RWT. Throughout this thesis, reference to the circumferential parametric direction will be made to U-direction. Similarly, V-direction refers to the longitudinal or radial direction of the rotor.

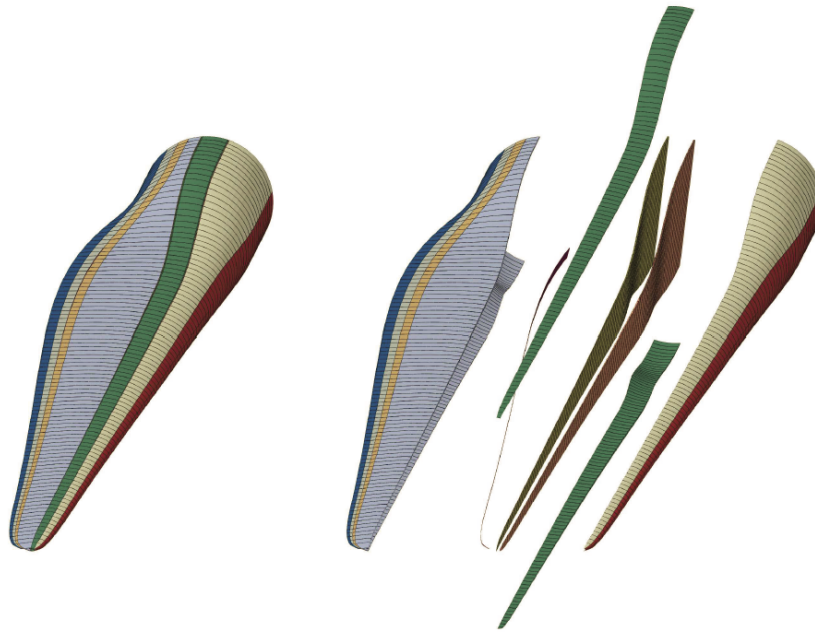


Figure 5.5: Regions of the wind turbine blade outer surface and internal structure (Bak et al., 2013).

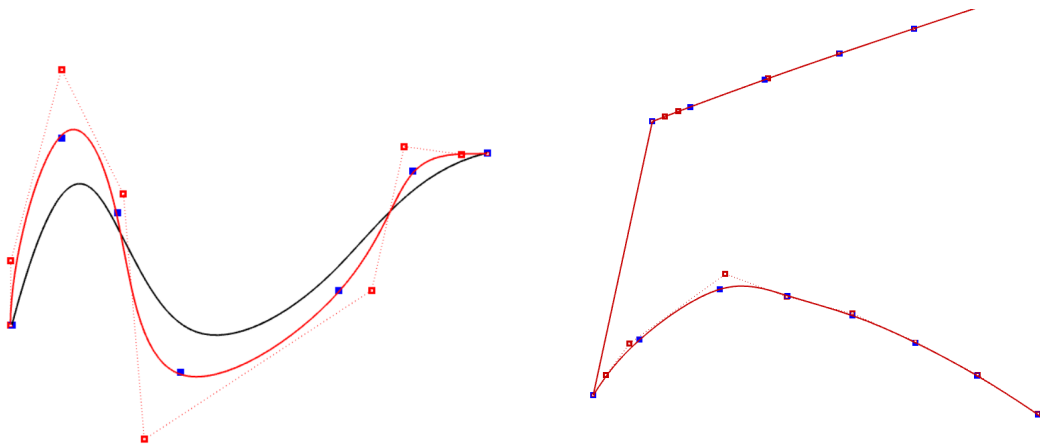
5.2.1 Importing Points

In the data repository, 101 cross sections with 109 points are located, which are utilized to create the airfoil sections of the wind turbine blade. These are the same Cartesian coordinates utilized for describe the geometry in the reference finite element analysis. However, the points are not a representation of the NURBS control polygon, but cubic spline interpolation points. In Figure 5.6a an exaggerated example shows how interpolation points and control points for the same curve has completely different coordinates. The red points are the control polygon for the NURBS curve, blue points are spline interpolation points and the black curve are the NURBS curve if the spline interpolation points enter as control points. This shows why cubic spline interpolation points can not be applied as control points for a NURBS curve. The cross sections utilized in the reference blade are cubic spline interpolation curves which is lofted to surface of polynomial degree $q = 3$.

Importing points to Rhino can be done from a text file, where the coordinates are on the form (x, y, z) delimited by commas. The cross section file from the data repository is on a form where the respective span position is the header, and a list of x- and y- coordinate pairs. Since the points are arrange differently, a script in Matlab creates a text file with the same convention as in *Rhino*. This new text file can then be imported directly and appear as a point cloud in the modelling environment.

5.2.2 Creating NURBS Control Polygon

Rhino is a NURBS-based CAD software that offers a variety of methods for creating free-form curves. One of them is to use picked points and draw a spline interpolation curve. In Rhino, the *IntrpCrv*-function allows the user to create a spline interpolation curve with a specified degree of the curve. However, the output from the function is a NURBS-curve with a corresponding control polygon, rather than a parametrization of the spline interpolation curve. This means that by choosing the degree $p = 3$ and uniform knots, the cross sections of the geometry model can be recreated in Rhino by picking all data points in a sequential order. Now the curves for the cross sections corresponds to the parametric U-direction, and their span corresponds to the V-direction of the model. Figure 5.6b shows an example from the trailing edge of a cross section close to the root, with interpolation points in blue and the NURBS curve and its corresponding control polygon in red. This illustrates different interpolation of the points used for the two mentioned free-form curve methods for this specific model.



(a) Interpolation points and control polygon for the same curve. (b) Display of points at the trailing edge of an arbitrary cross section.

Figure 5.6: NURBS curves with control polygon in red and interpolated points in blue.

5.2.3 Model Coarsening

When the cross sections are established, a surface is lofted through them in order to create the exact geometry of the wind turbine blade outer surface utilized as basis for the reference structural analysis. However, one of the scopes in this thesis is to use an IGA model with fewer numbers of degrees of freedom than the reference model. This means that the geometry model needs to be coarsened. Rhino has a built in function which allows for model coarsening by entering the desired number of control points and calculating the deviation from the original model. Robert McNeel & Associates reports that the calculation-algorithm tests for the distances between knots and halfway between knots of the two surfaces. As men-

tioned, the original geometry model uses 101 cross sections with 109 points (109x101). With the *Rebuild* function, several combinations of points in U- and V- direction are executed by the means of finding a combination that has low deviation from the original surface, as well as being coarser. The user is allowed to decide calculate the maximum deviation from the input surface while trying new combinations. In Table 5.1 some of the combinations of U and V are presented with their respective deviation from the original surface. With these criteria, the best combinations is found to be 50x60, which gives a maximum deviation of 2.3cm.

U	V	Maximum Deviation [<i>m</i>]
60	70	0.041
50	61	0.0233
70	60	0.035
60	60	0.043
51	60	0.022
50	60	0.0228
49	60	0.025
40	60	0.079
50	59	0.028
60	50	0.046
30	40	0.091

Table 5.1: Combinations for model coarsening.

5.2.4 Internal Structure in Rhinoceros

The internal structure described in Section 5.1 needs to be added to the new blade model created in Rhino. In addition to the trailing edge, the three internal webs are modelled as separate independent patches. The trailing edge is simply created from lines which connect the start and the end of the cross sections. Creating the shear webs is more complex, since their position somewhat irregular. As mentioned earlier, the internal structure is recreated with information of the three distances a , b and c in the data repository. Figure 5.7 shows the three mentioned distances as a plot along the span, which illustrates the complexity of the web structure, since their position is dependant on three varying distances, in addition to the varying chord length. Also, *Rhino* is primarily a free form surface modeller where technical drawing features like distance, angle and tangential constrains is less intuitive to add.

The procedure for creating the patches of the webs in *Rhino* starts with identifying the leading and trailing edge of the respective cross section. Between these points the chord line defined. Using *snap* options like *Tangent* and *SmartTrack*, the distance between the leading edge and pitch axis, c , is found on the the chord line. Similarly, b is found as a distance from the pitch axis and back towards the leading edge of the air foil. The with of the caps is

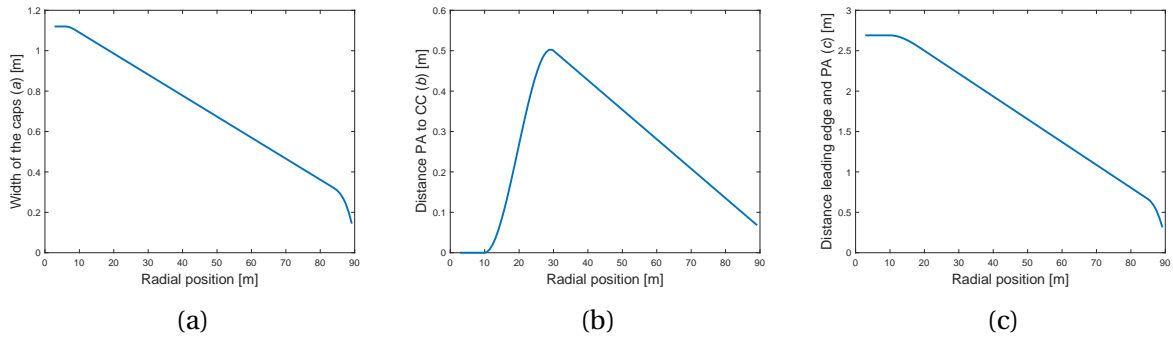
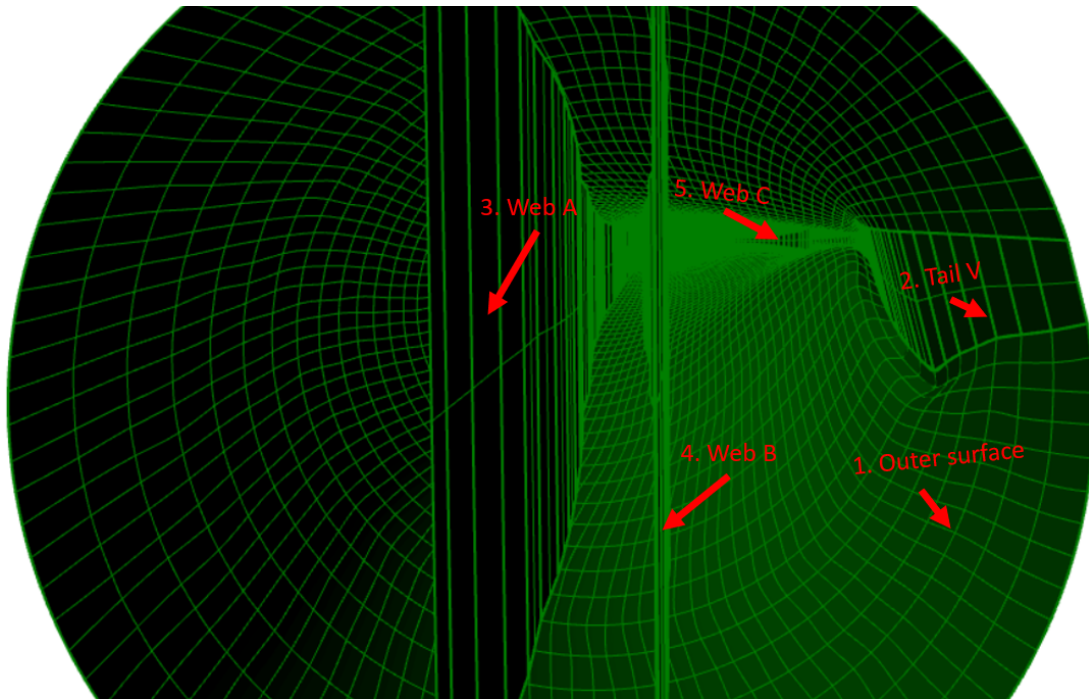


Figure 5.7: Definitions of internal structure.

then found from the distance a . The distances b and c are created with the default settings of the *Line*-function is created in Rhino with the *Line*-function. Since the distance a is a width from a centre, the option *BothSides* is selected in order to create a line from the midpoint. The third web is created from a point at 13% of the chord line, as shown in Figure 5.4. When the point at the chord line for all webs is identified, the *orthogonal* snap option is selected and a line is drawn to the cross section outer surface. A surface is then lofted through each web cross section, creating the internal structure of the blade. A detailed view of the blade consisting of five surfaces are shown in Figure 5.8.

Figure 5.8: Wind turbine blade in *Rhino* fitted with webs

For the analysis, a multipatch IGA model coupled with bending strips is also to be tested. This has to be accounted for in the modelling. As mentioned, coupling with bending strips requires a shared control point between the two coupled patches. This is only the case for the

patch representing the edge of the blade, i.e. some modifications of the current model needs to be executed. In order to keep the correct width of the caps as well as perpendicular webs to the x-axis, the modification is done by snapping a control point from the outer surface to the end of each webs. This obviously adds some change to the geometry, which is reported in Tables A.1-A.5, where the maximum deviation from the rebuilt cross sections is now found to be 8.8cm. Furthermore, a script written by Golay (2008) calculates the distance between two surfaces, finds a deviation 9.23cm between original geometry model and the rebuilt surface with manipulated control points. Figure 5.9 shows an arbitrary modified cross section with the pitch axis, cap centre and width. The detailed view shows the shared control point of the web and outer surface. that this is a modified where the outer surface is fitted to the webs. As can be seen in the detailed view, the outer surface and web has a shared control point. The deviation between the outer surface before and after attaching control points to the webs is calculated in the same way as with the *Rebuild* function.

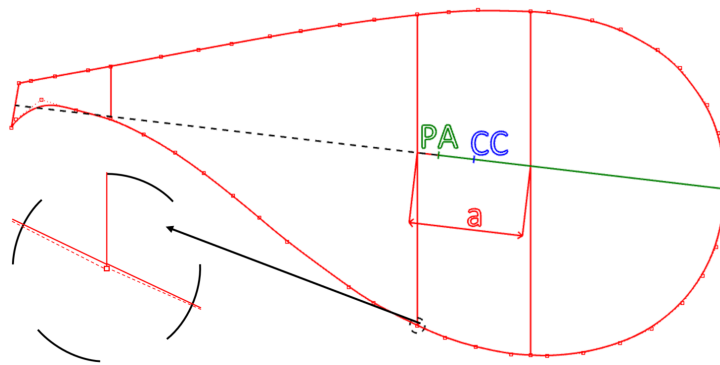


Figure 5.9: View of arbitrary wind turbine blade cross section in *Rhino* with the pitch axis (PA), cap centre (CC) and width (a).

5.3 Material Parameters

Thomsen (2009) states that wind turbine blades are most commonly designed using polymer matrix composites (PMC), in a combination with monolithic and sandwich composites. PMC is a composite material consisting of a polymer matrix reinforced by fibres bound together. Monolithic composites are continuous reinforced thermoplastic composites, and sandwich composites consists of two strong and stiff faces separated by a thick, lighter and weaker core. The DTU 10MW RWT blade layout is defined with a core material and three different plies, one unidirectional and two multidirectional. For the sandwich core material, balsa wood is applied, and the multidirectional plies has fibres going in two and three directions. The polymer matrix composite laminae consists of an epoxy matrix and E-glass fibres as reinforcement. All ply properties are denoted with suitable subscripts, with respect to type of property (fibre or matrix), plane, direction and sense of strength (tensile and compressive), as described in Chapter 4. Hinton et al. (2004) provides the mechanical properties

of the epoxy matrix and E-glass fibres used for the blade, and is presented in Table 5.2-5.3.

Property	Value	Unit
Young's modulus E_m	4.0	GPa
Poisson's ratio ν_m	0.35	-
Shear modulus G_m	1.4815	GPa
Mass density ρ_m	1140.0	kg/m ³

Table 5.2: Mechanical properties of the epoxy matrix (Hinton et al., 2004).

Property	Value	Unit
Longitudinal Young's modulus E_{f1}	75.0	GPa
Transverse Young's modulus E_{f2}	75.0	GPa
Major Poisson's ratio ν_{f12}	0.2	-
In-plane shear modulus G_{f12}	31.25	GPa
Transverse shear modulus G_{f23}	31.25	GPa
Mass density ρ_m	1140.0	kg/m ³

Table 5.3: Mechanical properties of the E-glass fibres (Hinton et al., 2004).

The composites under consideration have laminae of two different fiber volume fractions, where lamina 1 and lamina 2 uses volume fraction 0.5 and 0.55, respectively. With the matrix and fiber properties in Table 5.2-5.3 and simplified micro-mechanical Equations 4.1-4.7, the mechanical properties of the two unidirectional laminae utilized for the blade are computed. The resulting properties of each laminae are presented in Table 5.4.

Property	Lamina 1	Lamina 2	Unit
Fiber volume fraction V_f	0.50	0.55	-
Young's modulus in fiber direction E_1	39.5	43.05	GPa
Young's modulus transv. to fiber dir. E_2	12.099 01	13.425 83	GPa
In-plane shear modulus G_{12}	4.538 64	5.046 98	GPa
Out-of-plane shear modulus G_{23}	4.538 64	5.046 98	GPa
In-plane Poisson's ratio ν_{12}	0.2750	0.2675	-
Out-of-plane Poisson's ratio ν_{23}	0.3329	0.3301	-
Mass density ρ	1845.0	1915.5	kg/m ³

Table 5.4: Calculated mechanical properties of the two different laminae types.

As mentioned earlier, the composite is built from three different plies, where the first is unidirectional, and the two others are multidirectional. For sake of simplicity, Bak et al. (2013) represents the mechanical properties E_1 , E_2 , G_{12} and ν_{12} for the multidirectional plies as a single orthotropic material, derived using classical lamination plate theory. The engineering

properties for the plies are presented as in Table 5.5 and for the core material as in table 5.6.

Multidirectional Ply	Uniax	Biax	Triax	
Fiber volume fraction V_f	0.55	0.50	0.50	-
Unidirectional lamina	Lamina 2	Lamina 1	Lamina 1	-
0 deg fibres	95	0	30	%
90 deg fibres	5	0	0	%
45 deg fibres	0	50	35	%
-45 deg fibres	0	50	35	%
Young's modulus E_1	41.63	13.92	21.79	GPa
Young's modulus E_2	14.93 01	13.92 83	14.67	GPa
Shear modulus G_{12}	5.047	11.50	9.413	GPa
Shear modulus $G_{13} = G_{23}^a$	5.046 98	4.538 64	4.538 64	GPa
Poisson's ratio ν_{12}	0.241	0.533	0.478	-
Mass density ρ	1915.5	1845.0	1845.0	kg/m ³

^aChosen as identical to respective laminae.

Table 5.5: Calculated engineering mechanical properties of the two different laminae types.

Property	Balsa direction	Value	Unit
Fiber volume fraction V_f	0.50	0.55	-
Young's modulus E_1	radial	0.050	GPa
Young's modulus E_3	tangential	0.050	GPa
Young's modulus E_3	axial	2.730	GPa
Shear modulus G_{12}	radial-tangential	0.016 67	GPa
Shear modulus G_{13}	radial-axial	0.150	GPa
Shear modulus G_{23}	tangential-axial	0.150	GPa
Poisson's ratio ν_{12}	radial-tangential	0.5	-
Poisson's ratio ν_{13}	radial-axial	0.013	-
Poisson's ratio ν_{23}	tangential-axial	0.013	-
Mass density ρ		110	kg/m ³

Table 5.6: Mechanical properties of the core material.

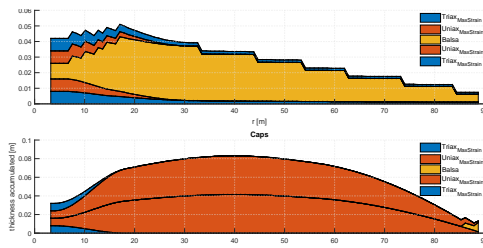
With the simplified material parameters in table 5.5-5.6, the total laminate stiffness matrix for each respective region is calculated as derived in Section 4.2. A Matlab code for calculating the **A**-, **B**- and **D**-matrix created by Badagi is modified in order to read properties of the different materials and region layup.

5.4 Regions Layup

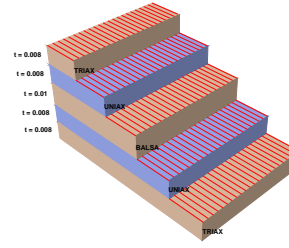
As mentioned earlier, the reference wind turbine blade consists of 11 circumferential and 100 radial regions. These regions are defined with varying thickness of the different plies and

core material from Table 5.5-5.6, which results in a unique laminate stiffness matrix for each region. Figure 5.10a shows how the layup of the leading panels and caps are distributed along the span of the wind turbine blade. These regions are neighbouring regions, i.e. stresses are transferred directly between them. As can be seen from the plot, the two neighbouring regions exhibits relatively large and abrupt thickness changes, which can be a precursor of stress concentrations. In Appendix B, the thickness of all plies in all circumferential regions are presented as plots along the span of the wind turbine blade.

In Section 4.2, the coupling stiffness matrix \mathbf{B} was presented as a part of the laminate stiffness matrix. When examining Equation 4.34, one sees that the presence of non-zero elements in the coupling matrix indicates that application of in-plane traction will lead to a curvature or *wrapping* of the plate, and similarly, an application of bending moment will generate an extensional strain. These effects are obviously undesirable. For this reason, the composite layup of the wind turbine blade is symmetric.



(a) Ply stack Leading panels and caps.



(b) Ply stack single region

Figure 5.10: Layup plot of circumferential blade regions leading panels and caps.

The regions are defined from as constant percentage of the chord length and constant distance, as shown in Figure 5.4-5.5. In the data repository, one can find the key points as Cartesian coordinates that separates the different regions. In order to read the layup data of the wind turbine blade model, these key points enters the analysis as boundaries each region. The coordinates are extracted from the data repository stored in a three dimensional matrix \mathbf{R} , where the first dimension stores information about the radial boundary, the second stores information about the circumferential boundary, and the third dimension tells whether the evaluated point is on the pressure or suction side of the blade. Now the evaluated point can be transformed to a point with Cartesian coordinates, which then is ran through a for-loop comparing it to the boundaries in the \mathbf{R} -matrix. When the region is decided for the evaluated point, suited material type, properties and thickness is registered for computation of the laminate stiffness matrix. The algorithms applied for this procedure can be fund in the Listings in Appendix C. Figure 5.11 shows how the regions of the blade are explained mathematically. Here iz , iy and ix are counting variables that decides the radial, side and circumferential region, respectively. The assignment of regions is summarized:

1. The evaluated point is transformed from the parametric u,v -space to the Cartesian x,y,z -space.
2. The counting index iz increases until the z -coordinate of the evaluated point reaches a radial boundary that it fits inside.
3. Decide weather the evaluate point is on the pressure or the suction side by calculating the distance between the y -coordinate of the evaluated point and each respective side. The side with the minimum distance is chosen to be iy (1 or 2).
4. Similarly to finding iz . Due to the twist of the blade, the circumferential boundary is dependant on whether the point is on the suction or the pressure side. With information of radial region and side, the circumferential region is found by adding up the counting index ix until the evaluated point reaches a boundary that it fits inside.
5. The variables iz , iy and ix is fed into the *struct* containing material properties and layup.

(a) An example of how the code interprets the assigned region is:

$$struct(p).layup(ix).t(:, iz) \Rightarrow struct(1).layup(4).t(:, 35),$$

where 1 is the number of the patch (Outer surface), $ix = 4$ indicates layup and material properties of trailing panels from Figure 5.11b, and $iz = 35$ decides the radial position of the region from Figure 5.11a.

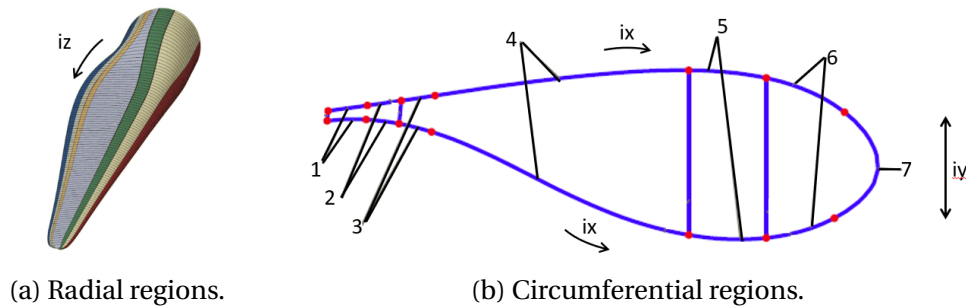


Figure 5.11: Region boundaries that enters the code.

During the modelling, the circumferential direction has been defined as the first axis in the reference system, whereas the reference finite element model has its first axis pointing in the radial direction. This means that the longitudinal strength is defined in the opposite direction for the two models. In order to solve this, the material is rotated 90 degrees in the stiffness function of the IGA model. Fiber direction is referring to the first axis of the model, as stated in Section 4.2. With a 90 degree angle added to fiber direction in the stiffness function, the conflict of material orientation between the two analysis models is corrected and further modifications is not needed for the NURBS model.

Chapter 6

Results and Discussion

In this chapter, results from the different tests will be presented. Also, a benchmark test has been performed for the adapted stiffness function which calculates the laminate stiffness for the regions. A finite element analysis has been performed as reference to the isogeometric structural analysis. This analysis applies the model which is provided in the DTU 10 MW RWT data repository, and is fitted with the boundary conditions and load cases applied in the IGA code. Two different patch coupling methods are also investigated. Penalty method is compared to the bending strip method in a simplified analysis model. Finally, IGA is performed on the blade model created in this thesis and compared to the reference model. This analysis applies the point search algorithm and couples patches with the penalty method, and has the same regional build up, laminate materials, boundary condition and load cases as the reference model.

6.1 Stiffness Function

With the classical lamination plate theory presented in Section 4.2, an algorithm that computes the **A**-, **B**- and **D**-matrix is created. A structure originally studied experimentally by Knight and Starnes (1985) is utilized as a benchmark to verify the algorithm. This is the same benchmark applied for Smith (2009) Example Problems guide. The test panel consists of 16 layers of unidirectional graphite fibres in an epoxy resin. Each layer is 0.142 mm (0.0056 in) thick and are arranged in the symmetric stacking sequence $\{\pm 45/90/0/0/90/\mp 45\}$ degrees repeated twice. The elastic material properties for the benchmark is presented in Table 6.1. It is worth noting that Knight and Starnes (1985) uses the United States customary units as input for the benchmark matrices i.e. the values with metric units are only approximations of the values with USCS units. The conversion factor applied for the benchmark is unknown, making it more convenient to use the reference matrices with USCS units:

Property	Value	Unit	Value	Unit
E_{11}	135	kN/mm^2	19.6×10^6	lb/in^2
E_{22}	13	kN/mm^2	1.89×10^6	lb/in^2
G_{12}	6.4	kN/mm^2	0.93×10^6	lb/in^2
G_{13}	6.4	kN/mm^2	0.93×10^6	lb/in^2
G_{23}	4.3	kN/mm^2	0.63×10^6	lb/in^2
ν_{23}	0.38	[-]	0.38	[-]

Table 6.1: Multidirectional and multistack composite laminate stiffness benchmark input.

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} 790.239 & 251.367 & 0 \\ 251.367 & 790.239 & 0 \\ 0 & 0 & 47.1831 \end{bmatrix} \times 10^3 \quad lb/in \quad ; \\
\mathbf{B} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad ; \\
\mathbf{D} &= \begin{bmatrix} 492.719 & 191.513 & 18.9245 \\ 191.513 & 517.951 & 18.9245 \\ 18.9245 & 18.9245 & 203.602 \end{bmatrix} \quad lb-in \quad ;
\end{aligned}$$

Using the material parameters in pounds per square inch, the algorithm computes the following \mathbf{A} -, \mathbf{B} and \mathbf{D} -matrix:

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} 790.239 & 251.367 & 0 \\ 251.367 & 790.239 & 0 \\ 0 & 0 & 47.1831 \end{bmatrix} \times 10^3 \quad lb/in; \\
\mathbf{B} &= \begin{bmatrix} 6.821 & 0 & 0 \\ 0 & -11.369 & 0 \\ 0 & 0 & 4.547 \end{bmatrix} \times 10^{-13} \quad ; \\
\mathbf{D} &= \begin{bmatrix} 492.719 & 191.513 & 18.9245 \\ 191.513 & 517.951 & 18.9245 \\ 18.9245 & 18.9245 & 203.602 \end{bmatrix} \quad lb-in;
\end{aligned}$$

As can be seen, all numbers in the \mathbf{A} and \mathbf{D} -matrix are the exact of the reference values. Also, the numbers in the diagonal of the \mathbf{B} are not equal to zero. However, these values are of several orders of magnitude lower than the other values, and are reflecting numerical rounding

errors.

6.2 Load Cases

For the comparison of IGA and FEM, two simple load cases are applied to the models. The first load case, LC1, is the self weight of the blade weight, working in the direction of natural gravity. The second load case, LC2, is a simulation of a flipped blade. This is simply modelled by changing the sign of the gravity constant. The gravity constant g is set to be $9.81 m/s^{-2}$. These load cases are the most simple load cases that can be adapted while still being realistic.

6.3 IGA Multipatch Coupling Methods

In this thesis, the multipatch wind turbine blade model is subject for analysis utilizing two different methods for coupling of NURBS patches. The scope of this analysis is to investigate how the multipatch coupling with penalty method compares to a model with bending strips. Since the input parameters are only of some importance for this analysis, computational time is prioritized and a simplified blade model with isotropic material properties is subject for analysis. These isotropic parameters, together with the simplified load case is presented in Table 6.2. The Young's modulus for the isotropic material is extracted from the longitudinal elastic modulus of the *uni*ax laminate material. Furthermore, the blade model has varying thickness, corresponding to the accumulated thickness of each respective blade region. The area load is only applied to the outer surface, acts in negative y -direction and has a magnitude of 5 400.

Parameter	Value
Young's modulus E	4.163×10^{10}
Poisson's ratio ν	0.28
Thickness t	Varying
Area Load F	-5 400

Table 6.2: Input parameters for multipatch coupling method analysis.

As mentioned in Section 3.4, the bending strip method requires a shared control point between the patches representing the outer surface and the webs. This is solved by modifying the control points of the outer surface, which obviously changes the geometry of the surface. Table A.1- A.5 stores information about the deviation from the original cross sections caused by the control point manipulation. Furthermore, the surface area deviation is only minor, and is found to increase by 0.06%. In addition, the result of the modified control points can be seen from the internal curves of the bending strip model in Figure 6.2, which represents the correspondence between control points in each respective cross section. Compared to

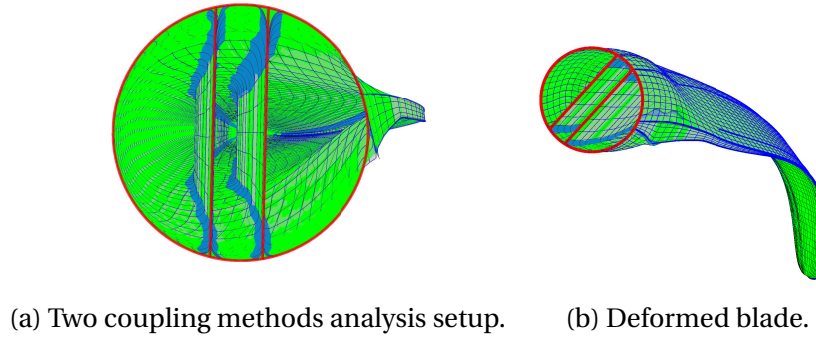


Figure 6.1: Two different models with different methods for coupling of NURBS patches. The bending strips model is fitted with light blue bending strips on the green surface, and the model coupled with penalty is the dark blue model.

the internal curves of the non-manipulated model, the bending strip model is highly irregular due to a bad parametrization. Nevertheless, the model modification does not seem to have much of an impact on the results of the tip deflection.

Table 6.3 shows that the tip displacement deviates by 0.58%. The most likely source of error for the deviation is the geometric change when the model is modified for bending strips. This indicates that the model modifications has minor influence on the tip displacement in this simple analysis and that the penalty parameter for the analysis is appropriate.

Coupling Method	Tip Displacement
Bending Strips Method	-11.4293
Penalty Method	-11.3626

Table 6.3: Tip deflection of the two IGA models.

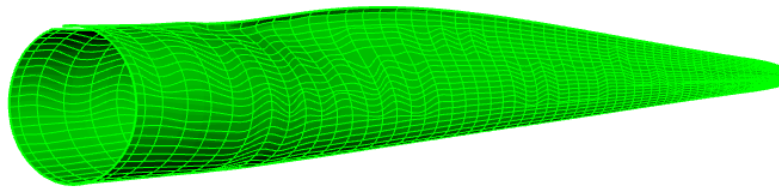


Figure 6.2: Outer surface with manipulated control points to fit internal webs.

6.4 Finite Element Method Reference Model

A complete finite element shell model of the blade's external and internal geometry and composite layup is located in the data repository is utilized in this thesis. The commercial computer aided engineering software *Abaqus/CAE*, or Complete Abaqus environment, is used

for the analysis. Material density and parameters are defined with the engineering constants presented in Table 5.5-5.6. The finite element model has the same composite layup as referred to in Section 5.4. This property of the blade is assigned to the model through the *composite layup*-option, where ply thickness and its orientation are assigned to each respective region. Moreover, the model is fitted with the load cases as presented in Section 6.2. Smith (2012) states that the *gravity load* option in the *Load* module, creates a gravity load with uniform acceleration in a fixed direction. With the density specified in the material definition, the gravity constants enter as acceleration in y -direction for both load cases. At the root of the blade, the clamped boundary condition in the root is applied on the outer surface, trailing edge and the two webs.

The FEM model has a verified mesh imported directly from the data repository. 8-node layered shell elements, *S8R*, are utilized for the outer surface, whereas solid 20-node hexahedral elements, *C3D20*, is utilized to represent the adhesive connecting, or gluing, the suction and the pressure side at the trailing edge. Because of the complex blade geometry blade, a semi-automatic meshing procedure is applied for generating the mesh. Unfortunately, the work hours behind this procedure is not reported. Finally, the whole blade is represented by a mesh of approximately 35 000 elements corresponding to 616 000 degrees of freedom. The complete setup of the analysis is shown in Figure 6.3, where the gravity load for LC1 indicated by yellow arrows and clamped boundary condition can be seen at the root. The solving procedure utilized in the analyzes is a linear static stress procedure in which inertia effects are neglected.

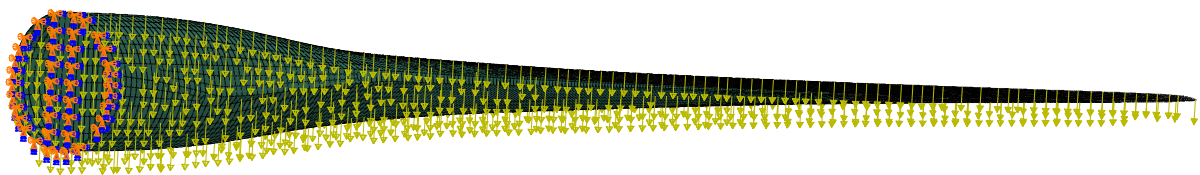


Figure 6.3: Setup of the finite element analysis with illustrative clamped boundary condition and gravity load.

The deformed configuration of the finite element analysis for the two load cases are shown in Figure 6.4. By means of the *query*-tool, tip displacement of the outer surface is extracted for the node located closest to the trailing edge. Both load cases results in the same tip displacement with different sign, as presented in Table 6.4. This is exactly as expected if one examine Equation 3.30. The only difference between the representation of the displacement vectors of the two load cases is the sign of the external force vector.

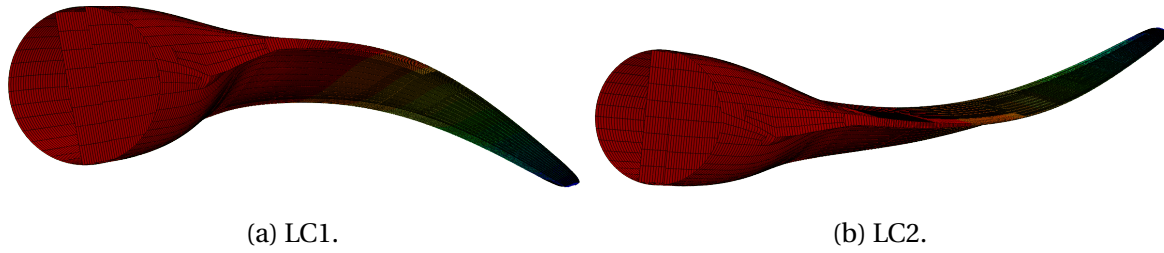


Figure 6.4: Deformed configuration of the reference FEM analysis for the two load cases.

Load Case	Tip Displacement
LC1	-1.3023
LC2	1.3023

Table 6.4: Tip deflection of the two load cases for the FEM analysis.

6.5 Structural Analysis of Wind Turbine Blade with IGA

The NURBS based Kirchhoff-Love shell is utilized for the isogeometric structural analysis of the wind turbine blade. In the research IGA-code, penalty method and point projection is utilized for multipatch coupling. From Rhino, the rebuilt coarsened NURBS model is imported to Matlab without further modifications. In the analysis, uniform knot vectors are created in both parametric directions. The final setup is shown in Figure 6.5, with the red curves representing the clamping condition imposed along the root of the blade. There are 3 000 control points in the outer surface NURBS patch. Furthermore, the trailing edge has 420 control points, Web C has 360 control points and both Web A and Web B has 540 control points. This adds up to a total of 4 860 control points, corresponding to 14 325 degrees of freedom.

Initial guesses for the point search algorithm in the parametric u -direction is added to the coupling parameter of each individual patch. As mentioned in Section 3.4, the point search algorithm searches for the orthogonal tangent plane of the opposite surface. For this model, there are two feasible solutions which fulfill this condition for the internal webs. Both the pressure and the suction side of the blade model has orthogonal tangent planes to the evaluated end points. The remedy for this is to apply initial guesses on the outer surface, close to the evaluated points of each respective patch. Additional notes to the point search algorithm input is the angle tolerance referring to the orthogonal tangential plane, which is set to be 10^{-8} . Similarly, the distance tolerance for the point is set to 0.01, which is less than the shortest distance if the point search algorithm were to find the point on the wrong side. For the initial guess in the parametric v -direction, the z -coordinate of the evaluated point is divided by the span of the blade. In Table 6.5, the initial guesses in the parametric u -direction is presented.

Patch	End 1	End 2
Trailing Edge	0	1
Web A	0.3617	0.6383
Web B	0.2340	0.8400
Web C	0.0638	0.9149

Table 6.5: Initial guesses for the point search algorithm in the parametric u-direction of the outer surface. Her the ends refers to the end points of the respective patches, in which the point search algorithm is applied for.

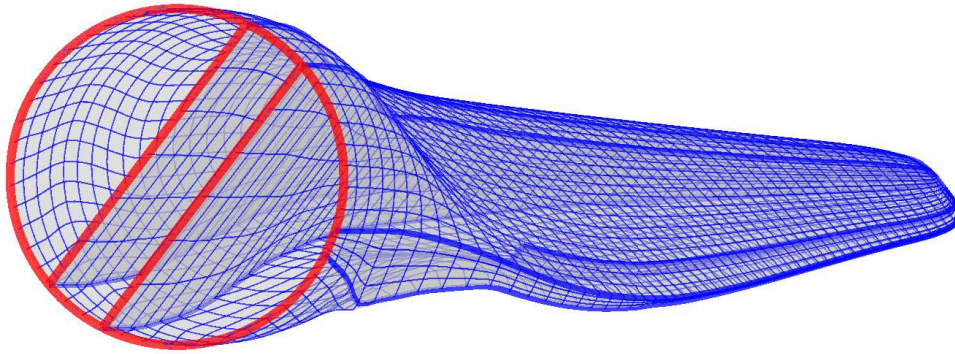
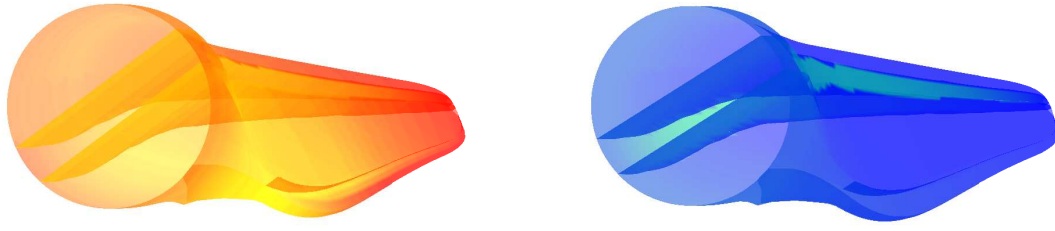


Figure 6.5: IGA model setup in the Matlab environment fitted with clamped boundary condition.

In Figure 6.6, the accumulated thickness and calculated effective longitudinal modulus for all regions are plotted as a contour plot. The material thickness contour plot in Figure 6.6a varies from red to yellow, with yellow being the thickest. One can see from the plot, that relative flat regions like the tail C, tail B and trailing panels are thicker than regions with more curvature, like the nose and leading panels. This is due to the fact that the load is carried through the shape in curved regions, rather than the material strength. The effective longitudinal modulus contour plot in Figure 6.6b varies from blue to green, where green is the stiffest. Here the caps region distinguishes from the other regions, and if one examines the composite layup plot for caps in Figure 5.10a, one can see that the dominating material is *Uniax*. As stated in Section 5.1, the main task of the caps is to withstand flapwise bending. In Table 5.5, it can be seen that this ply has the largest longitudinal modulus. Moreover, the other regions show a relatively small change in longitudinal stiffness. Also, the priorities of the shear webs is not longitudinal stiffness, which also can be seen from the plot by its distinct dark blue colour.

The load cases in Section 6.2 is added to the analysis at each respective patch as a function handle. With the function handle, all Gaussian points are fitted with the same load as in the reference finite element analysis. The load function utilizes the same algorithm applied to decide the region. When the Gaussian point is assigned to a region, the layup is interpreted with respect to thickness and mass density. Furthermore, the load magnitude f_a is calculated from:



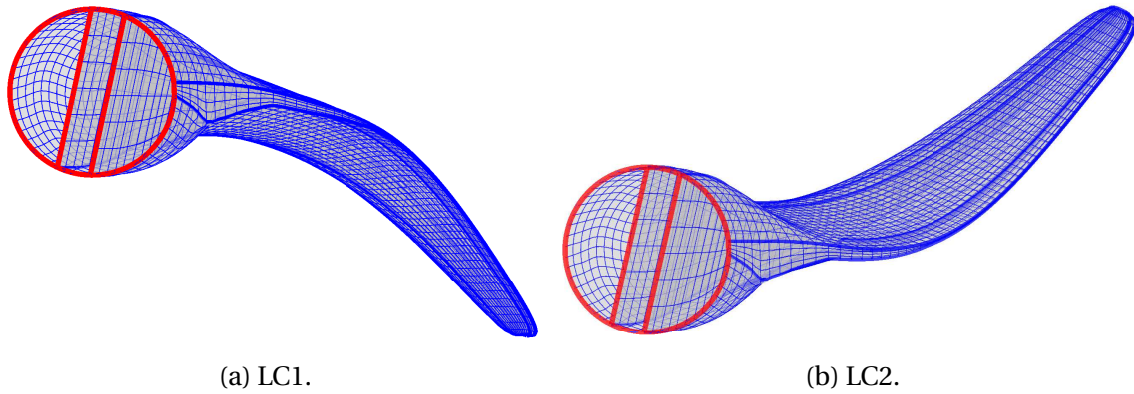
(a) Accumulated thickness of all regions. (b) Effective longitudinal stiffness contour plot.

Figure 6.6: Contour plot of accumulated thickness and effective longitudinal modulus as a result of the regional layup defined in the IGA code.

$$f_a = \rho g t = g \sum_{k=1}^n \rho_k t_k, \quad (6.1)$$

where k is the ply number, n is the number of plies, ρ_k is the mass density and t_k is the thickness of the respective ply. For the IGA-code, the load magnitude is described as force per area.

In Figure 6.7, a scaled deformation plot of the blade is displayed for LC1 and LC2. As for the finite element reference analysis, the deformation for the two load cases are the same but with different signs. This implies that there are no abnormalities in the IGA code. The tip deflection is extracted for the isogeometric analysis and shown in Table 6.6.



(a) LC1.

(b) LC2.

Figure 6.7: The scaled deformed configuration of the IGA model for both load cases.

Load Case	Tip Displacement
LC1	-1.2979
LC2	1.2979

Table 6.6: Tip deflection of the two load cases for the FEM analysis..

A plot of the displacement along the span for LC1 is created and compared to the finite element analysis in Figure 6.8. This shows a displacement pattern which is virtually identical.

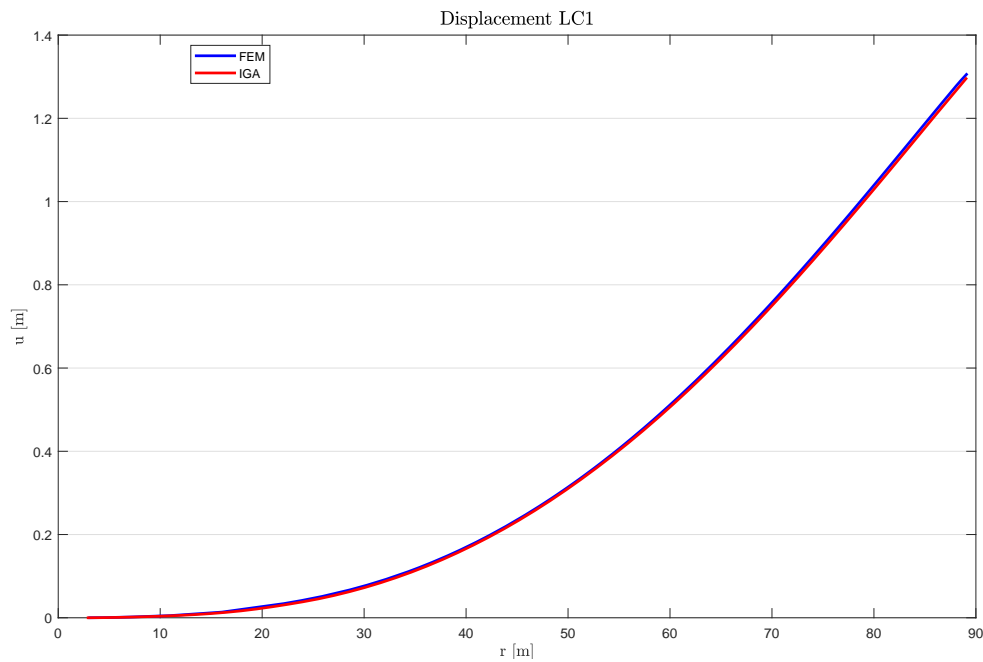


Figure 6.8: Displacement plot along the span for the two analyzes for LC1. The displacement pattern are the same and has only minor deviation.

From the displacement plot, one can see that the displacement is somewhat underestimated for all positions. Nevertheless, the displacement vector obtained in IGA is in good accordance with the reference results from the finite element method.

Chapter 7

Conclusions and Outlook

IGA has been performed on a non-matching multipatch model of the DTU 10MW Reference Rotor. From the CAD model applied to obtain the finite element model, Rhino has been implemented as a tool to create a coarsen model for IGA. As a result of the geometrical modelling, a coarsen multipatch NURBS model with only minor deviations from the original model has been created.

Two different methods of multipatch coupling has been investigated for a simple IGA model. Penalty coupling method has been compared to coupling with bending strips, where the bending strip model has been modified in order to achieve a common interface between the internal patches. Performing IGA on the modified blade model illustrated one of the advantages of IGA, where changes to the geometry did not need a new mesh for the analysis. Moreover, the modification done to the control points of the outer surface influenced the area, which again increases the external load vector. Also, the two models have different internal stiffness, which also affects the displacement. These effects causes an of increases the tip displacement by 0.58%. Nevertheless, this deviation is connected with the geometrical change and not the method of coupling. The penalty parameter $\alpha = 10^3$ is considered as acceptable for further analysis of the blade model.

The reference FEA has been modified in order to compare with IGA. In addition to the regional definitions in the model, clamped boundary condition and gravity load case has been added to the analysis framework. Results obtained from the FEM analysis for both load cases shows a realistic displacement pattern and tip deflection magnitude, and is utilized as reference in further analyzes.

With basis in the reported reference blade, IGA is performed on a multipatch coarsen NURBS model. A function for assigning the correct region to the parametric NURBS surface has been created with the purpose of assigning composite layup and corresponding material parameters. Furthermore, a code that utilizes composite laminate plate theory has been adopted in order to evaluate the element laminate stiffness. Compared to the reference model, IGA

is found to perform with great efficiency and accuracy. The number of degrees of freedom is reduced significantly in IGA, and the magnitude of the tip deflection is found to deviate by 0.34%.

The static structural analysis done in this thesis is only one of many possible analyzes for this type of model. Buckling analysis is commonly done on shell models, and with the exact geometry provided in the blade model from this thesis. This analysis has been done for the reference blade, and would interesting to perform for the IGA model as well. Also, non-linear, fatigue and dynamic structural analysis are other analyzes that could be done for this model.

One of the greatest advantages of IGA is that the meshing procedure between design and structural analysis is omitted. This creates an opportunity for the design to be modified, analyzed and optimized without having to script the mesh generation procedure. Design of wind turbines are generally based on parameters like span, chord length, airfoil shape and thickness. By the means of these parameters, a *parametrized* IGA blade model can be subject for multidisciplinary design optimized only varying a selected number of parameters, like the work done in Herrema et al. (2017)

References

- Apostolatos, A., Schmidt, R., Wüchner, R., and Bletzinger, K.-U. (2014). A nitsche-type formulation and comparison of the most common domain decomposition methods in isogeometric analysis. *International Journal for Numerical Methods in Engineering*, 97(7):473–504.
- Arfken, G. B. and Weber, H. J. (1999). Mathematical methods for physicists.
- Bachynski, E. (2017). Basic aerodynamics for wind turbines. Compendium.
- Badagi, V. K. Abd matrix. MATLAB Central File Exchange. Retrieved: 04.06.2018.
- Bak, C., Zahle, F., Bitche, R., Taesong, K., Yde, A., Henriksen, L., Andersen, P. B., Nataranjan, A., and Hansen, M. H. (2013). Design and performance of a 10 MW wind turbine. *Wind Energy*, 124.
- Boehm, W. and Prautzsch, H. (1985). The insertion algorithm. *Computer-Aided Design*, 17(2):58–59.
- Chamis, C. C. (1983). Simplified composite micromechanics equations for hygral, thermal and mechanical properties.
- de Boor, C. (1972). On calculating with b-splines. *Journal of Approximation Theory*, 6(1):50 – 62.
- European Union (2009). Directive 2009/28/ec of the european parliament and of the council on the promotion of the use of energy from renewable sources and amending and subsequently repealing directives 2001/77/ec and 2003/30/ec.
- Golay, P. (2008). Surface deviation.
- González-Viñas, W. and Mancini, H. L. (2015). *An introduction to materials science*. Princeton University Press.
- Gopalakrishnan, S. (2016). *Wave propagation in materials and structures*. CRC Press.
- Hau, E. (2013). *Wind turbines: fundamentals, technologies, application, economics*. Springer Science & Business Media.

- Hearn, E. J. (1977). *Mechanics of materials : an introduction to the mechanics of elastic and plastic deformation of solids and structural components*.
- Herrema, A. J., Wiese, N. M., Darling, C. N., Ganapathysubramanian, B., Krishnamurthy, A., and Hsu, M.-C. (2017). A framework for parametric design optimization using isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 316:944–965.
- Hinton, M. J., Kaddour, A. S., and Soden, P. D. (2004). *Failure criteria in fibre reinforced polymer composites: the world-wide failure exercise*. Elsevier.
- Hughes, T. J., 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):4135–4195.
- Hughes, T. J., Reali, A., and Sangalli, G. (2010). Efficient quadrature for NURBS-based isogeometric analysis. *Computer methods in applied mechanics and engineering*, 199(5):301–313.
- IPCC (2014). Climate change 2014 synthesis report. contribution of working groups i, ii and iii. *Intergovernmental Panel on Climate Change*. [Core Writing Team, R.K. Pachauri and L.A. Meyer (eds.)]. IPCC, Geneva, Switzerland, 151 pp.
- Kiendl, J., Ambati, M., De Lorenzis, L., Gomez, H., and Reali, A. (2016). Phase-field description of brittle fracture in plates and shells. *Computer Methods in Applied Mechanics and Engineering*, 312:374–394.
- 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. M. (2011). *Isogeometric Analysis and Shape Optimal Design of Shell Structures*. Phd thesis, Technische Universität München, Lehrstuhl für Statik.
- Klingenberg, W. (1978). *Wilhelm Klingenberg*. Graduate Texts in Mathematics. Springer Science & Business Media.
- Knight, N. and Starnes, J. (1985). Postbuckling behavior of axially compressed graphite-epoxy cylindrical panels with circular holes. *Journal of Pressure Vessel Technology*, 107(4):394–402.
- Ko, K. and Sakkalis, T. (2014). Orthogonal projection of points in cad/cam applications: an overview. *Journal of Computational Design and Engineering*, 1(2):116–127.
- Lovadina, C., Reali, A., and Sangalli, G. (2014). What is Isogeometric Analysis. Presentation for Sintef.
- Moan, T. (2003). *Finite element modelling and analysis of marine structures*. Department of Marine Technology, Norwegian University of Science and Technology.

- Piegl, L. and Tiller, W. (1995). *The NURBS book*. Monographs in visual communication. Springer, Berlin.
- Powell, P. C. and Housz, A. I. (1998). *Engineering with polymers*. CRC Press.
- Robert McNeel & Associates. Rhinoceros.
- Rogers, D. F. (2001). *An Introduction to NURBS : With Historical Perspective*. The Morgan Kaufmann Series in Computer Graphics. Morgan Kaufmann.
- Roylance, D. (2000). Laminated composite plates. *Massachusetts Institute of Technology Cambridge*.
- Schoenberg, I. J. (1946). Contributions to the problems of approximation of equidistant data by analytic functions. *Quart. Applied Mathematics*, 4:45–99.
- Smith, M. (2009). *ABAQUS/Example Problems Manual, Version 6.9*. Simulia.
- Smith, M. (2012). *Abaqus/CAE User's Manual*. Simulia, 6.12 edition.
- The European Wind Energy Association (2017). The european offshore wind industry - key trends and statistics 2016. Technical report, Andrew Ho and Ariola Mbistrova.
- Thomsen, O. T. (2009). Sandwich materials for wind turbine blades — present and future. *Journal of Sandwich Structures & Materials*, 11(1):7–26.
- Timoshenko, S. P. and Woinowsky-Krieger, S. (1959). Theory of plates and shells.
- Toselli, A. and Widlund, O. (2006). *Domain decomposition methods-algorithms and theory*, volume 34. Springer Science & Business Media.
- Versprille, K. J. (1975). *Computer-aided Design Applications of the Rational B-spline Approximation Form*. PhD thesis, Syracuse University, Syracuse, NY, USA. AAI7607690.
- Weisstein, E. W. Fundamental forms. From MathWorld—A Wolfram Web Resource. Last visited on 15/5/2018.

Appendix A

Modelling

The maximum deviation for the manipulated bending strips model is reported for the coarsen NURBS model. Control points in the respective cross sections is moved around manually in order to create a common interface between the outer surface and internal webs. The deviation from the origin cross section is calculated with the *CrvDeviation*-function and presented in Tables A.1- A.5.

# Section	1	2	3	4	5	6	7	8	9	10	11	12
Deviation ($\times 10^{-3}$)	7.3	8.1	7.2	6.7	7.8	6.8	7.1	8.5	5.3	3.9	5.7	4.9

Table A.1: Manipulated cross sectional deviation for section 1-12.

# Section	13	14	15	16	17	18	19	20	21	22	23	24
Deviation ($\times 10^{-3}$)	5.8	3.4	4.4	5.0	4.5	3.4	3.9	3.0	4.7	3.4	4.2	3.6

Table A.2: Manipulated cross sectional deviation for section 13-24.

# Section	25	26	27	28	29	30	31	32	33	34	35	36
Deviation ($\times 10^{-3}$)	3.6	2.3	2.3	2.6	2.4	2.4	2.2	2.2	2.1	2.0	1.8	1.5

Table A.3: Manipulated cross sectional deviation for section 25-36.

# Section	37	38	39	40	41	42	43	44	45	46	47	48
Deviation ($\times 10^3$)	1.6	1.7	1.6	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.4	1.4

Table A.4: Manipulated cross sectional deviation for section 37-48.

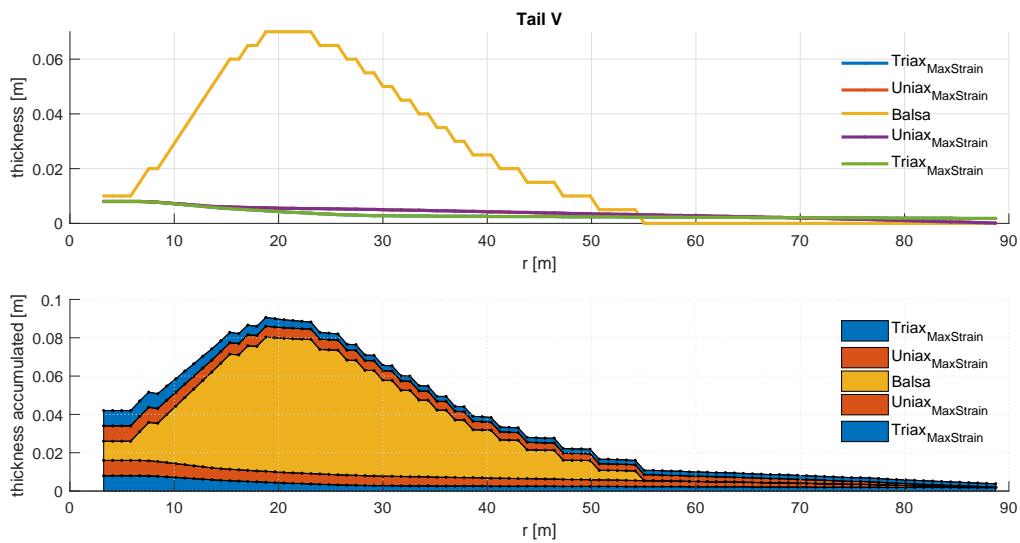
# Section	49	50	51	52	53	24	55	56	57	58	59	60
Deviation ($\times 10^3$)	1.4	1.2	1.1	1.1	8.7	1.0	.82	.81	.72	.69	.56	.31

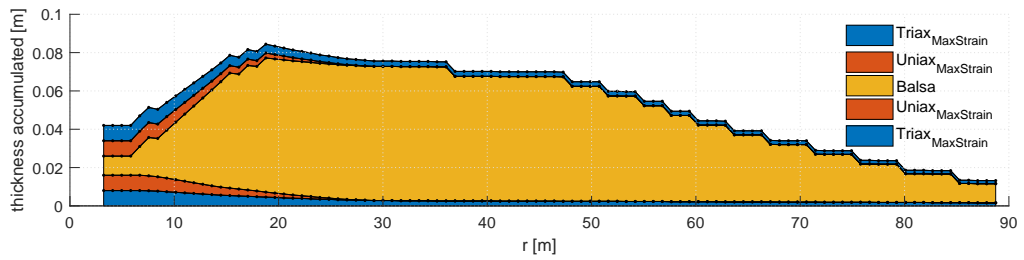
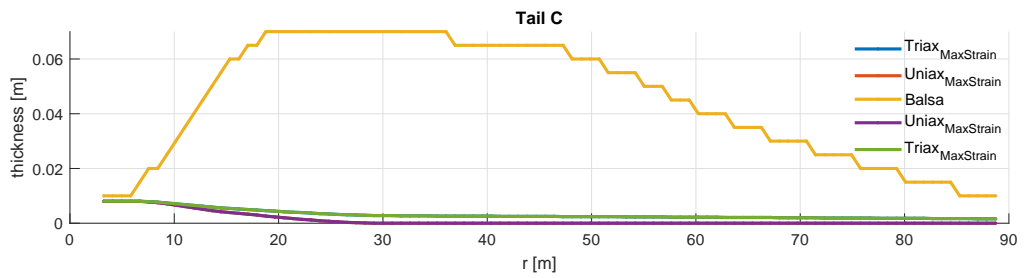
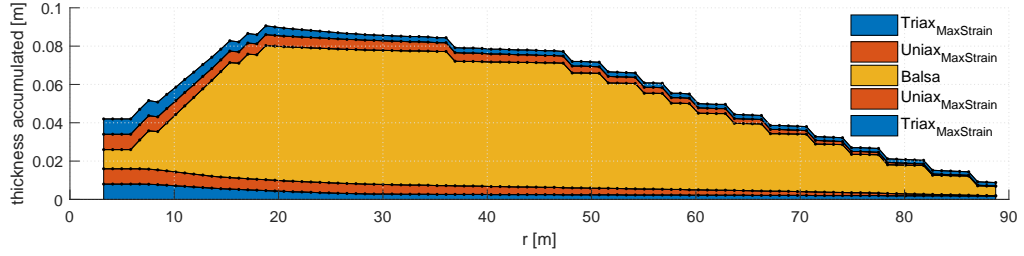
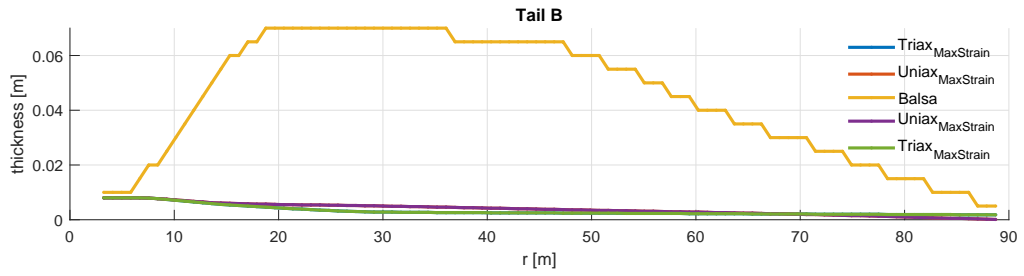
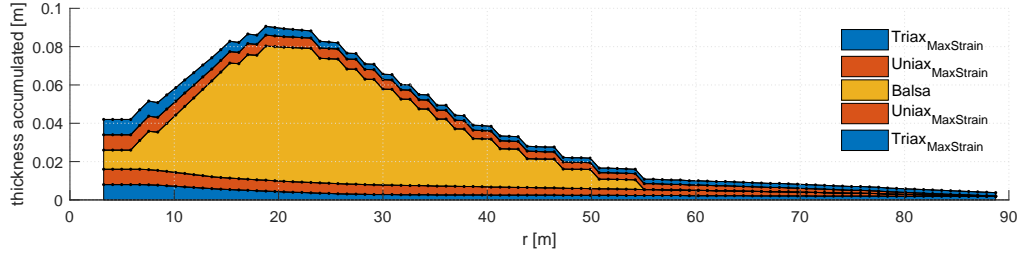
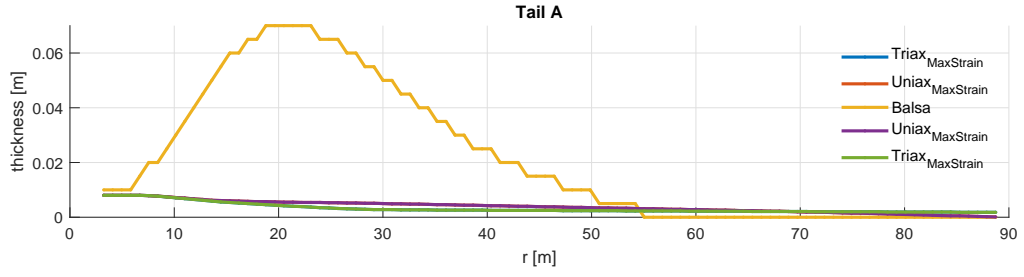
Table A.5: Manipulated cross sectional deviation for section 49-60.

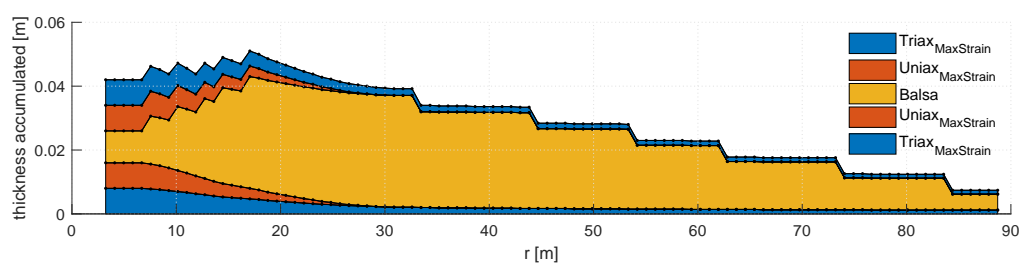
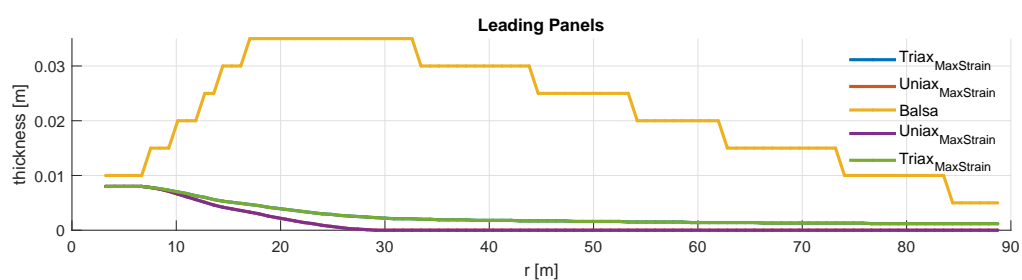
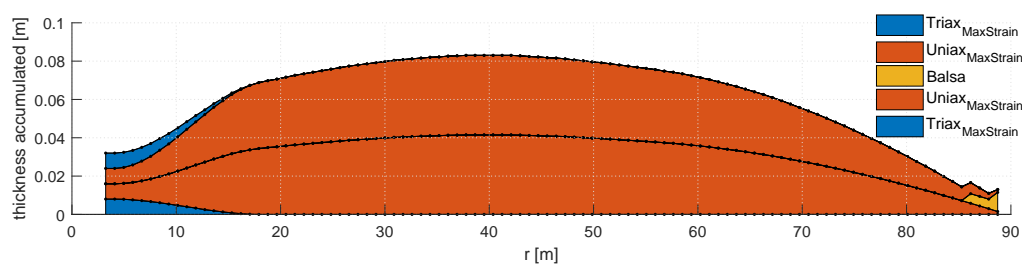
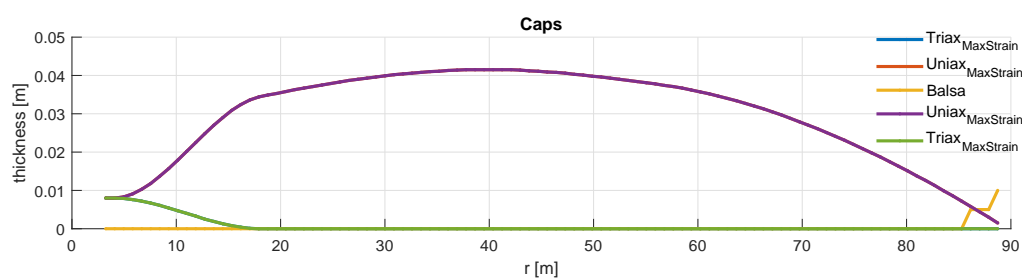
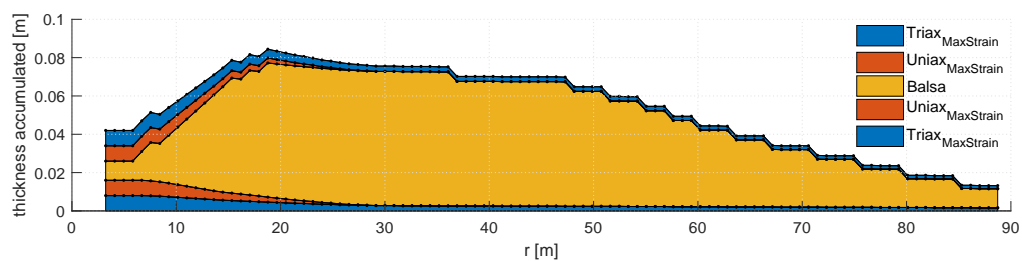
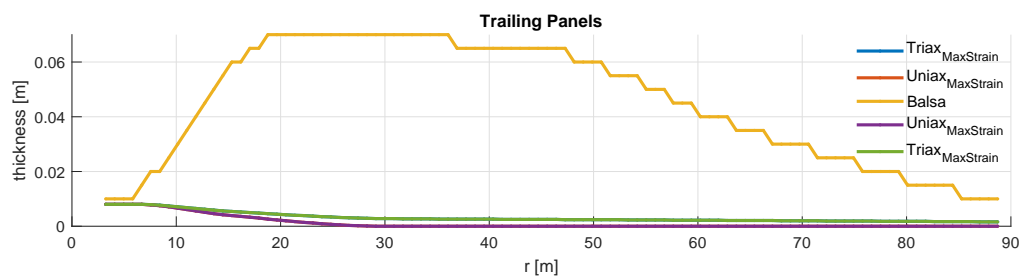
Appendix B

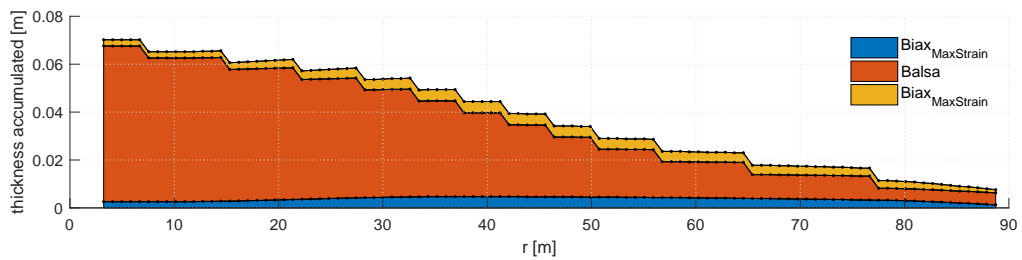
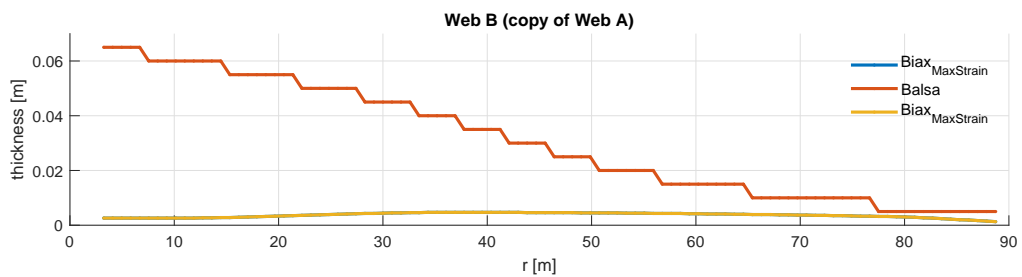
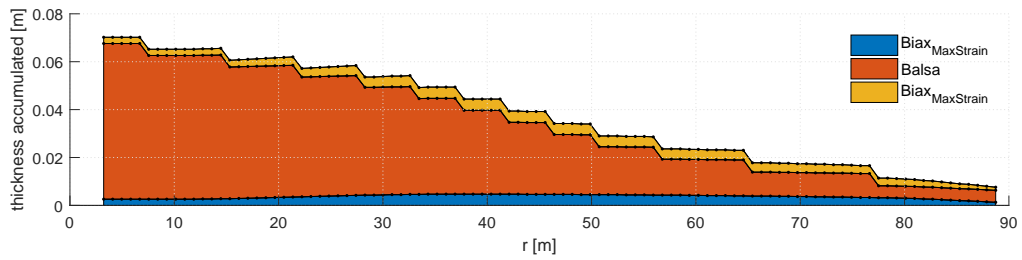
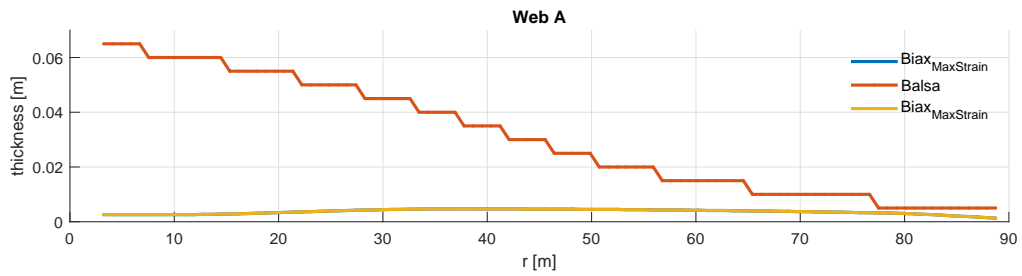
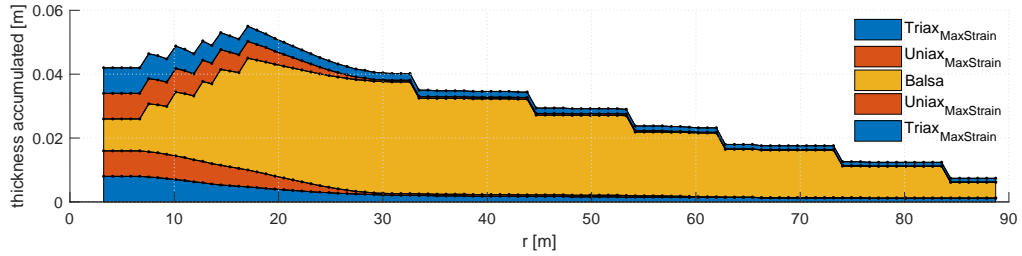
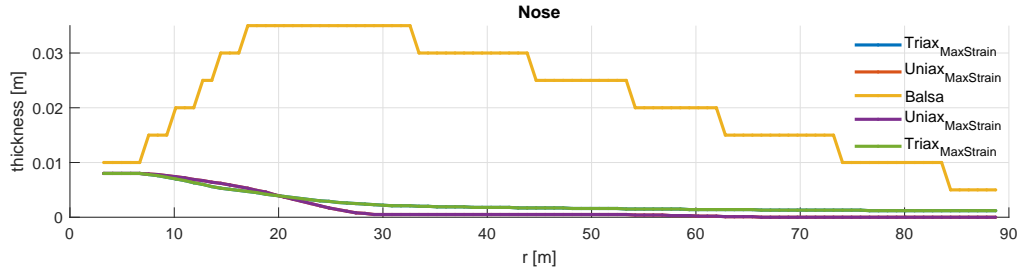
Composite Layups

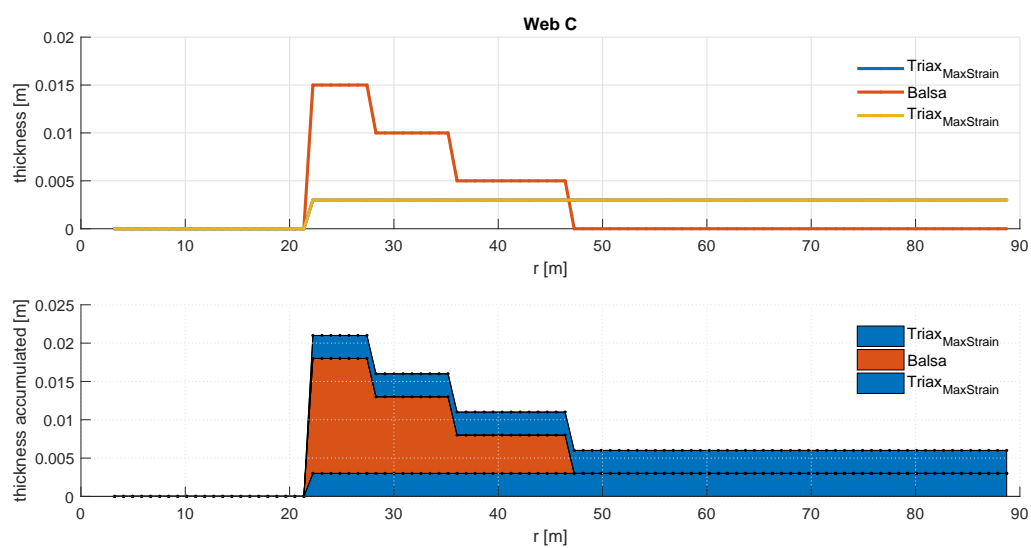
In this chapter, the composite layup for the DTU 10MW reference rotor is presented in terms of the stacking-sequence of layers representing multidirectional plies and core material. The circumferential regions and structural parts are presented as thickness plots of each respective ply and material, together with an illustrative accumulated thickness plot along the span. Plots are extracted from the data repository accompanied with the report. The name of the region is presented in the title of the plot, and the materials are labeled in the legend. It is worth noting that some regions uses the same layup.











Appendix C

Matlab

In the following chapter, the Matlab-codes used to assign the correct region to a Gaussian point, the stiffness function and load function is presented. For assignment of regions, the two matrices R_x and R_y are utilized to store regional boundaries in respective x- and y- coordinates. The first dimension of the matrix is the radial boundary and the second dimension is the circumferential boundary. The third dimension assigns the side of the blade. First the counting index for assignment of radial position is decided as:

```
1 for iz = 1:nz
2     if (z<rs(iz))
3         break
4     end
5 end
```

Listing C.1: Identify span position.

where iz is the counting index for the radial position, nz is the number of radial regions, z is the z-coordinate of the evaluated point and rs is the array of coordinates for radial boundaries. When the radial position is assigned, the side of the blade is decided as:

```
1 for iy=1:2
2     disty(iy)=abs(y-Ry(iz,1,iy));
3 end
4 [~, indexy]=min(disty);
5 iy=indexy;
```

Listing C.2: Identify side of the blade.

where $iy = 1$ is the suction (upper) side and $iy = 2$ is the pressure (lower) side, $S(2)$ is the y-coordinate of the evaluated point and $R_y(iz,1,iy)$ is the boundary for Tail A for the given radial position for the two blade sides. The side closes to the evaluated point is chosen as iy . Finally, the circumferential region is decided similarly to the radial position as:

```

1 for ix=1:nx
2     if x<Rx(iz , ix , iy)
3         break
4     end
5 end

```

Listing C.3: Identify circumferential region.

where ix is the counting index and nx is the number of circumferential regions and x is the evaluated point. The three counting indexes enters the layup structure where the correct material properties are assigned to the evaluated point. In the modelling, it is only the outer surface patch which uses circumferential regions, i.e the other patches are only considers the radial position with respect to region assignment.

An adapted code is used to compute the A-, B- and D- matrix making up the composite laminate stiffness. The code reads the region layup, stores the strength properties in a stacking sequence computes the composite laminate stiffness matrix. The stacking sequence is extracted from the *layup*-struct, and the material properties are extracted from the *matprops*-struct As mentioned in Section 5.4, the wind turbine blade has rotated material orientations to begin with, which solved by rotating the material orientations inside the stiffness function. The stiffness function for the evaluated point with assigned regional layup is then given as:

```

1 comm=layup (ix) . materials ;
2 M=length (layup (ix) . materials) ;
3
4 % Initializing
5 t=zeros (1 ,M) ;
6 E1=zeros (1 ,M) ;
7 E2=zeros (1 ,M) ;
8 E3=zeros (1 ,M) ;
9 n12=zeros (1 ,M) ;
10 n13=zeros (1 ,M) ;
11 n23=zeros (1 ,M) ;
12 G12=zeros (1 ,M) ;
13 G13=zeros (1 ,M) ;
14 G23=zeros (1 ,M) ;
15
16 for i=1:M
17     if strcmp(comm(i) , 'Uniax_{MaxStrain} ')
18         E1 ( i)=matprops . E1 (1) ;
19         E2 ( i)=matprops . E2 (1) ;

```

```

20         E3(i)=E2(i);
21         n12(i)=matprops.nul2(1);
22         n13(i)=n12(i);
23         n23(i)=matprops.nu23(1);
24         G12(i)=matprops.G12(1);
25         G13(i)=G12(i);
26         G23(i)=matprops.G23(1);
27         t(i)=layup(ix).t(i,iz);
28     elseif strcmp(comm(i),'Triax_{MaxStrain}')
29         E1(i)=matprops.E1(2);
30         E2(i)=matprops.E2(2);
31         E3(i)=E2(i);
32         n12(i)=matprops.nul2(2);
33         n13(i)=n12(i);
34         n23(i)=matprops.nu23(2);
35         G12(i)=matprops.G12(2);
36         G13(i)=G12(i);
37         G23(i)=matprops.G23(2);
38         t(i)=layup(ix).t(i,iz);
39     elseif strcmp(comm(i),'Biax_{MaxStrain}')
40         E1(i)=matprops.E1(3);
41         E2(i)=matprops.E2(3);
42         E3(i)=E2(i);
43         n12(i)=matprops.nul2(3);
44         n13(i)=n12(i);
45         n23(i)=matprops.nu23(3);
46         G12(i)=matprops.G12(3);
47         G13(i)=G12(i);
48         G23(i)=matprops.G23(3);
49         t(i)=layup(ix).t(i,iz);
50     elseif strcmp(comm(i),'Balsa')
51         E1(i)=matprops.E1(4);
52         E2(i)=matprops.E2(4);
53         E3(i)=E2(i);
54         n12(i)=matprops.nul2(4);
55         n13(i)=n12(i);
56         n23(i)=matprops.nu23(4);
57         G12(i)=matprops.G12(4);
58         G13(i)=G12(i);
59         G23(i)=matprops.G23(4);

```

```

60         t(i)=layup(ix).t(i,iz);
61     end
62 end
63
64 % Restructuring layup
65 % Remove materials of zero thickness
66 AllProps = [t; E1; E2; E3; n12; n13; n23; G12; G13; G23];
67 colsWithZero=any(AllProps==0);
68 AllProps = AllProps(:,~colsWithZero);
69 % Return to array notation
70 t=AllProps(1,:);
71 E1=AllProps(2,:);
72 E2=AllProps(3,:);
73 E3=AllProps(4,:);
74 n12=AllProps(5,:);
75 n13=AllProps(6,:);
76 n23=AllProps(7,:);
77 G12=AllProps(8,:);
78 G13=AllProps(9,:);
79 G23=AllProps(10,:);
80
81 %Begin CLPT computation
82 N=length(t);
83 all_ang = 90.*ones(1,N); % Rotate due to conflict in orientation from
    reference.
84
85 h=zeros(N+1,1);
86 H=sum(t)/2;
87     for i=1:N; h(i+1)=h(i)+t(i); end
88     for i=1:N+1; z(i)=(h(i)-H); end
89 lstf=zeros(3*N,3);
90 z=round(z,10); % Remove some of numerical error
91
92 %Begin laminate stiffness calculation
93 for k = 1 : N
94     %%Compliance Matrix elements
95     S = [1/E1(k)      -n12(k)/E1(k)  0 ;
96          -n12(k)/E1(k)  1/E2(k)      0 ;
97          0              0            1/G12(k)];
98

```

```

99      %%Stiffness Matrix elements
100      C = S^(-1);
101
102      %Reduced Stiffness
103      Q = zeros(3,3);
104      Q(1,1) = C(1,1) - (C(1,3))^2 / C(3,3);
105      Q(2,2) = C(2,2) - (C(2,3))^2 / C(3,3);
106      Q(1,2) = C(1,2) - C(1,3)*C(2,3) / C(3,3);
107      Q(2,1) = Q(1,2);
108      Q(3,3) = C(3,3);
109
110      %Transformation Matrix
111      m = cosd (all_ang(k));
112      n = sind (all_ang(k));
113
114      T = zeros(3,3);
115      T(1,1)=m^2;
116      T(1,2)=n^2;
117      T(1,3)=2*m*n;
118      T(2,1)=n^2;
119      T(2,2)=m^2;
120      T(2,3)=-2*m*n;
121      T(3,1)=-m*n;
122      T(3,2)=m*n;
123      T(3,3)=m^2-n^2;
124
125      %Transformed Stiffness
126      q=zeros(3,3);
127      q([1:2],[1:2])=Q([1:2],[1:2]);
128      q(3,3)=2*Q(3,3);
129
130      q=T^(-1)*q*T;
131      for i = 1:3
132          q(i,3)=q(i,3)/2;
133      end
134
135      lstf([3*k-2:3*k],[1:3])=q([1:3],[1:3]);
136  end
137
138  A=zeros(3,3);

```

```

139 B=zeros(3,3);
140 D=zeros(3,3);
141
142 for i=1:3
143     for j=1:3
144         q([1:3],[1:3])=lstf([1:3],[1:3]);
145         A(i,j) = q(i,j) * (z(2) - z(1));
146         B(i,j) = 1/2*(q(i,j) * (z(2)^2 - z(1)^2));
147         D(i,j) = 1/3*(q(i,j) * (z(2)^3 - z(1)^3));
148
149         for k = 2 : N
150             q([1:3],[1:3]) = lstf( [3*k-2:3*k] , [1:3] );
151             A(i,j) = q(i,j) * (z(k+1) - z(k)) + A(i,j);
152             B(i,j) = 1/2*(q(i,j) * (z(k+1)^2 - z(k)^2)) + B
                (i,j);
153             D(i,j) = 1/3*(q(i,j) * (z(k+1)^3 - z(k)^3)) + D
                (i,j);
154         end
155     end
156 end
157
158 if length(all_ang)==0
159     A= A.*[1 1 0 ; 1 1 0 ; 0 0 1];
160     B= B.*[0 0 0 ; 0 0 0 ; 0 0 0];
161     D= D.*[1 1 1 ; 1 1 1 ; 1 1 1];
162 end

```

Listing C.4: Stiffness function for a point with assigned region.

Using the same procedure, the load magnitude for the evaluated point computed from the load function given as:

```

1 comm=structure_p.layup(ix).materials;
2 M=length(structure_p.layup(ix).materials);
3 t=zeros(1,M);
4 rho=zeros(1,M);
5 for i=1:M
6     if strcmp(comm(i),'Uniax_{MaxStrain}')% comm(i) =={'Uniax_{
    MaxStrain}'}
7         rho(i)=structure_p.matprops.density(1);
8         t(i)=structure_p.layup(ix).t(i,iz);
9     elseif strcmp(comm(i),'Triax_{MaxStrain}')% comm(i) =={'Triax_{
    MaxStrain}'}

```

```

10         rho(i)=structure_p.matprops.density(2);
11         t(i)=structure_p.layup(ix).t(i,iz);
12     elseif strcmp(comm(i),'Biax_{MaxStrain}') % comm(i) == {'Biax_{
        MaxStrain}'}
13         rho(i)=structure_p.matprops.density(3);
14         t(i)=structure_p.layup(ix).t(i,iz);
15     elseif strcmp(comm(i),'Balsa') % comm(i) == {'BALSA'}
16         rho(i)=structure_p.matprops.density(4);
17         t(i)=structure_p.layup(ix).t(i,iz);
18     end
19 end
20
21 % Remove materials of zero thickness
22 AllProps = [t; rho];
23 colsWithZero=any(AllProps==0);
24 AllProps = AllProps(:,~colsWithZero);
25 % Return to array notation
26 t=AllProps(1,:);
27 rho=AllProps(2,:);
28
29 % Load calculateion
30 g=-9.81;
31 f=t*rho'.*g;

```

Listing C.5: Load function for a point with assigned region. Here the sign of the gravity constant is changed according to load case.